

# A LATENT REPRESENTATION OF USERS, SESSIONS, AND SONGS FOR LISTENING BEHAVIOR ANALYSIS

**Chia-Hao Chung**

National Taiwan University  
b99505003@ntu.edu.tw

**Jing-Kai Lou**

KKBOX Inc.  
kaelou@kkbox.com

**Homer Chen**

National Taiwan University  
homer@ntu.edu.tw

## ABSTRACT

Understanding user listening behaviors is important to the personalization of music recommendation. In this paper, we present an approach that discovers user behavior from a large-scale, real-world listening record. The proposed approach generates a latent representation of users, listening sessions, and songs, where each of these objects is represented as a point in the multi-dimensional latent space. Since the distance between two points is an indication of the similarity of the two corresponding objects, it becomes extremely simple to evaluate the similarity between songs or the matching of songs with the user preference. By exploiting this feature, we provide a two-dimensional user behavior analysis framework for music recommendation. Exploring the relationships between user preference and the contextual or temporal information in the session data through this framework significantly facilitates personalized music recommendation. We provide experimental results to illustrate the strengths of the proposed approach for user behavior analysis.

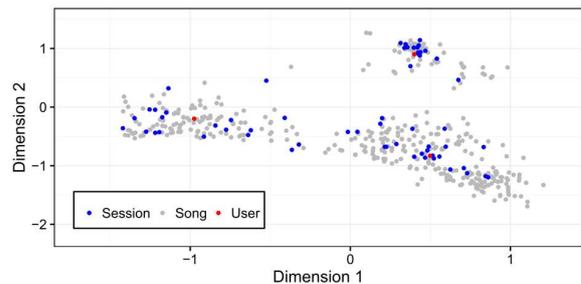
## 1. INTRODUCTION

Analyzing the listening behavior of users involves identifying and representing the music preferences of users. However, the music preference of a user is dynamic and varies with the listening context [1], such as time, location, user's mood, etc. How to take the contextual information into consideration for music recommendation is an important research issue [1]–[3]. In this work, we focus on the analysis of dynamic listening behavior and use the obtained information to personalize music recommendation.

Music and user preference are commonly represented using a taxonomy of musical genres, such as hip-hop, rock, jazz, etc. In this approach, the preference of a user is represented as a probability distribution of the genres that the user listened to. Although simple and easy to implement, the main drawback of this approach is that there is not a uniform taxonomy for music, making genre identification ambiguous and subjective [4]. In addition, it often lacks the kind of granularity needed to distinguish between songs of the same genre.



© Chia-Hao Chung, Jing-Kai Lou, Homer Chen.  
Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** Chia-Hao Chung, Jing-Kai Lou, Homer Chen. "A Latent Representation of Users, Sessions, and Songs for Listening Behavior Analysis", 17th International Society for Music Information Retrieval Conference, 2016.



**Fig. 1.** A two-dimensional latent space representation of three users listening to 403 songs in 60 sessions. Each user, session, or song is represented by a point in the latent space, and each user is surrounded by the songs played by the user and the listening sessions during which the songs are played.

In practice, it is often required to describe the music preference of a user with a fine precision. This requires a good metric to measure the similarity between songs. Therefore, besides the uniformity and granularity requirements, a music preference representation scheme has to provide an effective similarity measurement. Many approaches based on latent representation have been proposed [5], [7] to meet all three requirements. These approaches represent songs and user preferences by one single scheme. The unified representation, which is a multi-dimensional vector, is learned from a listening record or a rating record. Each dimension of the vector represents a latent feature of songs and user preferences. Therefore, each song or user is an object represented by a vector in a latent space, making the evaluation of similarity between songs or the matching between songs and users a simple matter of distance measure between vectors.

The music preference of a user may change with the listening session [3], [8]. A listening session here refers to a sequence of songs (and the associated time code) which a user continuously listened to. It contains information related to the listening experience of the user. To account for the dynamic nature of music preference, we incorporate the notion of session into the learning stage of a latent representation. In our approach, each session of the listening record of a user is also represented as an object in the latent space. The contextual information, such as the time of day and the device used for music listening, associated with each session enables the analysis of user preference at a fine level.

