A MUSIC RECOMMENDER FOR A GROUP OF PEOPLE

Jun-ichi Suzuki
Nihon University
junichi@kthrlab.jp

Naoyuki Suetsugu
Nihon University
suetsugu@kthrlab.jp

Tetsuro Kitahara
Nihon University
kitahara@chs.nihon-u.ac.jp

ABSTRACT

A variety of techniques for music recommendation have been proposed so far, but only a few attempts have been made to generate a playlist from music collections owned by multiple users. In this paper, we achieve a system that generates a playlist that include songs unknown to each user from the music collections stored in individual tablets owned by multiple users, in order enable users to enjoy an encounter with unknown songs. Experimental results showed that the playlist generated by our system contained songs that were unknown but match the participants' preferences.

1. INTRODUCTION

There are often opportunities, such as home parties and drives, where multiple people gather and enjoy listening to the same background music (BGM) at the same time. In such situations, some one will select and play back the BGM. However, it is not easy to select BGM that pleases as many people as possible.

There have been studies on music recommendation for a group of people [1] [2] [3], but they did not deal with support of encounters with unknown songs. If the system generates a playlist that contains songs unknown to but potentially favored by each participant, he/she may enjoy to discovery new songs.

Here, we propose a system that recommends songs that a person favors and that another person do not know but possibly favors. The posssiblity of favor is estimated through the similarity to his/her favorite artists. We implemented this system as software running on Android devices communicating with each other via Bluetooth.

2. SYSTEM OVERVIEW

The purpose of this study is to devise a device that will select appropriate BGM in a situation where a group of persons (e.g., family, friends) listen to the same BGM at the same time and in the same place. In such a situation, There are two technical issues:

Issue 1: Intergrate the music collections that are distributed across more than one device seamlessly.

Issue 2: Determine how likely each musical piece would be preferred by those other than the owner of that piece.

To address Issue 1, the devices will communicate with each other through Bluetooth, and data from each device's music collection is integrated at the master device. After the master device generates a playlist, the device storing each song plays it back automatically by connecting to a Bluetooth speaker.

To address Issue 2, a similar artist search function in the Last.fm Web API is used. Under the assumption that people may favor an artist if similar to their own favorite artists, our player may select songs from artists similar to those on a person's favorite artist list.

Our player runs based on the following procedure. Below, we suppose that there are n persons who are going to listen to BGM together. Their Android devices are denoted as $D_0, D_1, \cdots, D_{n-1}$, where D_0 is the master device.

1. Each person launches our player on his/her Android device, then the master device searches all devices executing our player in the same place and establishes a connection to every device.

2. The master device requests each device (including itself) to send information about the device owner's favorite artists to the master device. The favorite artists on device D_i is denoted by $F(D_i) = \{A_{i,1}, A_{i,2}, \cdots\}$. Up-to-five artists can be sent from each device, and can be specified by directly inputting artist names or automatically by counting playback frequencies. The master device integrates the information and obtains $F_{\text{all}} = \bigcup_{i=0}^{n-1} F(D_i)$.

3. The master device searches for artists similar to each in F_{all} using the Last.fm Web API, where the names of up to 100 similar artists can be obtained. Let $S(A)$ represent this list for artist A. Then, $S_{\text{all}} = \bigcup_{A \in F_{\text{all}}} S(A)$ is obtained.

4. The master device asks each device whether it has songs of any artists in S_{all}. Suppose D_i has such songs. It notifies the master device the most frequently played one in songs by that artist. This song is added to the playlist, defined as a sequence of tuples $(M_k, D(M_k))$, where M_k is the k-th song and $D(M_k)$ is the device where it resides.

