On Effective Personalized Music Retrieval by Exploring Online User Behaviors

Zhiyong Cheng Jialie Shen Steven C.H. Hoi
School of Information Systems, Singapore Management University, Singapore, 178902
jason.zy.cheng@gmail.com, jialie@gmail.com, chhoi@smu.edu.sg

ABSTRACT

In this paper, we study the problem of personalized text based music retrieval which takes users' music preferences on songs into account via the analysis of online listening behaviours and social tags. Towards the goal, a novel Dual-Layer Music Preference Topic Model (DL-MPTM) is proposed to construct latent music interest space and characterize the correlations among (user, song, term). Based on the DL-MPTM, we further develop an effective personalized music retrieval system. To evaluate the system's performance, extensive experimental studies have been conducted over two test collections to compare the proposed method with the state-of-the-art music retrieval methods. The results demonstrate that our proposed method significantly outperforms those approaches in terms of personalized search accuracy.

Keywords

Topic Model, Semantic Music Retrieval, Personalized

1. INTRODUCTION

Over the past decades, empowered by fast advances in digital storage and networking, we have witnessed ever increasing amount of music data from various domain applications. Meanwhile, with the proliferation of mobile devices (e.g., mobile phones and laptops) and cloud-based music service, the development of personalized music information retrieval techniques has gained greatest momentum as a means to assist users to explore large scale music collections based on "personal preference". In music information retrieval, there are two widely accepted and yet independent paradigms: content-based music retrieval [33] and text-based music retrieval [31, 28]. Due to a wide range of real applications, text-based music retrieval has been recently emerging as a popular paradigm. With this technique, users can compose several keywords to describe their music information needs and current contexts, with the expectation that the music search engine returns a list of suitable songs. However, existing methods in this paradigm only consider the relevance between songs and search keywords, while largely ignoring user's personal music preference. In fact, how an individual perceives a song is very subjective, heavily depending on his/her emotional and cultural background [30]. For example, given a query "*sad*", *whether a song is relevant* or *the relevance level of the song* with respect to "sad" is dependent on the user's personal perception on this song. Thus, for music retrieval, it is crucial to take user's personal music preference into account and effectively model the correlations among (user, song, term). In fact, the significance of leveraging user music preference has been widely recognized in the development of smart music information systems [8, 27]. However, few researches focus on 1) investigating the effects of user music preferences on search performance improvement; and 2) designing advanced schemes to catch and model such effects and exploit them in personalized music search systems.

Indeed, effective integration of user's music preference to improve retrieval performance generally requires a comprehensive understanding of user's music preference on songs with respect to search keywords. A naive approach is to leverage the assistance of end users to manually label songs with various music concepts. However, this approach could be very expensive in terms of time and expertise. In recent years, the rapid growth and popularity of online social music services such as Last.fm[1] and Pandora[2] provide excellent sources to harvest large scale user behavior information. When interacting with the social music portals, users leave rich digital footprints, which contain the details of personal music listening history, such as *which song was played by which user at what time for how long*. Through analyzing user's listening behaviors, we could obtain comprehensive information related to user music preference or taste, e.g., *which songs are played frequently by a certain kind of users* and *what are the favorite levels of a user on different kinds of songs*. Besides, in those social music portals, songs are tagged by users with different types of concepts, which reveal the semantic contents of songs. The social tags in Last.fm almost cover all the concepts that users usually use to describe songs, and have been used for text-based music search [17, 22]. The listening history of users and social tags provide us reliable sources to learn the correlations among (user, song, term), which can be used to support music search at personal level.

[1]http://www.lastfm.com
[2]http://www.pandora.com/

SIGIR '16, July 17–21, 2016, Pisa, Italy.

DOI: http://dx.doi.org/10.1145/2911451.2911491

Motivated by discussion above, in this work, we focus on designing a music retrieval system to facilitate personalized music search by jointly exploiting user listening behaviors and music tags extracted from popular social music portals. To achieve the goal, we propose a novel dual-layer topic model called *D*ual *L*ayer *M*usic *P*reference *T*opic *M*odel (DL-MPTM), which discovers two sets of latent topics - *latent music dimensions* and *latent semantic subtopics*. In this model, user's music preference is represented as mixtures of *latent music dimensions*, which are discovered based on the co-occurrence of songs in playlists and co-occurrence of *latent semantic subtopics* across songs. The latent semantic subtopics are represented as the mixtures of terms. Accordingly, the correlations among (user, song, term) can be captured by the associations of the two sets of latent topics. Based on the model, we further develop a personalized text-based music retrieval system. Comprehensive experiments have been conducted to examine the performance of the method by comparing with a set of competitors over two test collections. The results demonstrate the effectiveness and robustness of our proposed method on different types of queries and datasets. In summary, the main contributions of our work are as follows.

- Instead of conducting large scale user study, we leverage online social music data (i.e., use listening history and music social tags) to study the problem of *personalized text-based music retrieval*, which has not been well studied in existing research.
- We propose a personalized text-based retrieval method based on a novel dual-layer topic model DL-MPTM, which captures user's music preference on songs with respect to a term via the connection of two latent semantic spaces.
- The proposed system has been fully implemented and tested. An extensive range of tests have been designed to investigate different factors that affect the performance of the proposed approach and its competitors. The results demonstrate the superiority of our method over other state-of-the-art approaches.

The remainder of this article is organized as follows: Section 2 gives a overview of related work. In Section 3, we introduce the proposed personalized music search system, including the DL-MPTM topic model and retrieval method. Section 4 introduces the experimental configuration, and Section 5 reports experimental results and main findings. Finally, Section 6 concludes the paper.

2. RELATED WORK

In this section, we review the literature in two closely related domains: personalized music retrieval and topic model.

2.1 Personalized Music Information Retrieval

Driven by numerous real applications, personalized information retrieval has attracted lots of research attentions in text retrieval community, and thus various approaches have been proposed in last decades [6, 19, 29]. However, very few works have been reported in the domain of personalized text-based music retrieval. Hoashi et al. [12] leveraged relevance feedback methods to refine users profiles for search performance improvement, while the method was designed for content-based music retrieval systems. In [34], Wang et al. proposed a tag query interface which enables users to specify their queries using multiple tags and with multiple levels of preferences. This method relies on user's efforts to specify the importance of query tags in each query session. In [30], Symeonidis et al. applied the high order singular value decomposition (SVD) method to capture the associations between $(user, tag, item)$. Based on the likeliness that user u will tag musical item i with tag t, musical items are recommended to user u. However, this method suffers from the high time complexity and thus is only applicable for small scale data. Hariri et al. [11] considered the problem of personalized text-based music retrieval, where users' history of preferences are taken into account in addition to their issued textual queries. The proposed system has not been compared with other music retrieval methods and evaluated under the standard information retrieval evaluation framework.

2.2 Topic Models

This section reviews hierarchical and multi-modal topic models, which are closely related to our work.

Hierarchical Topic Model. Latent Dirichlet Allocation (LDA) [5] is an unsupervised algorithm to discover the "latent topics" underlying a large scale of text collections. Each document is modeled as a mixture of the topics and each topic is a mixture of words. In recent years, several hierarchical topic models were proposed to gain the relations between topics, such as nested Chinese Restaurant Process (nCRP) [3], tree-informed LDA [15] and nHDP [23]. A common feature of these hierarchical topic models is that they all focus on modeling the parent-child and sliding relations between topics. In these models, all topics (parent topics and child topics) are represented as the mixture of words and thus in the same semantic space. Distinguished from these models, the proposed model in this paper discovers two sets of latent topics under two different latent spaces: the latent topics in a high-level latent space are the mixtures of the latent topics in a low-level latent space.