In addition, each user, session, or song can be plotted as a point in a two-dimensional latent space, as illustrated in Fig. 1. Clearly, this provides an intuitive way to visualize or analyze the relationship between songs and user preferences. We exploit this feature of latent space for listening behavior analysis.

To the best of our knowledge, this paper is among the first that introduce the notion of session to the representation learning for music recommendation and proposes an approach that generates the latent representation of users, sessions, and songs (Sections 3 and 4). The proposed latent representation is a powerful basis for visual analysis of user preference in a two-dimensional space and enables the discovery of a user's listening behavior that would otherwise be difficult to do with conventional representations (Section 5). In addition, we propose an effective method to evaluate the performance of a latent representation for listening behavior discovery (Section 6).

## 2. REVIEW

Statistical approaches that analyze the dynamic nature of music preference have been reported in the literature. Herrera *et al.* [2] adopted a circular statistic method to identify the temporal pattern of music listening. Zheleva *et al.* [3] proposed a session-based probabilistic graphical model to characterize music preference and showed the usefulness of session to capture the dynamic preference of a user. The importance of these two pieces of work is that they show users' listening behaviors (patterns) can be discovered and used to predict future user behaviors.

Approaches that generate a latent representation of users and songs for music recommendation have been reported. Dror *et al.* [5] adopted a matrix factorization method to characterize each user or song as a low-dimensional latent vector and to approximate the user preference (i.e. a rating) as the inner product of a user vector and a song vector. The temporal dynamics of music preference and the taxonomy of musical genres are considered jointly to improve the performance of music recommendation. Moore *et al.* [7] proposed a dynamic probabilistic embedding method to generate a representation of users and songs for music preference analysis. Each user or song is represented as a point in a two-dimensional space, and the position of each point is allowed to gradually change over time. The trajectory of a point shows the long-term variation in music preference. Recently, Chen *et al.* [9] introduced a network embedding method to enhance music recommendation. The social relationship between users is exploited to learn a latent representation of social listening, which is fed to a factorization machine to improve the performance of music recommendation.

## 3. LATENT REPRESENTATION LEARNING

In our latent space approach, a network that describes the relationship between users, sessions, and songs stored in a listening record is first constructed, then a network embedding method is applied to learn the latent

representation from the network. The details are described in this section.

### 3.1 Network Construction

The basic idea to construct a network that describes the relationship between users and songs is to consider each user or song as an object in the network and connect each user with the songs the user listened to [9]. To account for the dynamic nature of music preference, we further incorporate listening sessions into the network construction and consider each user, session, or song as an object in the network. A user is connected with all sessions of the user, and a session is connected with all songs appearing in the session. This makes the network capture the dynamic music preferences of users.

### 3.2 Network Embedding

A network embedding method aims at learning the latent representation of objects in a network. Such representation captures the relationship between the objects in the network. Objects having a similar neighborhood in the network are represented by similar vectors. In our approach, the DeepWalk algorithm [10] is applied to learn the latent representation. This algorithm consists of a random walk procedure and an update procedure.

A network consists of a set of vertices representing the objects and a set of edges connecting related vertices. The random walk procedure uniformly samples a random vertex as the root of a random walk, and then uniformly samples a random vertex from the neighbors of the current vertex as the next vertex until the maximum walking length  $L$  is reached. The procedure repeats until each vertex serves as the root of  $R$  random walks, where  $R$  is a predetermined number. The total number of random walks generated in this procedure is thus equal to the number of vertices in the network multiplied by  $R$ . The vertices visited in a random walk are processed next.

