

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/320385043>

Sequence-based context-aware music recommendation

Article in *Information Retrieval* · October 2017

DOI: 10.1007/s10791-017-9317-7

CITATION

1

READS

53

3 authors:



Dongjing Wang

Zhejiang University

11 PUBLICATIONS 38 CITATIONS

[SEE PROFILE](#)



Shuiguang Deng

Zhejiang University

121 PUBLICATIONS 1,148 CITATIONS

[SEE PROFILE](#)



Guandong Xu

University of Technology Sydney

197 PUBLICATIONS 881 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Project

Short text [View project](#)

Sequence-based context-aware music recommendation

Dongjing Wang^{1,2} · Shuiguang Deng¹ · Guandong Xu²

Received: 13 October 2016 / Accepted: 2 October 2017
© Springer Science+Business Media, LLC 2017

Abstract Contextual factors greatly affect users' preferences for music, so they can benefit music recommendation and music retrieval. However, how to acquire and utilize the contextual information is still facing challenges. This paper proposes a novel approach for context-aware music recommendation, which infers users' preferences for music, and then recommends music pieces that fit their real-time requirements. Specifically, the proposed approach first learns the low dimensional representations of music pieces from users' music listening sequences using neural network models. Based on the learned representations, it then infers and models users' general and contextual preferences for music from users' historical listening records. Finally, music pieces in accordance with user's preferences are recommended to the target user. Extensive experiments are conducted on real world datasets to compare the proposed method with other state-of-the-art recommendation methods. The results demonstrate that the proposed method significantly outperforms those baselines, especially on sparse data.

This paper is an extended version of a 2-page poster, namely "Wang D, Deng S, Liu S, and Xu G. Improving Music Recommendation Using Distributed Representation", in the 25th International Conference on World Wide Web, 2016. In this new manuscript, we propose a new embedding learning model as well as three new recommendation approaches, conduct another three sets of experimental evaluations, and also provide the link of the dataset used in our work.

✉ Guandong Xu
Guandong.Xu@uts.edu.au

Dongjing Wang
tokyo1@zju.edu.cn

Shuiguang Deng
dengsg@zju.edu.cn

¹ College of Computer Science and Technology, Zhejiang University, Hangzhou, China

² Advanced Analytics Institute, University of Technology Sydney, Sydney, Australia

Keywords Recommender systems · Context-aware · Sequence-based · Embedding · Neural network

1 Introduction

Nowadays, the developments of the Internet and mobile technology are leading to a rapid growth in the digital music market, and there is an enormous amount of music content available on the Internet. For example, Apple Music offers more than 30 million pieces of music.¹ Besides, people's mobile phones or portable music players, such as iPhone and iPod, usually store more than 1000 pieces of music. Therefore, it is becoming more important than ever to help users find music pieces that meet their requirements. Similar to recommender systems applied in various domains (Linden et al. 2003; Resnick et al. 1994; Lacerda et al. 2015; Forsati et al. 2015), music recommendation (Celma 2010) has greatly benefited from the algorithmic advances of the recommender system community, e.g., collaborative filtering and content-based approaches, which have been predominantly adopted by simply treating music piece as a classic item, e.g., a book or movie, to solve the recommendation problem via user's long-term music preferences.

However, people usually have different preferences and requirements under different contexts, and it has been proven that contextual information like physical surroundings, emotional state, time, presence of other people, past and future events can help recommender systems better understand and satisfy users' real-time requirements (Schedl et al. 2014). Hence, the notion of context-aware recommendation has become the focus of many research works. According to the classification in (Adomavicius and Tuzhilin 2011), there are three types of contexts in recommender systems: completely observable context, partially observable context, and unobservable context. The contextual information is completely observable if all the relevant contexts are known or given explicitly. On the other hand, the contextual information is partially observable if only a part of the knowledge is available. As for the unobservable type of context, no explicit information is available about the contextual factors. In this condition, the recommender system should infer contextual information from available data and knowledge.

In particular, listening to music is a kind of typical context-dependent behaviors because people usually prefer different kinds of music under different contexts (Kaminskas and Ricci 2012). For instance, people usually like loud, energetic music when working out, and enjoy quiet music when resting. However, with the popularity of mobile devices like smartphones, people can listen to music whenever and wherever they want, which makes it difficult to acquire the real-time contextual information directly. Therefore, the context in music recommendation, on which our work focuses, is usually partially observable or even unobservable. However, the context may not be captured with a static set of factors, but rather, it is dynamic and can be inferred from users' interactions with the system. More specifically, users' contextual preferences are reflected in the sequence of music pieces played or liked by them in their current interactions with the system, such as listening sequences and playlists (Hariri et al. 2012). On the one hand, the sequences of music liked or played by users reflect their specific contextual preferences for music during the corresponding period of time. Therefore, it becomes feasible to infer users' contextual preferences from their music listening sequences. On the other hand, the music listening

¹ <http://support.apple.com/en-us/HT204951>.

sequences also reflect the intrinsic features similarity of music pieces and the representations of music pieces can be learned from users' music listening sequences.

In this paper, we present a context-aware music recommendation approach that can infer the users' general and contextual preferences for music from her/his listening records, and recommend music suitable for her/his current requirements. In detail, our approach consists of three steps. Firstly, the proposed approach infers music pieces' latent low dimensional representations (embeddings) from users' music listening sequences using neural network models. In this way, the music pieces that have similar intrinsic features yield similar embeddings. Secondly, users' general and contextual preferences for music are inferred from their complete listening records and her/his current interaction session (music pieces recently listened to by a user) using the learned embeddings. Finally, music pieces that conform to the user's general and contextual preferences are recommended to satisfy her/his real-time requirements. Specifically, compared with existing sequence-based recommendation methods (Rendle et al. 2010; Wang et al. 2015), the proposed approach can (1) capture more co-occurrence information instead of only adjacent relation in the sequences, and (2) exploit both listening sequences and user-music interaction matrix by combining the embedding techniques with collaborative filtering methods, which make it have better performance, especially on sparse data. The main contributions of this paper are summarized as follows:

- We devise models based on neural network to learn the real-valued, low dimensional representations of music pieces from users' music listening sequences.
- We propose a context-aware music recommendation method, which can infer users' general and contextual preferences for music and recommend appropriate music pieces in accordance with their preferences.
- We conduct extensive experiments to evaluate the proposed method on real world datasets, and the results show that our method outperforms baseline methods, especially on sparse data.

The remainder of this paper is structured as follows. Section 2 describes the related works. In Sect. 3, we introduce the motivation for our work. In Sect. 4, we introduce the proposed approaches in detail. Then, experimental evaluations are provided in Sect. 5. Finally, the conclusion and future work are given in Sect. 6.

2 Related work

In this section, we describe related work in context-aware music recommendation, as well as embedding models that also motivated our work.

2.1 Context-aware music recommendation

Previous works on context-aware music recommendation fall into two categories according to the type of the contexts: environment-related context approach and user-related context approach.

Environment-related context based approach Such works are based on the fact that the environment has an influence on users' state of mind or mood, and therefore may influence users' preferences for music (North and Hargreaves 2008). For instance, people tend to listen to different types of music in different seasons (Pettijohn et al. 2010). Consequently,

music recommendation approaches with environment-related parameters perform better than those ones without contextual information. The environment-related contexts include time (Dias and Fonseca 2013), location (Schedl et al. 2014; Cheng and Shen 2016), weather (Park et al. 2006), and hybrid contexts (Knees and Schedl 2013). Dias and Fonseca (2013) incorporated temporal information in session-based collaborative filtering method to improve the performance of music recommendation. Kaminskas et al. (2013) explored the possibility of adapting music to the places of interest (POIs) that the users are visiting. Park et al. (2006) presented a context-aware music recommender system where the environment related contexts include noise, light level, weather, and time. Hariri et al. (2012) adopted an LDA model to infer the topic probability distribution of music with tags, and discovered the pattern of topics in music listening sequences, which can be used as context to improve the performance of music recommendation.