Multi-modal LDA. Due to the success of LDA in single modality scenarios, it is extended to support multi-modal case, such as mmLDA [1], Corr-LDA [4], tr-mmLDA [24], MDRF [14], and factorized multi-modal topic model [32]. The basic philosophy behind these multi-modal LDA models is the existence of shared latent topics that are the common causes of the correlations between different modalities. In mmLDA [1], the image and text words are generated from two non-overlapping sets of hidden topics. For an image, the two sets of topics follow the same topic distribution. Corr-LDA [4] was designed so that image is the primary modality and is generated first, and each caption word is forced to be associated with an image region and is generated based on the topic of this image region. Tr-mmLDA [24] uses a latent variable regression approach to learn a linear mapping between the topic distributions of two modalities. Factorized multi-modal topic model [32] and Multi-modal document random field (MDRF) [14] generalize the modeling of two modalities to multiple modalities. In our personalized music retrieval system, there are two modalities - audio and text. Besides, because social tags are usually incomplete, the text document (formed for a song) is not complete as a corresponding document to the audio content of the song. The Corr-LDA has the merits that the topics of text words are indeed a subset of topics that occur in the corresponding

Table 1: Notations and their definitions

Notation	Definition
v	audio term in the audio word vocabulary
t	text term in the text vocabulary
y	index variable - indicating a text word is associated with which audio word
υ, z	latent music dimension and latent subtopic
s, u	song and user
v_s, w_s	audio word and text word in documents
N_u^l	number of times observing $\upsilon = l$ in user u's profile
N_l^k	number of times observing subtopic k in $\upsilon = l$
N_l^s	number of times observing song s in $\upsilon = l$
N_k^v	number of times observing audio term v in $z = k$
N_k^t	number of times observing text term t in $z = k$
V, T	vocabulary size of audio and text terms
L, K	number of music dimensions and number of subtopics
N, M	number of users and number of songs
$\boldsymbol{\theta_u}$	music interest of user u characterized by multinomial distribution over music dimensions
$\boldsymbol{\theta_\upsilon}$	property of music dimension υ characterized by multinomial distribution over subtopics
$\theta_{u,l}$	probability of $\upsilon = l$ specific to user u
$\theta_{l,k}$	probability of $z = k$ specific to $\upsilon = l$
$\phi_{l,s}$	probability of song s specific to $\upsilon = l$
$\phi_{k,v}$	probability of audio term v specific to subtopic k
$\phi_{k,t}$	probability of text term t specific to subtopic k
$\boldsymbol{\alpha}, \boldsymbol{\gamma}$	Dirichlet priors to distributions θ_u and θ_υ
$\boldsymbol{\phi_s}, \boldsymbol{\phi_v}, \boldsymbol{\phi_t}$	multinomial distributions over songs, audio terms and text terms, respectively
$\boldsymbol{\beta_s}, \boldsymbol{\beta_v}, \boldsymbol{\beta_t}$	Dirichlet priors to multinomial distributions ϕ_s, ϕ_v, and ϕ_t, respectively

image (song in our context), and an audio segment could be associated with multiple text words, which is reasonable for the annotations of an audio segment. Thus, we use the Corr-LDA as a basic component in our model. Obviously, our model is very different from these multi-modal LDA models in terms of the dual-layer structure.

3. PERSONALIZED TEXT-BASED MUSIC RETRIEVAL SYSTEM

This section presents a detailed introduction of the DL-MPTM model and the associated retrieval method.

3.1 Dual-Layer Music Preference Topic Model

In this study, we aim at designing a personalized text-based music retrieval system for searching songs, which are not only relevant to the query but also effectively satisfy user's personal music information needs. Consequently, the core research problem is how to effectively model user's music preference on songs with respect to the search keywords. Users usually prefer different types of music tracks, which can be reflected from the songs they often listen to. Meanwhile, people's music preferences on songs are highly associated with the semantics embodied by the audio contents of songs. Based on the semantics, user's music preferences can be extracted by analyzing the semantics of songs listened by the users. Further, given that the semantics of songs are modeled by song's contents and user generated annotations (e.g., social tags), the correlations among (user, song, term) can be estimated. To achieve the goal, we propose a dual-layer LDA model, which *characterizes the song's semantics based on the associations between audio contents and tags* and *models user's music interests based on the songs and their semantics*. To ease understanding of the model, we firstly introduce two important concepts.

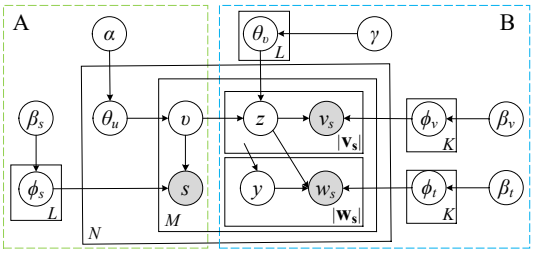

Figure 1: The graphical model representation of the DL-MPTM model. Note that the variable y are conditioned on V, the number of audio words.

- **Latent Semantic Subtopic:** Latent semantic subtopics (or subtopic for short) are the latent topics discovered (in the second layer - Part B in Fig. 1) based on the association between song's audio contents and annotations or text words. The subtopics are modeled using the multinomial distributions of audio words and text words. An audio word could be thought as a short audio segment (see Sect. 4.1).
- **Latent Music Dimension:** Latent music dimensions (or music dimensions for short) are a set of latent topics discovered (in the first layer - Part A in Fig. 1) based on co-occurrence of songs and their subtopic distributions. Users' music interests are modeled using the multinomial distributions of music dimensions. A music dimension is in turn a multinomial distribution of subtopics.

3.1.1 Model Description

Figure 1 illustrates the graphical representation of *D*ual *L*ayer *M*usic *P*reference *T*opic *M*odel (DL-MPTM). The model consists of two main components: Part A (the first layer) and Part B (the second layer). The second layer (Part B) is a Corr-LDA model [4], which discovers subtopics based on the co-occurrence of music contents (audio words and text words) in the same song. Besides, this model discovers the associations between audio contents and text words. The first layer (Part A) is a topic model to explore music dimensions υ based on the co-occurrences of songs in the same user's profile and the subtopics associated with these songs. Each subtopic z is represented by a multinomial of audio words and a multinomial distribution of text words; each music dimension υ is represented by a multinomial distribution of songs and a multinomial distribution of subtopics. The set of subtopics in the second level is shared across different music dimensions. These subtopics, which are represented by the distribution of text words or audio words, are used to characterize the music dimensions. User music interests are represented by the multinomial distribution of music dimensions. Because the music dimension is discovered based on the co-occurrence of subtopics of songs and the subtopics are discovered based on the co-occurrence patterns of songs' contents, the dual-layer topic model discovers the latent music dimensions and subtopics in a mutual reinforcement process.

From the generative perspective, a song s with text words $\boldsymbol{w_s}$[3] and audio words $\boldsymbol{v_s}$ preferred by a user u, namely, an

[3]In the paper, a notation in bold type denotes a vector or matrix.

observation of $(u, s, \boldsymbol{w_s}, \boldsymbol{v_s})$, is assumed to be generated by first choosing a music dimension υ (e.g., a certain music style) from music interest $\boldsymbol{\theta_u}$ of user u. Then based on the selected topic υ, song s is drawn according to $\phi_{\upsilon,s}$, which represents the likelihood for user u to select song s in the music dimension υ. The audio words $\boldsymbol{v_s}$ and text words $\boldsymbol{w_s}$ of song s are generated according to the subtopic distributions $\boldsymbol{\theta_\upsilon}$ of the music dimension υ. The generation process of audio words and text words is to firstly generate all the audio words, and then subsequently generate all the text words. Specifically, for each audio word v_s, a subtopic z is sampled and the audio word is generated accordingly based on ϕ_{z,v_s}. After obtaining all the audio words, for each text word, an audio word v_s is first selected and the text word w_s is generated, conditioned on the subtopic that generated the audio word. For details about the sampling process of the second layer (Part B), please refer to [4]. More formally, the process of user's profile generation is as follows:

1. For each music dimension $\upsilon \in \{1, ..., L\}$, draw a multinomial distribution $\boldsymbol{\phi_{\upsilon,s}} \sim Dir(\cdot|\boldsymbol{\beta_s})$;
2. For each subtopic $k \in \{1, ..., K\}$:
   (a) Draw a multinomial distribution $\boldsymbol{\phi_v} \sim Dir(\cdot|\boldsymbol{\beta_v})$;
   (b) Draw a multinomial distribution $\boldsymbol{\phi_t} \sim Dir(\cdot|\boldsymbol{\beta_t})$;
3. For each user u, draw a multinomial distribution $\boldsymbol{\theta_u} \sim Dir(\cdot|\boldsymbol{\alpha})$;
4. For each music dimension υ, draw a multinomial distribution $\boldsymbol{\theta_\upsilon} \sim Dir(\cdot|\boldsymbol{\gamma})$;
5. For each user u:
   - For each song $s \in \{1, ..., M\}$ in a user u's profile:
     (a) Draw a music dimension υ from the music interest distribution of user $\boldsymbol{\theta_u}$;
     (b) For each audio word $v_s \in \boldsymbol{v_s}$ in the song:
        i. Draw z from the subtopic distribution $\boldsymbol{\theta_\upsilon}$ of music dimension υ;
        ii. Draw v_s from the audio word distribution $\boldsymbol{\phi_v}$ from subtopic z ;
     (c) For each word w_s in the song (suppose there are n audio words in this song, and let z_i denote the sampled topic for the i-th audio word in previous step):
        i. Draw $y \sim \text{Unif}(1, 2, ..., n)$[4];
        ii. Draw w_s from the text word distribution $\boldsymbol{\phi_t}$ from the subtopic z_y ;

Based on the connection of two layers of topic models, DL-MPTM thus specifies the conditional joint distribution on song s and a term t given a user u and the latent variables:

$$P(s, t|u, \boldsymbol{\theta_u}, \boldsymbol{\theta_\upsilon}, \boldsymbol{\phi_s}, \boldsymbol{\phi_v}, \boldsymbol{\phi_t}) = \sum_{\upsilon=1}^{L} P(\upsilon|u, \boldsymbol{\theta_u}) P(s|\upsilon, \boldsymbol{\phi_s}) \sum_{z=1}^{K} P(z|\upsilon, \boldsymbol{\theta_\upsilon}) P(t|z, \boldsymbol{\phi_t}) \tag{1}$$

This equation estimates how correlative user u, song s, and term t could be, and thus can be used for personalized text-based music retrieval, which is introduced in Sect. 3.2.

[4]Unif$(1, 2, ...n)$ denotes the sampling of a value from 1 to n with equal probability

3.1.2 Model Inference

In the DL-MPTM model, $\alpha, \gamma, \boldsymbol{\beta_s}, \boldsymbol{\beta_v}$, and $\boldsymbol{\beta_t}$ are Dirichlet priors and pre-defined. The parameters needed to be estimated include: (1) user interest (user-music dimension) distribution $\boldsymbol{\theta_u}$, (2) music dimension - subtopic distribution $\boldsymbol{\theta_\upsilon}$, (3) music dimension - song distribution $\boldsymbol{\phi_s}$, (4) subtopic-term distribution $\boldsymbol{\phi_t}$ and (5) subtopic-audio word distribution $\boldsymbol{\phi_v}$. Several algorithms have been developed to approximate the parameters in variants of LDA. In our implementation, collapsed Gibbs sampling [10] is used to estimate these parameters, as this method has been successfully applied in many large scale applications of topic models [9, 10]. Notice that in the learning of a model, Gibbs sampling iteratively updates each latent variable given the remaining variable until it converges.

Preliminary. Given a user music profile corpus D with user set U, for each user $u \in U$, a playlist $\{s_1, s_2, ..., s_n\}$ records his/her playing behaviors or music profile. Each song s contains a sequence of text words $\boldsymbol{w_s}$ and a sequence of audio word $\boldsymbol{v_s}$. In the Gibbs sampling process, the playlists of users are sampled in sequence. Let $\boldsymbol{S}$ be the sampling sequence in the Gibbs sampling process, which is the concatenation of songs in the playlists of all the users. Similarly, let $\boldsymbol{V}$ and $\boldsymbol{W}$ denote the corresponding sampling sequences of audio words and text words. $\boldsymbol{\Upsilon}$ and $\boldsymbol{Z}$ denote the set of latent music dimensions and subtopics corresponding to the song sequence and audio words sequence, respectively. Besides, $\boldsymbol{Y}$ is the assignment indicators of the word sequence $\boldsymbol{W}$. $\boldsymbol{S_{\neg i}}$ denotes $\boldsymbol{S}$ excluding the i-th song s_i in $\boldsymbol{S}$. Similar notation is used for other variables.

Music Dimension υ Sampling for a Song For the sampling of latent music dimension $\upsilon_i = l$ for s_i, the probability is

$$P(\upsilon_i = l|\boldsymbol{\Upsilon_{\neg i}}, \boldsymbol{S}, \boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{V}, \boldsymbol{W}) \propto \frac{\alpha_l + N^l_{u,\neg i}}{\sum_{l=1}^{L}(N^l_{u,\neg i} + \alpha_l)} \cdot \frac{\beta_s + N^s_{l,\neg i}}{\sum_{s=1}^{M}(N^s_{l,\neg i} + \beta_s)} \cdot PLS(l, s_i) \tag{2}$$

$$PLS(l, s_i) = \frac{\prod_{k=1}^{K}\Gamma(\gamma_k + N^k_l)}{\Gamma(\sum_{k=1}^{K}(N^k_l + \gamma_k))} \cdot \frac{\Gamma(\sum_{k=1}^{K}(N^k_{l,\neg i} + \gamma_k))}{\prod_{k=1}^{K}\Gamma(\gamma_k + N^k_{l,\neg i})} = \frac{\prod_{k=1}^{K}(\gamma_k + N^k_l - 1)!}{\prod_{k=1}^{K}(\gamma_k + N^k_l - n_{l,k,s_i} - 1)!} \cdot \frac{(\sum_{k=1}^{K}(N^k_{l,\neg i} + \gamma_k - n_{l,k,s_i}) - 1)!}{(\sum_{k=1}^{K}(N^k_{l,\neg i} + \gamma_k) - 1)!} \tag{3}$$

where N^l_u denotes the number of times that music dimension l is observed in u's playlist. N^k_l is the number of times that subtopic k is observed in music dimension l. Notice that the exclusion of $\upsilon = l$ will cause the changes of N^k_l for all $k = [1, K]$. $N^k_{l,\neg i}$ denotes the number of times latent subtopic k is observed in latent music dimension l by excluding l assigned to song s_i, and $N^k_{l,\neg i} = N^k_l - n_{l,k,s_i}$. n_{l,k,s_i} denote the number of times the subtopic k is observed in music dimension l due to s_i. $PLS(l, s_i)$ denotes the effects of the exclusion of $\upsilon = l$ on the distribution of subtopics in the music dimension l. $\Gamma(\cdot)$ is the Gamma function.