The latent representation of each vertex  $v_i$  is initially a  $d$ -vector of random variables, where  $d$  denotes the dimension of the latent space. The update procedure [11], [12] takes the random walks one by one as input and progressively refines the latent representation of objects in two steps. The first step creates a probability formula for each vertex in a random walk, starting from the first one. Specifically, for a vertex  $v_i$  in a random walk and a neighborhood window  $w$ , the conditional probability that the set of vertices  $\{v_{i-w}, \dots, v_{i-1}\}$  appears in the backward window of  $v_i$  and  $\{v_{i+1}, \dots, v_{i+w}\}$  appears in the forward window of  $v_i$  is expressed as

$$P(\{v_{i-w}, \dots, v_{i-1}, v_{i+1}, \dots, v_{i+w}\} | v_i),$$

which is called the co-occurrence probability because it indicates the likelihood that these two sets of vertices are in the neighborhood of  $v_i$  in a random walk. If the order of the vertices in each window is ignored, the co-occurrence probability can be rewritten as

$$P(\{v_{i-w}, \dots, v_{i-1}, v_{i+1}, \dots, v_{i+w}\} | v_i) = \prod_{j=i-w, j \neq i}^{i+w} P(v_j | v_i). \quad (1)$$

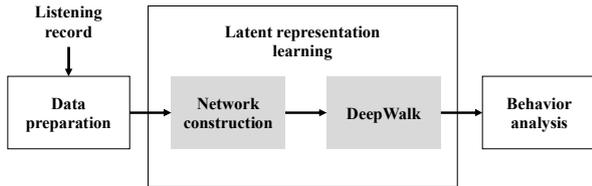


Fig. 2. Overview of our approach.

In the second step, the vector of  $v_i$  is optimized by maximizing  $P(v_j | v_i)$ . To enhance computational efficiency, a binary tree with all vertices of the network as the leaves is constructed to convert the maximization to a tree traversal process [13]. A path of the tree traversal is specified by a sequence of nodes  $\{b_1, b_2, \dots, b_k\}$  in the binary tree, where  $k$  is the length of the path,  $b_1$  is the root of the binary tree and  $b_k$  is the leaf node representing  $v_j$ . Then the conditional probability  $P(v_j | v_i)$  can be rewritten as

$$P(v_j | v_i) = \prod_{l=1}^{k-1} P(b_{l+1} | v_i, b_l). \quad (2)$$

Each conditional probability  $P(b_{l+1} | v_i, b_l)$  is modeled by a logistic function and can be rewritten as

$$P(b_{l+1} | v_i, b_l) = 1 / (1 + e^{-\Phi(v_i) \cdot \Psi(b_l)}), \quad (3)$$

where  $\Phi(v_i)$  maps  $v_i$  to its vector, and  $\Psi(b_l)$  maps  $b_l$  to its vector. Then, a stochastic gradient descent method [14] and a back-propagation algorithm [15] are applied to optimize  $\Phi$  and  $\Psi$ . The update procedure repeats until the optimization for each vertex in each random walk is processed. The optimized  $\Phi$  is the latent representation of each vertex in the network.

#### 4. OUR APPROACH

The three basic steps of the proposed approach are shown in Fig. 2. The first step involves the preparation of a listening record, the second step involves the construction of a network and the learning of a latent representation from the network, and the third step involves the analysis of user behavior. The first and second steps are described in this section, and the third step is described in Section 5.

##### 4.1 Preparation

We obtain a listening record of one hundred thousand users from a leading online music service provider [18] and use it in this work. The listening record contains every listening event of these users from January 1, 2015 to June 30, 2015, and each event contains seven fields: timestamp, user, session, listening device, song title, artist(s) of the song, and music tag(s) of the song. All users are anonymized to maintain privacy.

A session, which indicates the listening experience of a user, is defined as a sequence of events of the user with the following constraints: The gap between any two neighboring events in a session is shorter than 10 minutes, and the listening device stays the same in a session.

The music tags are used for visual analysis in Section 5, showing either genre or language information of a song. A genre tag indicates the musical style of the song, and a

	#events	#users	#songs
Training set	33,790,690	33,292	441,796
Testing set	8,797,016	19,831	219,377

Table 1. Data statistics.

language tag indicates the language of the song. We also obtain the popularity of each song by taking the logarithm of its playcount for the visual analysis.

