

Music Exploration by Impression Based Interaction

Hiroki Fujino
fujino@sict.i.u-tokyo.ac.jp

Kôiti Hasida
hasida.koiti@i.u-tokyo.ac.jp

Yusuke Matsubara
matsubara@sict.i.u-tokyo.ac.jp

Graduate School of Information Science and Technology, The University of Tokyo
Hongo 7-3-1, Bunkyo-ku, Tokyo 113-8656, Japan

ABSTRACT

Search and recommendation systems help users find their favorites among an abundant amount of songs available on distributors such as iTunes. Nevertheless, it is still hard for us to efficiently retrieve the right music. This paper proposes an interactive music exploration system to help them reach their favorite songs by means of interactive feedbacks submitted by the user. The recommendation component of our system is based on the feedback component accepts the user's feedbacks in terms of impression features such as "more *ballad*-like" and "less *pop*-like". The proposed concepts have been implemented as an online demonstration system using a real world dataset.

Keywords

music retrieval, personalization, interactive systems design

1. INTRODUCTION

Recent capacity enlargement of portable audio player and development of music distribution service like iTunesMusicStore enable us to save and play a large amount of music. For example, there are over million of songs on iTunes now.

On the other hand, We have had to select song we would like to listen to from a lot of songs. This prompted the development of recommendation services that predict user's preference and recommends songs to the user. Examples include Last.fm and Spotify, which recommend songs to the users who have same preferences like based on songs selected by the users. Other music recommendation systems, such as Pandora, recommend songs that have similar features to songs selected the user.

Whether the user wants to listen to a particular song can change based on various factors, such as time, place or mood. [1] Even if some songs match user's general preference, the user might not want to listen to these songs in a particular context. Although existing services recommend songs based on user's preference, they may not be able to recommend songs user want to listen to at the real time. From this

background, the research that recommends songs along with user's emotion has been conducted. Also, different users may express different preferences as to what kind of music they want to listen to in a certain mood. Therefore, it is necessary that the recommended songs are personalized by the user's feeling and preference.

In information retrieval, the method which uses the interaction between user and system has been proved to be effective for improving retrieval accuracy.[3] Also, we think that when listening to music, the user would like to give feedback based on impression, such as more *rock*-like song. Then, We think that the method which uses the interaction using terms of impression features such as "more *ballad*-like" and "less *pop*-like" between user and system is an effective way. This paper proposes an interactive music exploration system to help us reach our favorite songs by means of interactive feedbacks to songs recommended based on our preferences and previous feedbacks.

The remainder of this paper is organized as follows. In Section 2, we discuss related work. In Section 3, we discuss the proposed system. In Section 4, we discuss preliminary experiment.

2. RELATED WORK

Real-time personalization by user feedback and relationship between music and emotion has been studied in previous work.

2.1 Relevance Feedback

Relevance feedback personalizes retrieval results based on feedback. In information retrieval, it is an established personalization technique that enables to improve user modeling by asking the user to specify which documents are relevant explicitly (i.e., whether the document satisfies his/her information need).

Moon et al. [2] apply relevance feedback to personalizing document retrieval. They create a user model by online supervised learning based on documents clicked by the user, then apply the user model to subsequent retrieval. They claim real-time personalization by relevance feedback produces easily and efficiently user models.

Pouli et al[4] apply real-time personalization by relevance feedback to multimedia retrieval. They create a user model by feedback which includes 'relevant' or 'non-relevant' by the user.

2.2 Relationship between music and emotion

Since user-generated contents include a large amount of



This work is licensed under a Creative Commons Attribution International 4.0 License.

ESIDA'17 March 13-13 2017, Limassol, Cyprus

© 2017 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-4903-1/17/03.

DOI: <http://dx.doi.org/10.1145/3038462.3038468>

descriptions of songs, music recommendation by using descriptions related to music on the internet has been studied. Tsuda et al[5] recommended songs by using music emotion data based on descriptions related music emotion on Yahoo! Music which is internet music site. They conducted a subject experiment and then claim that music emotion is effective to music recommendation.

