Music Playlist Recommender System AFT-IS

Shobu Ikeda
Ritsumeikan University
1-1-1, Noji-Higashi, Kusatsu
Shiga 525-8577, JAPAN
Telephone number, incl. country code
is0152px@ed.ritsumei.ac.jp

Kenta Oku
Ryukoku University
1-5 Yokotani, Seta oe-cho, Otsu
Shiga 520-294, JAPAN
Telephone number, incl. country code
okukenta@rins.ryukoku.ac.jp

Kyoji Kawagoe
Ritsumeikan University
1-1-1, Noji-Higashi, Kusatsu
Shiga 525-8577, JAPAN
Telephone number, incl. country code
kawagoe@is.ritsumei.ac.jp

ABSTRACT

Previously, we had proposed a playlist recommendation method to suggest music by considering the change in acoustic features in the song so that the transition between songs becomes smooth. Our previous method uses the last two songs in a playlist to make recommendations for following songs that have a smooth transition of acoustic features from the current songs. However, in this previous method, if two or more songs are not given, it is not possible to recommend a suitable next song. Our method proposed in this paper considers the transition of the music that was last played and recommends the next song. Moreover, we have developed a playlist recommender system using our proposed method. As the user inputs information necessary for creating a playlist, the system outputs a playlist with a smooth transition of acoustic features.

CCS Concepts

• Information systems→Recommender systems

Keywords

Music Recommendation, Playlist Recommendation, Recommender System.

1. INTRODUCTION

Portable audio players, such as iPods or smartphones, and music cloud services like Google Play Music or Spotify, enable users to easily access a large number of songs anytime and anywhere. In this environment, users often find it difficult to select appropriate songs from a large number of songs. Therefore, a song recommender system that is personalized for each user' situation is necessary. In addition, studies on song recommendation have been conducted. When users want to enjoy music, they often listen to playlists.

A playlist is a sequence of songs [1]. Many music playlist recommendation methods have been investigated [2, 3, 4]. The purpose of playlist recommendation is to generate a music sequence that matches the characteristics of the target playlist.

ICCAE 2018, February 24–26, 2018, Brisbane, Australia

ACM ISBN 978-1-4503-6410-2/18/02…$15.00

DOI: https://doi.org/10.1145/3192975.3193019

Previously, we proposed the first method [4] for recommending the next song based on acoustic feature transition to generate a playlist. The next song is selected to naturally transition from the two most recent songs included in the playlist. In our previous method, we constructed the music feature space based on the music's acoustic features. When the first song and the second song are given, the third song is selected in the direction of the vector constructed from the two song points. Therefore, the third song smoothly transitions from the two previously played songs. After that, we proposed the second method [5] to recommend the next song so that the acoustic feature transitions smoothly using only the song that was last played. Our proposed method divides the song into multiple songs, and every acoustic feature is extracted. After that, based on the extracted acoustic features, a song set is mapped to a two-dimensional feature space. Our method selects the next song to be recommended in the two-dimensional feature space. From the evaluation of the experiment by subjects, our method has shown effectiveness.

In this paper, we describe a playlist recommender system named AFT-IS that we have developed based on our proposed method. The user inputs parameters necessary for creating a playlist and the system outputs a playlist with a smooth transition of acoustic features.

2. RELATED WORK

This section describes related work on the automatic playlist generation and recommender system.

The main approach of playlist generation is to find songs similar in characteristics to the seed song. Flexer et al. [3] proposed a method focusing on the transition of songs included in the playlist. Our previous study [4] had also considered the music acoustic feature transition in the playlist. Our previous method had recommended the next song on the basis of the preceding two songs in the playlist. the next song is recommended based on the latest songs during playlist playback, so that the recommendation can respond flexibly to the current playback situation. However, there was an insufficient correspondence with the songs whose characteristics changed. An instance of this issue was the gradual rise in music tone with the passage of time in a quiet melody. Therefore, we proposed a new music playlist recommendation method [5] which considers the acoustic feature transition in the music.