The listening record is split into training set and testing set. We adopt the real-life split strategy [16] and split the listening record into two parts: before and after 00:00:00, June 1, 2015. The last 80 sessions of each user in the first part are selected as training data, and the first 20 sessions of each user in the second part are selected as the testing data. Users with insufficient sessions or sessions with less than 5 songs in either set are discarded. In addition, because a cold start problem [16] may occur if songs in the testing set are not in the training set, we discard such events from the testing set. Table 1 shows the data statistics. The testing set is used for performance evaluation in Section 6.

##### 4.2 Representation Learning

Using the training set, a user-session-song network is constructed as described in Section 3.1, and the DeepWalk algorithm is applied to generate the latent representation of users, sessions, and songs from the network. We set the parameters of the learning algorithm as the following: The length of a random walk  $L$  is 40, the number of random walks starting from a vertex  $R$  is 20, and the window size  $w$  is 6. A two-dimensional latent representation is generated for visual analysis (Section 5), and a 128-dimensional latent representation is generated for performance evaluation (Section 6).

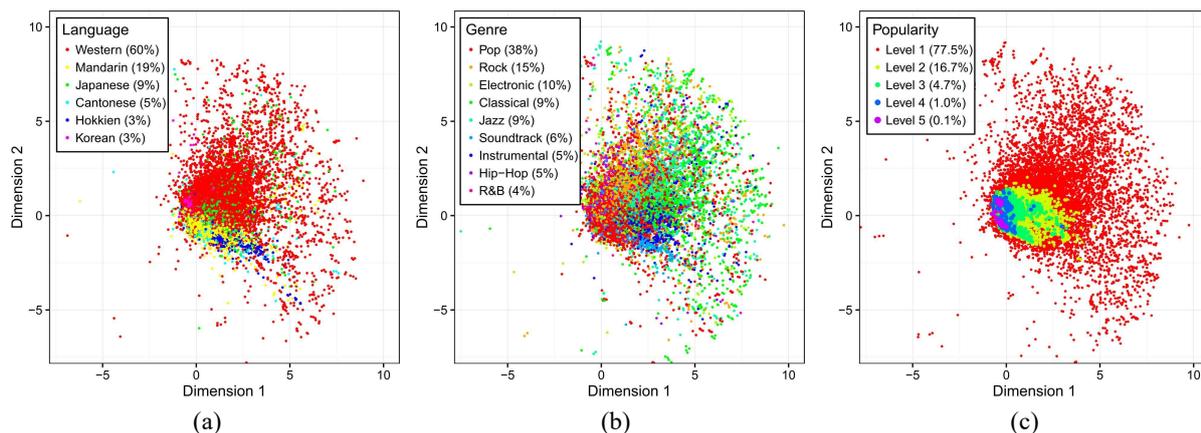
#### 5. VISUALIZATION AND ANALYSIS

A two-dimension latent space provides an intuitive way to visualize the relationship between songs and user preferences. We perform a visual analysis of the general trend of listening preference and then the individual listening behavior in such space. The details of these operations are described in this section.

##### 5.1 General Trend of Listening Preference

We can get an idea of the general trend of listening preference with respect to the song properties (the language of songs, the musical genre, and the song popularity) by examining the distribution of songs in the latent space. The two-dimensional latent space learned from the training set is plotted in Fig. 3. In each plot, 25,000 songs are plotted, and each song is marked with a specific color according to the property of the song.

In Fig. 3 (a), songs are colored according to the language of songs. There are songs in six different languages in our dataset where Western songs take a large proportion. We can see that songs of the same language are close and form a cluster in the latent space. This suggests that the user listening preference is strongly related to the song language and that a user tends to listen to songs of the languages that the user is familiar with.



**Fig. 3.** The two-dimensional latent space learned from the training set is shown in three plots. In each plot, 25,000 songs are randomly selected, and each song is specified by a point and marked with a specific color according to the property of the song. The proportion of songs with each property is showed in the legend. (a) Each song is marked according to the song language. (b) Each song is marked according to the genre, and (c) Each song is marked according to the popularity (Level 5 indicates the most popular songs), and songs with high popularity are overlaid over those with low popularity.

