Project Report
Music Recommendation Chat Bot

Team 5

Subhodeep Maji	201301129
Gaurav Bansal	201301133
Maulik Bopalia	201301142
Sai Sruthi Talluri	201301143
NIrjhar Bala	201301193

5th December, 2016

Table of Contacts

Introduction

The music recommendation chatbot is an multifunctional assistant that helps you in finding music that matches your taste. Many a times we feel like listening to songs that are similar to our favorites but do not know how to find them. Here is how, we can find them by asking the bot to suggest songs based on our liking of song, artist, genre, and even mood. The chatbot works on the principles of Machine Learning and Natural Language Processing where the prior helps it find the music with similarity and later helps in understanding, processing and responding to the requests of the user. We use the principles of Artificial Intelligence for the proper and complete implementation of the chatbot.

Project Overview

Why only a chatbot for this purpose?

We have seen that the user for the chatbot does not know the song he wants to listen or sometimes even the genre of songs he wants to listen according to his mood. The best way a computer scientist can help is via a chatbot. Clearly in this scenario as the user in not sure what he wants to listen to , his queries are essentially fuzzy and need further clarification and cross questioning. We know that the queries are short and if user is asked further he may be able to give more features that will make the recommendation easy. Also constant feedback and updating of the user's likes, dislikes and interests are captured and asked for which can best be taken by the chatbot. The chatbot takes care of these by saving the preferences in the database of its songs along with these additional features. The chatbot apart from helping user also serves the user with his direct queries making quick and easy look up for the user for a particular song.

Audience

The targeted audience for the chatbot are any individual with the liking of music and would like to know more songs that are similar to his likes. A user can also by a simple one who just wants to listen to music according to his mood. Who does not want to listen to music when we are happy, sad, angry(well that means in any mood)? The chatbot also helps in just finding the songs that are in a particular genre or by a artist . It also recommends songs newly trending or any new releases which may help a user to build a taste. This chatbot is basically for anyone who wants to listen to music but isn't sure what exactly to listen to.

Proposed Solution

The solution for the problem has been divided into several components and we have tried to solve using the some concepts of Machine Learning(ML) and Natural Language Processing(NLP). The part of connecting and designing of the features is done using the principles of Artificial Intelligence(AI). The AI helps us solve the problem in separate components and use ML and NLP in solving them.

Segments of the problem

Initial Setup

The user when logs into the chatbot or makes a profile, we provide an option for him/ her to import his playlist from the system for easy and better recommendations. The chatbot then stores the information like maximum playcounts, ratings, artists, genres into its database. This helps in initial profiling of the user for the bot to start its task easily.

Initial Recommendations

To start with the system the user is given suggestions in the form of direct buttons(4) of the songs that have maximum play count from the user's playlist(iTunes) and of the most popular of all time to know the user's likes. Later the song suggestion on the start are changed according to the feedback received. The user if does not select the option of import playlist the new and popular are suggested to the user and the system tries learning from only from feedback.

Mood Based

The database of the chatbot will contain the songs from various genres. These genres can be categorized to suit various moods of the users.But since all songs of a particular genre may not correspond to a mood we shall take user feedback to get confirmation of the songs that match mood for recommendations. Overall the mood attribute is loosely couples with the genre with global information, and particular user's mood suitability will be stored locally

Artist Based

This one of the most famous and important recommendations. We shall keep track of the user's favourite artists may be from the knowledge of initial playlist and later by the feedback, and recommend him/her about any all new releases of that artist. Further mood and other attribute wise similar songs can be suggested to the user having different artists. Then this can be used for recommending new artists to the user.

Genre Based

Again this is very similar to that of artist based but we have a broader domain to classify hence further questions are to be asked to the user for proper recommendations. These include the asking the user to specify the years or time period in that genre. Any specific interests of artist in that particular genre etc. This will help in proper profiling and also in the continuous feedback needed to help the chatbot perform better.

Categories like New Releases / Popular / Top rated / Random

If the user wants he can restrict his music to new releases since many users like listening to the latest music. Also popular music will be music that is rising in ratings or is being most frequently queried. Top rated will be like a classic list where only the top rated music based on the weighted ratings giving the user an option to choose from the best of the best. Also to be able to introduce the user to new music which the user may not know he likes thus expanding our knowledge of the user.

Playlist Recommendations

Instead of recommending songs the chatbot will sometimes generate playlists. This can be achieved by observing which songs are often listened together by clustering the songs based on the listeners. Thus songs listened to by similar users can be obtained. We shall use this data to generate playlists which will further help is in knowing the user.

Methods used to solve the segments

Machine learning

Machine learning is used for all the basic recommendation of songs to the user. The bot learns the type of songs a particular user likes based on many factors , computes similarity and recommends song similar to them. The learning and constant improvement of the songs is a part of machine learning.

