On Effective Location-Aware Music Recommendation

ZHIYONG CHENG and JIALIE SHEN, Singapore Management University, Singapore

Rapid advances in mobile devices and cloud-based music service now allow consumers to enjoy music anytime and anywhere. Consequently, there has been an increasing demand in studying intelligent techniques to facilitate context-aware music recommendation. However, one important context that is generally overlooked is user's venue, which often includes surrounding atmosphere, correlates with activities, and greatly influences the user's music preferences. In this article, we present a novel venue-aware music recommender system called VenueMusic to effectively identify suitable songs for various types of popular venues in our daily lives. Toward this goal, a Location-aware Topic Model (LTM) is proposed to (i) mine the common features of songs that are suitable for a venue type in a latent semantic space and (ii) represent songs and venue types in the shared latent space, in which songs and venue types can be directly matched. It is worth mentioning that to discover meaningful latent topics with the LTM, a Music Concept Sequence Generation (MCSG) scheme is designed to extract effective semantic representations for songs. An extensive experimental study based on two large music test collections demonstrates the effectiveness of the proposed topic model and MCSG scheme. The comparisons with state-of-the-art music recommender systems demonstrate the superior performance of VenueMusic system on recommendation accuracy by associating venue and music contents using a latent semantic space. This work is a pioneering study on the development of a venue-aware music recommender system. The results show the importance of considering the influence of venue types in the development of context-aware music recommender systems.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Query Formulation, Search Process; H.5.5 [**Sound and Music Computing**]: Systems

General Terms: Algorithms, Design, Experimentation, Human Factors

Additional Key Words and Phrases: Venue-aware, music recommendation, music concept, topic model

ACM Reference Format:
Zhiyong Cheng and Jialie Shen. 2016. On effective location-aware music recommendation. ACM Trans. Inf. Syst. 34, 2, Article 13 (April 2016), 32 pages.
DOI: http://dx.doi.org/10.1145/2846092

1. INTRODUCTION

Music plays an important role in our daily lives. In recent years, rapid advances in mobile devices and cloud-based music services like Pandora and Spotify have brought about a fundamental change in the way people consume music. Mobile devices become mainstream platforms allowing people to enjoy favorite music anytime and anywhere: More than one in ten American adults now owns an iPod or MP3 players. Rich online music resources enable us to gain instant on-demand access to millions of songs. Moreover, the increasingly fast growth of music service has raised countless demands

This work is supported by Singapore Ministry of Education under Academic Research Fund Tier-2 (MOE Ref: MOE2013-T2-2-156).
Authors' addresses: Z. Cheng and J. Shen (corresponding author), School of Information Systems, Singapore Management University, 80 Stamford Road, Singapore 178902; emails: zy.cheng.2011@phdis.smu.edu.sg, jlshen@smu.edu.sg.

DOI: http://dx.doi.org/10.1145/2846092

for advanced information retrieval techniques to facilitate effective music search and management.

Intelligent recommender systems, as a promising technology for music search, aim to assist users in exploring large-scale music collections by identifying *suitable* songs based on their preferences [Shen et al. 2013]. Users generally prefer music players that can automatically recommend the playlists fitting their preferences based on current contexts (e.g., mood, location, event, and activity). In North et al. [2004], a user study about the daily music usage pattern found that local environments can significantly affect users' listening behaviors. Consequently, Lee et al. [Lee and Downie 2004] suggest that it is essential to take contextual information into account when designing modern music recommender systems. Indeed, a wide range of contextual information has been recently explored in music recommender system development [Kaminskas and Ricci 2012]. These contexts include both environment-related (e.g., location and time) [Braunhofer et al. 2013; Baltrunas et al. 2011; Cheng and Shen 2014; Schedl et al. 2014; Schedl and Schnitzer 2014] and user-related contexts (e.g., activity and emotion) [Cai et al. 2007; Wang et al. 2012]. The studies have demonstrated that the incorporation of contexts in recommendation can effectively enhance the user's satisfaction on recommendation results. As a matter of fact, location is one of the most crucial contexts and has significant influence on a user's music preference [Braunhofer et al. 2013; North et al. 2004]. Several previous studies attempted to recommend music to specific geo-locations [Reddy and Mascia 2006; Ankolekar and Sandholm 2011]. In addition, Baltrunas et al. [2011] built an in-car music player for recommending music suitable to the landscapes passed when driving a car. Kaminskas et al. conducted a series of studies on retrieving songs suited for Places of Interest (POI) based on emotional tags [Braunhofer et al. 2013; Kaminskas and Ricci 2011; Kaminskas et al. 2013]. However, one important context that is generally ignored in current research is user's venue. To the best of our knowledge, no existing approaches can effectively recommend music based on common venues, such as *office*, *library*, *gym*, *mall*, and the like.

Venue, referring to *the place where an activity or event happens*, is an important location-based context and becomes more and more important in music recommender system design and development. On the one hand, people usually listen to music in different types of venues in the course of everyday life [North et al. 2004]. On the other hand, people would enjoy different types music at different types of venues, where different surrounding environments and atmospheres can be found. Thus, venue type has important influence on a user's song selections, and suitable songs can be very helpful in creating a nice atmosphere for a particular venue. For example, *night clubs*, *restaurants,* and *shops* often use music to help them create the right atmosphere for their customers. Furthermore, users' activities, which also play a critical role in determining users' song preferences [Wang et al. 2012; North et al. 2004; Levitin and McGill 2007], highly correlate with venue type. In fact, when users are engaging in the same or similar activity, the songs they prefer or play share many common musical characteristics [Wang et al. 2012]. For example, low tempo and middle-pitch-range music is usually selected to assist users in concentrating or thinking, whereas up-tempo music is a natural choice for physical exercise in the gym. This study mainly focuses on the effects of venue types instead of geo-locations (a geo-location refers to a point pinpointed by geographic coordinate) because users' music preferences are more likely to be influenced by the atmosphere and environment of venue types. For examples, a user would prefer similar types of music when he is working out whether the gym is near his office or his home, although these gyms have different geo-locations. In addition, when conducting different activities in a venue, users often like the same type of music, such as when reading and writing in a library. To support efficient music access, listeners frequently organize songs into different playlists that are suitable

for various venue types. For example, in a popular music streaming service website Grooveshark[1], venue types are very common titles of user playlists. It often happens that the same song appears in many different playlists named for the same venue type but created by different users (refer to Section 4.1.2). This observation suggests that users share similar understandings and views about the musical content suitable for a particular venue type. To simplify presentation, unless otherwise indicated, *venue* in this article refers to *venue type* hereafter.

Motivated by these earlier discussions, we study the problem of recommending suitable songs for different types of venues by exploring the correlation between the music *features* and the *characteristics* of these venues. In general, a venue owns distinct *characteristics*, such as ambience and atmosphere. Songs with certain *features* that fit those *characteristics* could be more suitable for this particular venue, such as energetic music for a gym and peaceful music for a library. According to one study [Lamont and Greasley 2009], users tend to label the pieces of music they like using high-level concepts, such as styles and emotions. This reveals that a human perceives and judges music based on the semantics embedded in musical contents. In many cases, music semantic meaning cannot be explicitly described and characterized using low-level spectral features due to the well-known "semantic gap" [Zhang et al. 2009a]. Acoustic content belonging to same or similar concepts could be highly diverse. Furthermore, a song could include a complex mixture of concepts at different levels. Therefore, the utilization of acoustic features or concepts for describing music preferences at a venue may not be effective and comprehensive enough to support high-quality recommendation.

In this article, we propose a smart music recommender system called VenueMusic, which can automatically generate a playlist matching a target venue appropriately [Cheng and Shen 2015]. Toward this goal, we approach the problem from a new perspective of effective topic modeling and develop a novel scheme called Location-aware Topic Model (LTM), which models the associations between the music content and venues in a *latent semantic space*. Similar to the standard Latent Dirichlet Allocation (LDA) [Blei et al. 2003], in the LTM, each topic is a multinomial distribution of music semantic concepts that captures the interactions between various music semantics. Each venue and each song is then represented by the multinomial distributions of these latent topics. Intuitively, the topic distribution of a venue characterizes the *relevant music properties* of songs that are suitable for this venue, and the topic distribution of a song reflects how general users perceive the music. Because both songs and venues are represented by the same latent topics, the suitability of a song for a venue can be directly measured. The LTM is trained based on a set of songs labeled with different venues. To enable the LTM to characterize the semantic meaning of a song, each song is represented as a "bag-of-words" document. This is different from existing methods [Yoshii et al. 2008; Cheng and Shen 2014] based on "bag-of-audio-words," which can not effectively express the semantic meanings of a song. In the VenueMusic system, each song is represented as a sequence of *music concepts*,[2] (i.e., a "bag-of-text-word" document). In particular, a Music Concept Sequence Generation (MCSG) method (Section 3.2) is proposed to generate the concept sequence of a song. As shown in Figure 1, each song is partitioned into small segments and then music concepts are extracted from each segment by learned concept detectors based on the acoustic features of the segment. With a concept filtering process to improve the detected quality (Section 3.2.2), the concepts of all segments in a song are concatenated to form the concept sequence of this song. As validated in our

[1]http://grooveshark.com.

[2]A music concept could be one or several text words that are usually used to describe music, such as *genre* and *mood* words.

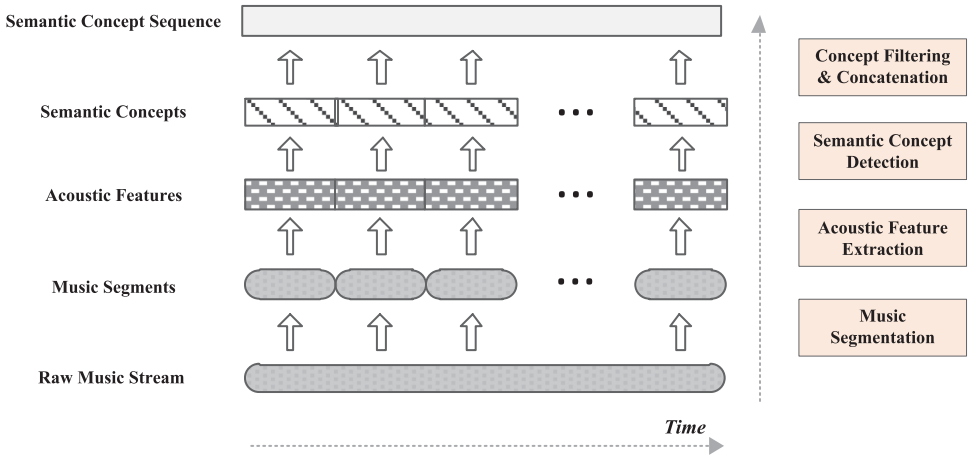


Fig. 1. The Music Concept Sequence Generation (MCSG) scheme.

experiments, song representation based on semantic concept sequences in the LTM is more effective than those using low-level "audio words."

To the best of our knowledge, no similar approach has been reported previously in the literature. Our main contributions can be summarized as follows:

—We propose a location-aware music recommender system that recommends music to match different types of common venues in everyday life. The system matches songs and venues based on their semantic features. This is the first attempt to develop venue-aware music recommendation methods.
—We propose a novel topic model, LTM, to capture the natural connections between the venue semantics and the music *content*. The latent semantic topics extracted by the LTM are used to characterize the *music features preferred in different venue types* as well as *the music features of songs*. With this approach, the suitability of a song to a venue can be quantitatively measured in a latent semantic space.
—We propose a *semantic concept sequence generation* scheme to represent a song as a set of concepts for topic modeling. In addition, an *infrequent concept pattern filtering method* is introduced to remove noisy concepts in the generated semantic concept sequence. The final semantic concept sequences of songs are effective in the training of LTM.
—We develop two large-scale music test collections and carefully design a set of experiments to evaluate and compare the performance of our system with a set of competitors over a wide range of venues. The core empirical results demonstrate the potential of our VenueMusic system developed in this study.

The rest of the article is organized as follows. Section 2 reviews the related work. The framework of the music recommender system is presented in Section 3. Section 3.3 introduces the LTM and provides details about algorithms for the model parameter inference. Section 4 describes the experimental configurations. The evaluation results are presented and analyzed in Section 5. Finally, Section 6 concludes the article with a discussion of the findings in this study and directions for future research.

2. RELATED WORK

In this section, we first give a general introduction on the techniques used in music recommendation and review recent developments in the domain of location-aware music

recommendation. Then, we review related work in topic models and analyze their limitations on modeling musical semantics related to location context.

2.1. Music Recommender Systems

The techniques used in music recommender systems can be broadly categorized into collaborative-based, content-based, social-based, and hybrid approaches [Adomavicius and Tuzhilin 2005; Kaminskas and Ricci 2012]. Collaborative-based methods [Schafer et al. 2007] estimate the similarity between users based on their listening records and recommend songs by referencing to the preferences of similar users. Content-based methods [Pazzani and Billsus 2007] compute the similarity between songs based on the music content or associated textual descriptions and recommend songs that are similar to those user liked in the past. Social-based methods [Kaminskas and Ricci 2012] rely on web mining techniques or social tags to compute similarity between the items to be recommended. Hybrid methods [Burke 2007] combine the techniques from the three basic approaches.

Recommending songs for particular locations can be viewed as a context-aware music recommendation problem, which is emerging as a promising research topic in recent years [Braunhofer et al. 2013]. Context-Aware Music Recommender Systems (CAMRs) aim to satisfy users' music needs by exploring local contexts. Many CAMRs have been developed to take various contexts into account, including environment-related (such as location and time [Baltrunas et al. 2011; Braunhofer et al. 2013]) and user-related context (such as activity and emotional state [Cai et al. 2007; Wang et al. 2012]), however, studies on exploiting location-related context information are still sparse. Gaye et al. [2003] designed a prototype of an interactive music system to generate electronic music for urban environments. The system heavily relies on hardware to collect various user-related (e.g., heart rate, arm motion) and environment-related contextual information (e.g., light, temperature, and noise, etc.). Lifetrak [Reddy and Mascia 2006] considers the location (represented by a ZIP code), time, weather, and activities to generate a playlist based on a user's music library. A mobile audio application, Foxtrot [Ankolekar and Sandholm 2011], allows a user to assign audio content to a specific geo-location, and play audio content associated with that particular location. Kaminskas et al. [Braunhofer et al. 2013; Kaminskas et al. 2013] conducted a series of studies on recommending music to POIs. They match the POIs and music by exploiting semantic relations between the POIs and music items using assigned emotional tags for both POIs and songs. Most of these studies relate location information with geographical coordinates. However, it is hard to capture correlations between music content and a specific geo-location. As a result, for a location, these systems can only recommend the songs liked by users in this location based on previous records [Reddy and Mascia 2006; Ankolekar and Sandholm 2011]. It is worth to mention that POIs in Kaminskas et al. [Braunhofer et al. 2013; Kaminskas et al. 2013] are places people do not visit frequently in everyday life.