User-related context based approach Compared with environment-related context, user-related context is users' state of mind or mood, which influences their preferences for music directly (Yang and Liu 2013). The user-related context includes activity (Wang et al. 2012), emotional state (Han et al. 2010; Deng et al. 2015; Yoon et al. 2012), and so on. Han et al. (2010) proposed a context-aware music recommendation system, in which music is recommended according to the user's current emotion state and music's influence on users' emotional changes. Deng et al. (2015) presented another emotion-aware music recommendation approach, which can infer users' emotion from her/his microblogs, and then recommend music pieces appropriate for users' current emotion. Yoon et al. (2012) extracted low-level features of music which may trigger human emotions from TV music program's audience rating information and the corresponding music audio data. Then they implemented personalized music recommendation system using selected features, context information and listening history.

2.2 Embedding

The proposed algorithm for learning the effective embeddings of music pieces in this paper can be seen as part of the literature on dimension reduction or representation learning in general (Bengio et al. 2013). In traditional representation learning models, the symbolic data, such as signal, words, and items, are represented as feature vectors using a one-hot representation. The object vectors have the same length as the whole object set, and the position of the observed object in the vector representation is set as one. However, these models suffer from many problems, such as dimension disaster and data sparsity, which limit their practicability to a great extent.

Therefore, neural network models, which can induce low dimensional distributed representation (embedding) of symbolic data by means of neural networks, have been proposed to solve these problems mentioned above. Specifically, embedding is a kind of feature learning technique, where symbolic data are mapped from a space with one dimension per symbolic data object to a continuous vector space with much lower dimension based on training dataset, and the learned low dimensional representation of the object is called its embedding. Note that the learned embeddings can effectively capture item's important relationships and features in training dataset. Especially, in natural language processing (NLP) domain, neural network models have been widely adopted to learn the effective embeddings of words and sentences (Collobert et al. 2011; Bengio et al. 2003). Such models make use of the word ordered in sentences or documents, to explicitly model the assumption that closer words in the word sequences (sentences) are statistically more dependent. Although inefficient training of the neural network models has been an

obstacle to their wider applicability in practical tasks when the vocabulary size may grow to several millions, this problem has been successfully addressed by recent advances in the field, particularly with the development of highly scalable skip-gram (SG) and continuous bag-of-words (CBOW) language models (Mikolov et al. 2013) for learning word representations. These powerful, efficient models have shown very promising results in capturing both semantic and syntactic relationships between words in large-scale text corpora, and obtained state-of-the-art results on many NLP tasks. Recently, the concept of embeddings has been expanded to many applications, including sentences and paragraphs representation learning (Djuric et al. 2015), text similarity measurement (Kenter and de Rijke 2015), document ranking (Nalisnick et al. 2016), information retrieval (Ai et al. 2016), trajectory data mining (Zhou et al. 2016), and recommendation (Wang et al. 2015, 2016).

3 Motivation

Listening to music is a kind of typical context-dependent behaviors because users usually prefer different types of music under different contexts (Kaminskas and Ricci 2012). For example, a user may prefer sad music when he/she is in bad mood and enjoy energetic music when working out. Therefore, contexts play an important role in predicting users' preferences for music and recommending appropriate music pieces. However, with the popularity of mobile devices like smartphones, users can listen to music anytime and anywhere, which makes it difficult to acquire the real-time contexts directly. Fortunately, the contexts may not be captured with a static set of factors, but rather, it is dynamic and can be inferred from users' interactions with the system. More specifically, users' contextual preferences are reflected in the music pieces played or liked by them in their current interactions with the system (Hariri et al. 2012), so it is feasible to infer users' contextual preferences for music from their music listening sequences. On the other hand, the music listening sequences also reflect the intrinsic features similarity of music pieces and the representations of music pieces can be learned from users' music listening sequences. In detail, our work is based on the three following observations from the preliminary analysis of users' listening data and existing works on music recommendation.

Observation 1: *Every user has specific preferences for music, which can be inferred from their listening records* (Celma 2010).

Every user has specific general preferences for music, which are determined by many factors, such as the user's country, gender, age, personality, education, work, and so on. For example, teenagers may enjoy popular or rock music rather than classical music. Moreover, users' general preferences for music can be inferred from their historical listening records, and then recommendation systems can generate music for users according to their general preferences.

Observation 2: *Each user has different contextual preferences for music under different contexts* (Knees and Schedl 2013).

Users' general preferences for music may be diverse and various, but people usually prefer only one or a few kinds of music under a certain context. For example, a user, who likes both light music and rock music, may prefer the former when resting. Therefore, compared with traditional recommendation methods, context-aware recommender systems can generate better results by capturing users' contextual preferences. Although contextual

preferences play an important role in music recommendation, it is usually dynamic and changeable, which makes it hard to acquire directly.

Observation 3: *Users' contextual preferences for music are reflected in their recent listening records* (Hariri et al. 2012).

As mentioned above, users' contextual preferences are usually influenced by different contextual factors such as mood, occasion, social setting, or the task at hand, and the sequences of music liked or played by users reflect their specific contextual preferences for music during the corresponding period of time. Therefore, it becomes feasible to infer users' contextual preferences from their music listening sequences.

Based on the three observations mentioned above, we need a model that can infer and model users' general and contextual preferences for music from users' listening data, and incorporating them into music recommendation. Those three observations become the main motivation for this work.

4 Proposed approach

In this section, we introduce the task formalization of the proposed context-aware music recommendation approach, and then describe the proposed methodology in detail, including music embedding learning, and context-aware music recommendation.

4.1 Formalization

Let $U = \{u_1, u_2, \dots, u_{|U|}\}$ be the user set and $M = \{m_1, m_2, \dots, m_{|M|}\}$ represent the music set, where $|U|$ and $|M|$ denote the total number of unique users and music pieces, separately. For each user u , her/his historical music listening sequence is a list of music records with corresponding timestamps and playing devices: $H^u = \{m_1^u, m_2^u, \dots, m_{|H^u|}^u\}$, where $m_i^u \in M$. The music pieces in each music sequence are ordered according to their playing timestamps. Moreover, u 's listening history H^u can be aggregated into sessions $S^u = \{S_1^u, S_2^u, \dots, S_{|S^u|}^u\}$ according to listening timestamps and playing devices. Here, u 's n th sessions are defined as $S_n^u = \{m_{n,1}^u, m_{n,2}^u, \dots, m_{n,|S_n^u|}^u\}$, where $m_{n,j}^u \in M$. For example,

Table 1 Listening record of user u

ID	Music name and player/singer	Playing time	Playing device	Session ID
m_1^u	Hero-Mariah Carey	2016/09/23 19:52	iPod	S_1^u
m_2^u	Without You-Mariah Carey	2016/09/23 19:56	iPod	S_1^u
m_3^u	My Heart Will Go On-Celine Dion	2016/09/23 20:00	iPod	S_1^u
m_4^u	For Elise-Guns Denver	2016/09/24 14:35	iPod	S_2^u
m_5^u	Voyage A Venice-Denver	2016/09/24 14:40	iPod	S_2^u
m_6^u	Knockin' on Heaven's Door-Guns N' Roses	2016/09/24 15:17	PC	S_3^u
m_7^u	Numb-Linkin Park	2016/09/24 15:22	PC	S_3^u
m_8^u	Don't Cry-Guns N' Roses	2016/09/24 15:27	PC	S_3^u

as shown in Table 1, u 's listening sequence is composed of 8 music pieces and corresponding timestamps and playing devices. Obviously, the top three items can be aggregated into one session. Although the playing timestamps of the other five music pieces are close to each other, they are played on two different devices, namely iPod and PC. Therefore, they are aggregated into two different sessions. More formally, the session of u is $S^u = \{S_1^u, S_2^u\}$, where $S_1^u = \{m_1^u, m_2^u, m_3^u\}$, $S_2^u = \{m_4^u, m_5^u\}$, and $S_3^u = \{m_6^u, m_7^u, m_8^u\}$.

Given user u 's music listening sequence, the task becomes recommending music that u would probably enjoy at present. More specifically, there are two challenges here: (1) how to infer and model users' general and contextual preferences for music from their listening records; (2) how to incorporate these preferences into recommendation to satisfy users' current requirements. To address these challenges, we first devise two models based on neural network for learning the low dimensional representations (embeddings) of music pieces. Then users' general and contextual preferences can be inferred by aggregating the embeddings of music pieces in their complete listening records and current interaction session, respectively. Finally, four embedding based recommendation approaches are proposed to recommend appropriate music.

