

# A Music Recommendation System Based on Annotations about Listeners' Preferences and Situations

Katsuhiko Kaji  
Graduate School of  
Information Science, Nagoya Univ.  
kaji@nagao.nuie.nagoya-u.ac.jp

Keiji Hirata  
NTT Communication  
Science Laboratories  
hirata@brl.ntt.co.jp

Katashi Nagao  
Center for Information  
Media Studies, Nagoya Univ.  
nagao@nuie.nagoya-u.ac.jp

## Abstract

*In this paper, we present a playlist generation scheme that uses lyrics and annotations to discover similarity between kinds of music and user tastes. It generates a playlist according to user preferences and situations. Additionally, users can provide some feedbacks to the system such as whether each tune is suitable for the preference and the situation. The system transforms the feature values concerning preferences and situations and adapts them to each user.*

*The playlists are generated through three phases. First, an initial playlist is found from databases by content-based retrieval. Second, transcoding improves the playlist according to the user's preference and situation. Finally, by interaction between the system and the user, the playlist becomes more suitable for the user.*

## 1. Introduction

Recently, many people can easily listen to music anywhere/anytime by portable digital music players that have large memory capacity, and there are many music files in the Internet.

Many researches on music recommendation and automatic playlist generation are actively in progress. Several researches on music recommendation [1][2][3] have concluded that collaborative filtering or the method to use musical metadata (genre, artist, etc.) efficiently recommends music. Though, there are several kinds of information that are difficult to acquire for each tune [4]. The method to collect such information immediately is necessary.

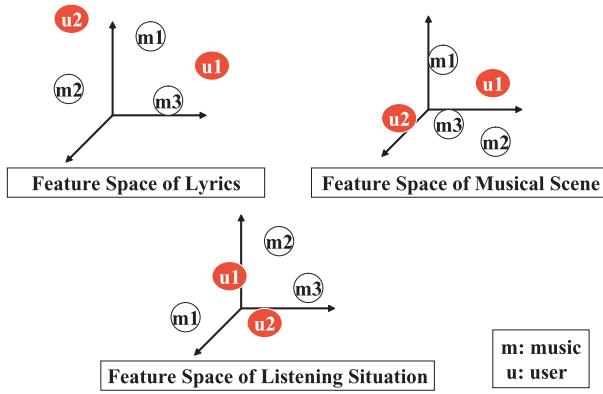
Several music distribution services such as iTunes Music Store are already available on the Web. In the near future, fixed charge services will be applied to music distribution. Many Web users will be able to share and recommend playlists without music data itself so that a playlist consists of a number of music IDs only. At present, Podcasting that enables iPod users to import many contents such as music and radio files automatically is already available. Additionally, audioscrobbler [5] is available that builds pro-

files of user's musical taste using listening history from media player. And many users are already enjoying Last.fm [6] that is personalised online radio station using their profiles collected by audioscrobbler. Like this, it is desired that users can automatically acquire suitable contents from vast amount of contents.

Based on the above considerations, we think that it becomes more important for people to express an individual mental state and situation and the relationship between listeners and tunes via playlists. However, these are inherently subjective and personal. In a typical technique for handling an individual mental state and situation, they are represented as attribute-value pairs for the attributes that are selected beforehand. For the relationship between listeners and tunes, conventional techniques have been developed for extracting features common among people, rather than personalized features. Actually, it is difficult to determine whether a tune is suitable for a situation of a listener in a mental state, for example, only with genre, artist, and acoustic information of sound.

As a starting point, we propose a playlist recommendation method that facilitates playlist generation and recommendation. We employed not only conventional musical metadata but also ones for taking account user's preference and situation into consideration. In the situation that the playlists are shared with many users, narrowing down by collaborative filtering is effective to consider each user's preference. In addition to collaborative filtering, it uses transcoding to convert an initial playlist into a more suitable one that reflects the preferences. Additionally, interactions between the system and the listener can polish his/her preferences.

We focus the annotation technology for treating such subjective and personal information. Annotation is appropriate for immediate collection and flexible description of such subjective and personal information. In the Semantic Web technology, annotation is widely recognized as a means for flexibly describing metadata of contents, such as genre, artist, personal and/or public comments [7][8].



**Figure 1. Feature spaces of lyrics, music scene and listening situation**

It can be said that interpersonal communication using text, drawing, and picture through Blog and SNS (Social Networking Service) becomes active. Though music is difficult for many people to compose and perform as they want, it is easily able to create playlists that reflect creators' idea and affection. In case of a playlist, its listener also can be a creator at the same time. Therefore we think that playlist will be one of media for future communication as well as text, drawing, and picture. We call it the playlist-mediated communication.