Many studies on music recommender systems have been conducted. Martin et al. [6] developed a music recommender system based on factorization machines that extracts information about the user's context from names of the user's playlists. Martin et al. calculated clusters of contextually similar tracks. Ja-Hwung et al. [7] proposed the hybrid recommendation system integrating user ratings and music low-level features. This system implements item-based collaborative filtering to predict user ratings.

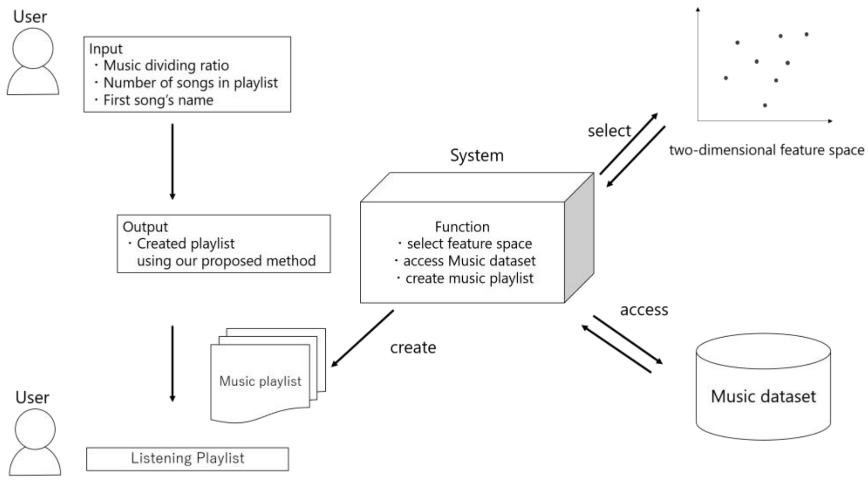

Figure 1. The composition of the system.

3. AFT-IS

AFT-IS is our recommended system. Figure 1 shows the composition of the system. First, the user inputs the necessary parameters. Then, the system receives the user's input parameters, accesses the music dataset and selects a feature space. Then, the system outputs a playlist based on the received information. At this time, the system uses our proposed method to generate the playlist. The user can listen to the outputted playlist.

3.1 Functions

The behavior of the AFT-IS changes according to the parameters entered by the user. In this section, we explain each system's behavior.

AFT-IS requests input of three parameters from the user. The three parameters are dividing parameters, number of songs in playlist, and first song's name.

The first parameter, dividing parameters, selects the music division ratio which is to be considered while creating a playlist. In AFT-IS, it has been made possible to decide from 3 types of division ratios of 10%, 30% and 50%. AFT-IS uses the ratio input by the user to select the two-dimensional feature space used to create the playlist.

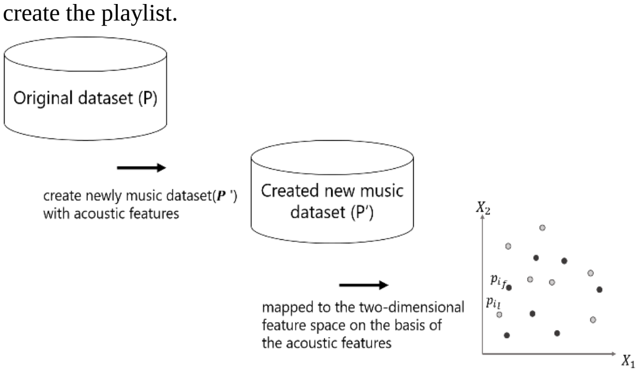

Figure 2. Base method's flow.

The number of songs in playlist, can be used to specify the number of songs in the playlist. The user inputs this parameter as an integer. The system creates a playlist with the number of songs as specified by the parameter received from the user.

The third parameter, first song's name, is a parameter where the user inputs the name of the first song of the playlist. AFT-IS creates a playlist with the song entered by the user as the seed song of the playlist.

3.2 System Operating Environment

In this session, we will explain the operating environment of ATF-IS.

