Spotify Music Recommendation System

Saikumar Reddy Sandannagari
Computer Engineering Department, San Jose State University, CA

Abstract:

In present scenario, Music has become an important media for every individual. With the wide availability of internet, Music applications are developing in a wide range. Applications like Spotify, Sound-cloud, Amazon-Music, Apple-Music made music availability easy. People are using these at a great deal. In every music application, recommendations of songs related to users interests plays a vital role in users selection of listening to songs. As a part of my project, I built a recommendation system for music using spotify data based on user searches which will help users to pick songs of his interest easily.

Introduction:

A recommendation system is used to predict the users interest on items based on his previous searches or interests. Recommendation systems have become an integral part in many ecommerce websites, music applications, Social media Platforms etc. There are mainly two types of recommendation system based on the approach. They are Collaborative filtering and Content based Recommendation system. I have chosen to build Content-based recommendation system to predict more accurate results by text based recommendation. Content-based recommendation mainly depends on users profile and item features for recommendations. This report will take you through all the steps i have followed to build this model and results.

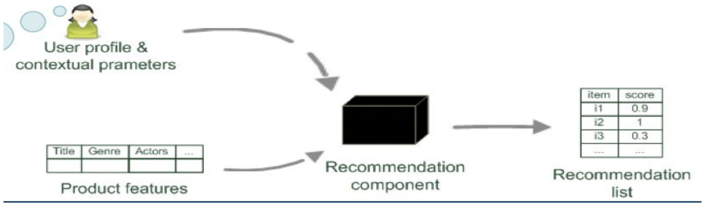
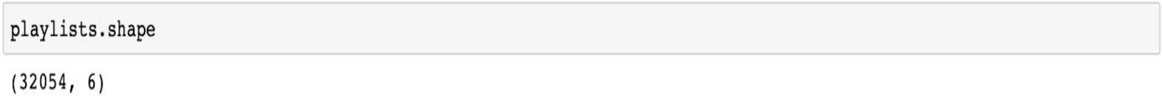


Fig. Content-based recommendation system

Model workflow:

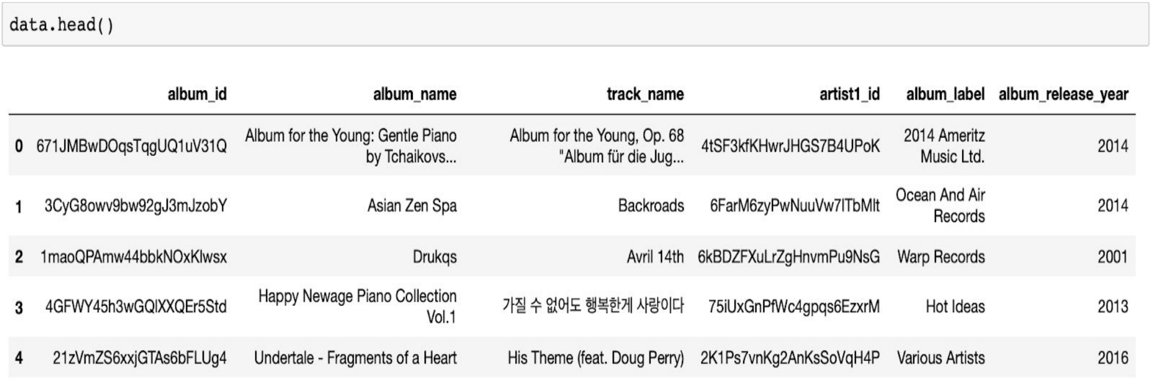
1. Data Acquisition:

Data Acquisition plays a vital role in Data Science Project. The efficiency for the project output depends on volume,Variety and Velocity of Data Collected. Spotify released Million Playlist Dataset(MPD) which consists of almost 1 million playlists. As this is a huge dataset, I considered around 30000 tracks with 6 features namely album_id, album_name, track_name, album_release_date, album_label, artist_id. This dataset contains release_date in MM/DD/YYYY format. I converted this to YYYY format and renamed it as release_year.



```
playlists.shape
```

```
(32054, 6)
```

