Dynamic Playlist Generation Based on Skipping Behavior

Elias Pampalk[1], **Tim Pohle**[1], **Gerhard Widmer**[1,2]

[1]Austrian Research Institute for Artificial Intelligence (OFAI)

[2] Department of Computational Perception, Johannes Kepler University Linz, Austria

ABSTRACT

We present and evaluate heuristics to adapt playlists automatically given a song to start with (seed song) and immediate user feedback. Instead of rich metadata we use audio-based similarity. The user gives feedback by pressing a skip button if they dislike the current song. Songs similar to skipped songs are automatically removed. We evaluate the heuristics with hypothetical use cases. Our results show that it is possible to drastically reduce the number of necessary skips.

BASIC ASSUMPTIONS

1. A seed song is given.
2. A skip button is easily accessible.
3. All the user is willing to do is press the skip button if the song currently played is a bad choice.

MUSIC SIMILARITY

We use a combination of spectral similarity [1, 2], fluctuation patterns [3], and other descriptors. Details are given in [4].

DATA

We run our experiments on a collection of 2522 tracks from 22 genres. The genres are user defined and inconsistent. In particular, there are two different definitions of trance. Furthermore, there are overlaps, for example, jazz and jazz guitar, heavy metal and death metal etc.

HEURISTICS

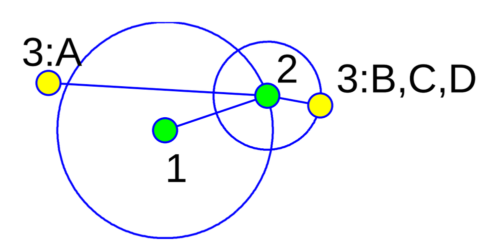
To model the user's preferences we use simple heuristics. Candidate songs are all songs in the collection which have not been played (or skipped) yet.

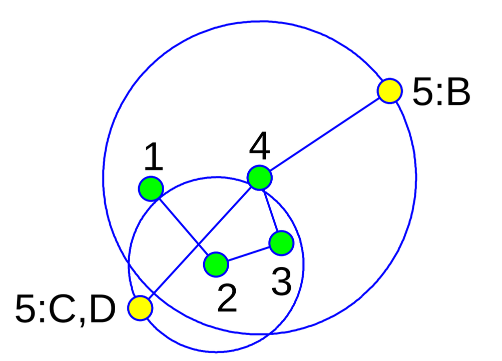
A. The songs closest to the seed song are played [5]. (The list is static.)

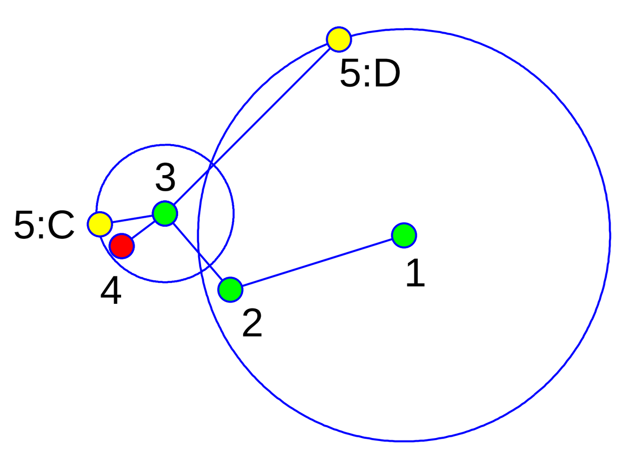
B. The song closest to the last accepted song is played.

C. The song closest to any of the accepted songs is played [6].

D. For each candidate song, let d_a be the distance to the nearest accepted, and let d_s be the distance to the nearest skipped. If $d_a < d_s$, then add the candidate to the set S. From S play the song with smallest d_a. If S is empty, then play the candidate song which has the lowest d_a/d_s ratio.







HYPOTHETICAL USE CASES

To evaluate the heuristics we define hypothetical use cases. We assume that the user wants to listen to 20 songs (approximately one hour of music). The number of skips are counted until these songs are played.

UC-1: The user wants to listen to songs similar to the seed. We measure this by equating similarity with genre membership. Any song outside of the seed's genre is skipped.

UC-2: The same as UC-1 except that for a not measurable reason the user dislikes one artist in the genre. This artist is randomly selected. Every time a song outside the seed's genre or from the unwanted artist is played, skip is pressed.

UC-3: The user's preferences change over time. We measure this as follows. Let A be the genre of the seed song and B a related genre which the user starts to prefer. The first 5 songs are accepted if they are from genre A. The next 10 are accepted if they are from either A or B. The last 5 are accepted if they are from B. We manually selected pairs of genres for this use case. The list of pairs can be found in the table to the right.

For UC-1 and UC-2 the evaluation is run using every song in the collection as seed. For UC-3 every song in genre A is used.

RESULTS

For UC-1 using random shuffle to generate the playlist would require up to 300 skips for half of the cases while heuristic A requires only 37, and heuristic D only 11. In general the performance increases from A to D. The large deviation between median and mean number of skips reflects the high number of outliers. For UC-3 the number of skips depends on the direction of the transition.

	Heuristic	Min	Median	Mean	Max
UC-1	A	0	37.0	133.0	2053
	B	0	30.0	164.4	2152
	C	0	14.0	91.0	1298
	D	0	11.0	23.9	425
UC-2	A	0	52.0	174.0	2230
	B	0	36.0	241.1	2502
	C	0	17.0	116.9	1661
	D	0	15.0	32.9	453

Table 1: *Skips for UC-1 and UC-2.*

Start	Goto	Median
Euro-Dance	Trance	20.0
Trance	Euro-Dance	4.5
German Hip Hop	Hard Core Rap	23.0
Hard Core Rap	German Hip Hop	14.0
Heavy Metal	Death Metal	28.0
Death Metal	Heavy Metal	3.0
Bossa Nova	Jazz Guitar	22.0
Jazz Guitar	Bossa Nova	6.0
Jazz Guitar	Jazz	18.5
Jazz	Jazz Guitar	29.0
A Cappella	Death Metal	271.0
Death Metal	A Cappella	350.0

Table 2: *Skips for UC-3 and heuristic D.*

OUTLOOK

The hypothetical use cases are very useful for fast evaluations. However, our next step will be to run small scale user tests. In particular, asking users to use the dynamic playlist generator on their private collection at home.

Directions for future work include improving the similarity measure and developing more accurate user profiles based on long term usage.

Acknowledgments

This work was supported by the EU project FP6-IST-507142 SIMAC. OFAI acknowledges the financial support of the Austrian ministries BMBWK and BMVIT.

References

[1] B. Logan and A. Salomon. A music similarity function based on signal analysis. In *ICME*, 2001.

[2] J.-J. Aucouturier and F. Pachet. Improving timbre similarity: How high is the sky? *Journal of Negative Results in Speech and Audio Sciences*, 2004.

[3] E. Pampalk. Islands of music: Analysis, organization, and visualization of music archives. MSc thesis, 2001.

[4] E. Pampalk, A. Flexer, and G. Widmer. Improvements of audio-based music similarity and genre classification. In *Proc ISMIR*, 2005.

[5] B. Logan. Content-based playlist generation: Exploratory experiments. In *Proc ISMIR*, 2002.

[6] B. Logan. Music recommendation from song sets. In *Proc ISMIR*, 2004.