In this study, the location contexts we consider are various types of popular venues where people often visit and enjoy music in everyday life, such as the library and the gym. Going beyond the geo-location information of latitude and longitude, each venue possesses its own distinguishing atmosphere or semantics. GeoShuffle [Miller et al. 2010] also considers the effects of the locations where users usually listen to music in their daily lives. The key difference is that in GeoShuffle, the location is captured based on GPS data, and the locations considered are restricted to the points in people's daily routines. Listening records are used to capture a user's music listening habits while in these routine paths. Therefore, its performance depends on both the regularity of user's daily routines and the quality of historical preference data. In our recent work [Cheng and Shen 2014], a Just-for-Me music recommender system was developed for effective

personalized music recommendation in different types of venues, together with the consideration of global music popularity trends. Just-for-Me applies an extended three-way aspect model and represents each song as a "bag-of-audio-words" document to learn the topics. In the extended three-way aspect model, users' music interests are represented as topic distributions, and topics are the distributions of songs, venues, and audio words. Inspired by the key research findings about the strong influence of venue type on users' music preference, in this study, we focus on the problem of recommending suitable songs based on different types of venues. The core innovation of our proposed VenueMusic system is an LTM that naturally associates venue types and music content in a latent semantic space using "bag-of-words"-based representation.

2.2. Topic Model

Topic modeling algorithms [Blei 2012] have been widely applied for discovering "latent topics" underlying a large set of documents. Each topic is a multinomial distribution over terms, and each document is in turn represented by a highly biased multinomial distribution of topics (biased to few topics). In a topic model, the latent topics provide an interpretable low-dimensional representations for the documents. Several previous studies adapted topic models for music search and recommendations [Cheng and Shen 2014; Hariri et al. 2013; Yoshii et al. 2008]. Early on, most topic models, such as Latent Dirichlet Allocation (LDA) [Blei et al. 2003], were unsupervised, aiming to maximize the likelihood of the document collection. In recent years, supervised variants of LDA have been proposed to discover latent topics that distinguish documents from different categories, such as supervised LDA [Mcauliffe and Blei 2008], DiscLDA [Lacoste-Julien et al. 2009] and MedLDA [Zhu et al. 2009]. These methods are suitable for rating prediction (e.g., to predict a movie rating [Mcauliffe and Blei 2008]) or for the classification problem that classifies a document into a certain category.

In Labeled LDA [Ramage et al. 2009], the terms in a document are directly assigned to the labels of the document, which indicates that the latent topics of a document are limited to its labels. The Author-Topic Model (ATM) [Steyvers et al. 2004] uses a topic-based representation to model both the contents of documents and the interests of authors. However, this model only focuses on the interests of authors; it cannot obtain document-specific topic-mixture proportions. To use ATM in location-aware music recommendation, a location is treated as an "author," and all the songs labeled with the location are generated based on the topic distribution of the location. There are two limitations to the method in location-aware music recommendation: (i) the model cannot capture the distinct characteristics of individual songs in a location because these songs are all generated from the same topic distribution, and (ii) for good performance, the ATM needs large numbers of "authors" to learn the latent topics. Whereas in our context a location refers to a type of venue (e.g., *library*), it is hard to collect enough data for thousands of venue types to learn such a model. Another related topic model is the location-aware topic model [Wang et al. 2007], which is used to explicitly model the relationship between locations and words. This model labels each word in a document with a location, but it cannot generate the topic distribution for a location. It is reasonable to assume that some textual keywords are related to a location, such as Personal Names ("Obama" is more likely related to the United States) or Regional Words ("CCTV" is more likely related to China).[3] However, it is hard to relate a short segment of music (e.g., 1 second) to a certain place. Thus, with different design goals, the topic models just discussed are not suitable for location-aware music recommendation tasks. By contrast, our proposed topic model can effectively discover the topic distributions of both songs and venues. Accordingly, the concepts relevant to venues and songs are mapped into the same latent space and can be directly matched in the space.

[3]Please refer to Table 2 in Wang et al. [2007] for more examples.

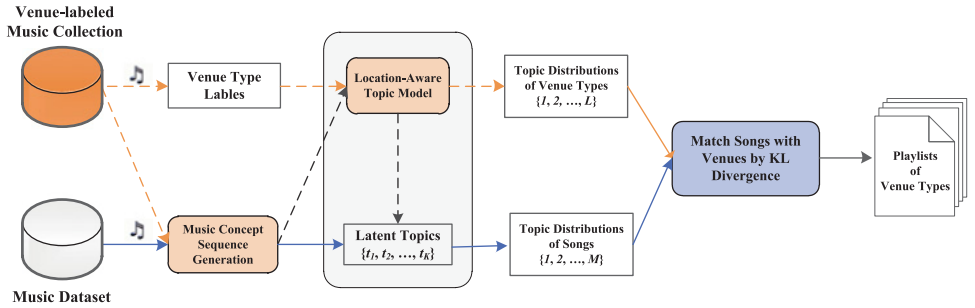


Fig. 2. The framework of VenueMusic System.

3. THE VENUEMUSIC SYSTEM

3.1. System Overview

The VenueMusic system consists of two main functionality modules: the MCSG and the LTM. Figure 2 illustrates details of the system architecture. Given a set of songs labeled based on their suitability to venues (venue-labeled music collection), each song is represented as a Music Concept Sequence (MCS) via the MCSG module. Then the LTM is trained to discover a set of latent topics that form a latent space. Both songs and venues are represented as topic vectors in this latent space. New songs in a music dataset are automatically converted into an MCS in the same way and mapped into the same latent space by the topic model. With the representations of songs and venues in the same space, the relevance (or suitability) of a song with respect to a venue can be directly measured. Since topic vectors are probabilistic distributions of the latent topics, the relevance between a song m and venue type l are evaluated using Kullback-Leibler (KL) distance. Specifically, a song m and a venue type l are both represented by the probabilistic distributions of K topics, and the KL distance is expressed as:

$$KL(l||m) = \sum_{k=1}^{K} l(k) ln \left(\frac{l(k)}{m(k)} \right), \tag{1}$$

where $l(k)$ and $m(k)$ are the probability of k-th topic in the topic distribution of l and m, respectively. The system is designed based on the key observation that a particular venue owns distinct *characteristics* or *atmospheres* that closely associates with the *events* or *activities* occurring in this venue. Typically, different types of music can be applied to match the atmosphere or activities in different venues [Wang et al. 2012; Kaminskas and Ricci 2011; Ricci 2012]. VenueMusic aims to model those rich and complex associations effectively and comprehensively via the LTM.

3.2. Music Concept Sequence Generation

The most straightforward scheme to generate a sequence of "word units" about music content is the "bag-of-audio-words," which has been explored in many studies [Yoshii et al. 2008; Riley et al. 2008; Hu et al. 2014]. However, this approach suffers from a few limitations. First, "audio words" are representative audio frames and thus have no semantic meanings. In the real world, people characterize music content using music semantic concepts (e.g., *mood*, *genre*, *instrument*, etc.) that reflect how humans perceive and interpret acoustic content. It is very difficult to connect the topics generated based on "audio words" with these music concepts. Second, the number of "audio words" is hard to determine. A small number of "audio words" will not be able to represent and distinguish different music content effectively, whereas a large number of "audio words" will lead to sparsity problems and low-efficiency indexing and learning.

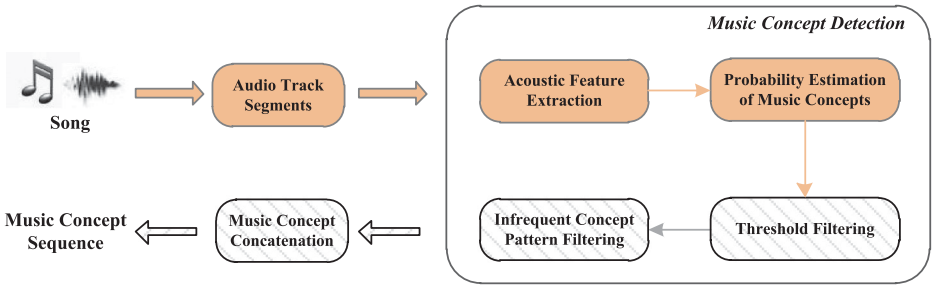


Fig. 3. Architecture of semantic concept sequence generation.

To address the issues of "audio words," we develop a method to extract semantic music concepts (e.g., genre, mood, and instrument) from the audio content to represent a song as an MCS, which is the concatenation of concepts in small segments of the song's audio stream. Alternatively, we can represent each song by assigning music concepts to the whole song. Compared to alternative methods, MCS has at least two advantages. (i) Good comprehensiveness: It contains all the possible music concepts expressed by the audio contents. And (ii) good differentiation: It can differentiate the relatively importance of concepts for a song. For example, in a song, the more segments a concept appears in, the more important or representative this concept is for the song. By aggregating a large set of songs for a venue, the latent associations between the music concepts for this venue can be mined from the MCSs of these songs. The quality of music concept sequence is very important for discovering such latent associations. To improve the concept detection quality, two post-filtering procedures are designed to reduce noisy concepts. As illustrated in Figure 3, MCS generation consists of three main steps:

(1) Partition a song into multiple segments.
(2) Estimate the probability of each music concept in each segment using concept detectors based on the extracted audio features and then filter the concepts to keep the most representative and confident concepts for the segment via two *filtering* methods.
(3) Concatenate the remaining concepts of each segment to form the MCS for this song.

The segments can be obtained by simply cutting the audio stream of a song into fixed-length windows or by detecting segments using music segmentation methods [Lu et al. 2001]. In our implementation, the former method is applied due to its simplicity. Since Steps (1) and (3) are straightforward, we focus on the description of Step (2). There are three key components in Step (2): *Audio Feature Extraction*, *Music Concept Probabilistic Estimation,* and *Concept Filtering*. Figure 4 is a comprehensive illustration of the system architecture of Music Concept Probability Estimation and Concept Filtering.

3.2.1. Audio Feature Extraction. For each segment, we extract four types of acoustic features:

—*Timbral feature:* This characterizes the timbral properties of music sounds. Timbral feature is calculated based on the short-time Fourier transform, including *Mel-Frequency Cepstral Coefficients* (MFCCs) [Logan 2000], *Rolloff*, *Flux*, *Low-Energy feature* [Tzanetakis and Cook 2002], and *Spectral Contrast* [Lu et al. 2006]. The total dimensionality is 23.
—*Spectral feature:* This describes the spectral properties of a music signal. They include *Spectral Centroid*, *Spectral Asymmetry*, *Kurtosis*, *Audio Spectrum Flatness*, *Spectral Crest Factors* [Brown et al. 2001], *Slope*, *Decrease*, *Variation*, *Frequency Derivative of*

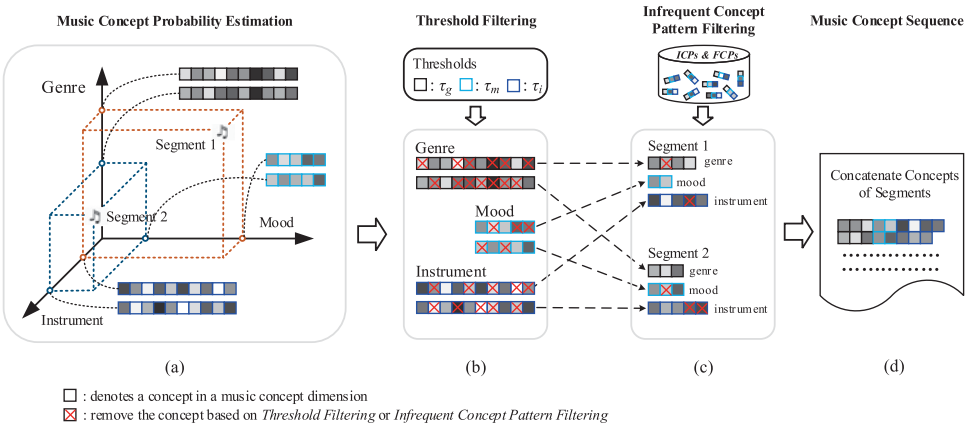


Fig. 4. Illustration of the music concept probability estimation and concept filtering.

Constant-Q Coefficients [Schörkhuber and Klapuri 2010], *and Octave Band Signal Intensities* [Essid et al. 2006]. The total dimensionality is 70.

—*Rhythmic feature:* This represents the patterns of a song over a certain duration. In this study, our rhythm feature includes *Beat Histogram*, *Rhythm Strength*, *Regularity, and Average Tempo* [Lu et al. 2006]. The total dimensionality is 12.

—*Temporal feature:* This characterizes the musical properties based on time domain signals. It includes *Zero Crossing Rate*, *Autocorrelation Coefficients* [Essid et al. 2006], *Waveform Moments* [Essid et al. 2006], and *Amplitude Modulation* [Essid et al. 2006]. The total dimensionality is 62.

Three public toolboxes are used to extract all these acoustic features: MIR Toolbox [Lartillot and Toiviainen 2007], Yaafe [Mathieu et al. 2010], and Essentia [Bogdanov et al. 2013][4].

3.2.2. Music Concept Probability Estimation. Music concept probability estimation aims to estimate the probabilities of various music concepts for a music segment, as illustrated in Figure 4(a). Suppose there are n music dimensions $\{C_1, C_2, \ldots, C_n\}$ (e.g., *genre*, *mood,* and *instrument*) and N_i concepts for each dimension C_i, then the probabilistic vector of a dimension C_i is $C_i = \{P_{i1}, P_{i2}, \ldots, P_{iN_i}\}, (0 \leq P_{ij} \leq 1, 1 \leq j \leq N_i)$, where P_{ij} is the probability that the segment belongs to j-th concept of C_i. Many existing regression and classification methods can be used to estimate P_{ij}. In our implementation, the SVM method in LIBSVM library is adopted for the task [Chang and Lin 2011].

