

**Faculty of Engineering & Information Technology**

**42913 - Social and Information Network Analysis**

**HYBRID MUSIC RECOMMENDER SYSTEM**

**Project Report**

**Subject Coordinator: A/Prof. Ying Zhang**

Submitted By

|  |  |  |
| --- | --- | --- |
| Group 30 | Vishnu Mohan Edala Dhanraj | 13069909 |

Table of Contents

[EXECUTIVE SUMMARY 4](#_Toc11016978)

[INTRODUCTION 4](#_Toc11016979)

[BACKGROUND 4](#_Toc11016980)

[Recommender Systems Function 4](#_Toc11016981)

[Prominent Recommendation System Techniques: 5](#_Toc11016982)

[CONTENT-BASED FILTERING: 5](#_Toc11016983)

[COLLOBORATIVE FILTERING: 6](#_Toc11016984)

[DEMOGRAPHIC: 7](#_Toc11016985)

[KNOWLEDGE-BASED: 7](#_Toc11016986)

[COMMUNITY BASED: 7](#_Toc11016987)

[INTERACTIVE RECOMMENDER SYSTEMS: 7](#_Toc11016988)

[HYBRID RECOMMENDER SYSTEMS 8](#_Toc11016989)

[BRIEF COMPARISON OF RECOMMENDATION TECHNIQUES 8](#_Toc11016990)

[Collaborative Technique: 8](#_Toc11016991)

[Content-Based: 8](#_Toc11016992)

[Demographic 8](#_Toc11016993)

[Utility- Based: 8](#_Toc11016994)

[Knowledge Based: 9](#_Toc11016995)

[RESEARCH METHODS 9](#_Toc11016996)

[EXISTING RECOMMENDER SYSTEMS 9](#_Toc11016997)

[SPOTIFY: 9](#_Toc11016998)

[APPLE MUSIC: 9](#_Toc11016999)

[DEEZER: 10](#_Toc11017000)

[DATASET 10](#_Toc11017001)

[DATA VISUALISATION 11](#_Toc11017002)

[LOAD TRACKS.CSV DATA 13](#_Toc11017003)

[LOAD ARTISTS DATA 14](#_Toc11017004)

[LOAD ARTISTS DATA and MERGE IT WITH TRACKS DATA 15](#_Toc11017005)

[Relationship Between Nods Artists and Tracks 15](#_Toc11017006)

[TOP Tracks listened on system by artist: AWOL 16](#_Toc11017007)

[To fetch all songs of an Album and its details 17](#_Toc11017008)

[FUTURISTIC APPROACH 20](#_Toc11017009)

[CONCLUSION 20](#_Toc11017010)

[REFERENCES 21](#_Toc11017011)

**Table of Figures**

[Figure 1 Architecture of Content Based Recommender System 5](#_Toc11011792)

[Figure 2 Content Based and Collaborative based Filtering 6](#_Toc11011793)

[Figure 3 Workflow of Interactive Recommender Systems 7](#_Toc11011794)

[Figure 4 Spotify's Market Growth 10](#_Toc11011795)

[Figure 5 Genre and its Count 12](#_Toc11011796)

[Figure 6 Load Tracks.csv into Database 14](#_Toc11011797)

[Figure 7 Node Relationships 15](#_Toc11011798)

[Figure 8 Nodes Relationship Graph 16](#_Toc11011799)

[Figure 9 Cipher Query of Artist "AWOL" 17](#_Toc11011800)

[Figure 10 Cipher Query to Compute Hip pop tracks of high listens. 18](#_Toc11011801)

[Figure 11 Query to Compute Electronic Genre Songs with Highest Listens. 19](#_Toc11011802)

[Figure 12 Songs with Most listen Count 20](#_Toc11011803)

# EXECUTIVE SUMMARY

Recommender systems has been in the lime light because of its application utility in e-commerce and business industries. The most type uses collaborative, content based on preference-based recommendation systems. With additional information of every item such as item review, user preference, interaction of user with the item, contextual information and its content features the user has all the feasibility to access the item and rate it. Thus, by providing a hybrid recommendation of providing both contents based as well collaborative based preferences based on the content features will increase the efficiency of the recommender system. This model proposed help in overcoming the cold-start problem that the other existing algorithms. Thus, the required experiments have been made on the dataset and suggestions has been provided based on the item features such as listen count, genre, track genre. This will provide an increased approach and feasibility to user in recommending lists.