The paper is organized as follows: Section 2 describes the detail of our playlist recommendation system. In Section 3, we evaluate how well our system works. Finally, Section 4 concludes the paper with mentioning future work and perspectives.

## 2. Playlist Recommendation System

It is generally said that several types of feature values are effective for music recommendation [9][10]. On the other hand, we suppose that listener's situation is crucial for music recommendation, in accordance with our ordinary method of enjoying music. For example, if some one is enjoying on the summer beach, he probably wants to listen to exciting tunes even if he normally likes soft music. It may occur that a listener does not like the tunes recommended by Last.fm [6] because we assume that a listening situation is dominant over the tune features handled by Last.fm.

Then we construct a playlist recommendation system using annotations based on listeners' preferences and situations. In this research, the following are adopted as musical feature values: lyrics, scene that tune expresses (musical scene), and listening situation.

### 2.1. Similarity between tunes using lyrics and annotations

Since information of musical scene and listening situation strongly depends on listeners' aspects, it is hard for automatic analysis to collect these information. Therefore several kinds of information of listener aspect are collected for

each tune by musical annotation system, and we use them to musical feature values. With the musical annotation system, we can associate each tune with musical scene and listening situation. The information of musical scene and listening situation are gathered from a multi-item questionnaire.

Since annotations associated to each tune do not contain equal amounts of information, annotation is made by users instead of by computer. In terms of musical scene and listening situation, the averages of annotations are regarded as the feature values of each tune. At the same time, keywords are extracted by TF\*IDF<sup>1</sup> from lyrics to map each tune to the feature space of lyrics.

Parameter  $m_i$  represents an  $i$ -th tune. Feature values  $l_{m_i}$ ,  $c_{m_i}$ , and  $s_{m_i}$  are the feature values of lyrics, musical scene and listening situation of tune  $m_i$ . These values are all vector value and  $l_{m_i}$  is defined as following formulas.

$$l_{m_i} = (f(m_i, k_0), f(m_i, k_1), \dots, f(m_i, k_n))$$

The number of keywords is  $n$  and  $k_i$  means the  $i$ -th keyword. The value of  $f(m_i, k_j)$  comes 1 when the lyric of music  $m_i$  contains keyword  $k_j$ , and 0 when the lyric doesn't contain the keyword.

Figure 1 shows feature spaces that feature values of lyrics, musical scene and listening situation is mapped. Each tune shown as  $m_1$ ,  $m_2$  and  $m_3$  and each users shown as  $u_1$ ,  $u_2$  are mapped to the feature spaces. Cosine coefficient of any two tunes was derived from feature spaces. I'm just guessing musical similarity  $\text{sim}(m_i, m_j)$  between music  $m_i$  and  $m_j$  can be calculated by each weighted cosine coefficient:

$$\text{sim}(m_i, m_j) = \alpha \cos(l_{m_i}, l_{m_j}) + \beta \cos(c_{m_i}, c_{m_j}) + \gamma \cos(s_{m_i}, s_{m_j})$$

Coefficients  $\alpha$ ,  $\beta$ , and  $\gamma$  are the weights of the feature spaces of lyrics, the musical scene, and the listening situation, respectively.  $\cos(a,b)$  denotes cosine coefficient. By using this function, the similarity between two objects can be calculated. Since there are a number of feature spaces, the weight of each feature space can be adjusted by the feature value that is enhanced. The system generates a particular playlist suitable for a user's situation. Then  $\gamma$  as the weight of the listening situation is bigger than  $\alpha$  and  $\beta$ .

Several studies on music recommendation methods have concluded that such musical metadata as genre or artist effectively recommend music. Our system can also import such effective metadata so that these feature values are mapped to multiple feature spaces.

### 2.2. Sequence of playlist generation

The system accumulates favorite tunes from histories of playlist generation. Assuming that a listener's ideal tune can be guess from his favorite tunes, the user can be mapped to the above feature spaces of lyrics and musical scene. A listener's situation can be mapped to the feature space of

<sup>1</sup>Term Frequency\*Inverse Document Frequency [11] is general method of keyword extraction.

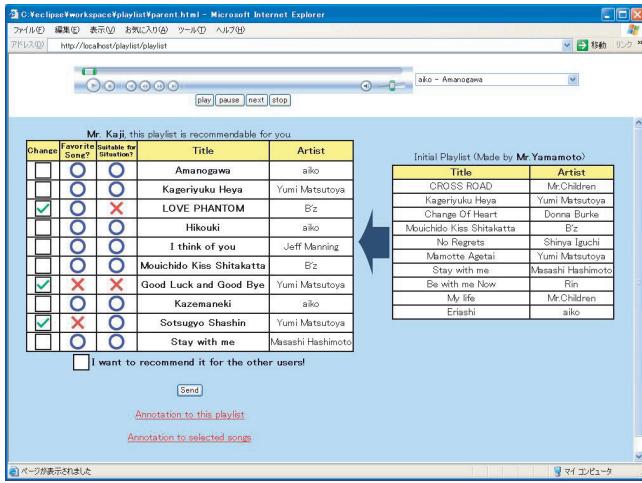


Figure 2. A generated playlist

listening situation when the system acquires it. Similarity between arbitrary tunes and listeners can also be calculated.