3.2.3. Concept Filtering. Generally, a music segment contains only a limited amount of concepts in a music dimension. For example, it is really rare that music is played using all kinds of instruments. Thus, effective and comprehensive music characterization might not be achieved by using all the concepts. How to select the most representative concepts and remove noisy concepts becomes very important. In VenueMusic, two different strategies are proposed for concept space refinement.

Threshold Filtering aims at removing those concepts with a probability lower than a predefined threshold. Specifically, for each concept dimension C_i, there is a predefined threshold τ_i. If $P_{ij} < \tau_i$, then the j-th concept in C_i is removed, where P_{ij} indicates the

[4]Specifically, Yaafe was used to extract the following features: *Spectral Crest Factors*, *Slope*, *Decrease*, *Variation*, *Frequency Derivative of Constant-Q Coefficients*, *Octave Band Signal Intensities*, *Beat Histogram*, *Autocorrelation Coefficients*, *Waveform Moments*, and *Amplitude Modulation*; Essentia was used to extract *Spectral Contrast*; and other features were extracted by MIR Toolbox.

Table I. Examples of Frequent Concept Patterns and Infrequent Concept Patterns Discovered in Our Dataset

Frequent Concept Patterns	Infrequent Concept Patterns
aggressive, guitar, rock	literate, snare, hiphop
literate, saxophone, country	humorous, clarinet, funk
rollicking, guitar, electronic	rollicking, snare, hiphop
passionate, violin, electronic	aggressive, clarinet, funk
aggressive, drumkit, alternative	humorous, drumkit, classical

Each concept pattern comprises a concept from each of the three music concept Types: *Mood*, *Instrument*, and *Genre*.

probability of the j-th concept of C_i in a segment. The threshold filtering is illustrated in Figure 4(b). This filtering process is conducted in each music dimension separately. The threshold for each concept dimension is determined empirically in experiments (refer to Section 4.4).

Infrequent Concept Pattern Filtering. Because existing music concept classification algorithms cannot obtain very accurate results [Downie 2014], it is possible that there are still misclassified concepts remaining after threshold filtering. To further improve the quality of generated concept sequences for songs, we propose an *Infrequent Concept Pattern Filtering* (ICPF) method. The underlying assumption is that there exist inherent interactions between concepts in different music dimensions, such as the use of *instruments* in different *genres* and the expressed *moods* of certain *instruments* and *genres*. Although a piece of music can contain or express any combination of concepts, some are very rare. For example, *guitar* is a popular instrument to express *passionate* mood in *rock* music, whereas *drumkit* has less chance of being found in *classical* music to express *humorous*. A *concept pattern* comprises a concept from each of the music dimensions. For example, suppose there are three music dimensions: *mood*, *instrument,* and *genre*, then {*passionate*, *guitar*, *rock*} is a concept pattern. Infrequent Concept Pattern (ICP) indicates those concept patterns that are rarely found or nonexistent in a large music corpus, such as {*humorous*, *drumkit*, *classical*}. The music dimensions and corresponding concepts used in this study are discussed in Section 4.1.1 and shown in Table III. Table I shows some examples of Frequent Concept Patterns (FCPs) and ICPs. The ICPF process removes suspicious concepts that cause such rare combinations. The intuition is that the appearance of an ICP is due to mis-detected concepts. Detailed steps of the ICPF process are as follows:

—**Step 1. Concept Pattern Construction:** For a segment of a song in the dataset, after concept probability estimation and threshold filtering, a set of concepts of different music dimensions are obtained. With the obtained concepts, all the concept patterns of this segment are formed based on the concept pattern definition. For example, suppose three music dimensions are considered and, for a segment, the obtained concepts are: three concepts in the first music dimension $\{c_{11}, c_{13}, c_{15}\} \in C_1$, two concepts in the second music dimension $\{c_{22}, c_{24}\} \in C_2$, and two concepts in the third music dimension $\{c_{32}, c_{37}\} \in C_3$. Then, in this segment, 12 concept patterns can be formed, such as $\{c_{11}, c_{24}, c_{37} | c_{11} \in C_1, c_{24} \in C_2, c_{37} \in C_3\}$.
—**Step 2. FCP Set and ICP Set Construction:** Count the frequency of each concept pattern formed by all the segments of songs in the dataset and then construct an FCP set and an ICP set based on the frequency of concept patterns (refer to Section 4.4).
—**Step 3. Noisy Concept Removal:** For each segment of a song, detect the ICPs and remove suspicious concepts that cause such ICPs using Algorithm 1. Specifically, for the set of concepts in an ICP of a segment, we remove the one that appears in the most number of ICPs (Lines 9–15) or the least number of FCPs (Lines 17–23) in this

ALGORITHM 1: Infrequent Concept Pattern Filtering Process of a Segment

Input: $\mathcal{S}_{fcp}$: FCP set; $\mathcal{S}_{icp}$: ICP set; $\mathcal{C}$: Concept set of a segment;
$\mathcal{P}$: $P_c(c \in \mathcal{C})$ is the estimated probability of concept c in the segment
Output: $\mathcal{C}$; // return the remaining concepts after filtering for the segment

1 Form all the concept patterns $\mathcal{L}$ with $\mathcal{C}$;
2 $\mathcal{C}_{temp} = []$; // define a empty set for concepts
3 **while** $\mathcal{S}_{icp} \bigcap \mathcal{L} \neq \emptyset$ **do**
4 **for** *each concept* $c \in \mathcal{S}_{icp} \bigcap \mathcal{L}$ **do**
5 $\mathcal{C}_{temp} = \mathcal{C}_{temp} \bigcup c$;
6 **for** *each concept* $c \in \mathcal{C}_{temp}$ **do**
 /* count the number of times c in a ICP of the segment */
7 Get m_c: the number of concept patterns $l \in \mathcal{S}_{icp} \bigcap \mathcal{L}$ containing c;
 /* count the number of times c in a FCP of the segment */
8 Get n_c: the number of concept patterns $l \in \mathcal{S}_{fcp} \bigcap \mathcal{L}$ containing c;
 /* get the concepts which appear in the most number of ICPs */
9 Set $m = max(m_c, \forall c \in \mathcal{C}_{temp})$;
10 $\mathcal{C}_{icp} = []$;
11 **for** *each concept* $c \in \mathcal{C}_{temp}$ **do**
12 **if** $m_c == m$ **then**
13 $\mathcal{C}_{icp} = \mathcal{C}_{icp} \bigcup c$;
 /* remove the concept which appears in the most number of ICPs */
14 **if** $|\mathcal{C}_{icp}| == 1$ **then**
15 Remove $c \in \mathcal{C}_{icp}$ from $\mathcal{C}$;
16 **else**
 /* get the concepts which appear in the least number of FCPs */
17 Set $n = min(n_c, \forall c \in \mathcal{C}_{icp})$;
18 $\mathcal{C}_{fcp} = []$;
19 **for** *each concept* $c \in \mathcal{C}_{temp}$ **do**
20 **if** $n_c == n$ **then**
21 $\mathcal{C}_{fcp} = \mathcal{C}_{fcp} \bigcup c$;
 /* remove the concept which appears in the least number of FCPs */
22 **if** $|\mathcal{C}_{fcp}| == 1$ **then**
23 Remove $c \in \mathcal{C}_{fcp}$ from $\mathcal{C}$;
24 **else**
 /* if there are more than one concepts appearing in the most number of ICPs
 and the least number of FCPs, remove the ones with smallest probability */
25 Remove the concepts $c \in \mathcal{C}_{fcp}$ with the smallest $P_c(\forall c \in \mathcal{C}_{fcp})$ from $\mathcal{C}$;
26 Re-form the concept patterns $\mathcal{L}$ with the remaining concepts $\mathcal{C}$;
27 $\mathcal{C}_{temp} = []$;
28 Return $\mathcal{C}$;

segment. If two concepts appear in the same number of ICPs and FCPs (i.e., both concepts appear in the most number of ICPs and the least number of FCPs), the label with lower probability (P_{ij}) will be removed (Line 25).

3.3. Location-aware Topic Model

In the real world, various songs could be suitable for a particular venue. A human possesses an amazing capability to judge *whether a song fits a venue* or *which song has higher suitability to a venue*. However, it is not easy to explicitly explain the reason in a straightforward way. Although people usually interpret music using various semantic

Table II. Notations and Their Definitions

Notation	Definition
m	a song (document)
t	a term in the vocabulary
w	a word (music cocnept) in documents[5]
z	a latent topic in LTM
N_m	the total number of words in the song m
V	the total number of terms in the vocabulary
N_l^k	the total number of times observing topic k in the venue l
N_m^k	the total number of times observing topic k in the song m
N_k^t	the total number of times observing term t with the topic k
N_{y_0}	the number of times that a word is drawn from venue types
N_{y_1}	the number of times that a word is drawn from songs
$y \in \{0, 1\}$	an indicator variable:
	if $y = 1$, w is drawn from the topic distribution of the song;
	if $y = 0$, w is drawn from the topic distribution of the associated venue
K, M, L	the total number of topics, songs, and venue types, respectively
$\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{Z}$	vectors for words, indicators, and topics, respectively
$\boldsymbol{\theta_m}$	a multinomial distribution over topics specific to song m
$\boldsymbol{\psi_l}$	a multinomial distribution over topics specific to venue l
$\boldsymbol{\phi_z}$	a multinomail distribution over words specific to the topic z
$\theta_{m,k}$	the probability of topic $z = k$ in the song m
$\psi_{l,k}$	the probability of topic $z = k$ in the venue type l
$\phi_{k,t}$	the probability of term t in topic $z_t = k$
π	the parameter of Bernoulli distribution $P(y = 1)$
$\boldsymbol{\eta}$	Beta priors ($\boldsymbol{\eta} = \{\eta_0, \eta_1\}$)
$\boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\beta}$	Dirichlet priors ($\boldsymbol{\alpha}$: K-vector, $\boldsymbol{\gamma}$: K-vector; $\boldsymbol{\beta}$:V-vector)

concepts, explanation based on concepts or a mixture of concepts could be inaccurate, less comprehensive, and confusing in many cases. One approach is to describe and model a venue's characteristics by combining the musical concepts that are suitable for the venue. In other words, one can map the venue and music items into a common musical concept space. The drawback of this method is its lack of an effective capability to model interactions between different concepts. Many music concepts are generally highly correlated and not independent of each other. In fact, they are intertwined together in a song to express certain semantics. For example, compiling the same song in different styles and using different instruments can create different atmospheres and give us different feelings. Music selection for a venue is highly related to the combinations and associations of multiple concepts. Motivated by this observation and discussion, we develop a novel topic model, the LTM, to facilitate a joint modeling of songs and venues under a latent topic space, in which the association and suitability between music and venues can be directly characterized and measured. In LTM, each latent topic is represented by a mixture of music concepts; in turn, songs and venues are the mixtures of topics. An LTM topic can be treated as a particular interaction between music concepts. The topics and their associations (i.e., the representation of a venue) explain the underlying reasons why people prefer certain songs at a certain type of venue. Table II lists the notations used in the following model description.

[5]We adopt the terminology of *term* and *word* in Heinrich 2005]: *Term* refers to the element of a vocabulary and *word* refers to the element of a document, respectively. A term can be instantiated by several words in a text corpus.

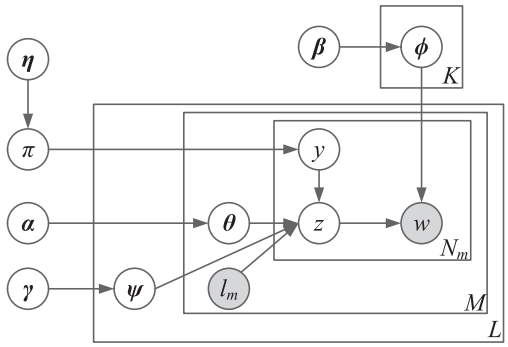


Fig. 5. Plate notation of the Location-aware Topic Model.

3.3.1. Model Description. The LTM is a generative probabilistic model used to characterize the associations between music contents and venue types. The associations are constructed via a set of latent semantic topics that are discovered from a venue-labeled music corpus. The corpus consists of a set of songs labeled with one or several venue labels, indicating that the song is suitable for these venues. The *common features* embedded in songs labeled with the same venue characterize the *music preference* of a venue. For the LTM, the music preference of a venue l is represented as a probabilistic distribution of latent topics, $\boldsymbol{\psi_l}$.[6] Meanwhile, each song m is also modeled as a probabilistic distribution of the same latent topics $\boldsymbol{\theta_m}$, which captures the *latent semantics* expressed by the song. Each latent topic z is a probabilistic distribution of terms or music concepts, denoted as $\boldsymbol{\phi_z}$, which effectively captures rich interactions between different music concepts. LTM can be represented by the graphical model shown in Figure 5. In the generation of a song m labeled with a venue l, each word w of the song m could be generated based on the music preference of the venue $\boldsymbol{\psi_l}$ or generated according to this song's properties $\boldsymbol{\theta_m}$. As shown in Figure 5, LTM contains a switch mechanism that controls the generation of words based on the topic distribution of the venue l_m or the song m. In particular, an indicator variable $y \in \{0, 1\}$ from Bernoulli distribution is parameterized by π associated with each word w. y acts as a switch: If $y = 0$, a topic z is drawn from $\boldsymbol{\psi_l}$ first, then word w is drawn from ϕ_z; otherwise, if $y = 1$, a topic z is drawn from $\boldsymbol{\theta_m}$ first, then word w is drawn from ϕ_z. Formally, the generative process of LTM is:

—For each topic $z \in \{1, \ldots, K\}$, draw a multinomial distribution $\boldsymbol{\phi_z} \sim Dir(\cdot|\boldsymbol{\beta})$;
—For each song $m \in \{1, \ldots, M\}$, draw a multinomial distribution $\boldsymbol{\theta_m} \sim Dir(\cdot|\boldsymbol{\alpha})$;
—For each venue $l \in \{1, \ldots, L\}$, draw a multinomial distribution $\boldsymbol{\psi_l} \sim Dir(\cdot|\boldsymbol{\gamma})$;
—For each song $m \in \{1, \ldots, M\}$ labeled with a venue $l_m \in \{1, \ldots, L\}$[7]:
  —For each word $w \in \{1, \ldots, N_m\}$ in the song m:
    (1) draw $y \sim Bernoulli(\cdot|\pi)$
      if $y = 0$, draw z from the topic distribution $\boldsymbol{\psi_{l_m}}$ of the venue l_m;
      if $y = 1$, draw z from the topic distribution $\boldsymbol{\theta_m}$ of the song m;
    (2) draw the word w from ϕ_z

[6]In this article, unless otherwise specified, notations in bold denote matrices or vectors, and notations in normal style denote scalars.
[7]It is possible that a song is suitable for several venues with similar properties. In such cases, for each word in the song m, a location l is chosen uniformly from the labeled venues.