We can also see that some clusters are located closely. For example, Mandarin songs and Hokkien songs are located closely or even mixed together. This indicates that a part of users who listen to Mandarin songs are likely to listen to Hokkien songs as well.

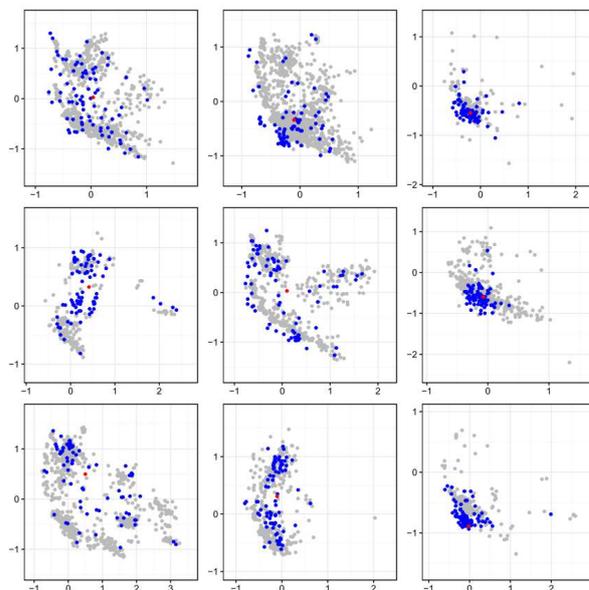
In Fig. 3 (b), songs are colored according to the musical genre. Clearly, songs of the same genre are located in the same area in the latent space. The similarity between genres is reflected on their distance. For example, Jazz songs are close to classical songs, and electronic songs are close to hip-hop songs. Combining Figs. 3 (a) and (b), we can see that Westerns songs contain many genres, such as hip-hop, rock, electronic, jazz, and classical songs, and Mandarin songs are mostly pop songs.

In Fig. 3 (c), songs are colored according to the popularity. The popularity of each song is obtained by taking the logarithm of the playcount of the song. For easy visualization, all songs are divided in to five levels in terms of popularity. Clearly, the most popular songs are near the origin of the latent space, and unpopular songs are far from the origin. It indicates that most users listen to the most popular songs, a typical long tail phenomenon of music listening [17]. Combining Figs 3 (a), (b), and (c), we can see that Western, Mandarin and Korean pop songs are more popular than other songs in our dataset.

## 5.2 Individual Listening Behavior

The individual behavior is analyzed through the distribution of sessions and songs associated with a user in latent space. A session is close to the songs that appear in the session, and sessions form a cluster if they contain similar songs. Fig. 4 shows the analyses for nine example users. In each plot, the sessions and songs associated with a user are plotted.

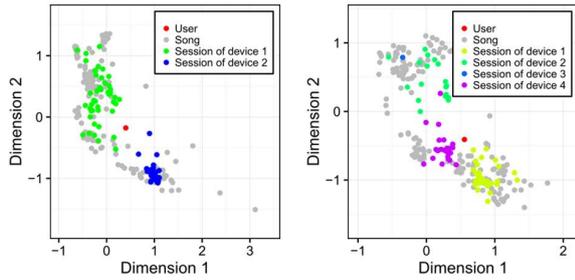
One important discovery is that users can be divided into two types, one with a single preference and the other with dynamic preference. For some users, the sessions are mostly located in one small area in the latent space. This



**Fig. 4.** Listening behavior analysis for nine example users. The sessions and songs associated with a user are plotted in each plot, where a red point represents a user, a blue point represents a session, and a gray point represents a song.

means that the users listens to the same songs most of the time and hence belong to the first type. For other users, the sessions have a wide distribution and form several clusters. This indicates that the users belong to the second type.