Liu et al[6] analyzed a correlation between user mood and music emotion. They collected posts from LiveJournal which allows a blogger to tag his or her post with a mood tag that shows which song matches his or her mood well. Then, they analyzed a correlation user mood and music emotion by using Music Emotion Recognition(MER) based on the collected posts and Last.fm datasets, with an emotion vocabulary consisting of 190 music emotion tags adopted by the professional music review website AllMusic. In their research, they created new music dataset and published it.

3. PROPOSED SYSTEM

In this section, we explain the proposed system. As Fig.1 shows, this system consists of three operations; music retrieval by the query, feedback by the user, and re-recommend a song.

1. Our system recommend music based on the query by the user, initialized user model randomly, and song features. It is described in detail in Section 3.1.
2. The user gives feedback to the song recommended. Our system has impression based feedback such as "I want to hear more *ballad*-like song". We use impression based feedback in this manner because it allows the user to express preferences more precisely than relevance feedback.
3. Our system updates the user model based on feedback by user. Then, our system predicts the best song the user would like to listen to and re-recommend new songs. It is described in detail in Section 3.2.2.

The purpose of this system is to help the user effectively find music he likes by the interaction with feedback and re-recommendation. Our system is developed using the Django framework. Django is an open source web application framework and is written in Python.

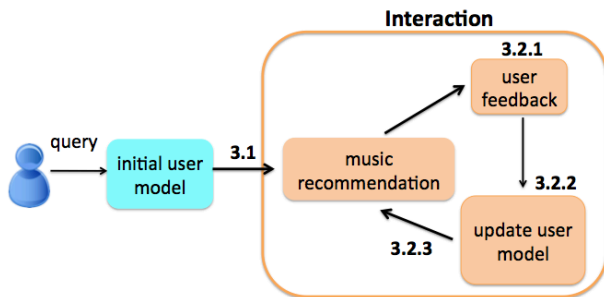


Figure 1: Proposed system

The user model gets updated based on the user's feedback to the recommended songs. Both the user and songs are modeled as emotion vectors in which each dimension corresponds to a keyword describing an emotion.

3.1 Music Retrieval

In the part of music retrieval by the query, our system recommends songs based on the user model and features of music described below.

3.1.1 User Model

The preference prediction value $f(u, m)$ of user u for music m is expressed by the following regression equation. Each music m is associated with a vector $x_m \in R^n$, each user u is associated with a vector $w_u \in R^n$, n is number of music songs.

$$f(u, m) = w_u^T x_m \quad (1)$$

3.1.2 Music Dataset

This system use Live Journal 2M dataset(LJ2M) created by Lis et al.[6] This dataset has the following content:

- 88,164 unique songs titles of 12,201 artists.
- Among 88,164 unique songs, 43 emotion tags with numerical value for 64,124 unique songs

Genres included in the dataset are *rock*, *pop*, and *hip-hop*. Each emotion tag has a numerical value indication the "affinity" (i.e., degree of association, in $[0, 1]$) between the song and a music emotion. The value is predicted by Lis who use state-of-the-art MER techniques and analyzed the content of the audio previews. In this system, we use 64,124 unique songs as search target songs and 43 emotion tags as features.

3.1.3 Search and Retrieval

In this paper, it is objective that construct an interactive music exploration system to help us reach our favorite songs by means of interactive feedbacks to songs recommended based on our preferences and previous feedbacks.

When the user enters a query, our system shows one of the songs with the highest initial relevance to the query. While it can be generalized to use any relevance metrics, for the sake of simplicity, we restrict queries to emotion words here and choose at random the initial song to show from the top 1000 list of the songs with the highest value in the corresponding dimension.

In order to evaluate whether our system recommends songs that the user wants to listen to in a specific situation or mood by interaction, we prepared three sample queries. These sample queries are typical impression words described in Section 3.2.1; "*ballad*", "*pop*", "*rock*". These three impression words are cluster names of 43 emotion words, which are described in Section 3.2.1.