# INTRODUCTION

Recommender systems are tools and techniques that provides suggestions for items to be of use to a user. The term “Items” referred here can be of anything like Music, Movie, events, things to buy, search queries, gadgets etc... With the growth and evolution of these Recommender systems, they have been termed as the most powerful and popular tools in the world of e-commerce.

Development of these recommender systems involves a lot of multi-disciplinary effect where experts from different fields like Artificial intelligence, Data Mining, statistics and Decision support systems would contribute their insights and suggestions that finally evolves as powerful tool.

In this report we have suggested a hybrid recommender system which works on both content based as well as collaborative Filtering methods.

BACKGROUND

## Recommender Systems Function

The Main functions of a recommender system are listed as follows,

1. Increase the conversion Rate: The primary purpose of creating a recommender system is to increase the number of users that take up the recommended items and consume them compared with the normal visitors looking for information. By providing them the required information needed they rate of using that system doubles.
2. Sell More Diverse items: The major role of any recommender system to give them the right suggestion. For e.g. In Netflix, the service provider should list movies and provide suggestions based on their existing history and likelihood. If they movies listed out are not likely in user’s taste, that would gradually reduce the rate of using that service provider.
3. Increase the User Satisfaction: A well designed recommender system will eventually end up in providing satisfaction to the customer. If the System keeps on suggesting items based on their taste and need, then the chances of accepting and viewing those recommendations are always high.
4. Elevate User Fidelity: Recommendation systems usually recognizes the old customer by the information and activities of the previous sessions and treat them with its existing refined models. Therefore, the longer the user visits the systems, the more refined its user model will be.
5. Better Understanding of user Likelihood: The System either explicitly get what the user wants or predicts the model based on its previous activities. For e.g. the system should screen itself about the content as to finalize whether to suggest it or not.

## Prominent Recommendation System Techniques:

Some of the prominent Recommendation System Techniques are listed as follows,

* Content- based
* Collaborative Filtering
* Demographic
* Knowledge-based
* Community-based
* Hybrid recommender Systems

The brief Study about each of the above prominent techniques, their advantages and disadvantages has been explained in the forthcoming segments.

# CONTENT-BASED FILTERING:

The System here recommends items, based on the similarity that the user liked in the past. The similarity of these items is calculated based on number of factors as wells features. For e.g. when the user listened to a Hard Rock song in the past, the system learns it from the history and suggests another of the same genre.