Subtopic Sampling of Audio and Text Word: Next we introduce the sampling of subtopic $z_j = k$ for an audio word $v_j = v$ in s_i and the sampling of all text words of the song s_i. Notice that the text words in s_i are sampled after sampling all the audio words in s_i, as the assignment of z_j to the words in a song is dependent on the subtopic sequence of audio words in this song. The probability of $z_j = k$ to an

audio word $v_j = v$ is:

$$\begin{aligned} &P(z_j = k|\boldsymbol{\Upsilon}, \boldsymbol{S}, \boldsymbol{Z_{\neg j}}, \boldsymbol{Y}, \boldsymbol{V}, \boldsymbol{W}) \\ &\propto \frac{\gamma_k + N^k_{l,\neg j}}{\sum_{k=1}^{K}(N^k_{l,\neg j} + \gamma_k)} \cdot \frac{\beta_v + N^v_{k,\neg j}}{\sum_{v=1}^{V}(N^v_{k,\neg j} + \beta_v)} \cdot PZ(k) \end{aligned} \tag{4}$$

$$\begin{aligned} PZ(k) &= \frac{\prod_{t=1}^{T}\Gamma(\beta_t + N^t_k)}{\Gamma(\sum_{t=1}^{T}(N^t_{k,\neg j} + \beta_t))} \cdot \frac{\Gamma(\sum_{t=1}^{T}(N^t_k + \beta_t))}{\prod_{t=1}^{T}\Gamma(\beta_t + N^t_{k,\neg j})} \\ &= \frac{\prod_{t=1}^{T}(\beta_t + N^t_k - 1)!}{\prod_{t=1}^{T}(\beta_t + N^t_k - n_t - 1)!} \cdot \frac{(\sum_{t=1}^{T}(N^t_{k,\neg j} + \beta_t - n_t) - 1)!}{(\sum_{t=1}^{T}(N^t_{k,\neg j} + \beta_t) - 1)!} \end{aligned} \tag{5}$$

where N^v_k is the number of times that subtopic $z_j = k$ is assigned to audio word $v_j = v$. $N^t_{k,\neg j}$ denotes the number of times that t is assigned to subtopic k before assigning k to the j-th audio word of song s_i, and $N^t_{k,\neg j} = N^t_k - n_t$. n_t denotes the number of times that term t is assigned to the subtopic of the j-th audio word in the current song s_i. Notice that the exclusion of $z_j = k$ for audio word v_j may influence the assignment of $z_j = k$ to multiple text terms and multiple times. Similar to $PLS(l, s_i)$, $PZ(k)$ denotes the effects of the exclusion of $z_j = k$ on the distribution of text terms in the subtopic k.

Parameter Estimation. Based on the state of the Markov chain $\boldsymbol{\upsilon}$ and $\boldsymbol{z}$, we can estimate the parameters:

$$\theta_{u,l} = \frac{\alpha_l + N^l_u}{\sum_{l=1}^{L}(\alpha_l + N^l_u)} \quad \theta_{l,k} = \frac{\gamma_k + N^k_l}{\sum_{k=1}^{K}(\gamma_k + N^k_l)} \tag{6}$$

$$\phi_{l,s} = \frac{\beta_s + N^s_l}{\sum_{s=1}^{M}(\beta_s + N^s_l)} \quad \phi_{k,t} = \frac{\beta_t + N^t_k}{\sum_{t=1}^{T}(\beta_t + N^t_k)} \tag{7}$$

$$\phi_{k,v} = \frac{\beta_v + N^v_k}{\sum_{v=1}^{V}(\beta_v + N^v_k)} \tag{8}$$

3.2 Retrieval Model

The goal of the retrieval model is to search a subset of songs that are relevant to a particular query. Let $q = \{t_1, t_2, ..., t_n\}$ represent user u's query consisting of n terms. The retrieval algorithm aims at ranking songs based on their relevance to the query according to u's music preference on the songs. Notice that the relevance level of a song with respect to a query is dependent on user's music taste. Given a query q issued by user u, for a song s, $P(s|q, u)$ denotes the likelihood or probability of user u preferring this song s with respect to the query q. Thus, candidate songs can be ranked in the descending order of their probabilities $P(s|q, u)$ with respect to the user and query (u, q). According to Bayes rule, $P(s|q, u)$ can be computed as:

$$P(s|q, u) = \frac{P(q, s|u)P(u)}{P(q, u)} \propto P(q, s|u) \tag{9}$$

where $P(q, s|u)$ represents the relevance of song s to query q based on user u's opinions on the song.

With the posterior estimation of $\boldsymbol{\theta_u}$, $\boldsymbol{\theta_\upsilon}$, $\boldsymbol{\phi_s}$, and $\boldsymbol{\phi_t}$ in the DL-MPTM, we have:

$$\begin{aligned} &P(q, s|u, \boldsymbol{\theta_u}, \boldsymbol{\theta_\upsilon}, \boldsymbol{\phi_s}, \boldsymbol{\phi_t}) \\ &= \sum_{\upsilon=1}^{L} P(\upsilon|u, \boldsymbol{\theta_u}) P(q, s|\upsilon, \boldsymbol{\theta_\upsilon}, \boldsymbol{\phi_s}, \boldsymbol{\phi_t}) \\ &= \sum_{\upsilon=1}^{L} P(\upsilon|u, \boldsymbol{\theta_u}) \prod_{i=1}^{n} P(t_i, s|\upsilon, \boldsymbol{\theta_\upsilon}, \boldsymbol{\phi_s}, \boldsymbol{\phi_t}) \end{aligned} \tag{10}$$

where $P(\upsilon|u, \boldsymbol{\theta_u})$ is the probability of user u selecting music dimension υ, and $P(q, s|\upsilon, \boldsymbol{\theta_\upsilon}, \boldsymbol{\phi_s}, \boldsymbol{\phi_t})$ is the joint probability of query q and s in the music dimension υ. In the derivation, we assume the query terms are independent from each other under this specific music dimension. Given the music dimension υ, s and t are independent, the joint probability of term t_i and song s in the music dimension υ can be estimated by multiplying the the probability of s and t_i in the music dimension υ: $P(s|\upsilon, \boldsymbol{\phi_s})$ and $P(t_i|\upsilon, \boldsymbol{\theta_\upsilon}, \phi_t)$.

$$P(t_i, s|\upsilon, \boldsymbol{\theta_\upsilon}, \boldsymbol{\phi_s}, \boldsymbol{\phi_t}) = P(s|\upsilon, \boldsymbol{\phi_s}) \sum_{z=1}^{K} P(t_i|z, \boldsymbol{\phi_t}) P(z|\upsilon, \boldsymbol{\theta_\upsilon}) \tag{11}$$

The probability of term t_i in music dimension υ can be obtained by the generative probability of term t_i in the subtopic space: $\sum_{z=1}^{K} P(t_i|z, \boldsymbol{\phi_t}) P(z|\upsilon, \boldsymbol{\theta_\upsilon})$. Based on Eq. 10 and Eq. 11, the probability of user u selecting s for query q can be estimated:

$$\begin{aligned} &P(q, s|u, \boldsymbol{\theta_u}, \boldsymbol{\theta_\upsilon}, \boldsymbol{\phi_s}, \boldsymbol{\phi_t}) \\ &= \sum_{\upsilon=1}^{L} P(\upsilon|u, \boldsymbol{\theta_u}) \prod_{i=1}^{n} P(s|\upsilon, \boldsymbol{\phi_s}) \sum_{z=1}^{K} P(t_i|z, \boldsymbol{\phi_t}) P(z|\upsilon, \boldsymbol{\theta_\upsilon}) \\ &= \sum_{\upsilon=1}^{L} \theta_{u,\upsilon} \cdot \prod_{i=1}^{n} \phi_{\upsilon,s} \cdot \sum_{z=1}^{K} \theta_{\upsilon,z} \cdot \phi_{z,t_i} \end{aligned} \tag{12}$$

Intuitively, for a specific music dimension υ, $P(\upsilon|u, \boldsymbol{\theta_u})$ denotes the preference of user u in this dimension; $P(s|\upsilon, \boldsymbol{\phi_s})$ denotes the likelihood of song s in this dimension; $\sum_{z=1}^{K} P(z|\upsilon, \boldsymbol{\theta_\upsilon}) P(t|z, \boldsymbol{\phi_t})$ denotes the likelihood of a term t in this dimension. Thus, $P(\upsilon|u, \boldsymbol{\theta_u}) P(s|\upsilon, \boldsymbol{\phi_s}) \sum_{z=1}^{K} P(z|\upsilon, \boldsymbol{\theta_\upsilon}) P(t|z, \boldsymbol{\phi_t})$ indicates the likelihood for user u to consider song s is relevant to term t in this music dimension.