In order to analyze the dynamic preference of an individual, we distinguish sessions by the context information (the device used for listening and the time of day). In Fig. 5, we color each session according to the device used for listening, and we can clearly see that the sessions form clusters according to the listening device. This indicates there is a strong relevance between the music preference and the listening device. For example, a



**Fig. 5.** The sessions are distinguished by the device used for music listening. Each plot illustrates a user whose music preference is related to the listening device.

user may listen to rock songs through computer and listen to pop songs through mobile phone. This kind of listening behavior can be found on many users in our dataset. In Fig. 6, we color each session according to the time of day, and each plot shows a user whose music preference is related to the time for music listening. However, this kind of listening behavior is not easy to be observed on users. Probably it is because the relationship between time and music preference is too complex to be explained.

An interactive system for individual behavior analysis can be designed based on the latent space. For example, when we select a session, the system would highlight the songs that appear in the session so that we can further discover the user preference in each session.

## 6. PERFORMANCE EVALUATION

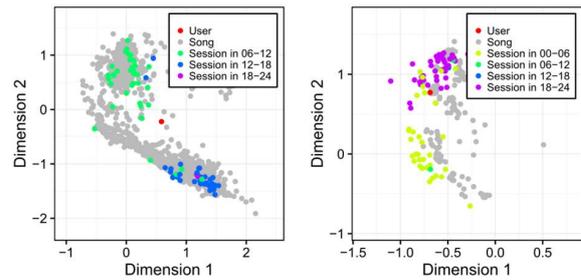
Two experiments are designed to evaluate the effectiveness of a latent representation for listening behavior prediction. The first one involves retrieving similar songs (i.e. songs that appear in the same session), and the second one involves recommending songs that match a user’s preference in a session. Each experiment is considered a retrieval problem whose goal is to retrieve songs relevant to a query object (user, session, or song) as many as possible. Specifically, a query object is selected according to a given testing session, and songs that appear in the testing session are considered relevant to the query object. For a query object, its  $k$  nearest neighboring songs in the latent space are retrieved, and the performance is evaluated in terms of how well the retrieved songs match the relevant songs. In each experiment, 200,000 sessions in the testing set are randomly selected for testing. Because the average length of the testing sessions is 20 songs,  $k$  (the number of retrieved songs) is set to  $\{10, 15, \dots, 50\}$ .

Two standard evaluation metrics for retrieval problem are applied here: Recall and precision:

$$Recall = \frac{|S_t \cap S_r|}{|S_r|} \quad (4)$$

$$Precision = \frac{|S_t \cap S_r|}{|S_t|} \quad (5)$$

where  $S_t$  is the set of songs that appear in a testing session and  $S_r$  is the set of retrieved songs. A high recall means that most of relevant songs are retrieved, and a high precision means that most of the retrieved songs are



**Fig. 6.** Each plot shows a user whose music preference is related to the time for music listening. The sessions tend to form clusters according to the time (hour of day).

relevant songs. The average recall and precision for all testing sessions are reported.

The performances of the following four methods are reported for comparison.

- **Random:**  $k$  songs are randomly selected from the dataset as the retrieved songs.
- **Popularity:** The popularity of each song is obtained by calculating its playcount in the training set, and the  $k$  most popular songs are considered the retrieved songs.
- **Matrix factorization (MF)** [6]: The vector  $\mathbf{p}_u$  for user  $u$  and the vector  $\mathbf{q}_i$  for song  $i$  are learned by solving the optimization problem

$$\min_{\mathbf{q}_i, \mathbf{p}_u} \sum_{u,i} (c_{ui} - \mathbf{q}_i^T \mathbf{p}_u)^2 + \lambda (\|\mathbf{q}_i\|^2 + \|\mathbf{p}_u\|^2), \quad (6)$$

where  $c_{ui} = 1 + \log(1 + y_{ui})$  is the confidence value of the playcount  $y_{ui}$  of user  $u$  for song  $i$ , and  $\lambda$  is a regularization parameter. Songs are retrieved according to the inner product between vectors.