The system generates a playlist through three phases. First, an initial playlist is found from playlist databases by collaborative filtering. To find similar listeners, listener similarity is calculated by cosine coefficient from triple feature spaces. We regard listeners as similar over a certain threshold. Then, the similarity of listener's situation and the situation in which each playlist is created is derived. From playlists over certain thresholds, one initial playlist is sought that considers listener and situation similarities.

Second is the transcoding phase where the suitability of the initial playlist is improved to match listener preference. In concrete terms, tunes that listener give feedback as unsuitable are removed from the playlist. At the same time, the system introduces several tunes that listener is not listened into the playlist at the rate of 30%. By this process, the listener is able to listen new tunes moderately.

Alternative tunes are found by using similarity between the user and each tune. In advance, the user is mapped to each feature space. From the user's preference and listening situation, similarity is calculated by using cosine coefficient and the weight of each feature space. This process is equal to the calculation of musical similarity.

Through the above process, an improved playlist is displayed, as shown in Figure 2. The generated playlist is displayed on the left side, and the playlist displayed on the right side is initial playlist sought by collaborative filtering. A playlist player is embedded above the playlist. Listeners can operate the player as well as general music players.

The user actually can enjoy the playlist and offer such feedback to the system as whether each tune is suitable for preferences and situation. Then the system presents a fine-tuned playlist according to these feedback. Besides, the system updates the user's preferences by using such feedback.

In concrete terms, the base vectors of each feature space are transformed. The following formula transforms the base vector of the lyrical feature space:

$$\text{base}_u = \text{normalize}(\text{base}_u + \delta \frac{l_f - l_u}{\|l_f - l_u\|} - \delta \frac{l_d - l_u}{\|l_d - l_u\|})$$

Lyrical base vector of user  $u$  is depicted as  $\text{base}_u$ . Feature value  $l_f$  means the average feature value of favorite music. On the contrarily,  $l_d$  is the average feature value of unfavorable music. The rate of transformation is depicted as  $\delta$ . Though the propriety the value of  $\delta$  is still under consideration, we allocate 0.1 to it. In this transformation, base vectors are expanded in the direction of the dislike feature value and flexed in the direction of the favorite feature value. As concerns the base vectors of the music scene and listening situation, they are also transformed by same method.

The system preserves user's feedback such as favorite tunes, tunes suitable for each situation and generated playlists. These are applied to collaborative filtering and transcoding.

### 2.3. Acquisition of listening history and its reflection to user profile

Generally, as ideal information introduced into user profiles, the history of appreciated contents is given. To collect the information, the system embeds a playlist player into the web browser that users can operate as popular music players. The player send users' operation histories to the system as frequency at which the music was listened to. Then, user profiles were updated with the information.

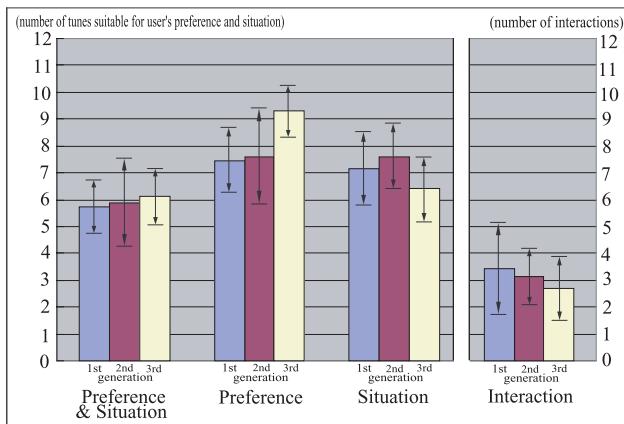
Such operation histories aren't useful for updating user preferences, but they are useful as musical annotations. Annotations, in the shape of listener aspects, are not equally collected by each tune. So music that lacks annotation can compensate by using information of operation histories. For example, when music is often listened to in certain situations, the music is suitable for those situations.

## 3. Experimentation

We used 230 pieces (100 from the RWC music database [12] and 130 from Japanese pop tunes). For preparation, annotations of musical scene and listening situation was collected for each tune.