3.2 Emotion Based Feedback

3.2.1 User Feedback

This proposed system has feedback based on the impression such as "I want to hear a more *ballad*-like music". However, if the user is given all 43 emotion tags as an option, it takes much time for the user to select one of 43 tags including a similar meaning. Therefore, we divide 43 tags into a specified number k of clusters. We use Fuzzy C-Means(FCM). This method is one of soft clustering, and one object can belong to two or more clusters. FCM searches for partitions that divide input data.

In this paper, we set the fuzzification parameter m to 1.5. The results are shown in Table.1.

Second, we name each cluster with labels that the user can easily understand. Cluster1 contains many emotion words representing *ballad* in music genre. Cluster2 contains many emotion words representing *pop* in music genre. Cluster3 contains many emotion words representing *rock* in music genre. Cluster4 contains emotion words of various meanings. In our system, cluster 1, cluster 2, cluster 3 which can be made meaningful was selected as a feedback option. Further, we add up and down elements such as "more" or "less" to each emotion, we decide total six choices such as Fig 2. $\uparrow$ means "more" and $\downarrow$ means "less." The user can feedback by clicking one of the six choices. Then, we calculated the cluster value of each song. The value is the average value obtained by multiplying the membership value for each cluster of each emotion word by the value of the emotion vector of each song. As described in Section 3.1.3, our system recommend songs with the highest value multiplied by the predicted value of the song and the cluster value related to the query.

Table 1: tag and membership value included in each cluster

cluster1	sweet(0.703), sad(0.943), calm(0.962), relaxed(0.765), romantic(0.752), gentle(0.929), tender(0.494), weary(0.687), peaceful(0.936), intimate(0.795), soft(0.841)
cluster2	bright(0.471), carefree(0.844), cheerful(0.757), confident(0.810), detached(0.416), fun(0.729), happy(0.748), hungry(0.392), lively(0.873), sexy(0.517), silly(0.756)
cluster3	aggressive(0.958), harsh(0.919), angry(0.796), complex(0.799), party(0.744), hostile(0.948), cold(0.427), messy(0.348), tense(0.411)
cluster4	ambitious(0.066), anxious(0.621), difficult(0.447), elegant(0.826), hopeful(0.964), innocent(0.650), lazy(0.540), reserved(0.629), reverent(0.554), smooth(0.423), thoughtful(0.739), warm(0.675)

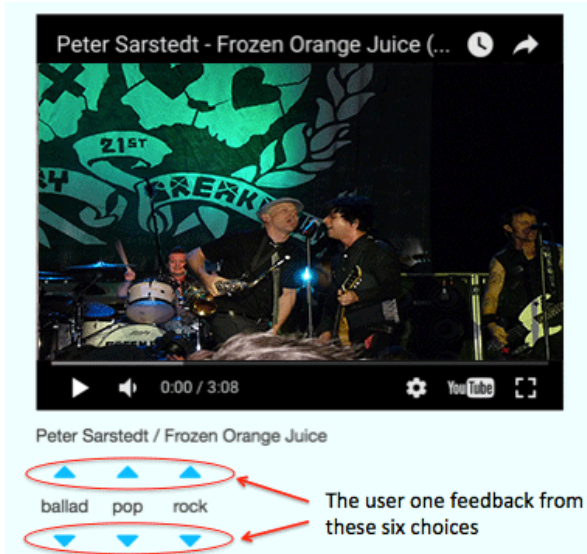


Figure 2: emotion feedback

3.2.2 Update User Model

In the section, we explain user model.

As described in the previous section, our system is given impression based feedback such as "more *ballad*-like" by the user.

We update the user model by learning-to-rank techniques, aiming to rank songs preferred by the user higher than the song that has been presented to the user.