According to the generation process, the probability of a word w in a song m under venue type label l is:

$$\begin{aligned} P(w|m,l) &= \pi P(w|\boldsymbol{\theta_m}, \boldsymbol{\phi}, m) + (1-\pi)P(w|\boldsymbol{\psi_l}, \boldsymbol{\phi}, l) \\ &= \pi \sum_z P(w|z, \boldsymbol{\phi})P(z|\boldsymbol{\theta_m}, m) + (1-\pi)\sum_z P(w|z, \boldsymbol{\phi})P(z|\boldsymbol{\psi_l}, l), \end{aligned} \tag{2}$$

where $P(w|\boldsymbol{\theta_m}, \boldsymbol{\phi})$ is the probability that the word w in m is generated according to the song's music properties, $P(w|\boldsymbol{\psi_l}, \boldsymbol{\phi})$ is the probability that the word w in m is generated based on the venue's music preference. π is the Bernoulli parameter or *mixing weight* that controls the generation process. From the generation process, we can easily find that the topic distribution of a song is determined by the word (i.e., music concept) occurrences in this song. The generated latent topics are meant to capture the difference between songs. At the same time, the *word co-occurrence patterns* or *hidden associations between the words/concepts* embedded in the songs of a venue are captured by the topic distribution of this venue. A venue's topic distribution can be regarded as the background distribution of the songs that are suitable for the venue, and the topic distribution of each song is a variation of the venue's topic distribution. As different songs are suitable for different venues, the topics are also tailored for discriminating the characteristics of different venues.

The proposed LTM discovers (i) each venue's music distribution over latent topics $\boldsymbol{\psi_l}$, (ii) each song's topic distribution $\boldsymbol{\theta_m}$, (iii) topic distribution over music concepts $\boldsymbol{\phi}$, and (iv) the mixing weight π. The generative model captures the associations between songs and venues via the generation of a venue-labeled music corpus. With the model hyperparameters $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\eta}\}$, the generation probability of a corpus D with the observed and hidden variables is:

$$\begin{aligned} P(D|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\eta}) = \int \cdots \int \prod_{m=1}^{M} \prod_{i=1}^{N_m} & P(w_i|z, \boldsymbol{\phi})P(\boldsymbol{\phi}|\boldsymbol{\beta}) \\ & P(z|\boldsymbol{\theta_m}, \boldsymbol{\psi}_{\boldsymbol{l_m}}, y)P(\boldsymbol{\theta_m}|\boldsymbol{\alpha})P(\boldsymbol{\psi}_{\boldsymbol{l_m}}|\boldsymbol{\gamma}) \\ & P(y|\pi)P(\pi|\boldsymbol{\eta})d\boldsymbol{\theta_m}d\boldsymbol{\psi}_{\boldsymbol{l_m}}d\boldsymbol{\phi}d\pi \end{aligned}. \tag{3}$$

3.3.2. Model Inference. In the LTM model, the estimation of the generation probability of a corpus involves a set of parameters as shown in Equation (3). Among them, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\eta}$ are hyperparameters and predefined. Their effects on the model are discussed in Section 3.3.4 and empirically studied in Section 5.4. The parameters to be estimated are (i) venue-topic distribution $\boldsymbol{\psi}$, (ii) song-topic distribution $\boldsymbol{\theta}$, (iii) topic-term distribution $\boldsymbol{\phi}$, and (iv) Bernoulli distribution parameter π. In addition, in the generation process, we also need to assign the indicator vector $\boldsymbol{Y}$ and latent topic vector $\boldsymbol{Z}$ to the sequence of words $\boldsymbol{W}$ in the corpus. Various approximate inference methods have been developed to estimate the parameters in variants of LDA, such as variation inference [Blei et al. 2003], expectation propagation [Minka and Lafferty 2002], and collapsed Gibbs sampling [Griffiths and Steyvers 2004]. Although Gibbs sampling is not necessarily as computationally efficient as approximation schemes such as variation inference and expectation propagation, it is unbiased and has been successfully applied in many large-scale applications of topic models [Griffiths and Steyvers 2004; Rosen-Zvi et al. 2010; Tang et al. 2012; Yin et al. 2014]. Following these studies, we apply collapsed Gibbs sampling to obtain samples of the hidden variable assignments and to estimate the unknown parameters $\{\theta, \psi, \phi, \pi\}$. In the collapsed Gibbs sampling, each latent variable is iteratively updated given the remaining variables. The

parameters $\{\theta, \psi, \phi, \pi\}$ are estimated based on the results of a constructed Markov chain that converges to the posterior distribution on z. The collapsed Gibbs Sampling process of LTM is described in Algorithm 2.

ALGORITHM 2: Collapsed Gibbs Sampling Process for LTM

Input: D: A venue-labeled music dataset;
K: number of topics;
Dirichlet priors: $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$;
Beta priors: $\boldsymbol{\eta}$
Output: Estimated parameters $\boldsymbol{\theta}$, $\boldsymbol{\psi}$, $\boldsymbol{\phi}$, π

```
Initialize Z and Y by assigning random values;
Count N_l^k, N_m^k, and N_k^t based on initialized Z;
Count N_{y_0} and N_{y_1} based on initialized Y;
for each Gibbs sampling iteration do
    for each song m = 1, ..., M do
        for each word w = 1, ..., N_m do
            Sample y_w ~ Bernoulli(·|π) based on π's value computed by Eq. 9;
            if y_w == 0 then
                Draw z_w according to Eq. 4;
            if y_w == 1 then
                Draw z_w according to Eq. 5;
            Update N_l^k, N_m^k, and N_k^t based on z_w = k;
            Update N_{y_0} and N_{y_1} based y_w;
Estimate model parameters θ, ψ, φ, and π according to Eq. 6, Eq. 7, Eq. 8 and Eq. 9,
respectively
```

Detailed derivation of the sampling process can be found in the Appendix. Here, we only show how to jointly sample $y_i \in \boldsymbol{Y}$ and $z_i \in \boldsymbol{Z}$ of a word $w_i \in \boldsymbol{W}$ conditioned on all other variables. y_i and z_i must be sampled jointly, because y_i decides whether to sample z_i from $\boldsymbol{\psi}$ or from $\boldsymbol{\theta}$. Formally, we define that $\boldsymbol{W}$ is a sequence of words during the sampling process, $\boldsymbol{Z}$ and $\boldsymbol{Y}$ denote the set of topics z and indicators y to the word sequence, respectively. $\boldsymbol{W_{\neg i}}$ denotes $\boldsymbol{W}$ excluding the i-th word w_i. Similar notation is used for other variables. For $\boldsymbol{W} = \{w_i, \boldsymbol{W}_{\neg i}\}$, $\boldsymbol{Z} = \{z_i, \boldsymbol{Z}_{\neg i}\}$, and $\boldsymbol{Y} = \{y_i, \boldsymbol{Y}_{\neg i}\}$, the joint probability of sampling $z_i = k$ and $y_i = 0$ is:

$$P(z_i = k, y_i = 0 | \boldsymbol{Z}_{\neg i}, \boldsymbol{Y}_{\neg i}, \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\eta}) \propto (\eta_0 + N_{y_0,\neg i}) \cdot \frac{\gamma_k + N^k_{l,\neg i}}{\sum_{k=1}^{K}(\gamma_k + N^k_{l,\neg i})} \cdot \frac{\beta_t + N^t_{k,\neg i}}{\sum_{t=1}^{V}(\beta_t + N^t_{k,\neg i})}. \quad (4)$$

Similarly, the joint probability of sampling $z_i = k$ and $y_i = 1$ is:

$$P(z_i = k, y_i = 1 | \boldsymbol{Z}_{\neg i}, \boldsymbol{Y}_{\neg i}, \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\eta}) \propto (\eta_1 + N_{y_1,\neg i}) \cdot \frac{\alpha_k + N^k_{m,\neg i}}{\sum_{k=1}^{K}(\alpha_k + N^k_{m,\neg i})} \cdot \frac{\beta_t + N^t_{k,\neg i}}{\sum_{t=1}^{V}(\beta_t + N^t_{k,\neg i})}, \quad (5)$$

where N^k_l denotes the number of times observing topic k in venue l, N^k_m denotes the number of times observing topic k in song m, N^t_k denotes the number of times that term t is observed with topic k. N_{y_0} and N_{y_1} denote the number of times that words are drawn

from venues and songs, respectively. Based on the state of the Markov chain $\boldsymbol{Y}$ and $\boldsymbol{Z}$, we can estimate the parameters:

$$\theta_{m,k} = \frac{\alpha_k + N_m^k}{\sum_{k=1}^{K}(\alpha_k + N_m^k)} \qquad \text{song} - \text{topic distribution} \tag{6}$$

$$\psi_{l,k} = \frac{\gamma_k + N_l^k}{\sum_{k=1}^{K}(\gamma_k + N_l^k)} \qquad \text{venue} - \text{topic distribution} \tag{7}$$

$$\phi_{k,t} = \frac{\beta_t + N_k^t}{\sum_{t=1}^{V}(\beta_t + N_k^t)} \qquad \text{topic} - \text{term distribution} \tag{8}$$

$$\pi = \frac{\eta_1 + N_{y_1}}{\eta_1 + \eta_0 + N_{y_1} + N_{y_0}} \qquad \text{Bernoulli distribution parameter} \tag{9}$$

3.3.3. Generalization to New Songs. To predict whether a new song $\hat{m}$ is suitable for a venue l, the topic distribution of this new song $\boldsymbol{\theta_{\hat{m}}}$ needs to be estimated. The generalization of the LTM to an unobserved song is the same as that of a standard LDA. First, topics are randomly assigned to words in the new song, then a set of iterations through Gibbs sampling is performed by updating locally for the word w_i of $\hat{m}$:

$$p(z_i = k|\hat{w}_i, \hat{\boldsymbol{Z}}_{\boldsymbol{\neg i}}, \hat{\boldsymbol{W}}_{\boldsymbol{\neg i}}, \boldsymbol{\alpha}) = \frac{\beta_t + N_k^t + \hat{N}_{k,\neg i}^t}{\sum_{t=1}^{V}(\beta_t + N_k^t + \hat{N}_{k,\neg i}^t)} \cdot \frac{\alpha_k + N_{\hat{m},\neg i}^k}{\sum_{k=1}^{K}(\alpha_k + N_{\hat{m},\neg i}^k)}, \tag{10}$$

where $\hat{N}_k^t$ counts the observations of term t with topic k in the new song and $N_{\hat{m}}^k$ counts the times of topic k observed in the new song. $\hat{N}_{k,\neg i}^t$ and $N_{\hat{m},\neg i}^k$ denote $\hat{N}_k^t$ and $N_{\hat{m}}^k$ excluding the i-th word w_i, respectively. The topic distribution of the new song can be calculated using Equatioin (6) after sampling. Notice that although the generalization of LTM is the same as in LDA, the topics of the LTM are *tailored for discriminating the characteristics of different venues* or *locationalized* in the training process, which is reflected in the topic-term associations $\phi_{k,t}$ (determined by N_k^t). The locationalized topic-term associations are propagated to the song-topic associations of the new song; as we observe in the equation, N_k^t dominates the topic sampling process compared to $\hat{N}_k^t$ and $N_{\hat{m}}^k$, because $\hat{N}_k^t$ and $N_{\hat{m}}^k$ are randomly assigned and $\boldsymbol{\alpha}$ is symmetric.[8]

3.3.4. Effects of Hyperparameters. In this section, we focus on studying the effects of hyperparameters on model performance. LTM has four important hyperparamers: $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\eta}$. $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ have a smoothing effect on multinomial parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ in LTM. $\boldsymbol{\gamma}$ has the same effect on $\boldsymbol{\psi}$ as $\boldsymbol{\alpha}$ on $\boldsymbol{\theta}$. From Equations (4) and (5), we can see that, in the sampling process, the elements of $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\beta}$ become pseudocounts for the corresponding song-topic associations, venue-topic associations, and topic-word associations, respectively. In the case of no prior knowledge on the topic distributions of location, songs, and the word distribution of topics, $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\beta}$ are often set to be symmetric. Generally, they are set to small values (e.g., $\alpha = 50/K$ and $\beta = 0.01$ as suggested in [Griffiths and Steyvers 2004]) such that the distributions are decided by the observations in corpus. In extreme cases, if we set β to a very large value, then the word distribution becomes uniform in each topic. As a result, the topic loses the discriminative power. Similarly, $\boldsymbol{\eta}$ has a smoothing effect on the Bernoulli distribution. As shown in Equations (4) and (5), η_0 becomes pseudocounts for N_{y_0}, and η_1 becomes pseudocounts for N_{y_1}. In our implementation, we find that unless very unbalanced

[8]In the case that we have no prior knowledge about the data, Dirichlet prior $\boldsymbol{\alpha}$ is set to be symmetric, which means $\alpha_k = \alpha$ for $1 \leq k \leq K$.

values are set to η_0 and η_1 (e.g., $\eta_0 \gg \eta_1$), the setting of $\boldsymbol{\eta}$ has very limited effects on the sampling process. The settings of the hyperparameters in our implementation are detailed in Section 4.4.

4. EXPERIMENTAL SETUP

We conducted a series of experiments to study the performance of the VenueMusic System and to address the following research questions:

—**RQ1**: Is it better to use latent topics to capture the associations between songs and venues compared to directly using low-level audio features or semantic concepts?
—**RQ2**: Is it better to represent songs as music concept sequences in the LTM than to represent songs as "bag-of-audio-words"?
—**RQ3**: Does the use of the ICPF process improve the final performance?
—**RQ4**: What are the effects of hyperparameters and topic numbers on the final performance of the LTM?