4.2 Music embedding learning

music2vec The music2vec is an model based on neural network that can learn the embeddings of music from user's complete music listening sequences. The graphical representation of this model is shown in Fig. 1. In this model, the embeddings of music are inferred using a skip-gram model (Mikolov et al. 2013) by maximizing the objective function over all music listening sequences in users' listening records. The key idea behind music2vec is that the sequences of music liked or played by people often reflect their specific preferences for music during the corresponding period of time, and the co-occurrences of music pieces in users' music listening sequences indicate the similarity or relevance between music pieces. For example, if there is a music listening sequence " $a \rightarrow b \rightarrow c \rightarrow d$ " and the context window size is set as 3, then music piece a co-occur with b or c (not d) in this listening sequence. In other words, music pieces with similar intrinsic features tend to be listened to together with the same music pieces, and should be represented closely in the low dimensional embedding space. In short, music2vec learns the embedding of music piece m_i^u from the neighbor music pieces $\{m_{i-c}^u : m_{i+c}^u\} \setminus m_i^u$ before

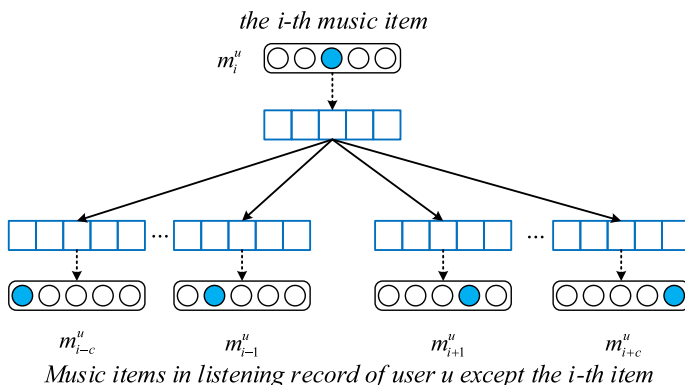


Fig. 1 The skip-gram model in music2vec

and after m_i^u in u 's complete listening sequence H^u . More formally, the objective function of music2vec is defined as follows:

$$L = \sum_{u \in U} \sum_{m_i^u \in H^u} \sum_{-c \leq j \leq c} \log p(m_{i+j}^u | m_i^u), \quad (1)$$

where c decides the size of the context window ($2c + 1$) for music listening sequences. Larger c results in more training examples, which can lead to higher accuracy at the cost of more training time. $p(m_{i+j}^u | m_i^u)$ represents the conditional probability of observing a neighbor music piece m_{i+j}^u given the current music piece m_i^u in H^u , which is formally defined using the soft-max function as follows:

$$p(m_{i+j}^u | m_i^u) = \exp(\mathbf{v}_{m_i^u}^T \cdot \mathbf{v}'_{m_{i+j}^u}) / \sum_{m \in M} \exp(\mathbf{v}_{m_i^u}^T \cdot \mathbf{v}'_m), \quad (2)$$

where \mathbf{v}_m and \mathbf{v}'_m are the input and output embeddings of music piece m , respectively, and M is the set of all music pieces. From (1) and (2), we can see that music2vec try to learn the embedding of music piece according to its neighbor music pieces in users' listening sequences, and music pieces with similar intrinsic features tend to have similar neighbor music pieces and yield similar embeddings. According to Observation 1, users' general preferences for music may be relatively diverse and various. However, music2vec does not explicitly take into account that users' complete listening sequences may contain many kinds of music, so it can only be used to represent music at user level. Therefore, we introduce a finer-grain session-based version of music2vec, namely session-music2vec, to represent music at session level.

session-music2vec As mentioned above, a user's complete listening sequence may contain music of many styles, but the user may be only interested in one kind of music during a period of time (Observation 2). To account for this fact, we propose a modified session-based neural network model. As shown in Fig. 2, this session-based neural network model tries to learn the embeddings using a skip-gram model (Mikolov et al. 2013) at session level instead of user level. Similar to music2vec, the key idea behind session-music2vec is that the sequences of music liked or played by people often reflect their specific contextual music preferences, and the co-occurrences of music pieces in users'

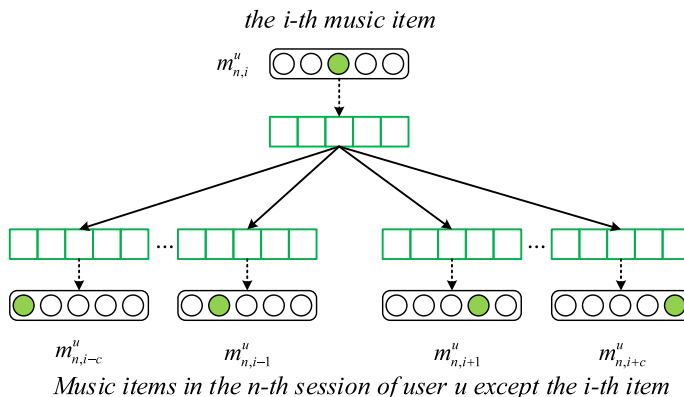


Fig. 2 The skip-gram model in session-music2vec

music listening sessions indicate the similarity or relevance between music pieces. In a word, this model attempts to learn the embeddings of music piece $m_{n,i}^u$ from the neighbor music pieces $\{m_{n,i-c}^u : m_{n,i+c}^u\} \setminus m_{n,i}^u$ before and after $m_{n,i}^u$ in u 's n th session S_n^u . More formally, the objective function of session-music2vec is defined as follows:

$$L = \sum_{u \in U} \sum_{S_n^u \in S^u} \sum_{m_{n,i}^u \in S_n^u} \sum_{-c \leq j \leq c} \log p(m_{n,i+j}^u | m_{n,i}^u), \quad (3)$$

where c is the length of the context window for music listening sessions. Similarly, $p(m_{n,i+j}^u | m_{n,i}^u)$ represents the conditional probability of observing a neighbor music piece $m_{n,i+j}^u$ given the current music piece $m_{n,i}^u$ in a certain session S_n^u , which is defined using the soft-max function as follows:

$$p(m_{n,i+j}^u | m_{n,i}^u) = \exp(\mathbf{v}_{m_{n,i}^u}^T \cdot \mathbf{v}'_{m_{n,i+j}^u}) / \sum_{m \in M} \exp(\mathbf{v}_{m_{n,i}^u}^T \cdot \mathbf{v}'_m), \quad (4)$$

where \mathbf{v}_m and \mathbf{v}'_m are the input and output embeddings of music m respectively, and M is the set of all music pieces.

Generally, a user may have an interest in music pieces of several styles, but usually tends to listen to music of specific styles within a session. Therefore, session-music2vec is better at learning the representations of music at the fine-grained level.

Learning In the learning phase, we need to maximize the objective functions of the log probability defined in (1) and (3) over users' complete listening sequences and sessions, separately. However, the complexity of computing the corresponding soft-max functions defined in (2) and (4) is proportional to the music set size $|M|$, which can reach millions easily. Two approaches of computationally efficient approximation of the full soft-max functions are negative sampling (Mikolov et al. 2013) and hierarchical soft-max (Morin and Bengio 2005). In this paper, we adopt negative sampling to compute the objective function, which approximates the original soft-max functions defined in (2) and (4) with the following formulae respectively:

$$p(m_{i+j}^u | m_i^u) = \log \sigma(\mathbf{v}_{m_i^u}^T \cdot \mathbf{v}'_{m_{i+j}^u}) + k \cdot E_{m_n \sim P_M} [\log \sigma(-\mathbf{v}_{m_n}^T \cdot \mathbf{v}'_{m_{i+j}^u})], \quad (5)$$

$$p(m_{n,i+j}^u | m_{n,i}^u) = \log \sigma(\mathbf{v}_{m_{n,i}^u}^T \cdot \mathbf{v}'_{m_{n,i+j}^u}) + k \cdot E_{m_n \sim P_M} [\log \sigma(-\mathbf{v}_{m_n}^T \cdot \mathbf{v}'_{m_{n,i+j}^u})], \quad (6)$$

where $\sigma(x) = 1/(1 + e^{-x})$, k is the number of negative samples, and m_n is the sampled music piece drawn according to the noise distribution P_M , which is modeled by empirical unigram distribution over all music pieces. Then stochastic gradient descent algorithm is used to maximize the optimized objective functions represented by (5) and (6). In each step, a pair of music pieces $(m_i; m_j)$ is sampled from music playing sequences, and meanwhile multiple negative pairs of music pieces $(m_n; m_j)$ are sampled from a noise distribution P_M . Specifically, for a sampled pair $(m_i; m_j)$, the gradient with respect to the embedding vector \mathbf{v}_{m_i} of music m_i will be calculated as:

$$\frac{\partial L}{\partial \mathbf{v}_{m_i}} \propto \frac{\partial \log p(m_j|m_i)}{\partial \mathbf{v}_{m_i}} \quad (7)$$

Note that the initial learning rate for the stochastic gradient descent algorithm is set as 0.025 and it will gradually be lowered as the training process repeats on the music dataset. Finally, each music piece can be represented with two kinds of embeddings, which are learned from users' complete listening sequences and sessions, respectively. Note that the embeddings are vectors of real numbers, and the dimension of the embedding decides its capacity of representing music. In other words, higher-dimension embeddings can capture more features of music pieces better at the cost of lower efficiency.

