BM-26: Best Music Recommender

Project Type - Demo

Team - Geethik Narayana, Gulshan Mandhyan, Niraj Goel, Sandeep Gottimukkala

1. What is the functionality of your tool?

The primary aim of this tool is, given a query playlist containing some seed songs, predict a list of recommended candidate songs ordered by relevance in decreasing order. This problem is called Automatic Playlist Continuation.

2. Why would we need it?

Music recommendation is often considered to be a difficult task for a variety of reasons. The challenges, for example include that there are often millions of available tracks, there is a lack of structured or consistent meta-data or other content information, and that we often only have limited amounts of customer feedback that can be used to personalize the recommendations. Playlists, i.e., lists of sequentially ordered tracks, represent one possible approach to deal with some of these challenges and help users explore the item space. A study carried out in 2016, by the Music Business Association, showed that playlists accounted for 31% of music listening time among listeners in the USA. Another study, conducted by MIDiA, revealed that as many as 55% of streaming music service subscribers create playlists. This evidence indicates the growing importance of playlists as a mode for music consumption, and indeed the crucial necessity of developing algorithms for automatic playlist continuation.

3. Does this kind of tool already exist?

A lot of previous research has been done on Automatic Playlist Generation and Music recommendation but very little work has been done in the field of Automatic Playlist Continuation. As of today, as far as we are aware, there are no commercial tools using the algorithms for automatic playlist continuation and most of the playlists are curated manually. Hence this a novel problem that is also a part of latest Spotify RecSys challenge.

4. How do you plan to build it?

The problem is expected to involve a lot of experimentation with different algorithms and approaches in order to obtain the best results. Following are the algorithms and techniques we plan to implement and experiment with:

1. Our baseline model will be Metadata Information Retrieval. We will use textual metadata such as title of the song, artist name, album name etc. to find songs by using some sort similarity measure.
    - We aim to investigate various similarity calculations and produce results for each of them - Cosine Similarity, Pearson Similarity and Spearman Similarity.
2. Our primary model will be based on Collaborative Filtering.
    - We aim to implement the memory-based collaborative filtering model using algorithms like kNN.
    - Here, we will model playlists as individual users and generate the playlist-song matrix to predict the song continuation.
    - As a next step in collaborative filtering, we will try to estimate the emotion associated with the playlist and extract it as a feature. (For this, the initial idea is that we would need a pre-trained model to estimate the keywords in our data to various emotions. This is a novel approach to collaborative filtering where we use additional features to establish similarity.)
    - The limitation of such an approach is what is called the "popularity bias", in that, the algorithm selects the small subset of popular items as the majority entries in the playlist.
3. Additional model we plan to use will be Collocated Artists- Greatest Hits method.
    - It lies on the assumption that different artists that are included in the same playlist have some measure of similarity between them. We plan to use this information to recommend tracks based on the frequency of collocation of the artists. This approach solves the popularity bias to an extent.

The primary dataset to be used is for this project is the Million Playlist Dataset (MPD) made available by the Spotify recsys challenge. The dataset contains 1,000,000 playlists created by users on the Spotify platform.

5. What existing resources can you use?

In terms of resources, apart from the challenge dataset we also intend to explore the use of other publicly available datasets. last.fm, AOTM, 8tracks are few of the potential candidates which we have identified. In addtion to the datasets, we also plan to make use of the opensource libraries for the algorithms and techniques which we are going to implement. A non exhaustive list of such libraries would be SciKit learn, NumPy, Pandas, Keras etc. We also expect to continue reviewing and referencing related previous work in literature and implementation.

6. How will you demonstrate the usefulness of your tool?

The novelty of this idea and its usefulness in various applications has already been established. For the evaluation purposes we intend to perform a train, test and validation split on our primary dataset and report the performance of our implementation using the following metrics:

- R-Precision
- NDCG
- Recommended song clicks.

The above mentioned metrics are same as the ones outlined by the original challenge and are also known standards for the evaluation of information retrieval/ recommendation systems.

7. References

- G. Bonnin and D. Jannach. A comparison of playlist generation strategies for music recommendation and a new baseline scheme. 2013.
- D. H. Park, H. K. Kim, I. Y. Choi, and J. K. Kim. A literature review and classification of recommender systems research. Expert Syst. Appl. , 39(11):1005910072, Sept. 2012.
- J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen. The adaptive web. chapter Collaborative Filtering Recommender Systems, pages 291324. Springer-Verlag, Berlin, Heidel- berg, 2007.
- G. Bonnin and D. Jannach. Automated generation of music playlists: Survey and experiments. ACM Comput. Surv., 47(2):26:126:35, Nov. 2014.
- M. Schedl, H. Zamani, C. Chen, Y. Deldjoo, and M. Elahi. Current challenges and visions in music recommender sys- tems research. CoRR , abs/1710.03208, 2017.
- https://recsys-challenge.spotify.com/