To answer these questions, we compared the performance of the LTM with four competitors: two content-based recommendation methods,[9] an LTM based on "audio words," and a LTM based on generated music concepts without the use of the ICPF process. We used two datasets, TC1 (Section 4.1.2) and TC2 (Section 4.1.3). The influence of parameters on final performance was carefully studied in TC1 (Section 5.4). We also compared LTM with other methods, such as Jaccard Similarity in Braunhofer et al. [2013],[10] Autotagger,[11] and two LDA variants (i.e., Author-Topic Model [Steyvers et al. 2004] and the location-aware topic model in Wang et al. [2007]). Because these methods are not designed for the current task (recommending songs to venue types[12]), their performance is very limited.[13] In this article, we only present the performance results of the four competitors (described in Section 4.3).

We describe the construction of two test collections in Section 4.1. We introduce experimental evaluation metrics in Section 4.2 and competitors in Section 4.3, followed by the experimental configurations in Section 4.4.

4.1. Test Collection Construction

Test collection plays an important role in large-scale performance evaluation and comparison. In this work, we carefully developed three test collections to facilitate empirical study; these can be accessed at http://www.mysmu.edu/faculty/jlshen/venuemusicdataset/dataset.rar.

4.1.1. Concept-Labeled Music Dataset. A dataset with songs labeled by music concepts was built for teaching SVM classifiers to estimate the probabilities of music concepts in each music segment (described in Section 3.2.2). In experiments, three music concepts

[9]Collaborative-based Filtering (CF) methods are not used in comparisons because CFs are suitable for cases with large number of users (venues in our case), but there are only eight venues in our experiments.
[10]In Braunhofer et al. [2013], a song and a POI are matched based on the similarity between manually labeled concepts. In our implementation, because no manual labels are available, we use the generated concept vectors of songs and venues (concept generation of venues is described in Section 4.3) for computing Jaccard similarity.
[11]Autotagger is used to classify each song into different venues.
[12]The two topic models are described in Section 2; Jaccard Similarity in Braunhofer et al. [2013] relies on manually labeled tags, and Autotagger is a classification method
[13]For all four methods, with the parameters of ATM and LATM tuned as described in 4.4, their average accuracies (precision@20) in TC2 are lower than 20%. The highest precision for the four methods is obtained by Jaccard similarity: 0.1875.

Table III. Types of Three Music Concepts Used in Experiments

Concept	Classes
Mood	aggressive, humorous, literate, passionate, rollicking
Genre	alternative, blues, classical, country, electronic, funk, hip-hop, jazz, metal, pop, reggae, rock
Instrument	trombone, trumpet, tuba, flute, clarinet, saxophone, piano, snare, drumkit, violin, cello, guitar

are used[14]: *genre*, *mood*, and *instrument*. The three types of concepts are selected because they are important concepts usually used to describe music preferences according to studies from psychology and cognition [Greasley and Lamont 2006; Rentfrow and Gosling 2003], and they are the most commonly used music concepts to annotate songs by ordinary users [Lamere 2008]. The five mood classes in the MIREX mood classification task[15] are used in the *mood* dimension. Twelve genres are used in the *genre* dimension,[16] and 12 instruments from four types of popular instruments [Zhang et al. 2009b] are used in the *instrument* dimension. The classes of each concept dimension are shown in Table III. For each class, 100 songs were carefully selected. A 30-second audio stream for each selected song was downloaded from 7digital.[17]

Details on the procedure of song selection for each concept dimension are described next.

—**Song Selection for *Mood*:** Songs for the mood dimension were collected from Allmusic,[18] an expert-based music website. Allmusic provides representative songs for various moods and genres.[19] There are 50 songs for each type. For the five classes of mood, each class represents a cluster of similar moods[20] (in Table III, a mood represents a mood class). The 50 songs provided by Allmusic for each mood in a mood class were collected first and then 100 songs were randomly selected for the class.

—**Song Selection for *Genre*:** *Blues*, *classical*, *country*, *electronic*, *jazz*, and *reggae* are clearly listed in Allumusic and provided 50 songs for each type. To obtain more songs from these genres and songs from other genres, we referred to DigitalDreamDoors,[21] which provides more than 200 music and movie lists. These lists are created through crowdsourcing, and the website allows people to review each list. Each list is revised regularly by the editor who creates the list based on users' comments. After collecting songs from corresponding genre lists on the website, three music hobbyists were asked to cross-check and select songs for each genre. A song was selected for a certain genre when the three evaluators agreed. Through this process, 100 songs were selected for each genre.

—**Song Selection for *Instrument*:** For each instrument, we searched (i) albums and songs of famous soloists on the instrument, such as *Taylor Davis* for *violin*, *Alison Balsom* for *trumpet,* and (ii) we searched albums and songs using keywords like "*guitar solo*," "*guitar music*," "*guitar songs*" in 7digital. After collecting the candidate

[14]Although there are only three types of music concepts in our implementation, more concepts can be used. When more music concepts are used, our model is expected to model the song and venue more accurately.
[15]http://www.musicir.org/mirex/wiki/2009: Audio_Music_Mood_Classification.
[16]According to the study of Rentfrow and Gosling [2003], 14 general genres are sufficient to represent user music preferences on the aspect of genre. In addition to the 12 genres used here, there are *sound track* and *religious* genres.
[17]https://www.7digital.com/.
[18]http://www.allmusic.com/.
[19]http://www.allmusic.com/genres; http://www.allmusic.com/moods.
[20]http://www.musicir.org/mirex/wiki/2009:Audio_Music_Mood_Classification.
[21]http://www.digitaldreamdoor.com/pages/about_us_ddd.html. Access on 27 December 2013.

songs, the same assessment procedure as used in genre music selection was conducted to select 100 songs for each instrument. The selected songs for an instrument include pure music, songs, solo pieces, and pieces mixed with other instruments.

The selection procedure, which first selects songs from reliable resources and then manually checks the songs by human subjects, guarantees data quality and saves time and labor. Note that a relatively simple procedure is adopted to verify the *genre* and *instrument* of a candidate song. This is because, in general, a song can be classified into a certain genre that a majority will agree on, and it is clear whether a song is played with a particular instrument or not. Because of the objective nature of the judgment on genre and instrument, it was easy for raters to agree on whether a song belonged to a genre or was played with a particular instrument. Similar to the song selection procedure used for each concept just described, candidate songs were first collected and then verified by human subjects.

4.1.2. Venue-Labeled Dataset (TC1). In this dataset, each song is labeled with one or several venue types. A song's labels indicate which venue types this song is suitable for. Eight representative types of venues in daily life were selected for the experiments: *library*, *gym*, *restaurant*, *bedroom*, *mall*, *office*, *bus/train*[22], and *bar*. These represent venues where people often enjoy music. The song candidates for each venue were collected from the corresponding playlists in Grooveshark. Grooveshark contains many playlists created by users, each titled with various contexts such as *gym playlist*, *bar music*, and the like. These labeled playlists in GrooveShark have been successfully used for activity classification [Wang et al. 2012]. Venue-labeled playlists imply that users have special preferences for music content in different venues and also provide us with a good source to collect data. In our implementation, for each venue, the playlists "$*venue*$ *songs,*" "$*venue*$ *music,*" and "$*venue*$ *playlist*" were retrieved in Grooveshark. Songs in the returned playlists were collected. Taking the venue *bar* as an example, "*bar songs*," "*bar music*," and "*bar playlist*" were used to search related playlists. For each venue, we collected songs from at least 150 playlists. More than 5,000 individual songs were collected on average for each venue. Many songs appear in multiple venue playlists. For example, the song "*The Hand That Feeds*" (Nine Inch Nails) appears in 48 playlists for *library*. Because the playlists of a venue are created by different users, the appearance of a song in multiple playlists implies that people have similar preferences for music for a particular venue. Songs for a venue are sorted in descending order based on the number of playlists they appear in. The top 500 songs in the sorted list of each venue were selected.

The selected 500 candidate songs for each venue were then evaluated by human subjects. Nine subjects volunteered for the evaluation. All the subjects are music hobbyists (five females and four males) with different educational backgrounds. Five were students, the other four were working professionals. During the evaluation, they were required to listen to each song of a venue and then rate the song. The rating guidelines are shown in Table IV. The subjects needed to listen to a song for at least 60 seconds before making the final decision.

We studied intersubject agreement by calculating Fleiss's kappa [Landis and Koch 1977] among the nine subjects for every venue. All kappa values are significantly higher than 0 (p-value < 0.001) with the lowest values for *restaurant* (0.076) and *mall* (0.09) and especially high for *bedroom* (0.337), *gym* (0.314), and *library* (0.287). The average kappa value over eight venues is 0.202 (± 0.100). The results indicate that subjects have statistically significant agreement on music for venues. To evaluate the precision and ranking performance of the methods, the song's ratings for a venue were

[22]Bus and train are used to represent *transportation*.

Table IV. Guidelines of Rating a Song for a Type of Venue

Score	Description
1 point	I absolutely will not listen to it in this type of venue
2 point	I can stand it in this type of venue
3 point	I do not mind listening to it in this type of venue
4 point	I like it in this type of venue
5 point	I like it very much in this type of venue

Table V. Number of Relevant Songs for Each Venue in TC1

Bar	Gym	Library	Office	Restaurant	Mall	Bus/Train	Bedroom
266	233	154	176	121	135	221	189

converted into three relevance levels. Specifically, if the majority of subjects (namely, 5 or more) give a rating of greater than 3 to a song for a particular venue, then the song is regarded as *relevant* for the venue; if the majority of subjects give a rating of less than 3 to a song for a particular venue, then the song is regarded as *irrelevant* for the venue; if a song does not rate as either relevant or irrelevant, it is regarded as *neutral*. The number of relevant songs for each venue is shown in Table V.

4.1.3. Large Music Dataset (TC2). Since TC1 is relatively small, another test collection (TC2) was developed for large-scale evaluation. TC2 contains 10,000 popular songs selected from Last.fm.[23] This collection was constructed as follows. Artists were collected from among the top 150 artists in each week (the most popular 150 artists in each week) from 20 February 2005 to 24 November 2013 in the category of *all places*.[24] Because the data in Last.fm are known to contain misspellings and mistakes, the collected artist list was checked by matching each artist name in AllMusic. After filtering, the list contained 531 artists. The songs of each artist were collected from the MusicBrainz database.[25] For songs in Last.fm, we collected the number of the song's listeners until 26 November, 2013. Finally, the top 10,000 songs by number of listeners were obtained and their audio tracks were downloaded from 7digital.

4.2. Evaluation Metrics

For recommender systems, the accuracy and ranking of relevant results are crucial. In particular, the relevance of the top results is most important because users usually listen to songs with top ranks in the sequence recommended by a playlist. In this study, precision at k (Precision@k), Average Precision at k (AP@k) and Normalized Discounted Cumulative Gain at k (NDCG@k) [Järvelin and Kekäläinen 2002] are used as evaluation metrics. $P@k$ is the proportion of relevant songs in the top k results, computed as:

$$Precision@k = \frac{\text{No. of relevant items in top } k \text{ results}}{k}. \tag{11}$$

AP averages the precision at each point of a relevant songs in the ranking list. It measures the quality of the whole ranking list.

$$AP@k = \frac{\sum_i^k Precision@i \cdot \delta(rel_i = 1)}{min(k, |rel|)}, \tag{12}$$

[23]http://www.last.fm.
[24]http://www.last.fm/charts/artists/top/place/all?limit=150.
[25]http://musicbrainz.org/. Access on 24 November, 2013.

where rel_i indicates the relevance of the i-th song in the ranking list. If the i-th song is relevant, $rel_i = 1$; otherwise, $rel_i = 0$. $\delta(\cdot)$ is a binary indicator function. $|rel|$ is the number of relevant songs in the dataset. Mean Average Precision (MAP) for a set of queries is the mean of the average precision scores for each query.

NDCG@k is widely used for measuring the rank accuracy, defined as

$$NDCG@k = \frac{1}{Z_k} \sum_{j=1}^{k} \frac{2^{r(j)} - 1}{log_2(j+1)}, \tag{13}$$

where j is the rank position, $r(j)$ is the rating value of j-th song in the ground-truth rank list, and Z_k is the normalization factor which is the discounted cumulative gain in the k-th position of the ground truth rank list. In the computation of NDCG@k, the rating values of relevant, neutral, and irrelevant items are 2, 1, and 0, respectively.

4.3. Competitors

In the following presentation, we use **CLTM_F** to represent our proposed method, which uses LTM based on the extracted music concept sequence with ICPF. We present the results of the following four competitors of CLTM_F to study the four research questions listed earlier.

Audio-Based Filtering (ABF): Each venue is represented by several representative audio feature vectors. Specifically, by representing the songs of a venue using the audio features described in Section 3.2.1, the K-means method is applied to generate k clusters. The feature vectors of the cluster centers are then used to represent the venue. The similarity between a representative vector of a venue and the feature vector of a new song is calculated by Euclidean distance. The best performance over these representative vectors of a venue is used to compare with other methods.

Concept-Based Filtering (CBF): In this method, the histogram of music concepts is used to represent songs and venues. Specifically, based on the generated music concept sequence of a song (described in Section 3.2), the occurrence times of music concepts in the signature are counted and normalized to generate a histogram vector, which is used to represent the song. By aggregating all the music concepts of all songs of a venue, the concept histogram of the venue can be obtained. Then the KL distance is used to compute the similarity between songs and the venue.

Audio Word based LTM (ALTM): This method uses "*bag-of-audio-words*" as input in the LTM. Specifically, each song in a corpus is segmented into small frames, and audio features are extracted from each frame. The K-means method is used to group the frames into clusters based on their audio features. The cluster centers are used as "audio words." Indexing each frame of a song with the closest "audio words," the song is represented as a sequence of audio words. In our implementation, an audio word is a 0.5s music frame.

CLTM: Compared with CLTM_F, this method doesn't have a module to support the ICPF process.