Algorithm 1 summarizes the whole procedure of personalized text-based retrieval method. DL-MPTM training process can be carried out in offline phase (line 1 - 2). Personalized music search is based on the obtained parameters in DL-MPTM, for a given query q, a rank list $\mathcal{L}$ can be returned (line 3 - 4).

Algorithm 1 DL-MPTM based personalized text-based music retrieval

Offline Phase: DL-MPTM model training
Input: User's music profiles: user-song documents, and song's contents
Output: $\theta_{u,\upsilon}$, $\phi_{\upsilon,s}$, $\theta_{\upsilon,z}$ and ϕ_{z,t_i}
1: Train the DL-MPTM model using the collapsed Gibbs sampling method
2: Estimate $\theta_{u,\upsilon}$, $\phi_{\upsilon,s}$, $\theta_{\upsilon,z}$, and ϕ_{z,t_i} using Eq. (6) - Eq.(8)

Online Phase: personalized music search
Input: A query $q = \{t_1, t_2, ..., t_n\}$
Output: A ranking list $\mathcal{L}$
3: Compute $P(q, s|u)$ using Eq. 12 based on the estimate parameters
4: Sort the songs into a ranking list $\mathcal{L}$ in the descending order of their probabilities $P(q, s|u)$

4. EXPERIMENTAL CONFIGURATION

In this section, we present the experimental settings for the performance evaluation, including test collections, query set with corresponding ground truth, competitors and performance metrics.

4.1 Test Collections

In order to achieve good repeatability of the experiments, test collections are developed based on two public datasets. Their details are as follows,

- **Taste Profile Subset (TPS)**[5] **[20]**: This dataset consists of more than 48 million triplets (*user, song, count*) gathered from user listening histories. Here, "(user, song, count)" refers to the number of times (*i.e., count*) the *user* played the *song*. It contains approximately 1.2 million unique users and covers more than 380,000 songs. From this dataset, we randomly select 10,000 users with their listening records for our experiments.

- **Lastfm-Dataset-1K (Lastfm-1K)**[6] **[7]**: This dataset contains (*user, timestamp, artist, song*) quadruples collected from the Last.fm using the public API. This dataset includes the listening history (until May 5th, 2009) of 992 users, 961,417 songs of 176,948 artists. Based on the quadruples records, we can also get the triplets (*user, song, count*) for this dataset.

In order to ensure quality of test collections, the p-core filtering method [2] is used to filter users and songs. The p-core of level k has the property, that each song was listened by at least k users and each user listened to at least k songs. In the experiments, k is set to 20. For the remaining songs, the 30 seconds audio samples were downloaded from 7digital[7], and their tags were crawled from Last.fm. Table 2 summarizes the details about the two datasets used in experiments. It is worth mentioning that two datasets have very different properties. Comparing with TPS, Lastfm-1K contains less users while each user has richer listening records. Thus, two datasets are used to examine the performances of personalized music retrieval systems in two scenarios: (1) with rich users' listening records available (Lastfm-1K), and (2) with limited users' listening records available (TPS), respectively.

Table 2: Details of two datasets used in experiments.

Dataset	#User	#Songs	#Artist	#Ave. Listened Songs per User
Lastfm-1K	992	7433	881	335.51
TPS	7022	2332	1094	15.96

The training of DL-MPTM model needs the played records of songs by users and the songs' contents, including textual content (e.g., textual words describing the song) and music content (e.g., audio words of the song). To facilitate the DL-MPTM training, we organize the related data into three types of documents. The description and generation process of the three types of documents are presented below.

User-Song Document For each user, a user-song document is generated based on his/her played records. The document is comprised by the concatenation of the songs (a "song" in a document is indexed by a unique ID) played by the users. For example, if a user u with profiles $(u, s_1, 2)$, $(u, s_2, 3)$, $(u, s_3, 1)$, the user's user-song document is $\{s_1, s_1, s_2, s_2, s_2, s_3\}$. It is worth noticing that the songs in the documents can be in any order of sequence. To accelerate the training process, the user-song document for each user is created by concatenating the songs that were played more than 2 times by the user, and each song only appears once. Thus, for each user, the user-song document is actually a playlist consisting of the songs that were preferred by the user in the past. For the users who are used as query users in experiments, half of the songs in their playlists are randomly selected as test songs and thus removed from the user-song document used in the training stage (see Sect. 4.1.1).

Song-Text Document The document contains the textual contents of the song, namely, the text words of a song used in the DL-MPTM. In our implementation, social tags are used to represent the text documents of songs. Our model is to capture the correlation of user, song, and term to facilitate personalized search. The tags of each song are collected from Last.fm using public API (Track.getTopTags). In our implementation, for each dataset, we filtered the tags that appeared in less than 10 songs. Besides, we also remove the tags which express personal preferences on the songs, such as "favorite songs", "favorite", "best song forever", etc. The remaining tags of a song are concatenated together and tokenized with a standard stop-list to form the text document for the song.

Song-Audio Document The document contains the audio content of a song, namely, the audio words used in the DL-MPTM. The audio contents of one song are represented by "bag-of-audio-words" document. An audio word is a representative short frame of audio stream in a music corpus. The general procedures to generate the audio words consists of three steps: (1) segment the audio track of each song in a corpus into short frames; (2) extract acoustic features from each short frame; and (3) apply a clustering algorithm (e.g., k-means) to group the short frames into n clusters based on their acoustic features. The cluster centers are the audio words generated for the corpus. By encoding each short frame of a song with the nearest cluster center (or audio word), then the song is indexed as a sequence of audio word document. In our implementation, we segment each song into 0.05s short frames without overlapping. Also, each song is converted to a standard mono-channel and 22,050 Hz sampling rate WAV format. Mel Frequency Cepstral Coefficients (MFCCs) [18] feature is used to generate the audio words. For each frame, a 13-d MFCCs vector with its first and second instantaneous derivatives are extracted, achieving a final 39-d MFFCs feature. We use K-means to generate the audio words. And for each dataset, we generate a vocabulary of 4096 audio words.

4.1.1 User-Specific Query, Test Collection, Ground Truth

In personalized music retrieval, a positive result should not only be relevant to the query but also be preferred by the query user[8]. In other words, to evaluate personalized music retrieval systems, we need to know (1) whether the result is relevant to the query, and (2) whether the user prefers the result. Therefore, user's preferences on all the songs in the test collection should be available in evaluation. To achieve

[5]http://labrosa.ee.columbia.edu/millionsong/tasteprofile
[6]http://www.dtic.upf.edu/ ocelma/MusicRecommendation Dataset/lastfm-1K.html
[7]https://www.7digital.com/

[8]The user who submits the query is called the query user. In personalized information retrieval, user and query should be in pairs. Afterwards, we use "query users" to refer to the users used in the search stage.

Table 3: Several examples for three types of queries.

1-Word Query	2-Word Query	3-Word Query
rock	chill, soft	00s, male, rock
metal	chill, mellow	00s, indie, mellow
piano	male, mellow	chill, mellow, rock
happy	drums, guitar	british, male, rock
rainy	country, guitar	melancholy, rock, sad
driving	danceable, harmonies	mellow, folk, tonality
energetic	alternative, guitar	guitar, rock, vocalists
romantic	emotional, romantic	chillout, mellow, rock

the goal, we create the query set and the test collection specific to each individual user. Firstly, a set of users are randomly selected from the datasets (Lastfm-1K and TPS) as query users. Then, for each user, a set of text queries are generated and a test collection for this specific user is created by randomly sampling half of the songs from his/her user-song document. In the user-specific test collection, the played times of songs can be used to estimate the user's preferences on these songs. Specifically, the relevance levels of a song with respect to a user-specific query are defined as follows,

- Non-relevant (0): song's text document does not contain all the query terms *or* the user listened to the song only once.
- Relevant (1): song's text document contains all the query term, *and* the user listened to the songs for 2 to 5 times.
- Highly relevant (2): song's text document contains all the query term, *and* the user listened to the songs for more than 5 times.