・OS version is Windows 10 pro 1709

・PC processor is Intel (R) Core™i7-4770 CPU @3.40 GHz

4. PLAYLISTS CREATION METHOD

AFT-IS is using our proposed method as the base method to create playlists. In addition, in this system, we implemented a function that extended the base method. In this section, we explain the base method and the extension method.

4.1 Base Method

Figure 2 shows the flow of base method. Suppose that we have an original music dataset of songs, $P = \{p_i\}$. First, the base method divides each song, $p_i \in P$ into the first half and the second half and a new dataset of songs P', which includes both the first half p_{i_f} and the second half p_{i_l}, is created. The division ratio at this time is 50%. Then, the acoustic features are extracted from every song included in P'. After that, the song set P' is mapped to the two-dimensional feature space on the basis of the acoustic feature. Finally, the next song is searched to be recommended on the mapped two-dimensional feature space. We use the method based on cosine similarity to map to the two-dimensional feature space. Previously, we had evaluated the mapping method of the two-dimensional feature space and got the highest evaluation [8]. For that reason, also used the method based on cosine similarity in the base method.

Figure 3 represents a procedure of dividing a song and creating a new dataset. First, each song included in the music dataset (P), given in advance, is divided into two songs. The dividing ratio of the base method is set to 50%. After that, base method registers each divided song in the newly created music dataset (P'). Then, the acoustic features of all songs included in the newly created

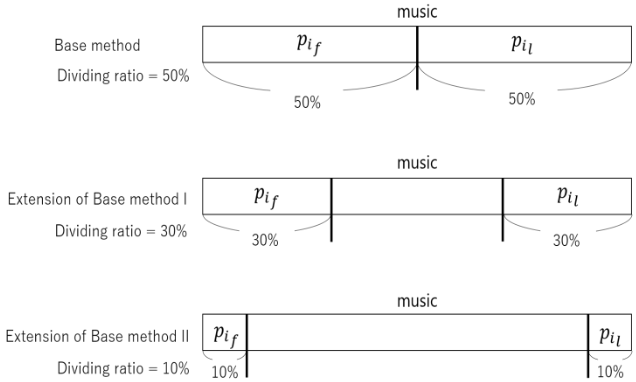

Figure 3. division ratio of song of each method.

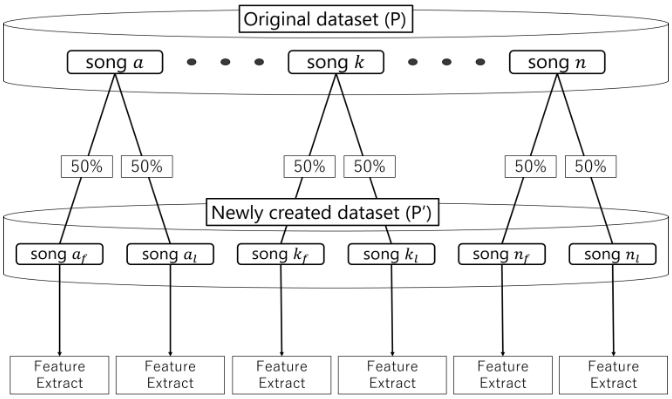

Figure 4. Divide music and create new dataset of base method.

music dataset are extracted. We extracted the acoustic features using MARSYAS [9]. MARSYAS is an open source software framework for audio processing. By using MARSYAS, we obtain a 34-dimensional acoustic feature vector from the set of songs P'. The feature amount can be broken down into Time zero-crossings, spectral centroid, flux and roll-off, and Mel-Frequency Cepstral Coefficients(MFCC).