We evaluated how well our system works with seven subjects by the following process: Listening situations are assumed freely to each subject and are input to our system with a situations declaration form. Then the system presents playlist suitable for subjects' preferences and situations. Next, the system let subjects express their preferences with buttons on a screen, and refined playlists are presented again. The process is repeated until the playlists become completely satisfied. The playlist generation process is repeated three times.

Our system can consider user's preference, situation and the number of interaction when generating a playlist as described in Section 2.2, and we evaluated our system under



**Figure 3. Results of experiments**

four conditions in terms of the parameters taken into account: preferences, situations, preferences & situations and the number of interaction.

There are four groups in Figure 3, each group is further made of three bars; a three-bar group corresponds to each condition. The three bars depict first, second, and third generations of refining a playlist, respectively. The arrowhead of each bar denotes standard deviation.

For the condition of preference only shown in the second group in Figure 3, the amount of tunes suitable user's preference increased; this means that the system could provide a better playlist by repeated playlist generation. Additionally, the number of interactions shown in the right most group decreased. The number of tunes suitable for preferences and situations shown on the left group also increased. Thus, subjects get satisfactory playlists provided a better playlist as a generation proceeded.

On the other hand, the number of tunes suitable for situations, shown at the third group from the left, indicated no tendencies after repeated playlist generation.

This is because when the system infers information of user's preference, we have faced two problems: (1) the number of situations is relatively larger than that of tunes (so called the sparseness problem), and (2) annotations are not added to all tunes impartially. We call the latter a biased annotation problem. To solve the problem, it is necessary to guide a user to pay attention to tunes that lack annotations. Moreover, it seems to be efficient to apply users' listening operation histories to the annotations. If a tune is listened to many people under certain situations, it can be regarded that the tune is suitable for the situation. Though, reliability of the information generated by computer is lower than the information created by users. When the information is applied to the system, we have to take account of the weight of each synonymous kind of annotation.

## 4. Summary

For the realization of playlist-mediated communication, we aim to encourage of playlist activity like generation, recommendation and appreciation. As a starting point, we have developed a playlist recommendation system. Using our system through the experiments, we have found that annotation can describe more precisely and dynamically an individual user's mental state and situation and each relationship of a user and a tune. At the same time, we found that collaborative filtering, transcoding and interaction are effective to generate playlists suitable for users. Additionally, we have accomplished the facilitation of playlist generation and recommendation.

Finally, we briefly mention future work. The playlists generated by our proposed system are supposed to be shared among many people. In the situations, we suggest that interaction through playlists among them helps their smooth communications. Therefore, we are implementing a playlist annotation system that enables users to add remarks to the playlists. Furthermore, we think it is important to develop a method to enhance the appreciation of the playlists.

## References

- [1] M. Anderson, M. Ball, H. Boley, S. Greene, N. Howse, D. Lemire and S. McGrath, "RACOFI: A Rule-Applying Collaborative Filtering System," *Proceedings of COLA'03*, 2003.
- [2] U. Shardanand and P. Maes, "Social Information Filtering: Algorithms for Automating "Word of Mouth"," *Proceedings of CHI*, 1995.
- [3] S. Pauws and B. Eggen, "PATS: Realization and user evaluation of an automatic playlist generator," *Proceedings of ISMIR*, 2002.
- [4] J. H. Lee and J. S. Downie, "Survey of Music Information Needs, Uses, and Seeking Behaviours: Preliminary Findings," *Proceedings of ISMIR*, 2004.
- [5] audioscrobbler, <http://www.audioscrobbler.com/>, 2002
- [6] Last.fm, <http://www.last.fm/>, 2002
- [7] Katashi Nagao, *Digital content annotation and transcoding*, Artech House Publishers, London, 2003.
- [8] J. R. Smith and B. Lugeon, "A Visual Annotation Tool for Multimedia Content Description," *Proceedings of the SPIE Photonics East, Internet Multimedia Management Systems*, pp.49–59, 2000.
- [9] E. Daniel, B. Whitman, A. Berenzweig and S. Lawrence, "The Quest for Ground Truth in Musical Artist Similarity," *Proceedings of ISMIR*, 2002.
- [10] B. Logan, A. Kositsky and P. Moreno, "Semantic Analysis of Song Lyrics," *IEEE International Conference on Multimedia and Expo (ICME)*, 2004.
- [11] G. Salton and C.S. Yang, "On the specification of term values in automatic indexing," *Journal of Documentation*, 29:351-372, 1973.
- [12] Masataka Goto, "Development of the RWC Music Database," *Proceedings of the 18th International Congress on Acoustics (ICA 2004)*, pp.I-553-556, 2004.