4.3 Context-aware music recommendation

As mentioned above, music2vec can learn the embeddings of music from users' complete listening records at user/general level, which is suitable for inferring and modeling users' general preferences for music. Session-music2vec learns the embeddings of music from aggregated sessions at session/contextual level, which can be used to infer and model users' contextual preferences for music. Specifically, we can infer and model the users' general preferences by aggregating the embeddings of music pieces in their listening records using music2vec. Similarly, users' contextual preferences can be learned from the embeddings of music pieces in their sessions using session-music2vec. Formally, as for a user u with complete listening sequences $H^u = \{m_1^u, m_2^u, \dots, m_{|H^u|}^u\}$, her/his general preference for music is defined as:

$$\mathbf{p}_g^u = \sum_{m_i^u \in H^u} \mathbf{v}_{m_i^u}^{m2v} / |H^u|, \quad (8)$$

where \mathbf{v}_m^{m2v} is the embedding of music piece m learned by music2vec.

Similarly, given u 's current session $S_n^u = \{m_{n,1}^u, m_{n,2}^u, \dots, m_{n,|S_n^u|}^u\}$, her/his contextual preference is defined as:

$$\mathbf{p}_c^u = \sum_{m_{n,i}^u \in S_n^u} \mathbf{v}_{m_{n,i}^u}^{s2v} / |S_n^u|, \quad (9)$$

where \mathbf{v}_m^{s2v} is the embedding of music m in current interaction session learned by session-music2vec.

With the embeddings of all music pieces, along with users' general and contextual preferences for music, we propose four embeddings based recommendation methods.

Music2vec-TopN (M-TN) Given a user u along with her/his general preference \mathbf{p}_g^u and listening record H^u , M-TN calculates cosine similarities between \mathbf{p}_g^u and all music pieces in the music set M using the embeddings learned by music2vec, and recommends the top N most similar music pieces to u . Formally, the predicted preference (pp) of the target user u for music piece m is defined as follows:

$$pp_{M-TN}(u, m) = \cos(\mathbf{p}_g^u, \mathbf{v}_m^{m2v}). \quad (10)$$

SessionMusic2vec-TopN (SM-TN) In this approach, we only incorporate the contextual preferences from current interaction session into recommendation. Specifically, SM-TN

calculates cosine similarities between u 's contextual preference \mathbf{p}_c^u and all music pieces in the music set M using the embeddings that are learned with session-music2vec, and recommends the top N most similar music pieces to u . The predicted preference of the target user u for music piece m is defined as:

$$pp_{SM-TN}(u, m) = \cos(\mathbf{p}_c^u, \mathbf{v}_m^{s2v}). \quad (11)$$

In fact, a user's preference for a certain music piece is determined by both her/his general and contextual preferences. Therefore, we propose another two context-aware recommendation approaches that incorporate both preferences.

Context-Session-Music2vec-TopN (CSM-TN) This method takes both u 's general and contextual preferences into consideration, and the predicted preference of the target user u for music piece m is defined as follows:

$$pp_{CSM-TN}(u, m) = \cos(\mathbf{p}_g^u, \mathbf{v}_m^{m2v}) + \cos(\mathbf{p}_c^u, \mathbf{v}_m^{s2v}), \quad (12)$$

where \mathbf{p}_g^u and \mathbf{p}_c^u are u 's general and contextual preferences for music, respectively, and \mathbf{v}_m^{m2v} and \mathbf{v}_m^{s2v} are the embeddings of music learned by music2vec and session-music2vec, separately.

Context-Session-Music2vec-UserKNN (CSM-UK) In this approach, we combine the learned embeddings of music using music2vec and session-music2vec with a traditional user-based collaborative method (UserKNN) (Resnick et al. 1994). Specifically, the similarity between users is calculated as follows:

$$sim(u, v) = \sum_{m \in M_u \cap M_v} 1 / \sqrt{|M_u| \times |M_v|} + \cos(\mathbf{p}_g^u, \mathbf{p}_g^v), \quad (13)$$

where u and v are two users, M_u and M_v are all music pieces listened to by u and v , separately, \mathbf{p}_g^u and \mathbf{p}_g^v are the general preferences of u and v , separately, and $\cos(\mathbf{p}_g^u, \mathbf{p}_g^v)$ is cosine similarity between \mathbf{p}_g^u and \mathbf{p}_g^v . The predicted preference of the target user u for music piece m is defined as follows:

$$pp_{CSM-UK}(u, m) = \sum_{v \in U_{u,K} \cap U_m} sim(u, v) / |U_{u,K} \cap U_m| + \cos(\mathbf{p}_c^u, \mathbf{v}_m^{sm2v}), \quad (14)$$

where u is the target user, $U_{u,K}$ is the set of top K users similar to u , U_m is the set of users who have listened to music piece m , \mathbf{p}_c^u is u 's contextual preference for music and \mathbf{v}_m^{s2v} is the learned embedding of music m using session-music2vec, and $\cos(\mathbf{p}_c^u, \mathbf{v}_m^{s2v})$ is cosine similarity between two vectors \mathbf{p}_c^u and \mathbf{v}_m^{s2v} .

Finally, the ranking of music pieces $>_{u, \mathbf{p}_g^u, \mathbf{p}_c^u}$ in our approach is defined as:

$$m_i >_{u, \mathbf{p}_g^u, \mathbf{p}_c^u} m_j : \Leftrightarrow pp(u, m_i) > pp(u, m_j), \quad (15)$$

where pp is the predicted preference of the target user u for music piece m . Then music pieces with high ranking scores are recommended to the target user.

5 Experiments

In this section, we experimentally evaluate the performance of the proposed recommendation methods. In detail, we first introduce dataset, baselines, experimental designs, and parameter settings. Then, we give an illustration of the learned embeddings using a visualization tool. Next, we evaluate the performance of different embedding-based recommendation methods and present valuable insights regarding how users' preferences and the embeddings' dimension can affect the performance of the proposed approaches. This is followed by a subsection on comparisons between our method and baselines. Finally, we investigate how the proposed method and baselines perform on datasets with different sparsities.

All experiments are performed on an Intel Core i3-2120 based PC running at 3.30 GHz, which has 8 GB of memory and operates on 64-bit Windows 8 operating system.

5.1 Dataset

To evaluate the proposed approach, we crawl 4,284,000 music listening records² of 4284 users from an online music service website named Xiami Music.³ On average, every user has 1000 listening records. The statistics of the dataset is shown in Table 2. Moreover, Fig. 3 illustrates the relationship between playing count (k) and the number of music pieces being played k times. We can see that, only minority music pieces are very popular and the majority of music are not popular, which conforms to the Long Tail Theory (Adamic and Huberman 2000).

5.2 Baseline methods

In last two decades, many algorithms have been proposed for *top-N* recommendation on binary data without rating, among which collaborative filtering (Linden et al. 2003; Resnick et al. 1994) is one of the most famous algorithms. Besides, *top-N* recommendation is actually a ranking problem. Therefore, six state-of-the-art recommendation algorithms, including temporal recommendation based on injected preference fusion (IPF) (Xiang et al. 2010), Bayesian personalized ranking (BPR) (Rendle et al. 2009), FISMauc (FISM) (Kabbur et al. 2013), factorizing personalized Markov chains (FPMC) (Rendle et al. 2010), hierarchical representation model (HRM) (Wang et al. 2015) together with a user-based collaborative filtering method (UserKNN) (Resnick et al. 1994) are used as baselines.