The high Level architecture of a content based recommender is shown below,

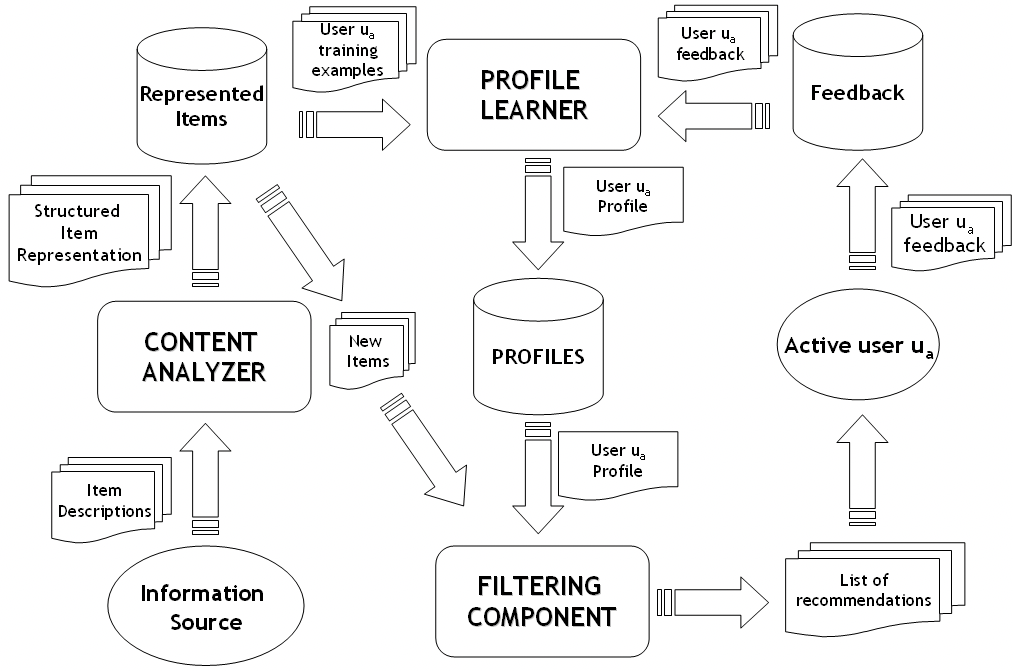


Figure 1 Architecture of Content Based Recommender System

The content-based filtering has several advantages than the collaborative one. Some of them are User Independence, Transparency and New Item suggestions.

They have some disadvantages too, as the items are to be categorized in a much more specified way and moreover it has a limit to analyze content.

# COLLOBORATIVE FILTERING:

The system here provides suggestions or recommendations to one user taking insights from other users who has similar tastes liked in the past. This similarity is being calculated by learning about the history of those users and their items respectively. Since the system generally builds on the model based on correlations between People-to-People, its being termed as Collaborative Technique.

There are two Approaches when it comes to CF, user based and Item based filtering.

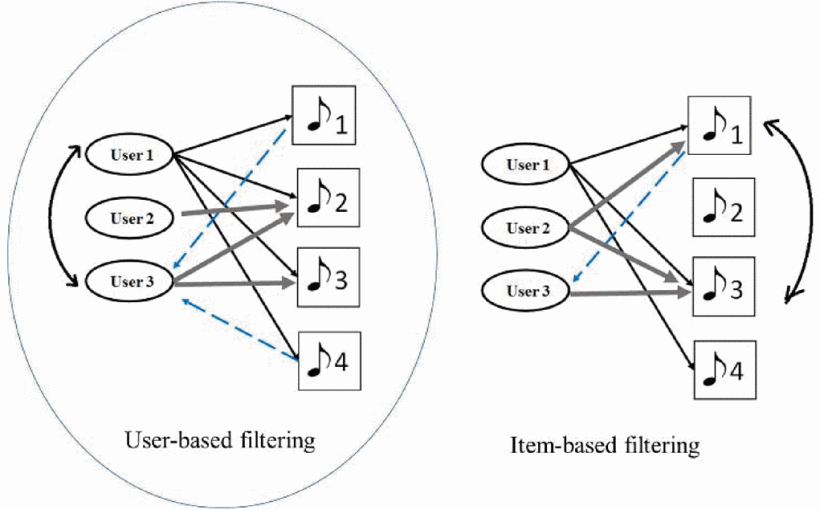


Figure 2 User Based and Item Based Filtering.

