the importance of song context in music playlists

Andreu Vall, Markus Schedl, Gerhard Widmer
Department of Computational Perception
Johannes Kepler University, Linz, Austria
andreu.vall@jku.at

Massimo Quadrana, Paolo Cremonesi
Politecnico di Milano, Italy
massimo.quadrana@polimi.it

Automated music playlist generation can be intuitively approached as a sequential problem. Thus, some playlists models take into consideration the *song context* (i.e., the current song and a number of previous songs) in order to predict the next song in a playlist. We conduct a numerical experiment on two datasets of hand-curated music playlists, where we compare playlists models that account for different song context lengths. The song context seems, at first, uninformative. However, we explain this effect by a strong bias in the data towards very popular songs and observe that, in fact, songs in the long tail are more accurately predicted when the song context is considered.

playlists models

training playlists:
$p_1 = (s_2, s_4, s_3, s_5, s_6)$
$p_2 = (s_4, s_3, s_6, s_5)$
$p_3 = (s_4, s_3, s_1)$

query playlist:
$q = (s_2, s_4, s_3, [s_5])$

song popularity (Pop.)

Next-song candidates are ranked by their popularity, regardless of the current and previous songs in the playlist. The popularity of a song is computed according to its relative frequency in the training playlists.

Patterns learned:
s_3 x 3, s_4 x 3, s_5 x 2, s_6 x 2, ...

Prediction: $q = (s_2, s_4, s_3)$ → s_3, s_4, s_5

song-based collaborative filtering (CF)

Next-song candidates are ranked according to their similarity to the current song, but previous songs in the playlist are ignored. Two songs are similar if they co-occur in training playlists.

Patterns learned:
$\{s_3, s_4\}$ x 3, $\{s_3, s_5\}$ x 2, $\{s_3, s_6\}$ x 2, ...

Prediction: $q = (s_2, s_4, s_3)$ → s_4, s_5, s_6

recurrent neural networks (RNN)

Next-song candidates are ranked according to the song scores predicted by an RNN as described in [1], considering the current and all the previous songs in the playlist.

Patterns learned:
$(s_4, s_3, -, s_5)$ x 2, $(s_4, s_3, -, s_6)$ x 2,
(s_4, s_3, s_1) x 1, ...

Prediction: $q = (s_2, s_4, s_3)$ → s_5, s_6, s_1

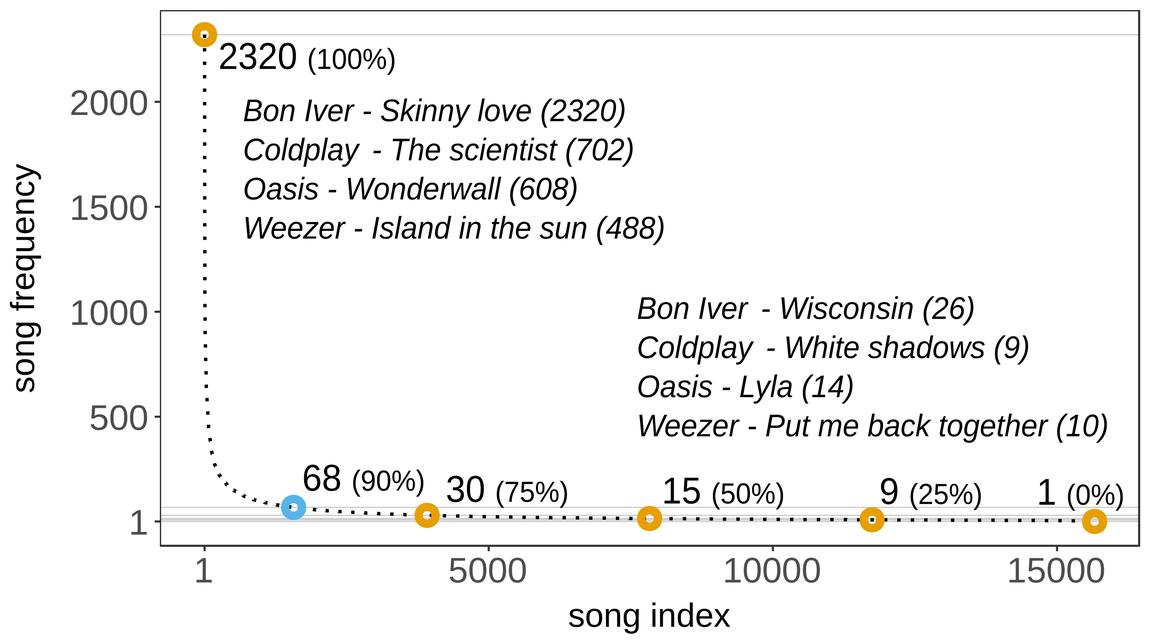
playlists datasets

	AotM-2011	8tracks
playlists	17,178	76,759
songs	7,032	15,649
artists	2,208	4,290

[1] Hidasi et al. 2016. Session-based recommendations with recurrent neural networks. In Proc. ICLR.