5.3 Experiment designs and evaluation metrics

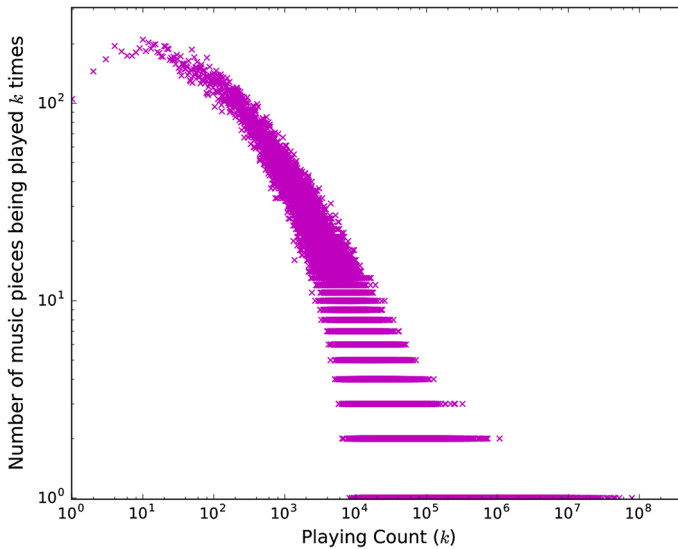
The goal of this experiment is to evaluate the performance of different recommendation approaches in making good recommendations given the users' listening sequences. Therefore, we split the whole dataset into training set and test set according to the idea of 5-fold cross-validation. In detail, we keep the complete listening records of the 80% users and the first half of each session (training session) of the remaining 20% users as the

² Dataset link: <https://1drv.ms/f/s!ApojZBGe9UzXgaI6x8pBf8JgN4PfZg>.

³ <http://www.xiami.com>.

Table 2 Statistics of dataset

#Users	#Songs	#Listening	#Listening per user	#Listening per song
4284	361,861	4,284,000	1000	11.8


Fig. 3 Playing count analysis of dataset

training set, and the following half of each session (testing session) of the remaining 20% users as the testing set. Specifically, the proposed approach infers a test user's general preference from all her/his training sessions, and then infers the test user's contextual preference from part of each of her/his testing session and generates recommendation according to both general and contextual preferences for music. The performance is evaluated for each user u on the testing session in the testing set. For each recommendation method, we generate a list of N music pieces for each user u , denoted by $R(u)$. The following four metrics (Baeza-Yates and Ribeiro-Neto 1999) are used to evaluate recommendation approaches.

HitRate is the fraction of hits, which means the recommendation list contains at least one music piece that the user u is interested in. For example, as for a listening record (u, m) in the test data, if the recommended list for u contains m , then it is a hit. The definition is given as follows:

$$\text{HitRate} = \#(\text{hits}) / \#(\text{recs})$$

where $\#(\text{hits})$ is the number of hits and $\#(\text{recs})$ is the number of recommendations.

Precision, Recall, and F1 Score Precision (also called positive predictive value) is the fraction of recommended music pieces that are relevant, and recall (also known as sensitivity) is the fraction of relevant music pieces that are recommended. F1 score is the harmonic mean of precision and recall. Formally, the definitions are given as follows:

$$\begin{aligned}
 \textit{Precision} &= |R(u) \cap T(u)|/|R(u)| \\
 \textit{Recall} &= |R(u) \cap T(u)|/|T(u)| \\
 F1 &= 2 \times \textit{Precision} \times \textit{Recall}/(\textit{Precision} + \textit{Recall})
 \end{aligned}$$

where $R(u)$ is the recommended music pieces and $T(u)$ is the music pieces that u has listened to in the test data.

5.4 Parameter settings

The detailed configurations and descriptions of the parameters in music2vec and session-music2vec are given in Table 3. Note that all the hyper-parameters are optimized on an independent validation set.

5.5 Illustration of the learned embedding

In order to show what the learned embeddings look like, some illustrations of the learned embeddings are given before the quantitative evaluations of our approach.

5.5.1 Illustrations of artists' embeddings

We firstly analyze the embeddings of some selected artists' music pieces with t-SNE (Maaten and Hinton 2008), which can effectively visualize high dimension data. More specifically, Table 4 shows several well-known artists and their tag information, and Fig. 4 shows the embeddings of top 10 music pieces played/sung by each artist in 2-dimensional space with t-SNE. From the results, we can draw two interesting conclusions.

Firstly, it is interesting to observe most music pieces of the same artist cluster tightly, and music pieces that are sung/played by artists of similar genre lie nearby in the 2-dimensional space. For example, at the bottom left of Fig. 4, the music pieces of Yuki Kajiura and Joe Hisaishi (9 and 10 in Table 4) form two obvious clusters. Besides, Gun N' Rose, Bon Jovi, and Bob Dylan (1, 3, and 4 in Table 4) are three famous rock singers, and the embeddings of their music pieces are very close to each other in the 2-dimensional space. The reason is that similar musicians' music pieces usually have similar genres, and music pieces of specific genres tend to be listened to by users that have similar general/contextual preferences. In other words, similar music pieces tend to appear in the same playing sequences. For example, a piece of rock music is likely to appear in the playing sequences of rock fans instead of classical music fans. Furthermore, these co-occurrences that reflect the features of music can be captured by our approach to learn the embeddings of music.

Secondly, some slight differences in styles are also reflected in the learned embeddings, which further demonstrates the effectiveness of the proposed approach. For example, music pieces of Yuki Kajiura, who is a Japanese instrumental soundtrack musician, are not very close to pieces of music sung/played by another soundtrack masters, Joe Hisaishi. The reason is that Yuki Kajiura's soundtracks have j-pop styles, while Joe Hisaishi mainly focuses on composing soundtracks with new age and classical styles.

Table 3 Parameter settings for training music2vec and session-music2vec

Parameter	Range of candidate values	Optimal value	Description
Window size ($2c + 1$)	[3–9]	5	The number of the music pieces in the context
Negative sample (k)	[1–30]	20	The number of “noise items” should be drawn (in order to increase the efficiency of training progress)
Down sample	[1e-5–1e-2]	1e-3	Higher frequency music pieces are randomly down sampled
α	[0.01–0.1]	0.025	The initial learning rate
iter	[1–10]	5	Number of iterations (epochs) over the corpus

Table 4 Basic information of some famous artists

No.	Artist	Tags in last.fm
1	Guns N' Roses	Rock, hard rock, classic rock, metal, 80s
2	Maroon 5	Pop, rock, pop rock, alternative, alternative rock
3	Bon Jovi	Rock, hard rock, classic rock, hair metal, 80s
4	Bob Dylan	Folk, rock, folk rock, classic rock, songwriter, 60s
5	Justin Bieber	Pop, rnb, r and b, hip-hop, black metal
6	Lady Gaga	Pop, dance, electronic, epic, female vocalists
7	Adele	Pop, soul, British, songwriter, female vocalists
8	Mariah Carey	Pop, rnb, soul, female vocalists, 90s
9	Joe Hisaishi	Sound track, Japanese, instrumental, anime, classical, piano
10	Yuki Kajiura	Sound track, Japanese, instrumental, anime, j-pop

5.5.2 Illustrations of users embeddings

While the visualization in Fig. 4 provides interesting qualitative insights about artists, we now provide a further quantitative display of some selected users. Figure 5 gives the visualization of the embeddings of music pieces in different users' sessions. From the results, we can draw two conclusions. Firstly, the music pieces listened to by each user form one or several clusters, which shows that users have different general preferences for music, and they usually enjoy one or several specific kinds of music (Observation 1). For example, user1 has relatively focused preferences while user2 has a broader range of interests. Secondly, the music pieces in each session cluster tightly, which shows that each user has different contextual preferences for music under different contexts (Observation 2–3).

In conclusion, the illustrations confirm our observations mentioned in Sect. 3 and show that embeddings learned by our method from music listening sequences depict the intrinsic features of music pieces effectively. On the other hand, the illustration also shows that the learned embeddings are useful for many other tasks, such as similarity measure, corpus visualization, automatic tagging, and classification.