The base method, given the first song $p_{t-1}$of the playlist, the next song p_t is recommended using the following procedure. First, a straight line p_{t-1}, passing through the first half $p_{t-1_f}$and the second half p_{t-1_l} of the song p_{t-1}, is detected on the two-dimensional feature space. Next, a search area is set on an extended line of the vector $(\overrightarrow{p_{t-1_f}\ p_{t-1_l}})$. The search area is a fan-shaped area with radius r and center angle θ with the extension line of the straight line as the center line. The next song p_t will be selected from the songs in which the first half of the songs are included in the search area. The Euclidean distance between the straight line p_{t-1} and the first half of each song included in the search area is calculated, and the song at the smallest distance is recommended as the next song p_t. Then, the system registers p_t as the second song of the playlist. The third song $p_{t+1}$of the playlist is determined using p_{t_f} and p_{t_l}. The system ends the procedure when the number of songs in the playlist reaches the number specified by the user through the input parameter. If there is no song in the search area, the song not included in the playlist and nearest to the song p_{t-1_l} is set as the recommended song p_t as an exceptional process.

The parameter of the base method is set to center angle $\theta = \pi / 2$, and radius r so that the search area contains up to 100 songs. The value of r changes each time a given song changes.

4.2 Extension Method of the Base

In this paper, we expanded the base method and created two extension methods.

4.2.1 Extension methods

The flow of extension method is roughly same as the base method, the extended part is in Figure 2, “Created new music dataset”.

Suppose that we have the original music dataset of songs $P = \{\mathrm{p_i}\}$. The extension method also divides each song $p_i \in P$ into the first half and the second half and a new dataset of songs P', which includes both the first half $p_{i_f}$and the second half p_{i_l}, is created. The division ratio of extension method is different from the base method.

The base method divides songs into first and second halves, which is the simplest division method. However, our previous study [5] suggested that accuracy will be affected by changing this division ratio. So, in this paper, we expanded the base and created two extension methods that changed the division ratio.

Figure 4 shows the division ratio of songs. The division ratio of the base method was 50%, while extension method ratios were 10% and 30%. In other words, p_{i_f} of the extension method *I* extracts only 30% of the first half of the song, and the p_{i_l} extracts only 30% of the second half of the song. The p_{i_f} of extension method *II* extracts only 10% of the first half of the song, while p_{i_l} extracts only 10% of the second half of the song. Parts not included in p_{i_f} and p_{i_l} of the extension methods *I* and *II* are not extracted as an acoustic feature. As for the method for extraction of acoustic feature extraction, we use MARSYAS [9] as well as the base method. Also, the method of mapping two-dimensional feature space uses the method based on cosine similarity as well as the base method.

4.2.2 Search for next song

Both the extension methods search for the following song to be registered in the playlist using the same method as the base method. Therefore, the extension methods request the user to input the first song in the playlist as the seed song. After that, extension methods search for songs based on the seed songs specified by the user and generate playlists.

The parameter of extension methods is also set to center angle $\theta = \pi / 2$, and radius r so that the search area contains up to 100 songs. The value of r changes each time a given song changes.

4.3 Comparative Evaluation

We evaluated the three types of performance, the base method and two extension method from the viewpoint of users as to whether playlists with smooth music transition are generated.

4.3.1 Dataset

In this evaluation, we used classical music from the "Classical masterpiece sound library" (http://classical-sound.seesaa.net/) as a dataset. We used 666 songs such as symphony, concerto, orchestral music, wind music, string orchestra etc. released on this site. We obtained permission for the use of music from the creator of this site.

4.3.2 Evaluation procedure

We conducted a subject experiment by the following procedure.

(1) Subjects arbitrarily select music from the music dataset. AFT-IS registers the music selected by the subject as the first song of the playlist.

(2) Three types of playlists I, II and III created by the base method and the two extension methods are presented to the subjects. At this time, every playlist is composed of five songs including the songs selected by the subject.

(3) The subject listens to the songs in the playlists I, II and III from the first song and ranks the playlists in the order of feeling that the transitions of the songs are smooth.

The above procedures (1) to (3) were performed three times for each subject. Also, in order to cancel the order effect, the method of playlists I, II and III was exchanged randomly every time to conduct the subject experiment.

4.3.3 Result

The number of subjects was 8. They are all in their 20's. Table 1 shows the number of times each method was ranked in the order of feeling that the transitions of the songs are smooth. Table 1 shows the number of times each method was ranked. In Table 1, the extension method *I*, dividing ratio is 30%, was the most frequently ranked the first and the number was 12. This result shows that the extension method *I* has the best recommendation with a probability of 50%. Also, the extension method *I* was the lowest frequently ranked the worst and the number was 4. This result shows that the possibility that the extension method *I* will make a low precision recommendation is 16%.