Recommendation system

The algorithm is as follows.
Every user has his own list of songs he has already listened to(history). Initially when a user joins, he is fed with recommendations from these already listened to ones or most popular songs and new releases from which the user chooses some songs.
Note that the user can also directly search for his songs. Now these songs form the history base of the user. The songs in which the user rated it as good or commented have higher weightage than any other songs. The songs in which user has rated bad or disliked have lower weightage than any other songs. Now the chatbot on the first tries the solve fuzzy query input given by the user and tries to arrive at a pretty accurate query in the form of a vector, the vector of the same type as a songs attribute vector and then :

1) The vector is then compared to the high weighted songs from the user and the first ‘k’ having the highest similarity (cosine) are suggested.
2) The above ‘k’ songs also have their own identifying vector. Songs universally, similar to the ones suggested are also recommended. (this makes sure the user is suggested with songs he does not commonly listen to.)

How are similar songs identified?

This is a part of the same item to item collaborative filtering that is used above. The above type of recommendation gives a high quality of recommendation. But Higher quality comes at a cost of higher computation time. To avoid this time to cause delay in the users chat box recommendation, the similarity is calculated in an offline process.
It can be described by the following algorithm.

```
For each song present in the (universal) list of songs i1
        For each customer c who listened to song i.
                For every other song i2 heard by the user
                        Compute similarity between i1 and i2
```

This similarity will be computed similar to cosine similarity. No wonder this will be a really computationally expensive process but all of this is done offline. This secondly also prevents the situations of cold start in the traditional collaborative filtering. This way the chatbot will be able to provide good recommendations by simply making a vector matrix (hence formed) lookup, saving a lot of time.

Weighted ratings

Weighted ratings is an idea to normalise the ratings given by people who have tendencies to rate in a different range. For example a good critique may always rate in the range of 4-8 out of 10 while a kid may always rate in the range of 8-10. So this makes the weightage of the user who rates in the greater range more, but this should not happen.
To tackle this problem we used this following solution.
We used square root of 10 as the mean index for rating system.
Every user’s rating is normalised by the formula of Normalised rating =root of 10*user’s rating/user’s average rating. And this normalised rating is used for all calculations instead of true rating.

Import playlist

To avoid the shortcomings and disadvantages of a Cold start we are going to use an option to Import the User’s any already saved playlist. What happens in a cold starts is that there is no data related to the user is available. Hence the recommendation system is at its weakest and the only way the users preferences can be collecting is a long process or randomly suggesting popular songs / recent ones. This process takes time to converge and also reduces the quality of recommendation. Adding the import playlist feature allows the chatbot to identify the user's special taste in music before hand, so as to use this information in giving out better recommendations.

Natural Language Processing

The chatbot for its proper functions requires a good amount of NLP to understand the user’s queries and to also to converse to it properly. The user may get annoyed if the chatbot keeps repeating the same phrase “Sorry I don’t understand what you said.” The NLP consists of Natural Language Understanding, part of speech tagging, Question Answering, topic segmentation, etc. We mainly require the natural language understanding that is understand exactly the type of query of the user. The direct query or indirect. For the direct query extraction of the words, and using tagging is enough. For queries of indirect type initially a extraction is needed and later the chatbot has to converse using a proper sentence that is pre programmed for certain tags.

Adding a Markovian Hint

It is obvious that the users taste develops gradually. In this case suggesting the user new songs based on the whole average of his previously heard and rated songs will be wrong. To solve this we include something like a discount factor, similar to the one that is used in solving MDP. The discount factor gives more weightage to the songs that have been recently heard by the user and not to the ones that were heard a long time ago. This slightly does improve the quality of recommendation.

Example Questions and Responses

Do you have an iTunes or Google Play Music rated playlist that you would like to import? This will help me give better suggestion.

Could you tell us some of your favorite artists?

X by Y is trending right now. Would you like to listen to it?

How much would you like rate the song you just were recommended?

How would you describe the mood of the song you were recommended now?

If you are busy and only want to listen to music in the background you can request a playlist.

What is your mood right now?

Would you like to listen to more music similar to this song?

You can choose a category of music that you want to listen to?

Is there a particular artist, genre of music you want to listen to?

Conclusions

The chatbot tries to recommend music in all the directions usually a user would like to listen to. To avoid initial delay in proper recommendation the playlist is used. The constant feedback helps in gaining of the knowledge of the user. The mood based and other queries are solved by asking the user all further questions to solve his doubts about how to search songs. The chatbot thus helps in times of fuzzy queries, immediate search, finding similar and new songs, etc. The chatbot can further be improved in recommendation by constant use and also when many users provide with proper feedbacks.