The definitions of relevance levels are based on the assumption that more times a user listen a song, higher preference level the user have on the song. The evidence that a user listened to a song more than two times indicates that the user shows some interests in the song. The songs listened to only once are regarded as irrelevant, since it could be a variety of reasons why users listen to a song only once. Notice that for a user, his/her listened songs, which are used in the *user-song document* in the topic model training stage, are removed from the test collections in the retrieval stage.

To test the performance of queries used in real scenarios, three types of text queries are developed for evaluation purpose: one-, two- and three-word queries, as users seldom issue long queries for music search in reality [21]. This strategy is also often applied in previous text-based music retrieval studies [21, 31]. For the one-word queries in each dataset, the most frequently used words are used as candidates. For the two- and three-word queries, the most frequent co-occurrent two and three words in tags are used as candidates, respectively. The *query users* and *user-specific queries* are carefully selected from these candidates to ensure that, for each user, the user-specific test collection contains sufficient relevant songs for his/her queries (for the fair comparisons of different retrieval methods) [21]. The query words cover the commonly used music concepts, such as *genre*, *instrument*, *mood*, and *era*. Table 3 shows the query examples used in the experiments.

Since the average number of songs listened by users in two datasets are very different, different numbers of users and queries can be generated in two datasets. The details about users and queries in both datasets are as below.

- **Lastfm-1K**: In this dataset, 124 users are selected as query users, and 96 different queries (30 one-word queries, 30 two-word queries, and 36 three-word queries) are selected. The selected queries are the same for all the users. The number of songs in this test collection of each user is at least 500. In total, there are 11,904 user-specific queries used in this dataset.
- **TPS**: Because the number of songs listened by users in this dataset is much smaller, few user-specific queries can be applied in order to make sure that there are enough positive songs (for each query) in the user-specific test collections. Finally, we select 20 users and 20 queries (8 one-word queries, 6 two-word queries, and 6 three-word queries) per user. Similarly, the queries of all the users are the same. The least number of songs in the test collection for each user is set to be 100. In total, there are 400 user-specific queries used in this dataset.

4.2 Experimental Setup

This section introduces the details about competitors, evaluation metrics and system parameters. To verify the effectiveness of the proposed personalized text-based music retrieval system, we compare it with popular and state-of-the-art text-based music retrieval methods, as well as the existing personalized music retrieval methods:

- **Text-based Music Retrieval (TMR)**: Based on the song-text documents, the standard tf-idf weighting scheme is used to compute the similarity between query and songs with the standard cosine distance in Vector Space Model (VSM) [26].
- **Weighted Linear Combination (WLC)**: Similar to the WLC described in [22], the first result returned by TMR is used as the seed for a content-based music retrieval (CBMR) method. Then the score of the TMR method and the CBMR method are linearly combined together to generate the final search results. In our experiment, the CBMR method described in [25] is used. Specifically, the "audio words" are treated as text terms, and then the standard VSM method is used to retrieve the music by using the seed song as query. The combination weights are carefully tuned to achieve the highest MAP in experiments.
- **Post-Hoc Audio-Based Reranking (PAR)** [16]: It was originally proposed to improve a text-based search engine that indexes songs based on related Web documents. Briefly, for each song s, the PAR approach computes a new score that combines the text-based rank of s, the text-based rankings of all the songs having s in their neighbourhoods, and the rank of s in all these neighbourhoods. The songs are then sorted according to this new score. In our implementation, the top 100 songs are re-ranked by the TMR method.
- **Personalized Retrieval Model (PRM) [11]**: To the best of our knowledge, this is the only scheme specially proposed for personalized text-based music retrieval in previous literature. It is an simple extension of LDA. The graphical representation of this model is illustrated in Fig. 2. This topic model captures the term associations based on their co-occurrences in the

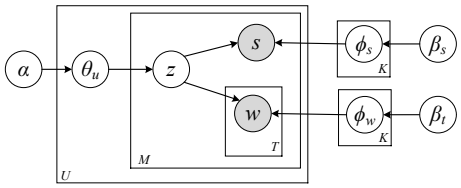

Figure 2: Graphical representation for PRM.

same song and the song associations based on their co-occurrences in the same user's profile under the same latent space. Each user's music preference is modeled as a multinomial distribution over a set of topics; and each topic has a distribution over the set of songs and terms. This model does not take the music contents into account.

Evaluation Metrics In information retrieval, users are more interested in results in the top positions. Therefore, we focus on the evaluation of top results in terms of accuracy. Several standard information retrieval metrics are used, including precision at k (Precision@k), Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain at k (NDCG@k) [13]. The relevance levels (i.e., 0, 1, and 2) are used to compute NDCG. For Precision@k and MAP, both relevant (i.e., 1) and highly-relevant (i.e., 2) results are regarded as positive results.

Parameter Setting In our implementation, the Dirichlet hyper-parameters of both topic models (DL-MPTM and PRM) are empirically set: $\alpha = 1.0$, $\gamma = 1.0$, $\beta_s = \beta_t = \beta_v = 0.01$. We carefully tune the latent topic numbers in both topic models. In DL-MPTM, the number of latent music dimension is tuned in $\{5, 10, 20, 30, 40, 50, 60\}$ and the number of latent sub-topics is tuned in {20, 40, 60, 80, 100, 150}. The number of latent topics in PRM is tuned in {20, 40, 60, 80, 100, 150}. Besides, the combination weight w in WLC retrieval methods are both tuned from 0 to 1 in steps of 0.1.

5. EXPERIMENTAL RESULTS

This section reports the experimental results of our methods and other competitors on retrieval performance. The reported results of DL-MPTM and PRM are based on the optimal numbers of latent topics in each dataset. The reported results based on MAP and NDCG in all the tables are truncated at 10, namely, MAP@10 and NDCG@10. The symbol (*) after a numeric value denotes significant differences ($p < 0.05$, a two-tailed paired t-test) with the corresponding second best measurement. All the results presented in below are the average values of queries over all the users.

5.1 Retrieval Performance

Effectiveness Table 4 reports the retrieval performance on different queries composed of one, two and three words on the two datasets. As can be seen, the proposed model outperforms all the other algorithms over both datasets. Larger performance gain can be achieved when considering more results in the top positions; in particular, the improvements in P@10, MAP and NDCG are statistically significantly compared to other algorithms. After comparing the results gained using TMR, WLC, and PAR, it is easy to find that for Lastfm-1K dataset, the consideration of audio features with text can improve the search results. Besides, PAR obtains better results than WLC does. However, in TPS dataset, the performance decreases when using acoustic features for re-ranking. It is worth noticing that the results of WLC in TPS are the same to TMR. It is mainly because the best performance of WLC can be achieved by only using text feature in WLC. Notice that the search datasets of query users in Lastfm-1K is much larger than that of users in TPS (see Sect. 4.1.1). Generally, it is difficult for the content-based method to achieve better search results over a smaller dataset, because finding songs with similar contents in smaller datasets is harder. Thus the audio feature does not work well when being applied to support search over the TPS dataset.

On the other hand, PRM and DL-MPTM achieve much better performance than the other three methods (TMR, PAR, and WLC), which have not taken the personal music preferences into account. It demonstrates the importance of user's music preference in facilitating effective music retrieval. DL-MPTM's performance improvement over PRM on both datasets demonstrates the effectiveness of our proposed dual-layers topic model in capturing the correlation of user, song, and term. In PRM, the correlation is modeled using the same latent space, which is discovered based on both the co-occurrence of songs in playlists and the co-occurrence contents of songs. In DL-MPTM, the correlation is captured by two layers of connected latent spaces: the low-level latent space (constructed by *latent semantic subtopics*) is discovered based on the co-occurrence contents of songs, the high-level latent space (constructed *latent music dimensions*) is discovered based on the co-occurrence of songs in playlists and the co-occurrence of latent subtopics across songs.