4.4. Experimental Configurations

In our experiments, TC1 was split into training set and test set. Specifically, for each venue, 70% relevant songs were randomly selected to construct the training set. The test data contain 1,000 songs, comprising 30% relevant songs for each venue and 552 randomly selected songs from TC1 (excluding relevant songs for venues). The representations of venues (for the ABF and CBF methods) were obtained based on the songs in the training set. ALTM, CLTM, and CLTM_F were also trained based on the training set of TC1. The learned models based on the TC1 training set were

directly used in TC2. We focus on the performance improvement achieved by CLTM_F over other methods.

4.4.1. Parameter Setting. To generate a concept sequence, a song is partitioned into segments of equal length, and the concepts detected in each segment are filtered by probability threshold τ and the ICPF process. The thresholds of genre and instrument were set to be the same, denoted by τ_{ig}, because they both have 12 classes. τ_m is used to denote the threshold of mood. Obviously, τ cannot be set to be too large or too small to preserve the balance of multiple concepts and to avoid misclassified concepts. In our experiments, we tuned τ_{ig} to $\{0.05, 0.1, 0.15\}$, τ_m to $\{0.15, 0.2, 0.25\}$. Similarly, the segment length should be neither too short nor too long. In our framework, segments are used to generate a music concept sequence to reflect the concept co-occurrence patterns in a song. Overly long segments result in few co-occurrence patterns, which affects statistical results; overly short segments require more computational resources. More importantly, if music partition length is too long, segments tend to contain duplicate information, and if the length is too short, segments often have less information about music semantics [Liu et al. 2005]. In our implementation, the segment length l_s is tuned in the range $\{0.5s, 1s, 2s, 3s, 4s, 5s\}$ based on the final performance.

According to experimental results on TC1, comparable performances were obtained by $\tau_{ig} \in \{0.1, 0.15\}$, $\tau_m \in \{0.2, 0.25\}$ and $l_s \in \{2s, 3s, 4s\}$. The frequent and infrequent patterns are defined based on the co-occurrence times of concept patterns {\$*genre*\$, \$*mood*\$, \$*instrument*\$} on TC1 with $\tau_{ig} = 0.1$, $\tau_m = 0.2$, and $l_s = 2s$. There are a total of 35,888 segments. We define the concept patterns with occurrence times large than 10% of the total number of segments as FCPs and concept patterns with occurrence times less than {1%, 2%, 3%, 4%} of the total number of segments as ICPs. Experiments show that a threshold of 1% can effectively remove noisy concepts, and using {2%, 3%, 4%} hurts the performance slightly. The experimental results presented here are based on $l_s = 3s$, $\tau_{ig} = 0.15$, and $\tau_m = 0.25$ for CLTM, and $l_s = 2s$, $\tau_{ig} = 0.1s$, and $\tau_m = 0.25$ for CLTM_F.

For topic models ALTM, CLTM, and CLTM_F, the hyperparameters were tuned to wide ranges, specifically, $\alpha, \gamma \in \{0.01, 0.05, 0.1, 1.0, 5.0\}$, $\beta \in \{0.01, 0.05, 0.1, 0.15, 0.20, 0.25\}$, and $K \in \{20, 50, 100, 200, 300, 400, 500\}$. $\boldsymbol{\eta}$ is empirically set at $\{10, 10\}$.[26] For ALTM, the number of audio words varies at $[1000, 5000]$ with a 1,000 interval. The best performance (based on Precision@20) with optimized parameters for each method is reported below. In Gibbs sampling, 400 iterations were run as burn-in iterations and then 100 samples with a gap of 5 were taken to obtain the final results.

5. EXPERIMENTAL RESULTS

In this section, we compare and analyze the performance of our proposed method and other competitors on TC1 and TC2 datasets.

5.1. Concept Classification

Music concept sequence generation plays a fundamental role in determining the effectiveness of the final performance of our model. For each music dimension, such as genre, mood, and instrument, a multi-SVM with RBF kernel was trained on 70% of randomly selected songs in each class and evaluated on the remaining items in the concept-labeled music dataset. The parameters γ and C were tuned in the range $\{2^{-10}, 2^{-9}, \ldots, 2^{10}\}$ and $\{10^1, 10^2, \ldots, 10^{10}\}$, respectively. Both parameters are chosen to maximize classification accuracy. Table VI shows the average classification and standard deviation. These accuracies are comparable to the results reported in Zhang et al.

[26]Different symmetric and asymmetric settings for $\boldsymbol{\eta}$ were tested.

Table VI. Accuracy of Music Concept Classification

Concept	Genre	Mood	Instrument
Accuracy	77.5 ± 3.2	60.9 ± 3.4	86.8 ± 1.2

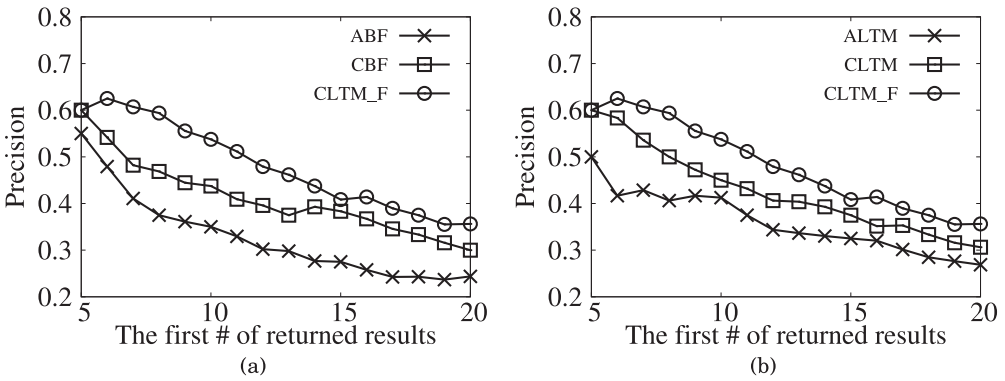


Fig. 6. Average precision@{5 – 20} comparison of different methods on Test Collection 1 (TC1).

[2009a], where a similar approach was applied in developing the CompositeMap music similarity measurement and demonstrated promising performance in large-scale music retrieval. Thus, the overall performance is acceptable and the results can be used in our system for effective location-aware music recommendation, as validated by the results shown in Sections 5.2 and 5.3. It can be expected that more accurate classification results will further boost the final recommendation performance of our method.

5.2. Performance Evaluation on TC1

In this section, we compare and analyze the performance of five methods on TC1. Figure 6(a) shows the average precision@ {5 – 20} of recommendations using acoustic features (ABF), concept histogram (CBF), and our method (CLTM_F). Comparisons were made between them to verify the advantages of using the LTM generated topic distributions to represent songs and venues (**RQ1**). From the figure, we can observe that CBF is consistently better than ABF, and CLTM_F clearly outperforms ABF and CBF with statistically significant improvement. The results demonstrate that low-level acoustic features alone cannot well represent the associations between music content and venues. Topics generated by the LTM, which associate music concepts in high-level semantic space, can better capture the connections between music content and venues. The comparisons among ALTM, CLTM, and CLTM_F shown in Figure 6(b) demonstrate the advantages of learning topics using music concepts over audio words (**RQ2**) and the usefulness of ICPF (**RQ3**). The ICPF process indeed improves the final performance, as CLTM_F outperforms CLTM. It implies that the process can obtain more suitable music concept sequences for a song. Furthermore, by comparing the results in Figure 6(a) and Figure 6(b), we observe that the performance of ALTM is only comparable to that of ABF, and CLTM is slightly better than CBF. The results indicate that the quality of music representation is crucial to the success of our LTM on venue-aware music recommendation.

Table VII shows the *Precision@20* and *AP@20* of our methods and other competitors in each venue. CBF achieves better results than ABF in *bar*, *bedroom*, *gym*, *library*, and *office*, but does not show any improvement in *restaurant*, *mall*, and *bus/train*. This is an interesting observation, one that in accord with the interperson annotation

Table VII. Precision and Average Precision Comparison across Different Venues on Test Collection 1 (TC1)

	Precision@20					AP@20				
Venue	ABF	ALTM	CBF	CLTM	CLTM_F	ABF	ALTM	CBF	CLTM	CLTM_F
Bar	0.3	0.3	0.35	0.3	**0.40**	0.118	0.226	0.173	0.197	**0.317**
Bedroom	0.25	0.35	0.35	0.4	**0.45**	0.189	0.151	0.297	**0.307**	0.301
Gym	0.25	0.35	0.35	0.4	**0.45**	0.104	0.216	0.325	0.294	**0.326**
Library	0.2	0.2	0.25	0.25	**0.30**	0.117	0.115	0.151	0.144	**0.176**
Office	0.15	0.2	0.3	0.25	**0.45**	0.057	0.085	0.141	0.169	**0.259**
Restaurant	**0.30**	0.15	**0.30**	0.25	0.25	0.151	0.053	**0.220**	0.089	0.105
Mall	0.2	0.2	0.2	0.2	0.2	0.077	0.073	0.069	0.118	**0.120**
Bus/Train	0.3	**0.40**	0.3	**0.40**	0.35	0.169	0.273	0.116	**0.288**	0.272
Mean	0.244	0.269	0.300	0.306	**0.356**	0.123	0.149	0.187	0.201	**0.235**

Table VIII. NDCG@20 Comparison of Different Methods across Different Venues on Test Collection 1 (TC1)

Venue	ABF	ALTM	CBF	CLTM	CLTM_F
Bar	0.291	0.428	0.349	0.400	**0.524**
Bedroom	0.381	0.33	0.492	0.515	**0.527**
Gym	0.272	0.432	0.506	0.508	**0.534**
Library	0.295	0.293	0.345	0.306	**0.385**
Office	0.173	0.222	0.308	0.358	**0.456**
Restaurant	**0.423**	0.169	0.361	0.270	0.331
Mall	0.295	0.221	0.206	0.225	**0.296**
Bus/Train	0.38	**0.507**	0.299	0.496	0.477
Mean	0.314	0.325	0.358	0.385	**0.441**

agreement analysis (Section 4.1.2). A possible explanation is that people may have more consistent preferences on the types of music they like in the former five venues than in the latter three venues. *Bar* and *gym* have special atmospheres where people tend to enjoy certain types of music. For these venues, the performance can be further improved by CLTM_F. It implies that for those venues where music concepts can be directly described to some extent, the CLTM_F topics can better capture the semantics of the venues. Similar to CBF, CLTM_F does not show any improvement in *mall*, and even performs worse in *restaurant*. This is partially because there are different kinds of *malls* and *restaurants*, and subjects annotated songs based on the types of mall and restaurant they frequently visit in daily life. Accordingly, the obtained relevant songs for the two venues are relatively diverse (low kappa values). Consequently, it is harder for the model to capture the associations between music and the two venues, resulting in poor performance. Particularly, CLTM and CLTM_F achieve better results in *bus*/*train*, where CBF does not show any advantages over ABF. This suggests that music concept-based LTM methods have the potential to capture the underlying reasons for music preference in the venue where the music concepts cannot be well explained (**RQ1**). Compared to CLTM, CLTM_F demostrates much more consistent performance, which clearly shows the effectiveness of ICPF (**RQ3**).

To evaluate and compare the ranking performance of the methods, NDCG@20 is calculated for all methods and presented in Table VIII. CLTM_F achieves the best results over the other methods at venues *bar*, *bedroom*, *gym*, *library*, and *office*. CLTM performs better than the other three methods on all venues except *restaurant* and *mall*. The results show the superiority of music concepts-based LTM methods on finding the most suitable songs for venues (**RQ1** and **RQ2**).

Table IX. Precision and Average Precision Comparison across Different Venues on Test Collection 2 (TC2)

	Precision@20					AP@20				
Venue	ABF	ALTM	CBF	CLTM	CLTM_F	ABF	ALTM	CBF	CLTM	CLTM_F
Bar	**0.95**	0.8	0.85	**0.95**	**0.95**	0.845	0.746	0.723	0.906	**0.950**
Bedroom	0.25	0.45	0.45	0.55	**0.65**	0.169	0.245	0.339	0.347	**0.491**
Gym	0.4	0.35	0.45	0.55	**0.65**	0.189	0.179	0.248	0.428	**0.515**
Library	0.35	0.3	0.6	0.6	**0.65**	0.257	0.154	0.344	0.437	**0.452**
Office	0.4	0.35	0.45	0.45	**0.50**	0.175	0.162	0.213	0.235	**0.269**
Restaurant	**0.30**	0.15	0.2	**0.30**	**0.30**	0.083	0.097	0.128	0.097	**0.156**
Mall	0.15	0.25	0.35	0.3	**0.45**	0.024	0.072	0.158	0.102	**0.200**
Bus/Train	0.2	0.2	0.5	0.45	**0.55**	0.067	0.047	0.217	0.359	**0.367**
Mean	0.375	0.356	0.481	0.519	**0.588**	0.226	0.213	0.296	0.364	**0.425**

Table X. NDCG@20 Comparison of Different Methods across Different Venues on Test Collection 2 (TC2)

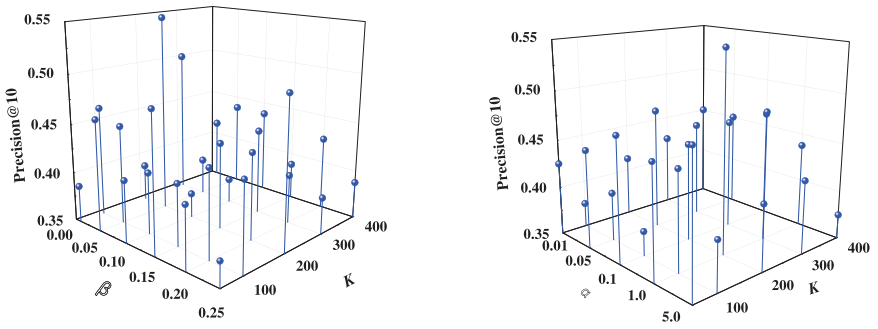
Venue	ABF	ALTM	CBF	CLTM	CLTM_F
Bar	0.91	0.853	0.859	0.955	**0.968**
Bedroom	0.365	0.451	0.555	0.541	**0.700**
Gym	0.393	0.402	0.485	0.619	**0.706**
Library	0.456	0.335	0.533	0.538	**0.625**
Office	0.383	0.389	0.456	0.438	**0.474**
Restaurant	0.251	0.253	0.308	0.272	**0.367**
Mall	0.117	0.227	0.382	0.275	**0.402**
Bus/Train	0.214	0.172	0.425	0.568	**0.586**
Mean	0.386	0.385	0.500	0.526	**0.604**

5.3. Performance Evaluation on TC2

We observed the achieved improvements of CLTM_F on music recommendation for specific venues in TC1, even though TC1 is a weakly labeled dataset.[27] To validate the real performance of the method on a large dataset, we evaluated its performance and compared it with other competitors on TC2. The results returned by each method were carefully evaluated by human subjects. Specifically, the five methods were used to recommend songs from TC2 for the eight venues. The top 20 recommended songs were collected and mingled together to form a single playlist for a venue. The optimal models of ALTM, CLTM, and CLTM_F obtained on TC1 were used. To fairly evaluate whether the songs in a playlist are suitable for the corresponding venue, seven human subjects were recruited (four females, three males), all with different educational background and all from Singapore and China (they are a different set of subjects from the subjects for TC1 annotation). The subjects were required to listen to the recommended songs in the corresponding venues[28] and rate them according to the rules described in Section 4.1.2. Each subject was required to assess each song in all playlists. With the collected ratings, each song in the results of a venues was judged as relevant, neutral, or irrelevant using the same method described in Section 4.1.2. Based on the relevance judgment of each song in the playlists for venues, *Precison@20*, *AP@20,* and *NDCG@20* were computed for each method in each venue. The results are shown in Tables IX and X.

