

Music Recommendation System

1stKanav Bhagat

DMG - CSE506

Indraprastha Institute of Information Technology

New Delhi, India

kanav16046@iiitd.ac.in

Abstract—This document intends to give the reader a detailed idea about the course project based on recommending music to the user using different data-mining and collaborative filtering techniques. The data set comprising around 10,000 entries are used along with different libraries like sci-kit learn, pandas and numpy. Total of three methods are used to recommend the music to the user i.e most popular songs, personalized approach using collaborative filtering and K-NN.

Index Terms—Data, Songs, recommendation, Co-occurrence matrix, User

I. INTRODUCTION

Streaming music online has become a trend these days and a huge number of people stream music online through applications like Spotify, Apple Music, Sawan etc. These music companies use some sort of music recommendation techniques to provide the best match for their listeners. These music recommenders filter the information and predict user's preferences. In this project, I used some of such techniques to make a music recommendation system for a specific user. The Million Songs dataset was used to build up the model. Here in total, there are two types of recommendation according to the choice of the user - *Global Top 50* and *Your Mix*. Global top 50 return the top 50 songs listen by users by eeping in the count of different users for each unique song as recommendation score. Whereas the Your Mix uses an item-item approach to filter the data by making a co-occurrence matrix of songs and counting the unique songs listened by the other entities who listen to the same song as the current user. The program creates and calculates the uniques entries in the co-occurrence matrix and gets give the score to each song lying in the co-occurrence matrix. The songs with the top 50 score are recommended to the user.

II. DATASET

A. The Million Song Dataset

The **Million Song Dataset** was used to explore and run models on. The Million Song Dataset is a freely-available collection of audio features and metadata for a million contemporary popular music tracks. Since the dataset was very large and took time to run queries on, a subset of it was used later which had a total of 10,000 samples. The dataset has data from the following sources:

- SecondHandSongs dataset - cover songs
- musiXmatch dataset - lyrics

- Last.fm dataset - song-level tags and similarity
- Taste Profile subset - user data
- thisismyjam-to-MSD mapping - more user data
- tagtraum genre annotations - genre labels
- Top MAGD dataset - more genre labels

The Dataset contains two file: *Triplet file and meta-data file*. The triplet_file contains [user_id, song_id, listen time]. The metadata_file contains [song_id, title, release_by, artist_name].

Both the files were loaded into the python script using pandas and later joined using *song_id*. The duplicates were removed from the data. After this, song and it's artist name was merged together in a single column and aggregated by the number of time the song is being listened by all the users. Duplicates from the users and songs columns were dropped.

III. MODELS PROPOSED

Three types of models were proposed -

- Naive Approach
- Content Based Approach
- Collaborative Based Approach

A. Naive Approach (Global Top 50)

To get the most popular songs, it counts the total number of times a particular song was listened by each user in the dataset and gives it a recommendation score. The most listened song will always top the list while the least listened song will be found at the bottom. It will return the most **50** popular song in the corpus No matter what the users' taste is, it will always recommend the most popular song. This approach is a naive approach and gives the same result for all the users. The **pros** of this approach is that it is easy to implement and can give a general music recommendation to the users. **cons** of this approach is that it is not personalized and will not account the songs usually listened by the user.

B. Content Based Approach (KNN- Recommendations)

In this type of method, this system is based on the item's description and the user's preference profile. The songs recommended are close to the most listened song of the user. Here choices of other users are not taken into account(even with the users with same interest). It makes recommendations by looking for music whose features are very similar to the tastes of the user. In K-NN, it makes a space of all the songs

according to the listen_count and artist_name of the song. It tries to find out the neighbourhood of each song. Once the feature space is created, we look for the most neighbouring songs. K-NN uses 6 nearest neighbours using cosine distance and top 5 songs are returned. Here songs from same artist and same name are commonly recommended.

C. Collaborative Based Approach (Your Mix)

In collaborative based approach, all the user collaborate to recommend the new music to a single user. It is also a personalized approach where users will be recommended songs. It is also a personalized approach where users will be recommended songs based on a number of time a user (any user other than current user), who has listened to that song, will also listen to another set of other songs. Following fig 1. tells a diagram which present the relation between item-item relationship.

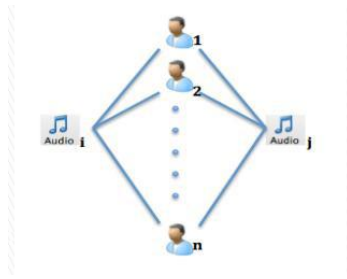


Fig. 1.

To further simplify this, based on what you like in the past, what another similar song that you will like based on what another similar user has liked. The key idea is songs that are often listened by the same user tend to be similar and are more likely to be listened together in future by some other user. In this approach, we generate a co-occurrence matrix-like given in the fig 2. After this, we get the items which are occurring more against the songs of the current user. The most occurring 10 songs are returned.

Product ID	1001	1002	1003	1004	1005
1001		1	3	1	1
1002	1		3		
1003	3	3		4	4
1004	1		4		2
1005	1		4	2	

Fig. 2.

IV. TERMINAL APPLICATION

A terminal application was made which lets the user to enter its user_id and can select any of the above recommendation system. The user can quit at any time. The terminal app gives the real world setup for the emulator to test the program. Following is a screenshot from the program:

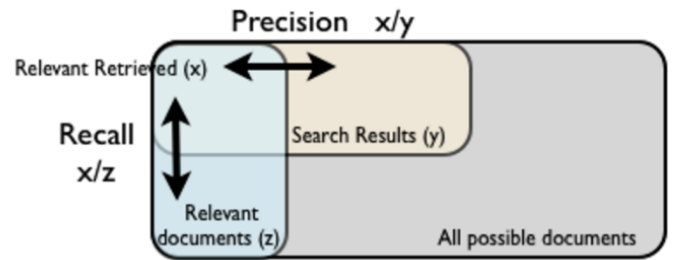


Fig. 3.

The user can press 1 to get the top 50 popular songs, press 2 to get the songs related to the collaborative approach and press 3 to get content based recommendation.

V. EVALUATION METRICS

We can compare all the above models using a precision-recall curve. Precision helps to get the top results that are relevant. If we could label the relevance and irrelevance of the songs recommended by the user, we could use the Precision-Recall curve and find the best model. According to my taste, the Collaborative approach works for the best since it recommended the songs I like more than any other approach. Since the relevance/irrelevance of a song is completely subjective thing, any model can be bet for a specific user. Fig 4. describes the Precision - Recall functioning.



Source: <http://aimotion.blogspot.com/2011/05/evaluating-recommender-systems.html>

Fig. 4.

The user can press 1 to get the top 50 popular songs, press 2 to get the songs related to the collaborative approach and press 3 to get content based recommendation.

- The pros of the popular approach is that it will always give a general approach and those songs which will have the most listen_count. This is a simple approach and has lowest time complexity. Still this approach doesn't give the personalized results.
- The content based K-NN recommendation gives personalized recommendations. The songs recommended are usually from the same artist. Also songs with similar titles also gets recommended more often.

- The collaborative approach also gives personalized results and takes account to the choices of other users also which makes it more reasonable and real - word.

REFERENCES

- [1] https://cse.iitk.ac.in/users/cs365/2014/_submissions/shefali/project/report.pdf
- [2] <http://millionsongdataset.com/>
- [3] <https://blogs.msdn.microsoft.com/carlnol/2012/06/23/co-occurrence-approach-to-an-item-based-recommender/>
- [4] <https://www.youtube.com/watch?v=18adykNGhHU>