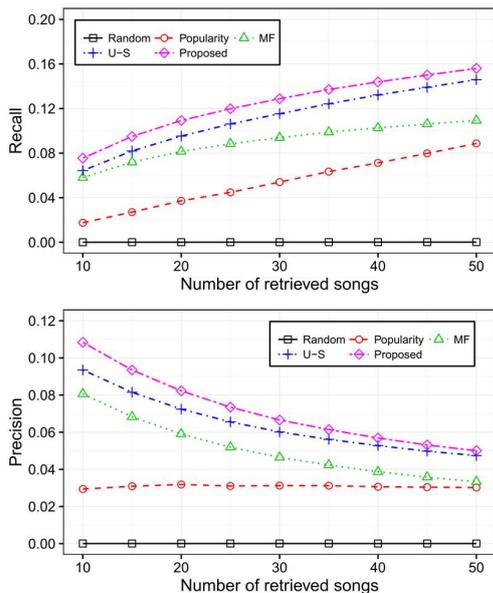
- **User-song network (U-S):** A simplified network, where a user directly connects to all songs the user listened to, is constructed. The vectors for users and songs are learned from the network using the DeepWalk algorithm. Songs are retrieved according to the Euclidean distance between vectors.

For fair comparison, the dimension of the latent representations (vectors) learned by MF, U-S, and the proposed method is fixed to 128.

### 6.1 Experiment for Retrieval of Similar Songs

This experiment evaluates the ability of the representation of songs to capture the similarity between songs. Because songs in a session usually have similar properties, this experiment is to find songs that appear in the same testing session. Specifically, the first song in each testing session is selected as a query object to retrieve its  $k$  nearest neighbor songs, and the remaining songs in the testing session are considered relevant to the query object.

The performances of various methods are compared in in Fig. 7. We can see that the proposed approach outperforms MF and U-S, showing the effectiveness of adding session objects into the learning stage of a latent representation. We can also see that the popularity-based method works slightly better than the random method, showing that many users tend to listen to the popular songs. This is consistent with our observation (i.e. the long tail phenomenon) in Section 5.1.



**Fig. 7.** Performance comparison of various methods for the retrieval of similar songs.

## 6.2 Experiment for Music Recommendation

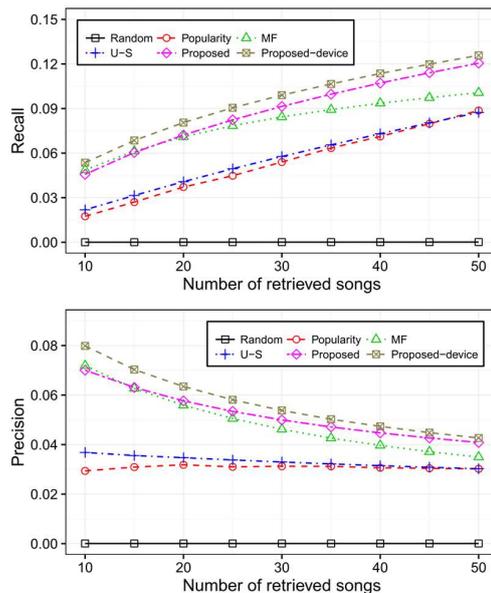
This experiment evaluates the effectiveness of a latent representation to recommend songs to a user in a testing session. The vector for the user is used to retrieve its  $k$  nearest neighbor songs, and the songs in the testing session are considered relevant songs.

As discussed in Section 5.2, a user may have dynamic preference, so only considering the user vector for every session of the user is not enough. With our observation that there are many users whose preferences are related to the listening device, we use the session vector to recommend songs. For each testing session, a reference session that belongs to the same user of the testing session and that is through the same listening device is selected as a query object, and the vector for the reference session is used to retrieve its  $k$  nearest neighbor songs.

The performance comparison is shown in Fig. 8. We can see that if the user vector is used, the proposed approach outperforms MF when  $k$  is higher than 20. If the session vector is considered, the performance of proposed approach is significantly improved. It shows the benefit of exploiting the contextual information (device information) to capture the dynamic music preference of a user.