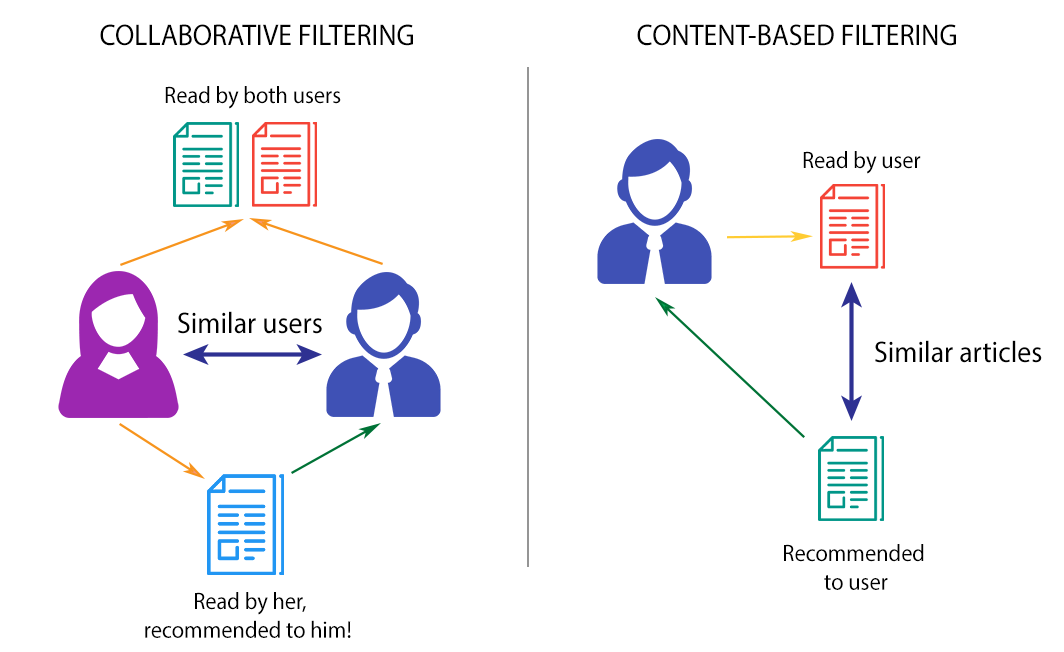


Figure 3 Content Based and Collaborative based Filtering

# DEMOGRAPHIC:

This kind of system provides suggestions based on the demographic profile of a user. The factors to be considered provides effective personalized solutions based on demographics. For e.g. Users are prone to information based on the age, Location or Language and the content provided on the sites may vary based on the demographic niches mentioned.

# KNOWLEDGE-BASED:

This Technique mainly helps in recommending how certain features of an item meet user needs and preferences and ultimately how the item will be useful to the user. There is a special module here as it estimates the proportionality between how much the user needs and how much the recommendations satisfy them. So, the Similarity here can be computed directly from the satisfaction of the user.

# COMMUNITY BASED:

This Type provides recommendations based on the preferences of the user’s Friends. These are also referred as social recommender systems. The recommendations are mainly provided by user friends and their likelihood. These techniques are mainly functional over the social networks and provides a simple comprehensive data about the social relations of users.

# INTERACTIVE RECOMMENDER SYSTEMS:

In this Type of recommender systems, the user was made through a set of preferences where its reads about the user preferences and need. Based on his data or interaction, it chooses the best of options or k nearby items and suggest as “items for you “to the user. The user finally chooses from the list, which is almost categorized on his requirements.

The Flow of interactive recommender systems can be shown by the following figure.



Figure 4 Workflow of Interactive Recommender Systems

# HYBRID RECOMMENDER SYSTEMS

This Type of recommender systems mainly work on the combination of the above-mentioned techniques. For e.g. when a hybrid system is formed from two techniques considered A and B, the main functionality here is to use the advantages of A technique to rule over the disadvantage that technique B possess. Thus, by the combination of providing two different techniques, the efficiency of the recommender systems output is considerably high, which enhances the user’s experience on using that system.

# BRIEF COMPARISON OF RECOMMENDATION TECHNIQUES

A brief Comparison of Recommendation techniques with respect to its background, Input , And its process are given below.

## Collaborative Technique:

**Back Ground:** It takes the Ratings from a group of users for a bunch of items

**Input:** Ratings or preference factor of Users of all the items in the database.

**Process:** Identify the users who has the same similarities of the chosen user and provide him the suggestions by the ratings and preference factors.

## Content-Based:

**Back Ground:** Selects the features of Items from the Database.

**Input:** User preference of items from the Database.

**Process:** Create a classifier model that fits the users rating behavior and develop predictions from the large database and supply them.

## Demographic

**Back Ground:** Demographic information about the group of users and their preference of items in the database

**Input:** Demographic information of factors to pull out the similarity quotient.

**Process:** Identify the users that share the same demographic features and extrapolate the information and their ratings of items.