Table 5 compares the performances of TMR, PAR, and DL-MPTM based on the top five search results in the ranking lists of one representative query in each type. The relevance level of each song in the top five positions is also shown. The results demonstrate that DL-MPTM achieves much better performance in task of searching user preferred songs with respect to the queries, comparing to TMR and PAR methods. For example, in response to the query "*guitar, pop*", DL-MPTM places three high-relevant songs at the top rank, compared with only one ranked by the TMR model at the 5th position and two ranked at the 3th and 4th positions by the PAR model.

Robustness By comparing the results of different query types (one-, two- and three-word queries), we can observe that the search performance is slightly decreased when the query complexity increases. For different types of queries, DL-MPTM achieves significant and consistent improvement over all metrics, showing a superior robustness across multi-word queries.

Music is usually described by different categories of music concepts, such as *mood*, *instrument*, *genre*, and *vocals*, which have been widely studied in music retrieval related research, such as classification and annotation. We examine the search performance of our method over other methods on different categories of music concepts. Table 6 presents evaluation results. One-word queries are classified into different music concept categories as shown in the table. We focus on the one-word query, since the two- and three-word queries could be the combination of different categories. The category "Other" contains queries, such as "*driving*", "*slow*", "*sexy*", which cannot be classified into other four categories.

Table 4: Retrieval performance for 1-word, 2-word, and 3-word queries

Dataset	Model	1-Word Query					2-Word Query					3-Word Query				
		P@3	P@5	P@10	MAP	NDCG	P@3	P@5	P@10	MAP	NDCG	P@3	P@5	P@10	MAP	NDCG
Lastfm-1K	TMR	.669	.671	.673	.668	.481	.664	.657	.664	.660	.483	.643	.647	.656	.649	.482
	WLC	.662	.654	.658	.669	.495	.652	.646	.653	.658	.489	.675	.669	.669	.650	.486
	PAR	.741	.735	.715	.728	.548	.734	.724	.713	.720	.541	.714	.710	.696	.704	.551
	PRM	.840	.818	.792	.824	.611	.839	.823	.795	.826	.602	.827	.814	.789	.819	.611
	DL-MPTM	**.842**	**.851***	**.858***	**.852**	**.663***	**.839**	**.845**	**.853**	**.846**	**.663***	**.846**	**.848***	**.848***	**.851***	**.664***
TPS	TMR	.550	.600	.557	.558	.434	.533	.575	.565	.564	.425	.490	.526	.533	.516	.412
	WLC	.550	.600	.557	.558	.434	.533	.575	.563	.565	.424	.490	.526	.531	.516	.411
	PAR	.492	.470	.455	.477	.365	.483	.495	.450	.468	.362	.486	.451	.443	.469	.363
	PRM	.648	.609	.583	.621	.492	.633	.565	.545	.595	.479	.567	.555	.557	.574	.486
	DL-MPTM	**.667***	**.640***	**.633***	**.645**	**.520***	**.658***	**.635***	**.629***	**.639***	**.534***	**.655***	**.625***	**.613***	**.619***	**.517***

Table 5: The top 5 songs in the ranking lists obtained by the TMR, PAR, and DL-MPTM models for 3 representative queries of a user "*user_000477*". The relevance level of each result is shown in the parentheses after each result, e.g., "(2)" indicates high relevance (see Sect. 4.1.1).

TMR	PAR	DL-MPTM
	metal	
System of a Down - thetawaves (1)	Rage Against the Machine - Bullet in the head (1)	**Rammstein - Du hast** (2)
System of a Down - I-E-A-I-A-I-O (1)	**Linkin Park - Valentine's day** (2)	A Perfect Circle - Over (1)
Linkin Park - Valentine's day (2)	Audioslave - Set it off (1)	**Nirvana - Smells like teen spirit** (2)
Korn - Did my time (1)	Goldfrapp - Cologne cerrone houdini (1)	Muse - Hysteria (1)
Metallica - Nothing Else Matters (1)	**Incubus - Anna molly** (2)	**AC/DC - Back In Black** (2)
	guitar, pop	
Dread Zeppelin - Misty mountain hop (0)	New Order - crystal (0)	**Oasis - Wonderwall** (2)
Dire Straits - Sultans of swing (1)	Dire Straits - Romeo and juliet (1)	**The Smashing Pumpkins - 1979** (2)
Dire Straits - Money for nothing (1)	**Linkin Park - Valentine's day** (2)	**The Cranberries - Zombie** (2)
Dread Zeppelin - Your time is gonna come (0)	**The Smashing Pumpkins - 1979** (2)	Blur - Song 2 (1)
Dire Straits - Brothers in arms (2)	Red Hot Chili Peppers - Mellowship slinky in b major (1)	Oasis - Live forever (1)
	guitar, rock, vocalists	
Lez Zeppelin - Communication breakdown (0)	The Smiths - Stretch out and wait (0)	**AC/DC - Back in black** (2)
Dread Zeppelin - Misty mountain hop (0)	New Order - Crystal (0)	**AC/DC - Highway to hell** (2)
Lez Zeppelin - Whole lotta love (0)	Interpol - The heinrich maneuver (0)	Dread Zeppelin - Heartbreaker (0)
Dire Straits - Sultans Of Swing (1)	Klaxons - Two receivers (0)	**The Cranberries - Zombie** (2)
Dire Straits - Money for nothing (1)	Dire Straits - Romeo and juliet (1)	**AC/DC - Hells bells** (2)

Table 6: Retrieval results for query categories. The best results for each category are indicated in bold.

Category	Modal	P@10	MAP	NDCG
Emotion	TMR	.684	.677	.487
	WLC	.659	.672	.498
	PAR	.721	.732	.550
	PRM	.790	.822	.606
	DL-MPTM	**.863**	**.858**	**.668**
Genre	TMR	.649	.639	.463
	WLC	.646	.644	.474
	PAR	.701	.718	.542
	PRM	.793	.831	.616
	DL-MPTM	**.852**	**.836**	**.647**
Instrument	TMR	.673	.665	.474
	WLC	.651	.669	.491
	PAR	.724	.735	.552
	PRM	.803	.835	.620
	DL-MPTM	**.871**	**.862**	**.673**
Vocals	TMR	.657	.651	.473
	WLC	.665	.672	.488
	PAR	.717	.720	.551
	PRM	.796	.830	.623
	DL-MPTM	**.842**	**.841**	**.648**
Others	TMR	.685	.687	.497
	WLC	.671	.683	.518
	PAR	.711	.729	.547
	PRM	.787	.808	.603
	DL-MPTM	**.850**	**.847**	**.666**

The significant improvements over other methods on P@10, MAP and NDCG show the effectiveness and robustness of DL-MPTM over different music concept categories.

Comparing the search performances of all the methods on the two datasets, they cannot achieve good performance when searching over the TPS dataset, because of the limited size of relevant results in each user's specific dataset. Notice that the number of training samples in the TPS dataset is also much smaller than that in the Last.fm-1K dataset. On the TPS dataset, the absolute performance gain achieved by DL-MPTM over other methods for all the metrics are at least comparable to those in the Lastfm-1K dataset. This demonstrates strong robustness of DL-MPTM on relatively small training datasets.