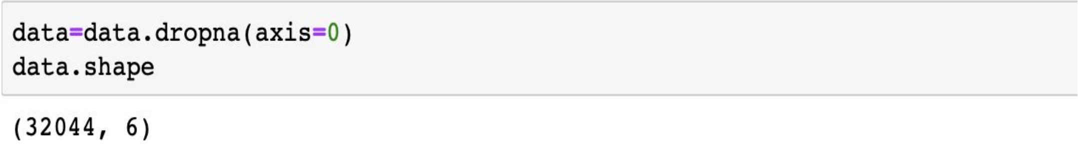


```
data.head()
```

	album_id	album_name	track_name	artist1_id	album_label	album_release_year
0	671JMBwDOqsTqgUQ1uV31Q	Album for the Young: Gentle Piano by Tchaikovs...	Album for the Young, Op. 68 "Album für die Jug...	4tSF3kfKHwrJHGS7B4UPoK	2014 Ameritz Music Ltd.	2014
1	3CyG8owv9bw92gJ3mJzobY	Asian Zen Spa	Backroads	6FarM6zyPwNuuVw7lTbMlt	Ocean And Air Records	2014
2	1maoQPAmw44bbkNOxKlwsx	Drukqs	Avril 14th	6kBDZFXuLrZgHnvmPu9NsG	Warp Records	2001
3	4GFWY45h3wGQlXXQEr5Std	Happy Newage Piano Collection Vol.1	가질 수 없어도 행복한게 사랑이다	75iUxGnPfWc4gpqs6EzxrM	Hot Ideas	2013
4	21zVmZS6xxjGTAs6bFLUg4	Undertale - Fragments of a Heart	His Theme (feat. Doug Perry)	2K1Ps7vnKg2AnKsSoVqH4P	Various Artists	2016

2. **<u>Data Cleaning:</u>**
   Data cleaning is the process of removing noisy records from collected dataset. As I collected raw data, There are some missing records and duplicates in the dataset. I followed following techniques to clean the data

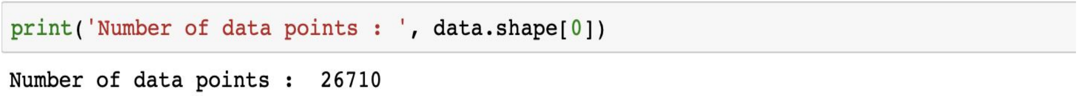
   I. <u>Removing Null values:</u> As i collected raw data, there may be some missing columns, These missing values cannot be predicted randomly as every column has its own importance, so I tried to remove those columns with Null values. After removing those data set has around 32044 data points.



```
data=data.dropna(axis=0)
data.shape
```

```
(32044, 6)
```

   II. <u>Removing Duplicates with same track_name:</u> After removing the null values, another important step in data cleaning is removing duplicates i.e data points with same track_name which will reduce the accuracy of model to predict the recommendations. After removing such duplicates, dataset has left with around 26000 data points.



```
print('Number of data points : ', data.shape[0])
```

```
Number of data points :  26710
```

3. Text Based Recommendation Systems:

For text based recommendation, I considered the feature “Album_name” of each track to recommend tracks. We consider different tracks t1, t2, t3 and their corresponding album_names a1, a2, a3 to find the similarity point using euclidean distance between them. The main concept of Text based recommendation is to convert album_name into vector of words which will make the similarity measure easy to calculate.

For this recommendation_System, I am using bag of words model. We will discuss this model in depth in the next section.

4. Bag of Words Model:

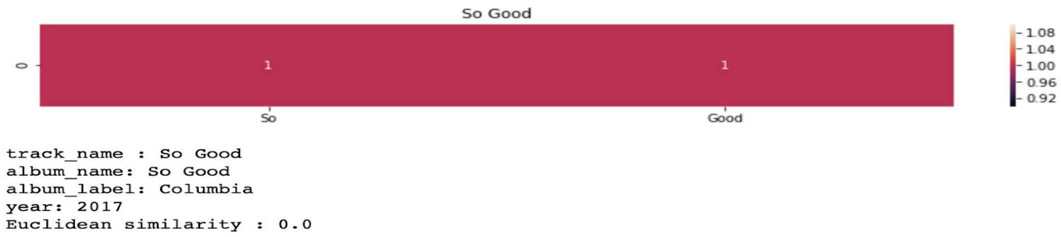
Bag of words(BOW) is a technique that describes the occurrence of a word within a document. Initially I create a corpus of document which is a collection of all the unique words from all the album_names. Then, I arrange each document/album_name as a vector of corpus of document that represents the value of occurrence of each word in the album_name. I am using feature extractor from Scikit Learn package to convert each document into a vector.

The vector that I get from this model is sparse and has integers which describes the occurrence of word in the particular album_name. Once I get vectors for all the album_names, I provide input to that model for which I would like to find recommendations and I used euclidean distance between the required album_name and all other album_names. I then print the track_name, album_name, artist_id and Euclidean similarity for top 5 recommendations for the given album_name..