## Utility- Based:

**Back Ground:** characteristic features of Items from the database.

**Input:** A utility function that describes the user’s preference from the database.

**Process:** Apply the utility function and determine the item’s priority from the database.

## Knowledge Based:

**Back Ground:** From categorizing the features of the items in the database, Finding the knowledge of how these items meet the user’s needs.

**Input:** Description of user’s priority or interests.

**Process:** Run a match between the list of items and user’s need and provide them the suggestions.

# RESEARCH METHODS

## EXISTING RECOMMENDER SYSTEMS

A brief Research on existing recommender systems are considered to check the exact functionality of how a recommender system fetches data directly from the database and provides suggestions for a new user and recommendations for existing users by their activity and track record.

Some of the recommender systems examined are

* SPOTIFY
* APPLE MUSIC
* DEEZER

## SPOTIFY:

Spotify is the leading audio streaming platform that facilitates DRM- protected music from renowned medial labels. Spotify generally works on three prominent techniques,

* **Collaborative Filtering:** Compares individual users taste with other similar users taste and provides suggestions.
* **Natural Language Processing:** It analyses the text or the language in each of its song and provide based on its language or core.
* Audio Modeling: Uses a raw Song or its texture, takes it, analyze it and compares it to other song textures.

One interesting note found during the research is, Spotify lacks the lyrics update during a song. This had some feedbacks on twitter and social networks and a study said that people switched over to apple music because of this.

## APPLE MUSIC:

Apple Music Utilizes similar kind of algorithmic techniques and methods. But there is this Connect features that sets apple apart from existing techniques. The connect Features helps in providing a platform with the Artists or Music labels and keep on sending updates of their new songs whenever it out.

## DEEZER:

Deezer has similar functionality and methods but has a distinct karaoke style lyric features which helps in providing a cool interface, which attracts users.

One more factor to consider among these are its UI/UX features which attracts users and keep them stay on. With that perspective Spotify stays on first with apple music and Deezer behind.

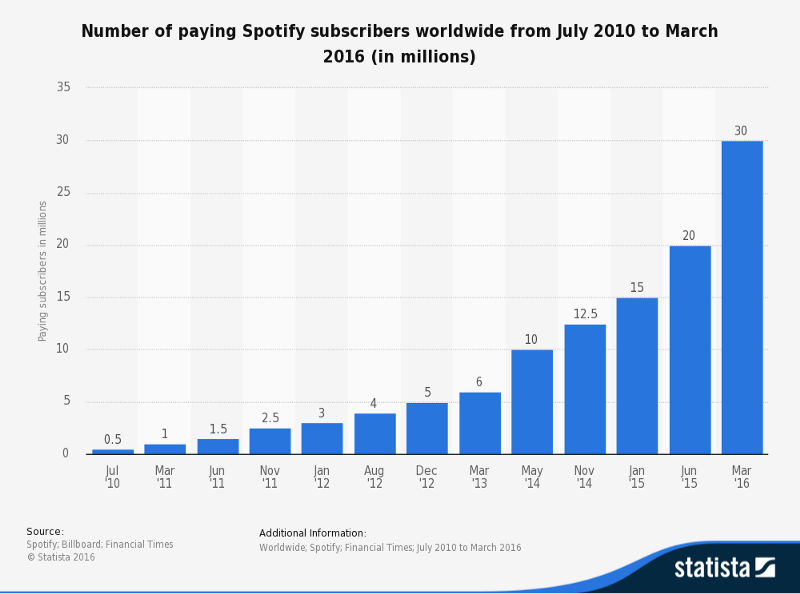


Figure 5 Spotify's Market Growth

# DATASET

Free Music Archive (FMA), a dataset that is suitable in providing, evaluating information about searching, organizing and accessing large music collections. The entire dataset is found to be 917 GB and we have extracted a subset out of it and finalized with a dataset of Tracks and its related information.

**Tracks.CSV** - This dataset provides information of 106574 tracks and its respective features such as ID, artist ID, Album name, Artist Name, Album Type, Genre, Number of listens - count.