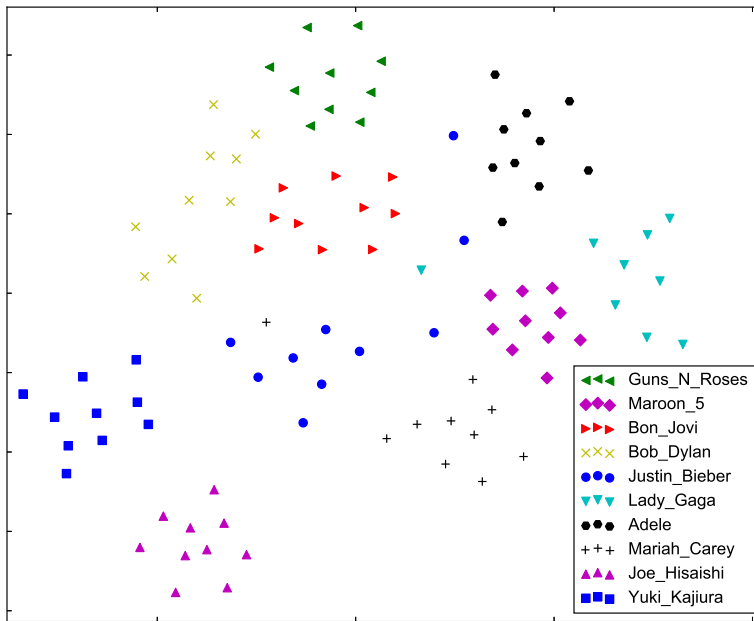


Fig. 4 Visualization of the embeddings of selected musicians' pieces

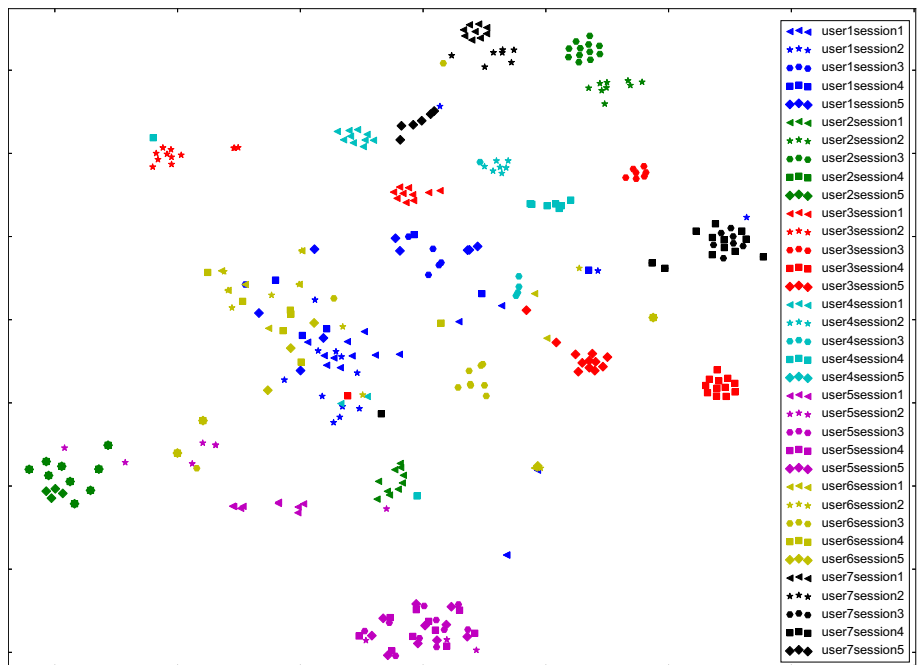


Fig. 5 Visualization of the embeddings of selected users' pieces

5.6 Comparison among embedding-based methods

We first empirically compare the performance of four proposed embedding-based recommendation methods, referred to as Music2vec-TopN (M-TN), Session-Music2vec-TopN (SM-TN), Context-Session-Music2vec-TopN (CSM-TN), and Context-Session-Music2vec-UserKNN (CSM-UK), separately. The results are shown in Fig. 6.

We have the following observations from the results: (1) M-TN, which only considers users' general preferences for music, performs the worst among all four methods. It indicates that users' general preferences are not the only factor in determining users' real-time requirements. (2) SM-TN, which only incorporates user's contextual preferences for music, has better performance than M-TN, and the relative performance improvement in term of precision by SM-TN is around 70%. However, SM-TN is not as good as the other two methods, especially CSM-UK. The reason is that users' general preferences also has an important influence on users' current preferences for music, although it is not the only factor. (3) CSM-TN and CSM-UK, which incorporate both users' general and contextual preferences for music, achieve better performance than the other two methods. As for CSM-TN, its performance is only slightly better than SM-TN, whose improvement is no more than 5%. The reason is that the combination of users' general and contextual preferences for music in CSM-TN is a simple weighted linear addition, which may be not reasonable enough. Moreover, CSM-UK has the best performance in all four embeddings

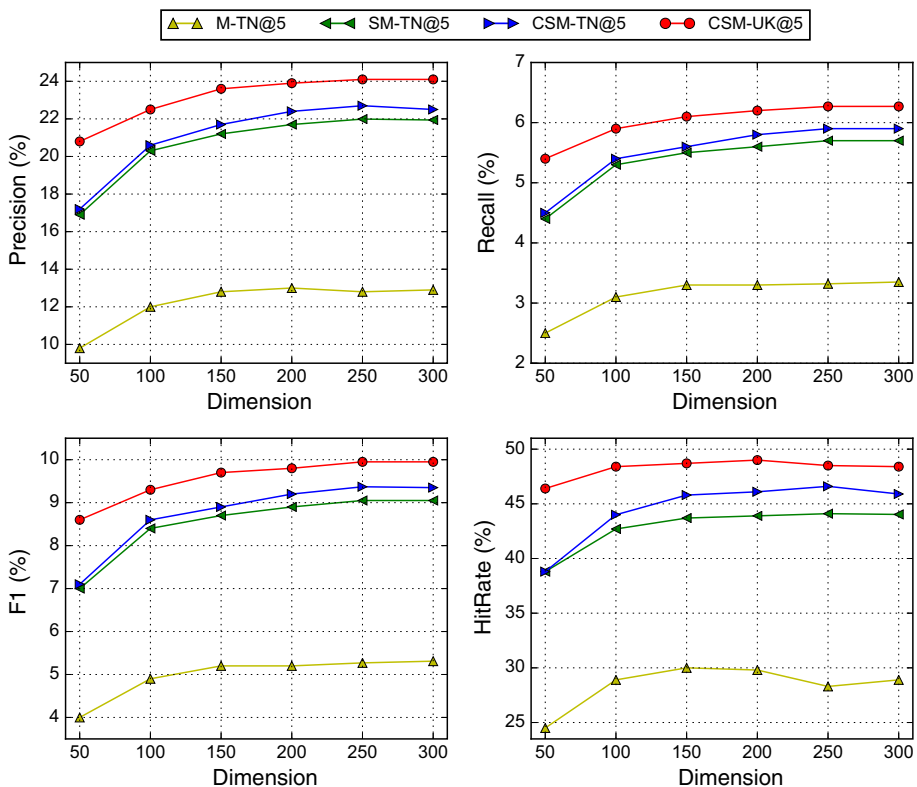


Fig. 6 Top-5 performance comparison among different embedding-based recommendation methods

based recommendation methods. Take the F1 score as an example, when compared with M-TN, SM-TN, and CSM-TN with dimensionality set as 50, the relative performance improvement by CSM-UK is around 115.0, 22.9, and 21.1%, respectively. One reason is that the combination of users' general and contextual preferences for music is more effective than the other methods, especially CSM-TN. The second reason is that CSM-UK utilizes similar users' preferences to assist recommendation. (4) As the dimension of embeddings gets high, the performances of all methods tend to get better, and the performance tends to be stable when the dimension exceeds 200. The reason is that the embeddings of higher dimension can capture more features and depict music pieces better at the cost of lower efficiency or even over-fitting. Finally, the dimension of the embeddings is set to 200 in consideration of both accuracy and efficiency.

5.7 Comparison with baselines

We further compare our methods with six state-of-the-art baseline methods, including hierarchical representation model (HRM), factorizing personalized Markov chains (FPMC), temporal recommendation based on injected preference fusion (IPF), Bayesian personalized ranking (BPR), FISMauc (FISM), and a user-based collaborative filtering method (UserKNN). For the sake of brevity, we compare the best performed CSM-UK with all baselines. The results are shown in Fig. 7.

