

PANDORA'S JUKEBOX

Following are some of the features with their approaches that can be added to our Music Recommender System for better user experience:

1. Start the exploratory phase if any of the top genre arms suffers a lot due to negative rewards.

Approach:

- ❑ Since there can be an issue where most arms have similar values (maybe a reason - suddenly user faces some sort of mood change for long time and top arms suffer)
- ❑ We can keep minimum threshold for estimated value of top arm.
- ❑ If it falls below the threshold value we can start the exploratory phase again.

2. Since our system is like - once a user starts using it works absolutely great as expected but if he/she visits our system again it starts afresh again. (As arm-updates and user preferences are initialized again)

Approach:

- ❑ We just need to store the estimated arm values and round number for each user.
- ❑ Then load (the user's estimated arm values and round number) once the user signs in.

3. We don't have timestamp in our dataset. Else we could have not only shown user songs from a similar artists but also particular era for better experience. (Eg: Rock songs from 1990s to 2000s)

Approach:

- ❑ Users generally tend to listen songs from certain era/decade for particular time/session.
- ❑ Thus we can after getting top similar items from user_preference we can recommend only those which belong to the decade he/she has been listening in past few hours.
- ❑ Also few users tend to listen songs of particular era than the other. We can maintain a list/ priority based data structure for implementing it.

4. Just as song similarity we can add artist similarity which would be good for suggesting new artist in the system.

Approach:

- ❑ Artists can compared based on the songs they have composed/released. (eg: using cosine similarity)
- ❑ Thus if a new artist (which matches the user's preference) releases his song then it could be recommended to the user even though yet his estimated arm value is low.

5. With timestamp in dataset we can handle newly released songs more properly.

Approach:

- ❑ Many users would like to listen newly released songs more frequently than the older ones so we can have a feature which gives weights to songs based on their added time.
- ❑ Thus giving priority to these songs (which can be filtered on basis on his/her user_preference) while recommending them.