**Artists.CSV**- this dataset has information about Artist name, Artist Location, Album name, Artist Handle Name, tags, Location.

These two nodes are Considered in building the recommendation systems. Relationships are made in such a way that; tracks could have been produced by number of artists and an artist could have produced a number of tracks.

# DATA VISUALISATION

Insights has been drawn from the Data set Tracks, in which certain correlations between columns or attributes designed.

Some of the major correlations which actually inferred from the dataset of the fma\_database are as follows,

**Album Tracks Vs Number of Tracks**



Figure 6Album Tracks Vs Track Genre

From the above graph, it can be easily inferred that the Experimental genre has the highest number of tracks among the others. Moreover, there are some songs which aren’t under any category.

Therefore, these tracks are not under any of those categories and are seem to Karaoke tracks under different times.

The different type of genre which the tracks are categorized in can be shown as,

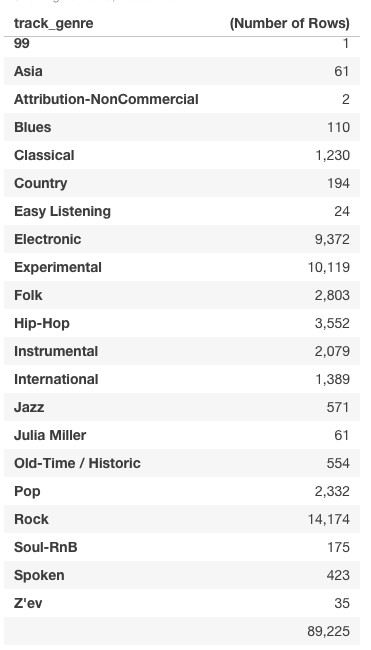


Figure 7 Genre and its Count

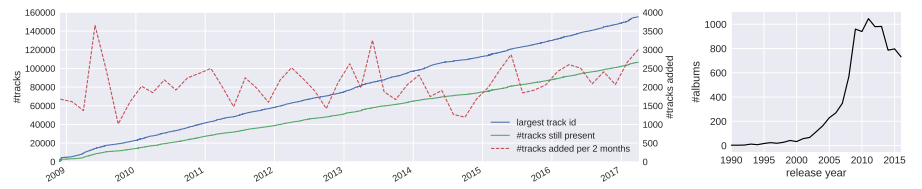


Figure Archived ablums since 2008

The data set has been analyzed and Visualized using Neo4j and all the Computations and queries are listed as follows,

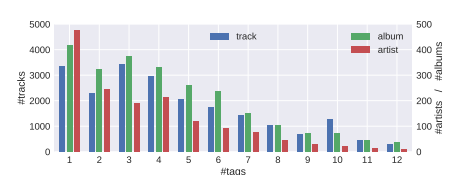


Figure Tracks vs Tags

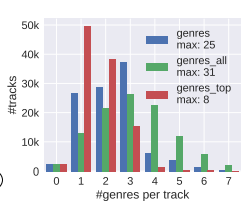
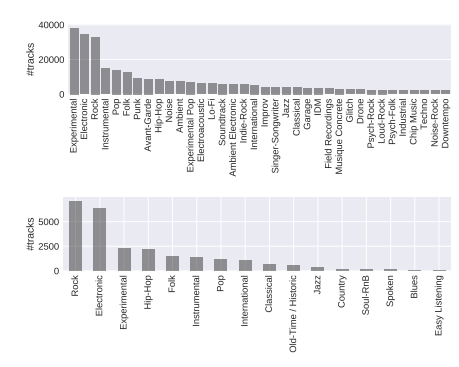


Figure Example of genres



## LOAD TRACKS.CSV DATA

Firstly, the Tracks Data has been Loaded onto the database. The Tracks data has Different Properties ranging from track\_id, Album\_id, Total number of listens, Album Title, Artist ID, Artist Name and Genre.