From the results of *Precision@20* and *AP@20*, we can see that CLTM methods (CLTM and CLTM_F) achieve more than 50% recommendation accuracy for *bar*, *bedroom*, *gym*, *library,* and *bus/train*, and show significant improvement over other methods

[27]It is possible that a song suitable for a venue was not labeled for the venue.
[28]We did not specify the exactly location for each venue.



(a) $\alpha = 0.5$ (b) $\beta = 0.01$

Fig. 7. Mean precision@10 of CLTM_F using different hyperparameter values.

in these venues except for *bar*, where all methods can achieve high recommendation accuracy. Compared to CLTM, CLTM_F presents more consistent performance and outperforms other methods across all venues, which implies the necessity of removing noisy concepts. As shown in the Table X, CLTM_F outperforms other methods in all venues on ranking performance, and it achieves significant improvement in *bedroom*, *gym*, *library*, *office,* and *bus*/*train*. The overall performance of CLTM is better than the other three methods, although its performance is not as stable as CLTM_F across different venues. The performances in *restaurant* and *mall* are still unsatisfactory. Because subjects judged the results based on the venues they went to, it is possible that the recommended songs are suitable for other types of malls or restaurants. To further study this problem, it will be necessary to classify the venues by finer granularity, such as specifying the particular types of mall and restaurants.

5.4. Influence of Hyperparameters

There are four hyperparamers in the LTM, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\eta}$. $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ have a smoothing effect on multinomial parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$. $\boldsymbol{\gamma}$ has the same effect on $\boldsymbol{\psi}$ as $\boldsymbol{\alpha}$ on $\boldsymbol{\theta}$. From Equations (4) and (5), we can see that, in the sampling process, the elements of $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\beta}$ become pseudocounts for the corresponding song-topic associations, venue-topic associations, and topic-word associations. Because we do not have prior knowledge on the topic distributions of location and song and the word distribution of topics, $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\beta}$ are set to be symmetric. $\boldsymbol{\gamma}$ and $\boldsymbol{\alpha}$ are set to the same value. The influences of $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\beta}$ on the final performance are associated with the number of topics K. Figure 7 shows the results of average recommended Precision@10 of CLTM_F on TC1 under different parameter settings. In general, it is better to set $\alpha \in \{0.05, 0.1\}$, $\beta \in \{0.01, 0.05\}$, and topic number is set at less than 200. Similarly, $\boldsymbol{\eta}$ has a smoothing effect on the Bernoulli distribution. As shown in Equations (4) and (5), η_0 becomes pseudocount for N_{y_0}, and η_1 becomes pseudocount for N_{y_1}. We tried symmetric and asymmetric settings for η_0 and η_1 in the experiment, but the value of $\boldsymbol{\eta}$ does not affect the final results. Because the term counts in the corpus are very large, as long as we do not set extremely unbalanced values for η_0 and η_1, their influence becomes negligible after only few iterations in Gibbs sampling.

6. CONCLUSION

In this article, we presented a location-aware music recommender system called VenueMusic. This system can effectively recommend suitable songs for common venues in daily life. We detailed an LTM that represents the music profiles of venues in a latent semantic space. A process of generating high-quality music concept sequences

for songs was described. The generated music concept sequences can effectively learn LTM to recommend songs for various types of venues. Two large datasets were constructed to evaluate the performance of our system. Experimental results demonstrate the effectiveness of our system.

As an effective and robust solution for location-aware music recommendation problems, VenueMusic's success provides an impetus for further research on this important topic and opens up a lot of interesting directions for further study. First, we plan to evaluate the system using larger and more complex datasets. Second, it will be interesting to integrate broader venue types into the system and take other kinds of personal context information (e.g., age and gender) into consideration. Another novel research direction is to extend the current system to other kinds of media data (e.g., video and text) and support different music search-related applications (e.g., ranking/reranking and personalization). Last but not least, music choice can be influenced greatly by the events and activities they involve. Therefore, we believe that studying the effects of events and activities and integrating event- and activity-related contexts into the design and development of high-performance location-aware music recommendation systems present very promising directions for scholarly investigation in the future.

APPENDIX

GIBBS SAMPLING DEVIATION FOR LTM MODEL

Please refer to Table II for the notations used in the following presentation. In the Gibbs sampling process of the LTM, we need to assign an indicator and a topic for each word in the corpus. $\boldsymbol{W}$ denotes the generation sequence of words. $\boldsymbol{Y}$ and $\boldsymbol{Z}$ denote the corresponding indicators and topics. The joint distribution can be factored as:

$$P(\boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{W}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\eta}) = P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\beta}) \cdot P(\boldsymbol{Z}, \boldsymbol{Y}|\boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\eta}). \tag{14}$$

Note that, on the right-hand side, the first term is independent of $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\eta}$, and the second term is independent of $\boldsymbol{\beta}$. Both terms of the joint distribution can be handled separately. We first derive the first term, which is the same as that in the standard LDA.

Derivation of $P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\beta})$. Given the association topic z_i of each word w_i in $\boldsymbol{W}$, we can derive that:

$$P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\Phi}) = \prod_{i=1}^{W} P(w_i|\phi_{z_i}) = \prod_{k=1}^{K}\prod_{t=1}^{V} \phi_{k,t}^{N_k^t}, \tag{15}$$

where N_k^t is the number of times that term t is generated by topic k. $\boldsymbol{\Phi}$ is a $K \times V$ matrix, in which each row ϕ_k is the term distributions of topic k over the vocabulary. $P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\beta})$ is obtained by integrating over $\boldsymbol{\Phi}$:

$$\begin{aligned} P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\beta}) &= \int P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\Phi}) P(\boldsymbol{\Phi}|\boldsymbol{\beta}) d\boldsymbol{\Phi} \\ &= \int \prod_{k=1}^{K} \frac{1}{\Delta(\boldsymbol{\beta})} \prod_{t=1}^{V} \phi_{k,t}^{N_k^t+\beta_t-1} d\boldsymbol{\phi_k} \\ &= \prod_{k=1}^{K} \int \frac{1}{\Delta(\boldsymbol{\beta})} \prod_{t=1}^{V} \phi_{k,t}^{N_k^t+\beta_t-1} d\boldsymbol{\phi_k} \\ &= \prod_{k=1}^{K} \frac{\Delta(\boldsymbol{N_k} + \boldsymbol{\beta})}{\Delta(\boldsymbol{\beta})} \end{aligned} \tag{16}$$

where $\boldsymbol{N_k} = \{N_k^t\}_{t=1}^V$. $\Delta(\cdot)$ is a multidimensional extension to the beta function [Heinrich 2005], which is defined as $\Delta(\alpha_1, \alpha_2, \ldots, \alpha_n) = \frac{\prod_{i=1}^n \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^n \alpha_i)}$, where $\Gamma(\cdot)$ is the gamma function.

Derivation of $P(\boldsymbol{Z}, \boldsymbol{Y}|\boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\eta})$. Since $\boldsymbol{Y}$ is only dependent on $\boldsymbol{\eta}$, the joint distribution $P(\boldsymbol{Z}, \boldsymbol{Y}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta})$ can be factorized into:

$$\begin{aligned} P(\boldsymbol{Z}, \boldsymbol{Y}|\boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\eta}) &= P(\boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) P(\boldsymbol{Y}|\boldsymbol{\eta}) \\ &= P(\boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{\psi}, \boldsymbol{\theta}) P(\boldsymbol{\psi}|\boldsymbol{\gamma}) P(\boldsymbol{\theta}|\boldsymbol{\alpha}) P(\boldsymbol{Y}|\boldsymbol{\eta}). \end{aligned} \tag{17}$$

Similar to the derivation of the distribution $P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\Phi})$, we can obtain the distribution $P(\boldsymbol{Y}|\boldsymbol{\eta})$ by integrating over π:

$$P(\boldsymbol{Y}|\boldsymbol{\eta}) = \int P(\boldsymbol{Y}|\pi) P(\pi|\boldsymbol{\eta}) d\pi = \frac{B(\eta_0 + n_{y_0}, \eta_1 + n_{y_1})}{B(\eta_0, \eta_1)}, \tag{18}$$

where $B(\cdot)$ is the beta function.

Analogous to $P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\Phi})$, the topic distribution $P(\boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{\psi}, \boldsymbol{\theta})$ can be derived as follows.

$$\begin{aligned} P(\boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{\psi}, \boldsymbol{\theta}) &= \prod_{i=1}^{W} P(z_i|y_i, \boldsymbol{\psi}, \boldsymbol{\theta}) \\ &= \prod_{i=1}^{W_{y_i}=0} P(z_i|\boldsymbol{\psi}, y_i = 0) \prod_{i=1}^{W_{y_i}=1} P(z_i|\boldsymbol{\theta}, y_i = 1) \\ &= \prod_{l=1}^{L} \prod_{k=1}^{K} P(z_i = k|y_i = 0, \boldsymbol{\psi_l}) \prod_{m=1}^{M} \prod_{k=1}^{K} P(z_i = k|y_i = 1, \boldsymbol{\theta_m}) \\ &= \prod_{l=1}^{L} \prod_{k=1}^{K} \psi_{l,k}^{N_l^k} \prod_{m=1}^{M} \prod_{k=1}^{K} \theta_{m,k}^{N_m^k} \end{aligned} \tag{19}$$

For a music track m labeled with l_m, l_i is set to l_m for every word $w_i, \forall i \in \{1, \ldots, N_m\}$ in this track. In the derivation of Equation (19), the words are divided into two parts: (i) words drawn from the topic distribution of venue l; $W_{y_i=0}$ denotes the total number of words of this part. And (ii) words drawn from the topic distribution of music track m; $W_{y_i=1}$ denotes the total number of words in this part. N_m^k is the number of times that topic k has been observed with a word in the document m; N_l^k is the number of times that topic k has been observed with a word of the documents labeled with venue l. Integrating out $\boldsymbol{\psi}$ and $\boldsymbol{\theta}$, we obtain:

$$\begin{aligned} P(\boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) &= \int P(\boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{\psi}, \boldsymbol{\theta}) P(\boldsymbol{\psi}|\boldsymbol{\gamma}) P(\boldsymbol{\theta}|\boldsymbol{\alpha}) d\boldsymbol{\psi} d\boldsymbol{\theta} \\ &= \int \prod_{l=1}^{L} \frac{1}{\Delta(\boldsymbol{\gamma})} \prod_{k=1}^{K} \psi_{l,k}^{N_l^k + \gamma_k - 1} d\boldsymbol{\psi_l} \prod_{m=1}^{M} \frac{1}{\Delta(\boldsymbol{\alpha})} \prod_{k=1}^{K} \theta_{m,k}^{N_m^k + \alpha_k - 1} d\boldsymbol{\theta_m} \\ &= \prod_{l=1}^{L} \frac{\Delta(\boldsymbol{\gamma} + \boldsymbol{N_l})}{\Delta(\boldsymbol{\gamma})} \prod_{m=1}^{M} \frac{\Delta(\boldsymbol{\alpha} + \boldsymbol{N_m})}{\Delta(\boldsymbol{\alpha})}, \end{aligned} \tag{20}$$

where $\boldsymbol{N_l} = \{N_l^k\}_{k=1}^K$ and $\boldsymbol{N_m} = \{N_m^k\}_{k=1}^K$. The joint probability in Equation (14) becomes:

$$\begin{aligned} P(\boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{W}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \eta) &= P(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{\beta}) \cdot P(\boldsymbol{Z}, \boldsymbol{Y}|\boldsymbol{\alpha}, \boldsymbol{\gamma}, \eta) \\ &= \frac{B(\eta_0 + n_{y_0}, \eta_1 + n_{y_1})}{B(\eta_0, \eta_1)} \prod_{l=1}^{L} \frac{\Delta(\boldsymbol{\gamma} + \boldsymbol{N_l})}{\Delta(\boldsymbol{\gamma})} \prod_{m=1}^{M} \frac{\Delta(\boldsymbol{\alpha} + \boldsymbol{N_m})}{\Delta(\boldsymbol{\alpha})} \prod_{k=1}^{K} \frac{\Delta(\boldsymbol{N_k} + \boldsymbol{\beta})}{\Delta(\boldsymbol{\beta})}. \end{aligned} \tag{21}$$

With the derivation of joint distribution, we can derive the full conditional distribution shown in Equations (4) and (5). We show the steps of deriving Equation (4). Equation (5) can be derived in the same way.