Learning to rank is the technique which is to construct a model or a function for ranking objects and major technique in ranking models for information retrieval[9]. Learning to rank techniques are categorized them into three groups by their input representation and loss function; pointwise approach, pairwise approach, and listwise approach. Our system uses pairwise approach. Pairwise approach uses the pair of objects where each object has a different relevance value from the other. The objective function is Equation (2).

$$\min \left(\frac{\lambda}{2} \|w_u\|^2 + \sum_{m \in M_t} L((x_m - x_{m_t}, \text{sign}(f(u, m) - f(u, m_t), w_u)) \right) \quad (2)$$

Here, x_{m_t} is feature vectors of recommended song at time t , M_t is sets of training songs which are assumed that the user prefer to recommended song at time t , λ is a regularization parameter and the function $\text{sign}(p)$ returns $+1$ for $p > 0$, -1 for $p < 0$. L is *Hingeloss* that is computed as Equation (3).

$$L((x_m - x_{m_t}, \text{sign}(f(u, m) - f(u, m_t), w_u))) = \max(0, 1 - \langle w_u \text{sign}(f(u, m) - f(u, m_t)), (x_m - x_{m_t}) \rangle) \quad (3)$$

Next, we explain how to create training data. In each iteration of feedback, we create training examples from the songs in the proximity of the most recently listened song. We first extract as candidate training examples all songs with a higher (or lower, in the case of negative feedback) emotion vector value in the dimension selected in the user's feedback. When the feedback is "more *ballad*", for example, we do not additionally require an explicit input of how far the user wants to go in the direction of *ballad*. Instead, we set a bound condition to tune the range, as defined in Equation 5 and shown in Figure 3.

$$b_t = \frac{v_f}{2^t} \quad (4)$$

Here, t is the number of interaction that the user has provided. v_f is standard deviation of feedback cluster at time t . (4) represent that as interaction increases, bound gradually decreases. In addition, we set the parameter d for creating the training data. d is the average standard deviation of all tags of each song. We choose songs whose Euclid distance from the feature vectors of song m_t is less than d .

3.2.3 Reranking

After the model of the user is updated, our system recommends songs with the highest value multiplied by the predicted value of the song and the cluster value related to the query.

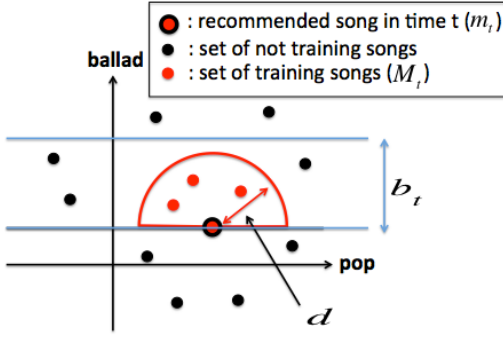


Figure 3: sample space of song features

4. PRELIMINARY EXPERIMENT

We implemented the music exploration system described above and performed a preliminary experiment to demonstrate the effectiveness of the proposed concepts.

Table 2 - 4 show series of songs recommended by our system through interaction at the preliminary experiment. Zeroth interaction means the initial search result. When feedback related to *ballad* or *rock* is performed, the value of the emotion vector of the song related to the feedback changed as intended. For example, in the second iteration of Table 2, the song with a larger *rock* value was recommended. However, when preferences regarding *pop* were given, the parameters of the song in the dimensions relevant to *pop* did not change as intended. This may be due to the lower membership values for the *pop* cluster overall relative to the other two; it is possible that the feedback influenced other dimensions primarily belonging to the other two clusters rather significantly.

In addition, the influence of the feedback is greatly influenced by the result of the interaction of the earlier one, which confirms the effect of the decaying factor shown in Equation (4).

