

Designing a Voting Mechanism in the GroupFun Music Recommender System

George Popescu

École Polytechnique Fédérale de Lausanne, Switzerland
georgepopescu2003@gmail.com

Abstract. In this paper we present the implementation of a truthful preference elicitation algorithm that favors item diversity in our group music recommender system, GroupFun. First, we discuss our methodology and relate to other group decision mechanisms already implemented in state of the art group recommender systems. Then, we formalize the implementation of the probabilistic weighted sum algorithmic in GroupFun. In addition to this we discuss user strategies and incentives for submitting ratings and show how PWS favors music diversity maximizing the group's overall satisfaction. Due to space constraints in the following we highlight only the main characteristics of our system presenting its design, interfaces and user actions.

1 “Home” Page in GroupFun

GroupFun is a Facebook application available at the address and hosted at EPFL. The “Home” page contains the visual identify of the GroupFun and three playlists: Top 8 GroupFun, Christmas and Lausanne Party. Three entities are samples of what GroupFun can have as output, as shown below.



Fig. 1. "Home" tab

2 My List

Users can create their own playlist from a number of about 10,000 songs. After the playlist is created, the user can rate the songs, as in the Figure 2. The music player, soundmanager, can help the user to take the right decisions. The user can edit his/her playlist and add/remove songs from the playlist.

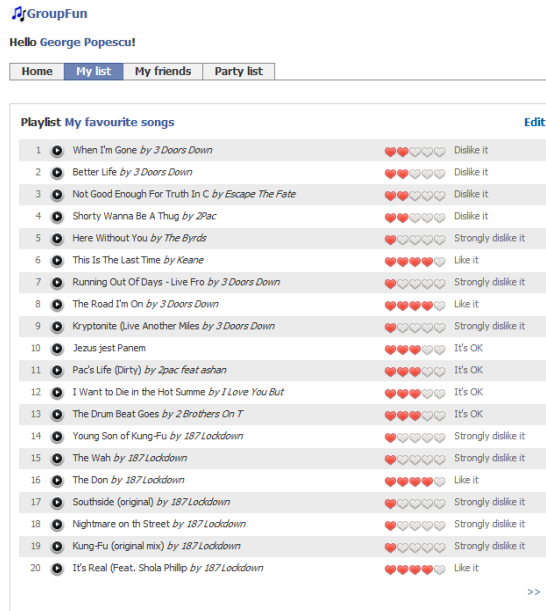


Fig. 2. "My list" tab

3 My Friends

The user can invite his friends to use the application and check their activity: whether they accepted or not the invitation and what are their music preferences. In the implementation, we used the standard Facebook request fb: multi-friend-selector, customized with 6 maximum invitations and 5 friends per column. The activity of user's friends in GroupFun is additionally available, in case that he wants to check their music preferences. This feature increases the interaction within a group of friends, as some users can rate the songs already rated in the system. A preview is available in the figures below.



Fig. 3. "My friends" tab

4 My Scrobbler

Using the Last.fm music recommender system called “Audioscrobbler”, we imported users data into GroupFun by taking advantage of the profile of each user's musical taste after recording details of the songs the user listens to, either from Internet radio stations, or the user's computer or many portable music devices. This information is transferred to Last.fm's database (“scrobbled”) and then scrobbled again into GroupFun. The profile data is then displayed on the user's profile page.

5 Discussion

The total number of users, n , we currently have logged into our system is between 200 and 300 users. This is not a strict limitation since a virtually infinite number of Facebook friends can access the shared space of GroupFun in the same time. Some users may be friends with one-another. In this regards they may be strongly (or weakly) influenced by other friends’ (or strangers’) choices.

GroupFun offers unbounded preference elicitation all users: they can choose which and how many songs as desired in constructing their individual playlist. Songs can be extracted from a list of popular songs (like billboards) or can be searched within a shared database: the filtering can be set upon artists’ names, song name, album, genre, etc. Users also may directly recommend music to other users or (sub) groups of users (eventually friends but not necessary) – they can send e-mail notifications to other user profiles using the same platform or they can interact verbally being in the same place.

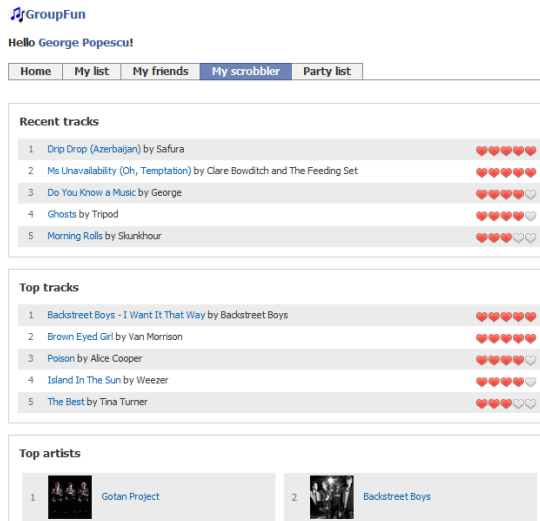


Fig. 4. "My scrobbler" tab

With respect to information overload, computational limitations and users' ratings we include an incomplete ranking of all items: users do not need to know all songs because they only vote for the songs they are interested in – giving their vote to songs which are relevant for them. The amount of user interaction through GroupFun is user-dependent: individuals rate either the available songs in GroupFun or the displayed ones. Rating all group songs would be very time consuming considering users' time and interest in doing so. Here we rely on recent research that has provided design guidelines for reducing users' effort by focusing on preference dynamics.

In terms of the length of the common playlist, it can virtually be infinite. However, this case does not represent an interest research case since songs will only need to be ordered by their final group score and users will get to see all songs (e.g. 300 friends composing a list of 1,000 songs). Since they will not be able to listen or visualize them all we aim at recommending top k songs based on the probabilistic weighted sum algorithm, where k is group-specified (the group agrees on a common value for k for a specific event, e.g. Birthday party).

6 User Feedback

We used GroupFun in a series of pilot tests and targeted user experiments. The general feedback we received was positive, users emphasizing the need to “discover other people's tastes” that “match their own”. From an interaction point of view our users mentioned the fact that they would prefer a system that is easy to use and intuitive, and would allow all members to have the same importance in choosing the common music playlist. The system's algorithm would then benefit the group by suggesting a list of songs that would favor everyone. Individuals also expressed their interest in using the predefined playlists, which would include recent songs, or are event-specific. Furthermore, our test users suggested that since both the music itself and the people forming the groups are equally important, it could be interesting to implement an algorithmic mechanism that would allow groups to compete among them-selves with respect to their music preferences, ratings, listening times, interaction, etc.