$$\begin{aligned} &P(z_i = k, y_i = 1|\boldsymbol{Z}_{\neg i}, \boldsymbol{Y}_{\neg i}, \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \eta) \\ &= \frac{P(\boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{W})}{P(\boldsymbol{Z}_{\neg i}, \boldsymbol{Y}_{\neg i}, \boldsymbol{W})} = \frac{P(\boldsymbol{W}|\boldsymbol{Z})}{P(\boldsymbol{W}_{\neg i}|\boldsymbol{Z}_{\neg i})P(w_i)} \cdot \frac{P(\boldsymbol{Z}, \boldsymbol{Y})}{P(\boldsymbol{Z}_{\neg i}, \boldsymbol{Y}_{\neg i})} \\ &\propto \frac{B(\eta_0 + N_{y_0}, \eta_1 + N_{y_1})}{B(\eta_0 + N_{y_0}, \eta_1 + N_{y_1,\neg i})} \cdot \frac{\Delta(\boldsymbol{\alpha} + \boldsymbol{N_m})}{\Delta(\boldsymbol{\alpha} + \boldsymbol{N_{m,\neg i}})} \cdot \frac{\Delta(\boldsymbol{\beta} + \boldsymbol{N_k})}{\Delta(\boldsymbol{\beta} + \boldsymbol{N_{k,\neg i}})} \\ &= \frac{\Gamma(\eta_0 + N_{y_0})\Gamma(\eta_1 + N_{y_1})}{\Gamma(\eta_0 + \eta_1 + N_{y_0} + N_{y_1})} \cdot \frac{\Gamma(\eta_0 + \eta_1 + N_{y_0} + N_{y_1,\neg i})}{\Gamma(\eta_0 + N_{y_0})\Gamma(\eta_1 + N_{y_1,\neg i})} \cdot \frac{\prod_{k=1}^K \Gamma(\alpha_k + N_m^k)}{\Gamma\left(\sum_{k=1}^K (\alpha_k + N_m^k)\right)} \\ &\cdot \frac{\Gamma\left(\sum_{k=1}^K (\alpha_k + N_{m,\neg i}^k)\right)}{\prod_{k=1}^K \Gamma(\alpha_k + N_{m,\neg i}^k)} \cdot \frac{\prod_{t=1}^V \Gamma(\beta_t + N_k^t)}{\Gamma\left(\sum_{t=1}^V (\beta_t + N_k^t)\right)} \cdot \frac{\Gamma\left(\sum_{t=1}^V (\beta_t + N_{k,\neg i}^t)\right)}{\prod_{t=1}^V \Gamma(\beta_t + N_{k,\neg i}^t)} \\ &\propto (\eta_1 + N_{y_1,\neg i}) \cdot \frac{\alpha_k + N_{m,\neg i}^k}{\sum_{k=1}^K (\alpha_k + N_{m,\neg i}^k)} \cdot \frac{\beta_t + N_{k,\neg i}^t}{\sum_{t=1}^V (\beta_t + N_{k,\neg i}^t)} \end{aligned} \tag{22}$$

ACKNOWLEDGMENT

We thank Dr. Haiyan Miao for helping to improve our English grammar. The authors would like to thank three anonymous reviewers for their valuable and insightful comments and helpful suggestions.

REFERENCES

Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions in Knowledge Data Engineering* 17, 6 (2005), 734–749.

Anupriya Ankolekar and Thomas Sandholm. 2011. Foxtrot: A soundtrack for where you are. In *Proceedings of Interacting with Sound Workshop: Exploring Context-Aware, Local and Social Audio Applications*. ACM, 26–31.

Linas Baltrunas, Marius Kaminskas, Bernd Ludwig, Omar Moling, Francesco Ricci, Aykan Aydin, Karl-Heinz Lüke, and Roland Schwaiger. 2011. Incarmusic: Context-aware music recommendations in a car. In *Proceedings of the International Conference on Electronic Commerce and Web Technologies*. Springer, 89–100.

David M. Blei. 2012. Probabilistic topic models. *Communications of the ACM* 55, 4 (2012), 77–84.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3 (2003), 993–1022.

Dmitry Bogdanov, Nicolas Wack, Gómez Emilia, Gulati Sankalp, Perfecto Herrera, Oscar Mayor, Gerard Roma, Justin Salamon, José R. Zapata, and Xavier Serra. 2013. Essentia: An audio analysis library for music information retrieval. In *Proceedings of the International Society of Music Information Retrieval*. Citeseer, 493–498.

Matthias Braunhofer, Marius Kaminskas, and Francesco Ricci. 2013. Location-aware music recommendation. *International Journal of Multimedia Information Retrieval* 2, 1 (2013), 31–44.

Judith C. Brown, Olivier Houix, and Stephen McAdams. 2001. Feature dependence in the automatic identification of musical woodwind instruments. *Journal of the Acoustical Society of America* 109, 3 (2001), 1064–1072.

Robin Burke. 2007. Hybrid web recommender systems. In *The Adaptive Web*, P. Brusilovski, A. Kobsa, and W. Nejdl (Eds.). Springer, Chapter 12, 377–408.

Rui Cai, Chao Zhang, Chong Wang, Lei Zhang, and Wei-Ying Ma. 2007. MusicSense: Contextual music recommendation using emotional allocation modeling. In *Proceedings of the ACM International Conference on Multimedia*. ACM, 553–556.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems Technology* 2, 3, Article 27 (2011), 27 pages. http://doi.acm.org/10.1145/1961189.1961199

Zhiyong Cheng and Jialie Shen. 2014. Just-for-me: An adaptive personalization system for location-aware social music recommendation. In *Proceedings of the ACM International Conference on Multimedia Retrieval*. ACM, 185–192.

Zhiyong Cheng and Jialie Shen. 2015. VenueMusic: A venue-aware music recommender system. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 1029–1030.

John Stephen Downie. 2014. MIREX 2014 evaluation results. (2014).

Slim Essid, Gaël Richard, and Bertrand David. 2006. Instrument recognition in polyphonic music based on automatic taxonomies. *IEEE Transactions on Audio, Speech, Language Process.* 14, 1 (2006), 68–80.

Lalya Gaye, Ramia Mazé, and Lars Erik Holmquist. 2003. Sonic city: The urban environment as a musical interface. In *Proceedings of the Conference on New Interfaces for Musical Expression*. National University of Singapore, 109–115.

Alinka E. Greasley and Alexandra M. Lamont. 2006. Music preference in adulthood: Why do we like the music we do. In *Proceedings of the International Conference on Music Perception and Cognition*. Citeseer, 960–966.

Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences* 101, Suppl 1 (2004), 5228–5235.

Negar Hariri, Bamshad Mobasher, and Robin Burke. 2013. Personalized text-based music retrieval. In *Workshops at Proceedings of the AAAI Conference on Artificial Intelligence*. Citeseer, 24–30.

Gregor Heinrich. 2005. Parameter estimation for text analysis. (2005). http://www.arbylon.net/publications/text-est.pdf.

Pengfei Hu, Wenju Liu, Wei Jiang, and Zhanlei Yang. 2014. Latent topic model for audio retrieval. *Pattern Recognition* 47, 3 (2014), 1138–1143.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems* 20, 4 (2002), 422–446.

Marius Kaminskas and Francesco Ricci. 2011. Location-adapted music recommendation using tags. In *Proceedings of the International Conference on User Modeling, Adaption, and Personalization*. Springer, 183–194.

Marius Kaminskas and Francesco Ricci. 2012. Contextual music information retrieval and recommendation: State of the art and challenges. *Computer Science Review* 6, 2 (2012), 89–119.

Marius Kaminskas, Francesco Ricci, and Markus Schedl. 2013. Location-aware music recommendation using auto-tagging and hybrid matching. In *Proceedings of the ACM Conference on Recommender Systems*. ACM, 17–24.

Simon Lacoste-Julien, Fei Sha, and Michael I. Jordan. 2009. DiscLDA: Discriminative learning for dimensionality reduction and classification. In *Proceedings of Advances in Neural Information Processing Systems*. MIT Press, 897–904.

Paul Lamere. 2008. Social tagging and music information retrieval. *Journal of New Music Research* 37, 2 (2008), 101–114.

Alexandra Lamont and Alinka Greasley. 2009. Chapter 15: Music preferences. In *Oxford Handbook of Music Psychology*, Susan Hallam, Ian Cross, and Michael Thaut (Eds.). Oxford University Press.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics* (1977), 159–174.

Olivier Lartillot and Petri Toiviainen. 2007. A matlab toolbox for musical feature extraction from audio. In *International Conference on Digital Audio Effects*. University of Bordeaux 1, 237–244.

Jin Ha Lee and J. Stephen Downie. 2004. Survey of music information needs, uses, and seeking behaviours: Preliminary findings. In *Proceedings of the International Society of Music Information Retrieval*. Universitat Pompeu Fabra, 193–198.

Daniel J. Levitin and James McGill. 2007. *Life Soundtracks: The Uses of Music in Everyday Life*. Technical Report.

Ning-Han Liu, Yi-Hung Wu, and Arbee L. P. Chen. 2005. An efficient approach to extracting approximate repeating patterns in music databases. In *Database Systems for Advanced Applications*. Springer, 240–252.

Beth Logan. 2000. Mel frequency cepstral coefficients for music modeling. In *Proceedings of the International Society of Music Information Retrieval*.

Lie Lu, Hao Jiang, and HongJiang Zhang. 2001. A robust audio classification and segmentation method. In *Proceedings of the ACM International Conference on Multimedia*. ACM, 203–211.

Lie Lu, Dan Liu, and Hong-Jiang Zhang. 2006. Automatic mood detection and tracking of music audio signals. *IEEE Transactions on Audio, Speech, Language Process.* 14, 1 (2006), 5–18.

Benoit Mathieu, Slim Essid, Thomas Fillon, Jacques Prado, and Gaël Richard. 2010. YAAFE, an easy to use and efficient audio feature extraction software. In *Proceedings of the International Society of Music Information Retrieval*. Ghent University, 441–446.

Jon D. Mcauliffe and David M. Blei. 2008. Supervised topic models. In *Proceedings of Advances in Neural Information Processing Systems*. MIT Press, 121–128.

Scott Miller, Paul Reimer, Steven R. Ness, and George Tzanetakis. 2010. Geoshuffle: Location-aware, content-based music browsing using self-organizing rag clouds. In *Proceedings of the International Society of Music Information Retrieval*. Ghent University, 237–242.

Thomas Minka and John Lafferty. 2002. Expectation-propagation for the generative aspect model. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 352–359.

Adrian C. North, David J. Hargreaves, and Jon J. Hargreaves. 2004. Uses of music in everyday life. *Music Perception* 22, 1 (2004), 41–77.

Michael J. Pazzani and Daniel Billsus. 2007. Content-based recommendation systems. In *The Adaptive Web*, P. Brusilovski, A. Kobsa, and W. Nejdl (Eds.). Springer, Chapter 10, 325–341.

Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 248–256.

Sasank Reddy and Jeff Mascia. 2006. Lifetrak: Music in tune with your life. In *Proceedings of the ACM International Workshop on Human-Centered Multimedia*. ACM, 25–34.

Peter J. Rentfrow and Samuel D. Gosling. 2003. The do re mi's of everyday life: The structure and personality correlates of music preferences. *Journal of Personal and Social Psychology* 84, 6 (2003), 1236–1256.

Francesco Ricci. 2012. Context-aware music recommender systems: Workshop keynote abstract. In *Proceedings of the International Conference Companion on World Wide Web*. ACM, 865–866.

Matthew Riley, Eric Heinen, and Joydeep Ghosh. 2008. A text retrieval approach to content-based audio retrieval. In *Proceedings of the International Society of Music Information Retrieval*. Drexel University, 295–300.

Michal Rosen-Zvi, Chaitanya Chemudugunta, Thomas Griffiths, Padhraic Smyth, and Mark Steyvers. 2010. Learning author-topic models from text corpora. *ACM Transactions on Information Systems* 28, 1 (2010), 4.

J. Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative filtering recommender systems. In *The Adaptive Web*, P. Brusilovski, A. Kobsa, and W. Nejdl (Eds.). Springer, Chapter 9, 291–324.

Markus Schedl, Georg Breitschopf, and Bogdan Ionescu. 2014. Mobile music genius: Reggae at the beach, metal on a Friday night? In *Proceedings of the ACM International Conference on Multimedia Retrieval*. ACM, 507–510.

Markus Schedl and Dominik Schnitzer. 2014. Location-aware music artist recommendation. In *Proceedings of the International Conference on MultiMedia Modeling*. Springer, 205–213.

Christian Schörkhuber and Anssi Klapuri. 2010. Constant-Q transform toolbox for music processing. In *Proceedings of the 7th Sound and Music Computing Conference*. 3–64.

Jialie Shen, Meng Wang, Shuicheng Yan, and Peng Cui. 2013. Multimedia recommendation: Technology and techniques. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 1131–1131.

Mark Steyvers, Padhraic Smyth, Michal Rosen-Zvi, and Thomas Griffiths. 2004. Probabilistic author-topic models for information discovery. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 306–315.

Jie Tang, Sen We, Jimeng Sun, and Hang Su. 2012. Cross-domain collaboration recommendation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1285–1293.

George Tzanetakis and Perry Cook. 2002. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing* 10, 5 (2002), 293–302.

Chong Wang, Jinggang Wang, Xing Xie, and Wei-Ying Ma. 2007. Mining geographic knowledge using location aware topic model. In *Proceedings of the ACM Workshop on Geographical Information Retrieval*. ACM, 65–70.

Xinxi Wang, David Rosenblum, and Ye Wang. 2012. Context-aware mobile music recommendation for daily activities. In *Proceedings of the ACM International Conference on Multimedia*. ACM, 99–108.

Hongzhi Yin, Bin Cui, Yizhou Sun, Zhiting Hu, and Ling Chen. 2014. LCARS: A spatial item recommender system. *ACM Transactions on Information Systems* 32, 3 (2014), 11.

Kazuyoshi Yoshii, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno. 2008. An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. *IEEE Transactions on Audio, Speech, Language Process.* 16, 2 (2008), 435–447.

Bingjun Zhang, Jialie Shen, Qiaoliang Xiang, and Ye Wang. 2009a. CompositeMap: A novel framework for music similarity measure. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 403–410.

Bingjun Zhang, Qiaoliang Xiang, Huanhuan Lu, Jialie Shen, and Ye Wang. 2009b. Comprehensive query-dependent fusion using regression-on-folksonomies: A case study of multimodal music search. In *Proceedings of the ACM International Conference on Multimedia*. ACM, 213–222.

Jun Zhu, Amr Ahmed, and Eric P. Xing. 2009. MedLDA: Maximum margin supervised topic models for regression and classification. In *Proceedings of the Annual International Conference on Machine Learning*. ACM, 1257–1264.

Received June 2015; revised September 2015; accepted November 2015