I also print the heat map for all recommendations which shows the basic difference in words between the album_names. In this way, I used bag of words model to predict the top n recommendations for given track based on its similarity in album_name with other tracks. This recommendation system mainly recommends others tracks from similar album_name for given track_name

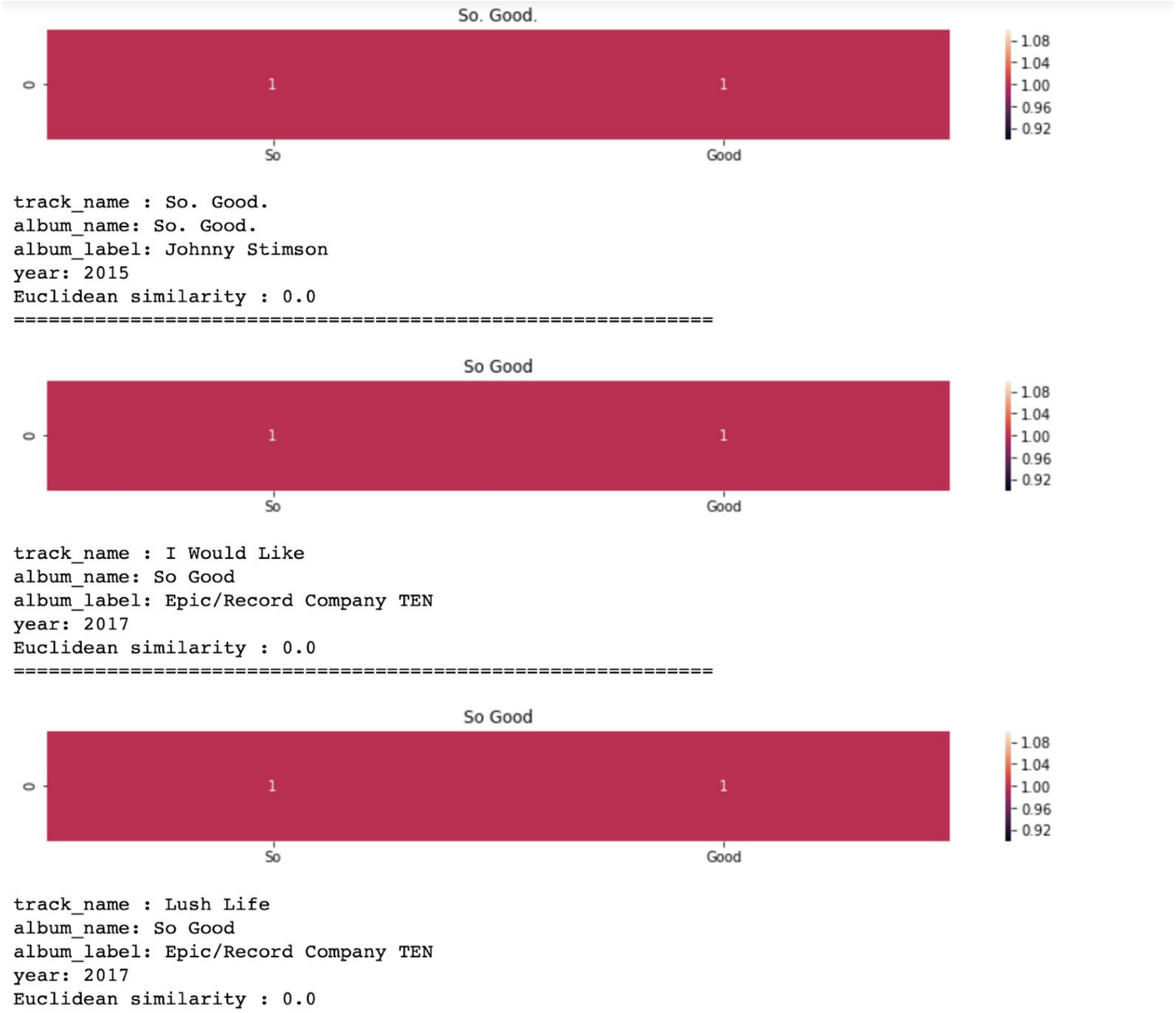
Results:

Input:



```
track_name : So Good
album_name: So Good
album_label: Columbia
year: 2017
Euclidean similarity : 0.0
```

Output



```
track_name : So. Good.
album_name: So. Good.
album_label: Johnny Stimson
year: 2015
Euclidean similarity : 0.0
==============================================================

track_name : I Would Like
album_name: So Good
album_label: Epic/Record Company TEN
year: 2017
Euclidean similarity : 0.0
==============================================================

track_name : Lush Life
album_name: So Good
album_label: Epic/Record Company TEN
year: 2017
Euclidean similarity : 0.0
```

Conclusion:

I have collected data from spotify Million Playlist Dataset and then followed data cleaning techniques to remove null values and duplicates from dataset. That dataset is used to run the bag of words model to find the recommendations for given track_name.

In this model Euclidean distance is calculated between the vectors of each track.The tracks having less euclidean distance are considered to be most similar.

References:

https://recsys-challenge.spotify.com/details
https://developer.spotify.com/documentation/web-api/