Table 2 shows the average rank of each method. This result shows the extension method is the highest performance to recommend the playlist which has smooth transitions of the acoustic features.

Table 1. The number of times each method was ranked

Method (Dividing ratio)	**1st rank**	**2nd rank**	**3rd rank**
Base method (Dividing ratio = 50%)	5	8	11
Extension method *I* (Dividing ratio = 30%)	**12**	8	**4**
Extension method *II* (Dividing ratio = 10%)	7	8	9

Table 2. The average rank of each method

	Base method	**Extension method *I***	**Extension method *II***
Average rank	2.15	**1.81**	2.03

5. CONCLUSION

In this paper, we developed a playlist recommender system AFT-IS. AFT-IS uses our proposed playlist recommendation method that takes into account the transitivity of the acoustic features of songs to create playlists. By inputting three parameters, the user can obtain a playlist that matches the user's requirements from the system. In addition to our previously proposed method, AFT-IS implements two extension methods, and it has become possible to generate the playlist in three ways. And we compared the performance of the three methods. From the subject experiment, the extension method *I* got the highest evaluation.

Future work will involve evaluating and improve AFT-IS based on the feedback received from users. Also, we improve AFT-IS to visualize playlist transition to the user. And we will implement the function that the user can select the next song by seeing the playlist transition in real time in the future.

6. ACKNOWLEDGMENTS

This work was partially supported by MEXT-Supported Programs for the Strategic Research Foundation at Private Universities 2013-2017 and 2015-2019. This work was supported by JSPS KAKENHI, Grant Number 15K12151. The song dataset used in this research was licensed by the site creator of "CMSL Classical masterpieces Sound Library" (http://classical-sound.seesaa.net/). We would like to thank Editage (www.editage.jp) for English language editing.

7. REFERENCES

[1] Geoffray Bonnin and Dietmar Jannach. 2014. Automated Generation of Music Playlists: Survey and Experiments. *Comput. Surveys* 47, 2 (nov 2014), 1–35.

[2] K. Tada, R. Yamanishi, S. Kato. 2012. Interactive Music Recommendation System for Adapting Personal Affection. In *Proc. International Conference on Entertainment Computing*, Lecture Notes in Computer Science (LNCS), Vol. 7522, 417-510.

[3] Arthur Flexer, Dominik Schnitzer, Martin Gasser, and Gerhard Widmer. 2008. Playlist generation using start and end songs. In *Proc. ISMIR Conference '08*, 173-178.

[4] S. Ikeda, K. Oku, K. Kawagoe. 2016. Music Playlist Recommendation Using Acoustic-Feature Transitions, In *Proc. the Ninth International C* Conference on Computer Science & Software Engineering*, 115-118.

[5] S. Ikeda, K. Oku, K. Kawagoe, 2017. Music Playlist Recommendation Using Acoustic-Feature Transition inside the Song, In *Proc. the 15th International Conference on Advances in Mobile Computing and Multimedia, Show2017*, 216-219,

[6] Martin Pichl, Eva Zangerle, Günther Specht, Improving Context-Aware Music Recommender Systems: Beyond the Pre-filtering Approach. In *Proc. ACM International Conference of Multimedia Retrieval 2017*, 201-208

[7] Ja-Hwung Su, Ting-Wei Chiu, An Item-Based Music Recommender System Using Music Content Similarity. *In Proc. Asian Conference on Intelligent Information and Database Systems 2016, Part II*, 179-190

[8] S. Ikeda, K. Oku, K. Kawagoe. 2017. Analysis of Music Transition in Acoustic Feature Space for Music Recommendation. In *Proc. the 9th International Conference on Machine Learning and Computing*, 77-80.

[9] George Tzanetakis and Perry Cook. 2000. MARSYAS: A Framework for Audio Analysis. Organized Sound, 4(3), 169-175