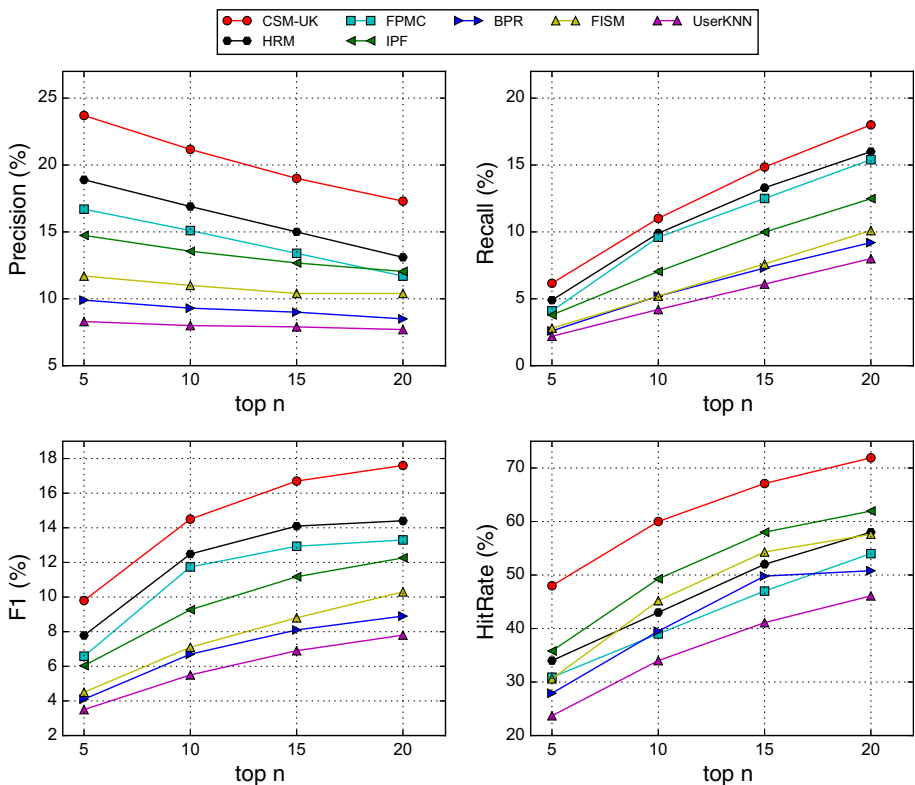


Fig. 7 Performance comparison with baselines

We have the following observations from the results. (1) Our method has the best performance. Take the F1 score as an example, when compared with HRM, FPMC, IPF, BPR, FISM, and UserKNN with the recommending number n set as 20, the relative performance improvements by CSM-UK are around 22.2, 32.4, 43.5, 97.8, 70.9, and 125.6%, respectively. The improvements show the effectiveness of our approach in learning the embeddings of music from playing sequences and inferring users' preferences for music as well as performing context-aware music recommendation. Especially, the proposed approach is better than HRM and FPMC because our approach can capture more co-occurrence information instead of only adjacent relation in the sequences, and fully exploit both listening sequences and user-music interaction matrix by combining the embedding techniques with collaborative filtering methods. (3) IPF performs better than BPR, FISM, and UserKNN because it incorporates both users' general preferences and contextual preferences for music, while other baselines (BPR, FISM, and UserKNN) only consider users' general preferences. However, IPF is not as good as our method. The reason is that our methods can fully utilize playing sequences and incorporate contextual information in a more effective way. Besides, the high sparsity of this dataset (99.72%) may result in the bad performance of baselines. Therefore, we further conduct a series of experiments on datasets with different sparsities in the next subsection. (4) The hitrate and recall for all the three strategies increase but the precision decreases when n is increasing. These results are in accordance with the intuitive and common sense. It requires system developer to select the proper n in order to balance the performances of hit-rate/recall and precision.

In conclusion, our method can effectively infer users' general and contextual preferences for music and incorporate both the user's general and contextual preferences into music recommendation to satisfy their real-time requirements.

5.8 Impact of data sparsity

To investigate the proposed method's ability to handle sparse data, we further evaluate our method and the baseline methods on datasets with different sparsities. Specifically, the datasets with different sparsities are generated by removing music pieces that have been played less than k_m times, where k_m are set to $\{0, 5, 10, 15, 20\}$. The results are shown in Fig. 8.

From the results, we can see that: (1) our method still has the best performance. Take the F1 score as an example, when compared with HRM, FPMC, IPF, BPR, FISM, and UserKNN with the sparsity being 97.94%, the relative performance improvements by CSM-UK are around 11.7, 38.5, 13.7, 68.9, 56.1, and 171.1%, respectively. This result proves our methods can handle sparse data in a more efficient way. Besides, it also verifies the importance of music playing sequences and users' contextual preferences. (2) With the sparsity increasing, all seven methods' performance shows obviously decreasing trends. However, the performance gap between baselines, especially the ones without considering music playing sequences (IPF, BPR, FISM, and UserKNN), and CSM-UK also gets larger. Take the F1 score as an example again, when compared with HRM, FPMC, IPF, BPR, FISM and UserKNN with the sparsity being 99.72%, the relative performance improvements by CSM-UK are around 25.9, 48.9, 62.4, 139.0, 117.8 and 180%, separately. This is because CSM-UK depends on both music listening sequences and user-music interaction matrix to perform recommendation, and it is less sensitive to the sparsity of user-music dataset. In brief, our method can handle sparse data better than baseline methods.

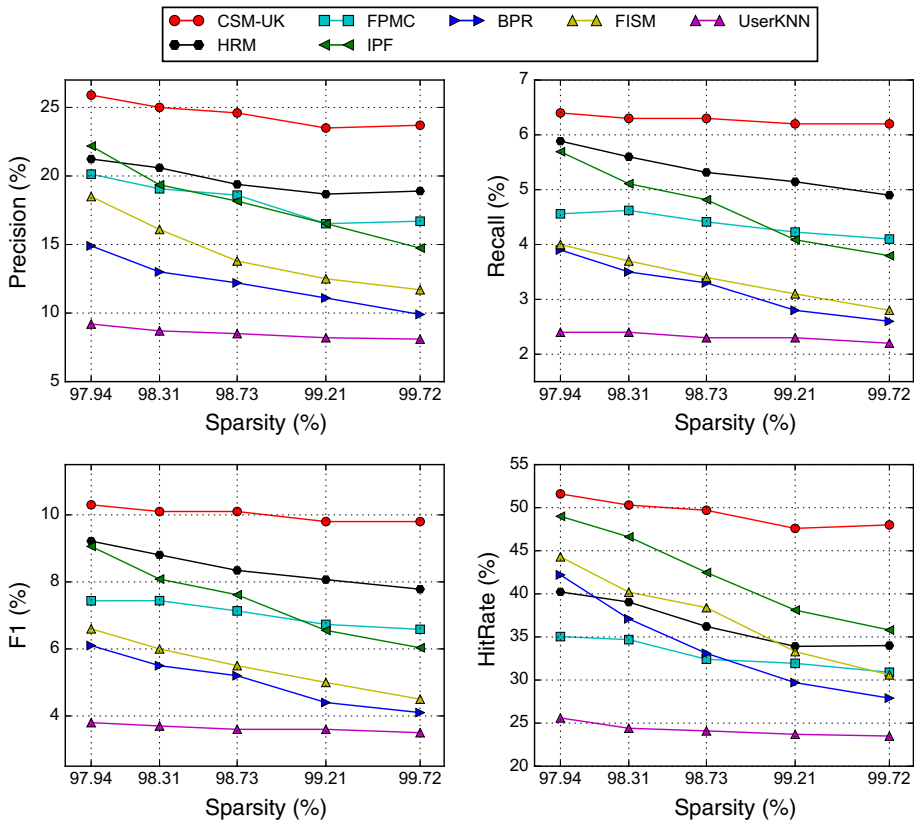


Fig. 8 Top 5 Performance over datasets with different sparsities

6 Conclusion and future work

This paper presents an approach for context-aware music recommendation, which infers and models users' general and contextual preferences for music from listening records, and recommends appropriate music fitting users' real-time requirements. Specifically, our approach first learns the embeddings of music pieces in low dimensional continuous space from users' music listening sequences using neural network models. Therefore, music pieces with similar neighbors or listened together yield similar representations. Then we infer and model users' general and contextual preferences from her/his listening records using the learned embeddings. Finally, music pieces, which are in accordance with users' general and contextual preferences, are recommended to the target user. Our work differs from prior works in two aspects: (1) the proposed approach depends on both listening sequences and user-music interaction matrix to perform recommendation, and it is less sensitive to the sparsity of user-music interaction data; (2) the proposed approach incorporates both users' general and contextual preferences for music into recommendation, which makes it perform better than baseline methods.

