

MINI PROJECT (2020-21)

ON

MUSIC RECOMMENDATION SYSTEM

**AUTHOR:** SHREYA GARG

**SECTION:** E

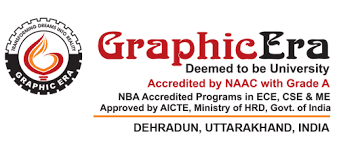
**ROLL NO:** 2014866(51)

**SEMESTER:** 5th

**BATCH:** 2019 - 23

INTRODUCTION

OBJECTIVE



MINI PROJECT (2020-21)

ON

MUSIC RECOMMENDATION SYSTEM

**AUTHOR:** SHREYA GARG

**SECTION:** E

**ROLL NO:** 2014866(51)

**SEMESTER:** 5th

**BATCH:** 2019 - 23

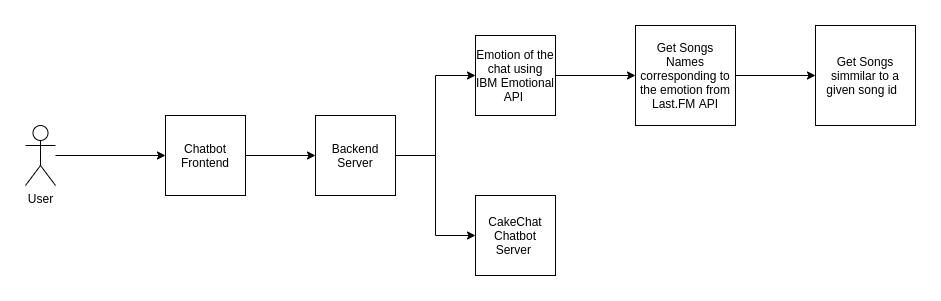
ABSTRACT

Currently, digital online music increase significantly, both in terms of content and users. Increasing the number of digital music content every month conduce a lot of song catalog data and becoming unstructured and making it difficult for users to choose the songs they want to listen to. To make it easier for users to optimize a large number of subscribed music catalogs, a user-centric music recommendation system is needed that allows users to be able to manage catalogs of digital music content according to their needs. This study examines how to implement song recommendation system using collaborative filtering method in digital online music. This study uses 20,000 users, 6,000 songs and 470,000 transactions rating. Through those research, it is discovered that user-based collaborative technique that could make one system for clients will gather those playlist they really want to hear.

INTRODUCTION

In this project, we would be building an extensive Chatbot service, to which you can talk to. And talking to a chatbot wouldn't be business-driven. It would just be casual conversations. Further, on top of it, the chatbot would also be recommending songs to the user based on the tone of the user. This song recommendation feature employs the use of MILLIONSONGDATASET. Also for tone/emotion analysis of the conversation we will be using the [IBM Tone Analyzer API](https://www.youtube.com/watch?v=wUb--6FPBik).

This is a collaborative filtering model where algorithm is to learn a function that can predict if a user will benefit from an item — meaning the user will likely listen to a song. In order to do so, this type of filtering only relies on the historical preference of users on a set of songs. It is based on rating (listen\_count: rating is based on how many times user listened the song).



**REQUIREMENTS:**

**LANGUAGE USED:** Python

**SOFTWARE REQUIREMENTS:**

* Python IDE
* Libraries: sckit-learn, tensorflow
* IBM Tone Analyser

**HARDWARE REQUIREMENTS:** Windows Desktop

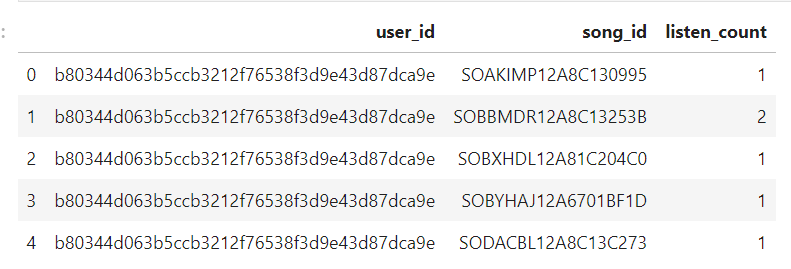
METHODOLOGY

### STEP 1: Read dataset that shows metadata of each song into Pandas data frame

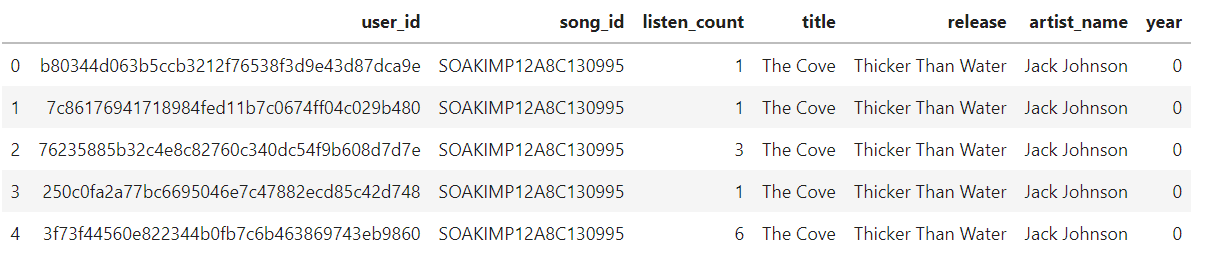




### STEP 2: Read dataset that shows how many times a user plays each song into pandas data frame



## STEP 3: Merge songs and songs to user dataset



### STEP 4: Get most listened songs

* First, we’ll combine song id and listen count (no of times user listened the song), then combine song title and listen count.
* Atlast, filter all by combining artist name and listen count.

### STEP 5: Create Pivot Table of User Vs Songs



STEP 6: Compress dataset by applying SVD

STEP 7: Create coorelation matrix

STEP 8: Print specified song list

Using the list and matrix we’ll recommend songs.

**REFERENCES:**

* <https://www.crio.do/projects/python-chatbot-api/> --- for understanding chatbot
* <https://towardsdatascience.com/the-keys-building-collaborative>
* <https://www.ijert.org/music-recommendation-system-using-content-and-collaborative-filtering-methods#:~:text=Collaborative%20filtering%20is%20based%20on,for%20different%20users%20or%20items>.
* Dataset: <http://millionsongdataset.com/>