Table 2: preliminary experiment result (user query = "pop")

iteration	feedback	song name	pop	ballad	rock
0	None	Good Lovin'	0.157	0.106	0.103
1	↓ <i>pop</i>	Walk Real Slow	0.104	0.176	0.085
2	↑ <i>rock</i>	Abnegating cecity	0.136	0.071	0.143
3	↓ <i>rock</i>	Dum Dum Boys	0.121	0.104	0.112
4	↑ <i>pop</i>	From The Heart	0.118	0.181	0.069
5	↑ <i>pop</i>	Pass Me The Bottle	0.125	0.109	0.109

Table 3: preliminary experiment result (user query = "rock")

iteration	feedback	song name	pop	ballad	rock
0	None	Institutions Of Deceit	0.111	0.071	0.138
1	↓ <i>rock</i>	The Cows Wrong	0.100	0.174	0.078
2	↑ <i>pop</i>	The Light	0.090	0.158	0.093
3	↑ <i>rock</i>	The Rulers Back	0.138	0.090	0.127
4	↑ <i>pop</i>	Rhythm Of The Night	0.154	0.089	0.125
5	↑ <i>rock</i>	Canada	0.096	0.163	0.080

Table 4: preliminary experiment result (user query = "ballad")

iteration	feedback	song name	pop	ballad	rock
0	None	Justine	0.083	0.185	0.066
1	↓ <i>pop</i>	Beginners Luck	0.135	0.092	0.137
2	↑ <i>rock</i>	Nemesis	0.121	0.065	0.143
3	↓ <i>rock</i>	May I Have This Dance	0.096	0.186	0.065
4	↑ <i>ballad</i>	My Foolish Heart	0.100	0.190	0.069
5	↑ <i>pop</i>	The Gift	0.116	0.111	0.116

5. CONCLUSIONS

This paper proposes an interactive music exploration system to help us reach our favorite songs by means of iterative feedbacks to song recommended based on our preferences and previous feedbacks. The recommendation component of our system is based on the feedback component accepts our feedbacks in terms of impression features such as "more *pop*-like" and "less *ballad*-like". Because previous research shows that there is a correlation between music emotion and user, feedbacks like this may be more effective than relevance feedback.

In future, to prove that our system is effective for music exploration, we will do the subject experiment and evaluate our system.

6. REFERENCES

- [1] Lee, Jin Ha, and J. Stephen Downie. "Survey of music information needs, uses, and seeking behaviours: Preliminary findings." ISMIR. Vol. 2004. 2004.
- [2] Moon, Taesup, et al. "Online learning for recency search ranking using real-time user feedback." Proceedings of the 19th ACM international conference on Information and knowledge management. ACM, 2010.
- [3] Shen, Xuehua, Bin Tan, and ChengXiang Zhai. "Implicit user modeling for personalized search." Proceedings of the 14th ACM international conference on Information and knowledge management. ACM, 2005.
- [4] Pouli, Vasiliki, et al. "Personalized multimedia content retrieval through relevance feedback techniques for enhanced user experience." Telecommunications (ConTEL), 2015 13th International Conference on. IEEE, 2015.
- [5] 津谷篤, et al. "印象の有無のみの感性評定を用いた状況に応じた楽曲選択法とその応用." 日本感性工学会論文誌 12.3 (2013): 397-407. (A, Tsuya, et al. "Music Selection Method Appropriate for Users' Situations and its Application Using Kansei Evaluation Only by the Existence of Impression")
- [6] Liu, Jen-Yu, Sung-Yen Liu, and Yi-Hsuan Yang. "LJ2M dataset: Toward better understanding of music listening behavior and user mood." Multimedia and Expo (ICME), 2014 IEEE
- [7] Katsavounidis, Ioannis, C-C. Jay Kuo, and Zhen Zhang. "A new initialization technique for generalized Lloyd iteration." Signal Processing Letters, IEEE 1.10 (1994): 144-146.
- [8] Hung, Ming-Chuan, and Don-Lin Yang. "An efficient fuzzy c-means clustering algorithm." Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on. IEEE, 2001.
- [9] Sculley, D. "Large scale learning to rank." NIPS Workshop on Advances in Ranking. 2009.