5.2 Effects of Parameters

In this section, we investigate the effects of parameters on the proposed retrieval method. In topic models, it is hard to accurately pre-define the number of topics, which has important effects on the results. In the DL-MPTM model, there are two sets of latent topics: the number of latent music dimensions in the first layer, and the number of subtopics in the second layer. Fig. 3a and Fig. 3b illustrate the effects of the two parameters, respectively. From the results, it can be observed that the number of latent music dimensions has strong impacts on the final performance, and it is optimal to set the number of music dimensions to [5, 20]. In contrast, we can observe the minor effects of sub-topic number, especially for Lastfm-1K dataset. Fig. 3c shows the effects of weights in the combination of acoustic similarity and textual similarity. From the results, we can find that for Lastfm-1K, the combination of acoustic similarity and textual similarity can slightly improve the performance when w is set to [0.6, 0.8]. However, the performance degradation is observed when the same similarity combination is applied to the TPS dataset. Notice that in the WLC method, the acoustic similarity is computed based on the first search result of TMR. Thus, the accuracy of the first search result has an important impact on the WLC performance. When the search accuracy of TMR is relatively high, WLC can improve the TMR performance further, such as the results observed in the Lastfm-1K dataset. This also suggests that when searching music using both text and audio query examples (using relevance feedback), performance could be improved by the combination of acoustic similarity and text similarity.

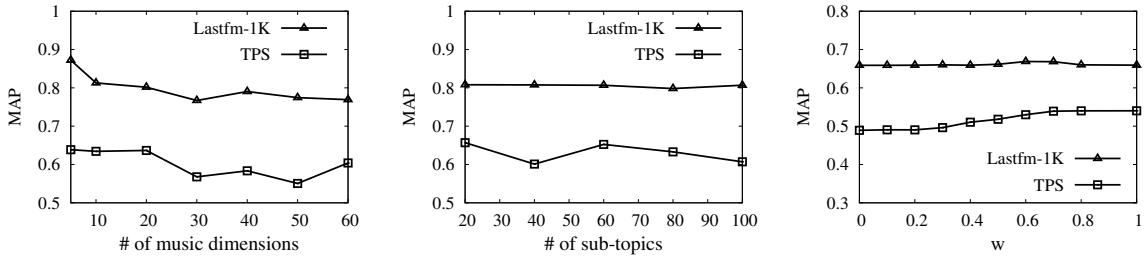

Figure 3: Effects of parameters in topic model based retrieval methods.

6. CONCLUSION

In this paper, we present a personalized text-based music retrieval system which exploits the user listening behaviors in social music services. The system can accurately estimate the relevance of a song with respect to a term subject to user's music preference. To achieve the goal, a Dual-Layer Music Preference Topic Model (DL-MPTM) topic model is proposed to leverage the user listening logs and social tags to learn the interactions among (user, song, term), which are applied for personalized text-based music search. To evaluate the performance of the personalized retrieval system, comprehensive experiments have been conducted on two public datasets. The comparisons with the state-of-the-art text-based retrieval methods and existing personalized music retrieval methods in experiments show that our method can significantly improve the search performance in terms of accuracy. The results also demonstrate the importance of effective integration of personal music preference in developing high performance music search engine, and verify the effectiveness of our proposed retrieval model.

7. REFERENCES

[1] K. Barnard, P. Duygulu, D. Forsyth, N. D. Freitas, D. Blei, and M. Jordan. Matching words and pictures. *The Journal of Machine Learning Research*, 3:1107–1135, 2003.

[2] V. Batagelj and M. Zaveršnik. Generalized cores. *arXiv preprint cs/0202039*, 2002.

[3] D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. *Proc. of NIPS*, 16:17, 2004.

[4] D. Blei and M. Jordan. Modeling annotated data. In *Proc. of ACM SIGIR*, 2003.

[5] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[6] F. Cai, S. Liang, and M. de Rijke. Personalized document re-ranking based on bayesian probabilistic matrix factorization. In *Proc. of ACM SIGIR*, 2014.

[7] O. Celma. *Music Recommendation and Discovery in the Long Tail*. Springer, 2010.

[8] Z. Cheng and J. Shen. Just-for-me: An adaptive personalization system for location-aware social music recommendation. In *Proc. of ACM ICMR*, 2014.

[9] Z. Cheng and J. Shen. On effective location-aware music recommendation. *ACM Trans. Inf. Syst.*, 34(2):13, 2016.

[10] T. Griffiths and M. Steyvers. Finding scientific topics. *PNAS*, 101(Suppl 1):5228–5235, 2004.

[11] N. Hariri, B. Mobasher, and R. Burke. Personalized text-based music retrieval. In *Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[12] K. Hoashi, K. Matsumoto, and N. Inoue. Personalization of user profiles for content-based music retrieval based on relevance feedback. In *Proc. of ACM MM*, 2003.

[13] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, 2002.

[14] Y. Jia, M. Salzmann, and T. Darrell. Learning cross-modality similarity for multinomial data. In *Proc. of IEEE ICCV*, 2011.

[15] D. Kim, G. Voelker, and L. Saul. A variational approximation for topic modeling of hierarchical corpora. In *Proc. of ICML*, 2013.

[16] P. Knees, T. Pohle, M. Schedl, D. Schnitzer, K. Seyerlehner, and G. Widmer. Augmenting text-based music retrieval with audio similarity. In *Proc. of ISMIR*, 2009.

[17] M. Levy and M. Sandler. Music information retrieval using social tags and audio. *IEEE Trans. Multimed.*, 11(3):383–395, 2009.

[18] B. Logan. Mel frequency cepstral coefficients for music modeling. In *Proc. of ISMIR*, 2000.

[19] A. Majumder and N. Shrivastava. Know your personalization: learning topic level personalization in online services. In *Proc. of WWW*, 2013.

[20] B. McFee, T. Bertin-Mahieux, D. Ellis, and G. Lanckriet. The million song dataset challenge. In *Proc. of WWW*, 2012.

[21] R. Miotto and G. Lanckriet. A generative context model for semantic music annotation and retrieval. *IEEE Trans. Audio, Speech, and Language Process.*, 20(4):1096–1108, 2012.

[22] R. Miotto and N. Orio. A probabilistic model to combine tags and acoustic similarity for music retrieval. *ACM Trans. Inf. Syst.*, 30(2):8, 2012.

[23] J. Paisley, C. Wang, D. M. Blei, and M. I. Jordan. Nested hierarchical dirichlet processes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(2):256–270, 2015.

[24] D. Putthividhy, H. Attias, and S. Nagarajan. Topic regression multi-modal latent dirichlet allocation for image annotation. In *Proc. of IEEE CVPR*, 2010.

[25] M. Riley, E. Heinen, and J. Ghosh. A text retrieval approach to content-based audio retrieval. In *Proc. of ISMIR*, 2008.

[26] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[27] M. Schedl and A. Flexer. Putting the user in the center of music information retrieval. In *Proc. of ISMIR*, 2012.

[28] J. Shen, H. Pang, M. Wang, and S. Yan. Modeling concept dynamics for large scale music search. In *Proc. of ACM SIGIR*, 2012.

[29] A. Singla, R. White, A. Hassan, and E. Horvitz. Enhancing personalization via search activity attribution. In *Proc. of ACM SIGIR*, 2014.

[30] P. Symeonidis, M. Ruxanda, A. Nanopoulos, and Y. Manolopoulos. Ternary semantic analysis of social tags for personalized music recommendation. In *Proc. of ISMIR*, 2008.

[31] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Towards musical query-by-semantic-description using the cal500 data set. In *Proc. of ACM SIGIR*, 2007.

[32] S. Virtanen, Y. Jia, A. Klami, and T. Darrell. Factorized multi-modal topic model. In *Proc. of UAI*, 2012.

[33] A. Wang. The shazam music recognition service. *Communications of the ACM*, 49(8):44–48, 2006.

[34] J. Wang, Y. Shih, M. Wu, H. Wang, and S. Jeng. Colorizing tags in tag cloud: a novel query-by-tag music search system. In *Proc. of ACM MM*, 2011.