Based on our current work, there are three possible future directions. First, we will attempt to connect microblog service (such as Twitter, Sina Weibo) with music service websites (Such as Xiami, Last.fm) to extract more side information and try to provide

better recommendation results (Rafailidis and Nanopoulos 2016; Manzato et al. 2016). Secondly, we also plan to use more advanced techniques to extract and model users' general and contextual preferences for music, and combine preferences with advanced techniques (Wu et al. 2015), to further improve the performance. Thirdly, we only evaluate our approach by offline experiments in this work, and we will explore if users' satisfaction can be increased when users listen to the recommended music by online experiments.

Acknowledgements This research work is supported in part by Key Research and Development Project of Zhejiang Province (No. 2015C01027), National Science Foundation of China (No. 61772461), Natural Science Foundation of Zhejiang Province (No. LR18F020003 and No. LY17F020014), and Australian Research Council (ARC) Linkage Project under No. LP140100937.

References

- Adamic, L. A., & Huberman, B. A. (2000). Power-law distribution of the world wide web. *Science*, 287(5461), 2115–2115.
- Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender systems handbook* (pp. 217–253). Springer.
- Ai, Q., Yang, L., Guo, J., & Croft, W. B. (2016). Improving language estimation with the paragraph vector model for ad-hoc retrieval. In *Proceedings of the 39th international ACM SIGIR conference on research and development in information retrieval* (pp. 869–872). ACM.
- Baeza-Yates, R., Ribeiro-Neto, B., et al. (1999). *Modern information retrieval* (Vol. 463). New York: ACM press.
- Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 1798–1828.
- Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3(Feb):1137–1155.
- Celma, O. (2010). Music recommendation. In *Music recommendation and discovery* (pp. 43–85). Berlin, Heidelberg: Springer.
- Cheng, Z., & Shen, J. (2016). On effective location-aware music recommendation. *ACM Transactions on Information Systems (TOIS)*, 34(2), 13.
- Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.
- Deng, S., Wang, D., Li, X., & Xu, G. (2015). Exploring user emotion in microblogs for music recommendation. *Expert Systems with Applications*, 42(23), 9284–9293.
- Dias, R., & Fonseca, M. J. (2013). Improving music recommendation in session-based collaborative filtering by using temporal context. In *2013 IEEE 25th international conference on tools with artificial intelligence* (pp. 783–788). IEEE.
- Djuric, N., Wu, H., Radosavljevic, V., Grbovic, M., & Bhamidipati, N. (2015). Hierarchical neural language models for joint representation of streaming documents and their content. In *Proceedings of the 24th international conference on world wide web* (pp. 248–255). ACM.
- Forsati, R., Moayedikia, A., & Shamsfard, M. (2015). An effective web page recommender using binary data clustering. *Information Retrieval Journal*, 18(3), 167–214.
- Han, B.-J., Rho, S., Jun, S., & Hwang, E. (2010). Music emotion classification and context-based music recommendation. *Multimedia Tools and Applications*, 47(3), 433–460.
- Hariri, N., Mobasher, B., & Burke, R. (2012). Context-aware music recommendation based on latent topic sequential patterns. In *Proceedings of the sixth ACM conference on recommender systems* (pp. 131–138). ACM.
- Kabbur, S., Ning, X., & Karypis, G. (2013). Fism: Factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 659–667). ACM.
- Kaminskas, M., & Ricci, F. (2012). Contextual music information retrieval and recommendation: State of the art and challenges. *Computer Science Review*, 6(2), 89–119.
- Kaminskas, M., Ricci, F., & Schedl, M. (2013). Location-aware music recommendation using auto-tagging and hybrid matching. In *Proceedings of the 7th ACM conference on recommender systems* (pp. 17–24). ACM.

- Kenter, T., & de Rijke, M. (2015). Short text similarity with word embeddings. In *Proceedings of the 24th ACM international conference on information and knowledge management* (pp. 1411–1420). ACM.
- Knees, P., & Schedl, M. (2013). A survey of music similarity and recommendation from music context data. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 10(1), 2.
- Lacerda, A., Santos, R. L., Veloso, A., & Ziviani, N. (2015). Improving daily deals recommendation using explore-then-exploit strategies. *Information Retrieval Journal*, 18(2), 95–122.
- Linden, G., Smith, B., & York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1), 76–80.
- Maaten, L. V. D., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605.
- Manzato, M. G., Domingues, M. A., Fortes, A. C., Sundermann, C. V., D'Addio, R. M., Conrado, M. S., et al. (2016). Mining unstructured content for recommender systems: An ensemble approach. *Information Retrieval Journal*, 19(4), 378–415.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).
- Morin, F., & Bengio, Y. (2005). Hierarchical probabilistic neural network language model. In *Proceedings of the tenth international workshop on artificial intelligence and statistics, citeSeer*, 5, 246–252.
- Nalisnick, E., Mitra, B., Craswell, N., & Caruana, R. (2016). Improving document ranking with dual word embeddings. In *Proceedings of the 25th international conference companion on world wide web, international world wide web conferences steering committee* (pp. 83–84).
- North, A., & Hargreaves, D. (2008). *The social and applied psychology of music*. Oxford: Oxford University Press.
- Park, H.-S., Yoo, J.-O., & Cho, S.-B. (2006). A context-aware music recommendation system using fuzzy bayesian networks with utility theory. In *International conference on Fuzzy systems and knowledge discovery* (pp. 970–979). Springer.
- Pettijohn, T. F., Williams, G. M., & Carter, T. C. (2010). Music for the seasons: Seasonal music preferences in college students. *Current Psychology*, 29(4), 328–345.
- Rafailidis, D., & Nanopoulos, A. (2016). Modeling users preference dynamics and side information in recommender systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(6), 782–792.
- Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence* (pp. 452–461). aAUAI Press.
- Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2010). Factorizing personalized Markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on world wide web* (pp. 811–820). ACM.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on computer supported cooperative work* (pp. 175–186). ACM.
- Schedl, M., Vall, A., & Farrahi, K. (2014). User geospatial context for music recommendation in microblogs. In *Proceedings of the 37th international ACM SIGIR conference on research & development in information retrieval* (pp. 987–990). ACM.
- Wang, D., Deng, S., Liu, S., & Xu, G. (2016). Improving music recommendation using distributed representation. In *Proceedings of the 25th international conference companion on world wide web, international world wide web conferences steering committee* (pp. 125–126).
- Wang, P., Guo, J., Lan, Y., Xu, J., Wan, S., & Cheng, X. (2015). Learning hierarchical representation model for nextbasket recommendation. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval* (pp. 403–412). ACM.
- Wang, X., Rosenblum, D., & Wang, Y. (2012). Context-aware mobile music recommendation for daily activities. In *Proceedings of the 20th ACM international conference on multimedia* (pp. 99–108). ACM.
- Wu, J., Chen, L., Yu, Q., Han, P., & Wu, Z. (2015). Trust-aware media recommendation in heterogeneous social networks. *World Wide Web*, 18(1), 139–157.
- Xiang, L., Yuan, Q., Zhao, S., Chen, L., Zhang, X., Yang, Q., et al. (2010). Temporal recommendation on graphs via long- and short-term preference fusion. In *Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 723–732). ACM.
- Yang, Y.-H., & Liu, J.-Y. (2013). Quantitative study of music listening behavior in a social and affective context. *IEEE Transactions on Multimedia*, 15(6), 1304–1315.

- Yoon, K., Lee, J., & Kim, M.-U. (2012). Music recommendation system using emotion triggering low-level features. *IEEE Transactions on Consumer Electronics*, 58(2), 612–618.
- Zhou, N., Zhao, W. X., Zhang, X., Wen, J.-R., & Wang, S. (2016). A general multi-context embedding model for mining human trajectory data. *IEEE Transactions on Knowledge and Data Engineering*, 28(8), 1945–1958.