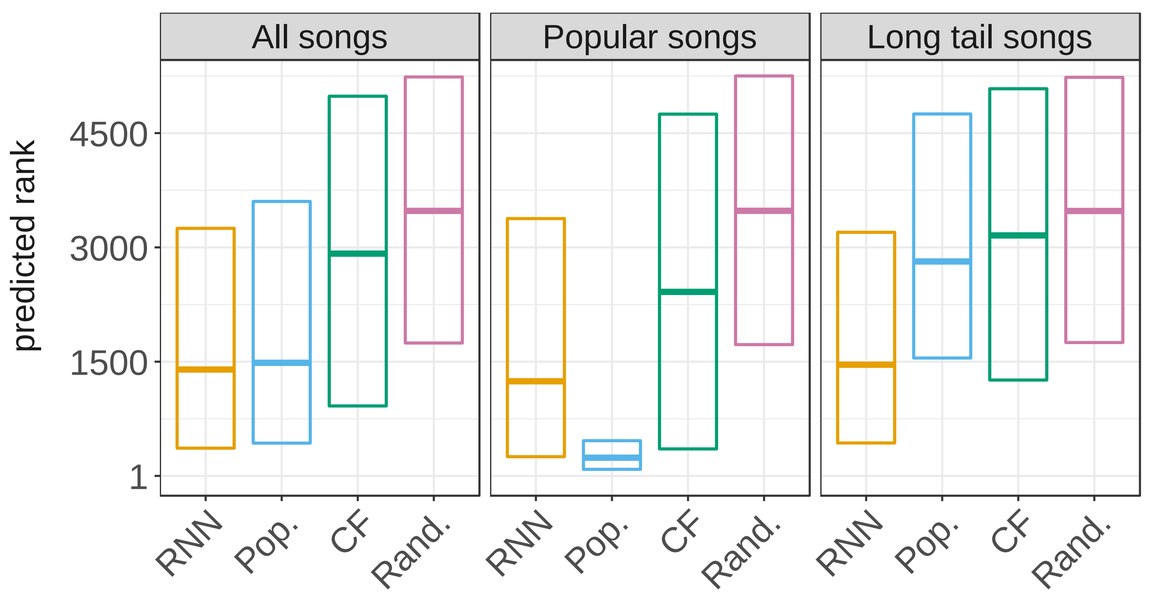
bias towards popular songs

Selected examples from the 8tracks dataset

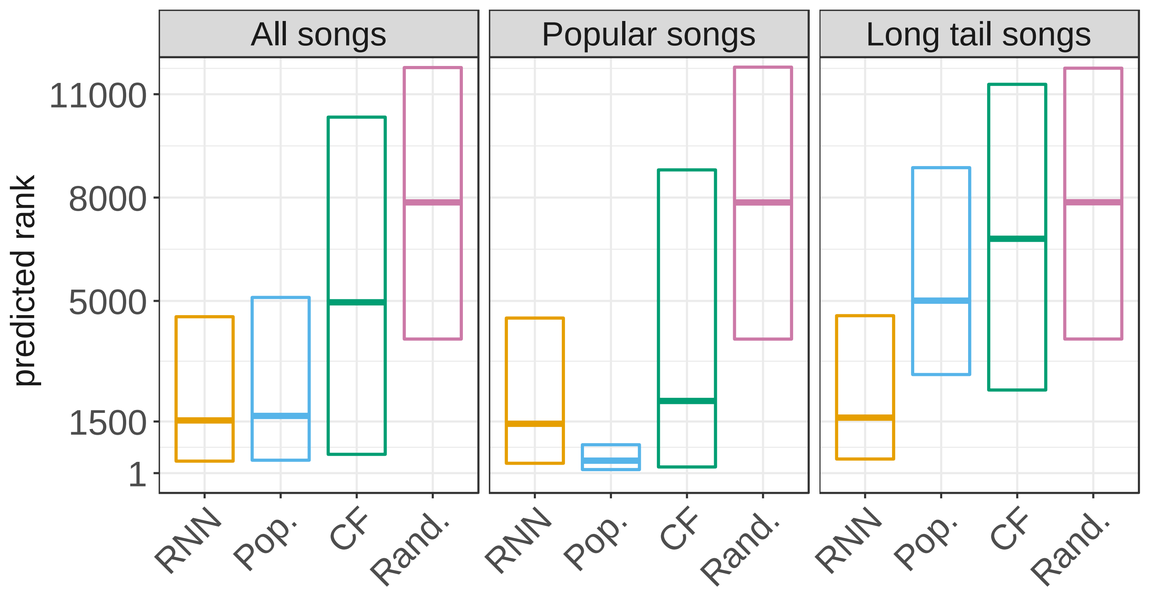


numerical experiment

AotM-2011 dataset



8tracks dataset



JOHANNES KEPLER UNIVERSITÄT LINZ

Department of Computational Perception