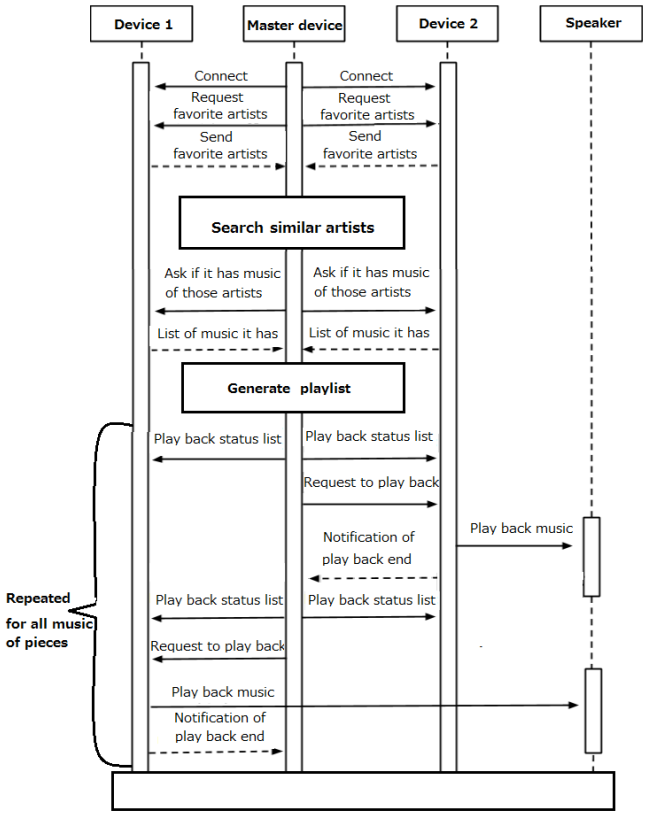

Figure 1. Flow diagram of the system.

5. For each song M_k, the master device requests $D(M_k)$ to play it back, at which point $D(M_k)$ connects to the Bluetooth speaker. At the same time, data such as the title and artist name of M_k are broadcasted to all devices and are displayed on each. Once the playback ends, $D(M_k)$ disconnects from the Bluetooth speaker to free it up for the next device. This process is repeated for each song in the playlist. The Advanced Audio Distribution Profile (A2DP) is used to connect to the Bluetooth speaker and the Serial Port Profile (SPP) is used.

3. EXPERIMENT

3.1 Experimental conditions

We conducted an experiment using this system. Three participants who knew each other made up a group, and they listened to BGM together while playing a card game. The total number of the participants was 12 (four groups). We used Android devices (a Google Nexus 7 with Android 4.4.2/4.4.4) and 120 music CDs taken from 10 of the best-reviewed artists in each of 13 genres on Amazon.co.jp. Because the information of favorite artists are not avaiable due to no playback history data inside, favorite artists were determined based on a preliminary survey.
Immediately after the playback of each song ends, the participants answered [Q1] whether they have listened to it and [Q2] whether they like it on a scale of 1 to 5.

3.2 Experimental results

The results are listed in Table 1. As shown in this table, the playlists contained 5.2 unknown pieces on average, and an average of 2.4 unknown pieces were preferred by the participants. In particular, Participants A, E, and J were pleased with half or more of the unknown pieces.

Participants D, F, and K did not highly evaluated any unknown songs. One possible reason for participants F and K would be that the number of the unknown songs were very small (one or two).

Table 1. Number of recommended songs, unknown songs, unknown and highly evaluated songs, and the average of ratings for Q2.

Participants		Recommended	Unknown	Unknown highly eval.	Pref. eval.
Group 1	A	12	7	4	2.8
	B	5	2	2	3.6
	C	5	1	1	4.6
Average		7.3	3.3	2.3	3.7
Group 2	D	6	5	0	1.8
	E	8	7	5	1.8
	F	2	1	0	2.0
Average		5.3	4.3	1.7	1.9
Group 3	G	7	5	3	2.6
	H	10	7	2	2.7
	I	8	5	3	3.3
Average		8.3	5.7	2.7	2.9
Group 4	J	14	9	6	3.6
	K	5	2	0	3.4
	L	14	11	3	3.3
Average		10	7.3	3	3.3
Total average		8	5.2	2.4	2.9

4. CONCLUSIONS

In this paper, we proposed an Android music player that enables a group of persons to share and listen to music at the same time and in the same place. Future work includes first conducting an experiment using the participants' devices and music collections and second, improving favorite artist identification by exploiting various other cues such as Tweets, Amazon shopping histories, and playback histories in other applications.

This work was supported by JSPS KAKENHI Grant Number 26240025.

5. REFERENCES

[1] Sihem Amer-Yahia et al., "Group recommendation: semantics and efficiency", In *Proc. VLDB Endow. 2, 1*, 754-765, 2009.

[2] Andrew Crossen et al., "Flytrap: intelligent group music recommendation", In *Proc IUI '02.*, 184-185, 2002.

[3] George Popescu and Pearl Pu. "What's the best music you have?: designing music recommendation for group enjoyment in groupfun", In *Proc CHI '12 Extended Abstracts*, 1673-1678., 2012.