## 7. DISCUSSION

Besides the network embedding method adopted in this work, factorization is another approach that can be applied to generate the latent representation of objects. A factorization method approximates the interactions (e.g. ratings or counts) between objects as the inner product of the vectors representing the objects. In contrast, as we can see in Eqn. (3), the network embedding method maps all objects into the same latent space ( $\Phi$ ) and makes similar objects close to each other in the latent space. This enables us to visually analyze the relationship between



**Fig. 8.** Performance comparison of various methods for music recommendation. Proposed-device indicates that the device information of each testing session is exploited.

objects in the latent space. Moreover, the network embedding method learns the relationship between objects that do not have explicit links between them, such as two users who listen to the same songs. Such relationship cannot be learned by the factorization method. A comprehensive comparison between the network embedding and factorization methods (e.g. user-session-song factorization) is an interesting topic for future work.

## 8. CONCLUSION

Knowledge of the behavior of music listeners is important to music recommendation. In this paper, we have described an approach to address this issue. The proposed approach generates the latent representation of users, sessions, and songs from a listening record. Such representation makes the relationship between these objects easy to analyze. We have performed a visual analysis of user behavior and preference in a two-dimensional latent space and illustrated the strengths of the proposed approach by comparing its music recommendation and retrieval performances with various methods. We have also shown that the information obtained from the two-dimensional analysis is useful for personalized music recommendation. The contextual information associated with each session enables both user preference analysis and music recommendation at a fine level.

## 9. REFERENCES

- [1] M. Kaminskas and F. Ricci, "Contextual music information retrieval and recommendation: State of the art and challenges," *Comput. Sci. Review*, vol. 6, no. 2, pp. 89–119, 2012.

- [2] P. Herrera, Z. Resa, and M. Sordo, "Rocking around the clock eight days a week: An exploration of temporal patterns of music listening," in *1st Workshop Music Recommendation Discovery*, 2010.
- [3] E. Zheleva, J. Guiver, E. M. Rodrigues, and N. Milic-Frayling, "Statistical models of music-listening sessions in social media," in *Proc. 19th ACM Int. Conf. World Wide Web*, pp. 1019–1028, 2010.
- [4] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Tran. Speech Audio Process.*, vol. 10, no. 5, pp. 293–302, 2002.
- [5] G. Dror, N. Koenigstein, and Y. Koren, "Yahoo! music recommendations: Modeling music ratings with temporal dynamics and item taxonomy," in *Proc. Fifth ACM Int. Conf. Recommender Syst.*, pp. 165–172, 2011.
- [6] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proc. 8th IEEE Int. Conf. Data Mining*, pp. 263–272, 2008.
- [7] J. L. Moore, S. Chen, T. Joachims, and D. Turnbull, "Taste over time: The temporal dynamics of user preferences," in *Proc. 14th Int. Soc. Music Inform. Retrieval Conf.*, pp. 401–406, 2013.
- [8] O. Carlsson, *Cluster User Music Sessions*, Master of Science Thesis, Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Sweden, 2014.
- [9] C.-M. Chen, P.-C. Chien, Y.-C. Lin, M. F. Tsai, and Y.-H. Yang, "Exploiting Latent Social Listening Representations for Music Recommendations," in *Proc Ninth ACM Int. Conf. Recommender Syst. Poster*, 2015.
- [10] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 701–710, 2014.
- [11] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv: 1301.3781*, 2013.
- [12] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Advances Neural Inf. Process. Syst.*, pp. 3111–3119, 2013.
- [13] F. Morin and Y. Bengio, "Hierarchical probabilistic neural network language model," in *Proc. Int. Workshop Artificial Intell. Stat.*, pp. 246–252, 2005.
- [14] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. 19th Int. Conf. Comput. Stat.*, pp. 177–186, 2010.
- [15] Y. LeCun, D. Touresky, and G. Hinton, "A theoretical framework for back-propagation," in *Proc. Connectionist Models Summer School*, pp. 21–28, 1988.
- [16] S.-Y. Chou, Y.-H. Yang, and Y.C. Lin, "Evaluating music recommendation in a real-world setting: On data splitting and evaluation metrics," in *Proc. IEEE Int. Conf. Multimedia Expo*, pp. 1–6, 2015.
- [17] O. Celma, *Music Recommendation and Discovery*, Springer Berlin Heidelberg, 2010.
- [18] KKBOX Inc., <https://www.kkbox.com>.