LOAD CSV WITH HEADERS FROM 'file:///tracks.csv'as line create (t:Tracks {trackid: line.track\_id, albumid: line.album\_id, listens: line.album\_listens, name: line.album\_title, artistid: line.artist\_id, aname: line.artist\_name, genre: line.track\_genre, Titlename: line.Title\_Track}) RETURN t.trackid as trackid, t.albumid as albumid, t.listens as listencount, t.name as albumname, t.artistid as artistid, t.aname as artistname, t.genre as genre, t.Titlename as Trackname

The data then Loaded is shown as ,

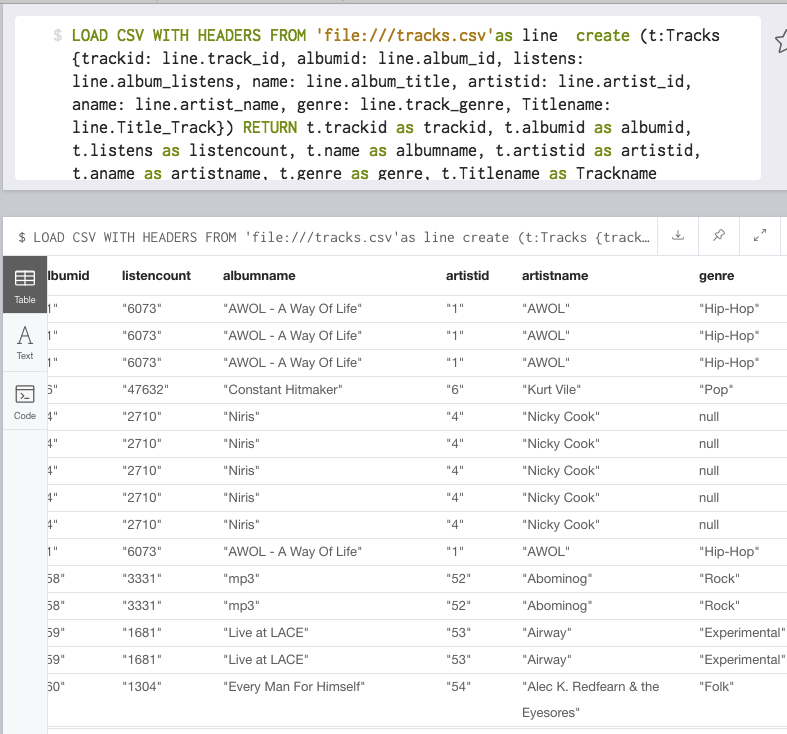


Figure 11 Load Tracks.csv into Database

## LOAD ARTISTS DATA

Then the Artists.csv data is considered and is imported and merged with the Tracks data. Thus, we have two nodes Artists and Track and the Cipher query to import the artists Data is shown as,

LOAD CSV WITH HEADERS FROM 'file:///Artists.csv'as line create (a:Artists {artistid: line.artist\_id, Favorites: line.artist\_favorites, artisthandle: line.artist\_handle, artistname: line.artist\_name, tags: line.tags}) RETURN a.Favorites as Favorites, a.artisthandle as Ahandles, a.artistid as artistid, a.artistname as artist\_name, a.tags as tags

## LOAD ARTISTS DATA and MERGE IT WITH TRACKS DATA

The loaded Artists data and tracks data is then merged. The nodes are formed with relationship “Produce” and “created by”

LOAD CSV WITH HEADERS FROM 'file:///Artists.csv'as line

MERGE (a:Artists {artistid: line.artist\_id, Favorites: line.artist\_favorites, artistname: line.artist\_name, tags: line.tags})

WITH a.Favorites as Favorites, a.artistid as artistid, a.artistname as artist\_name, a.tags as tags, line

MATCH (t:Tracks { artistid: line.artist\_id }),(a:artist { artistid: line.artist\_id })

Create (a)-[:produces]->(t)

RETURN a.Favorites as Favorites, a.artistid as artistid, a.artistname as artist\_name, a.tags as tags

The Schema of the Graph DB is shown below,

Two relationships have been established. Tracks “created by” artists and Artists “Produced” tracks

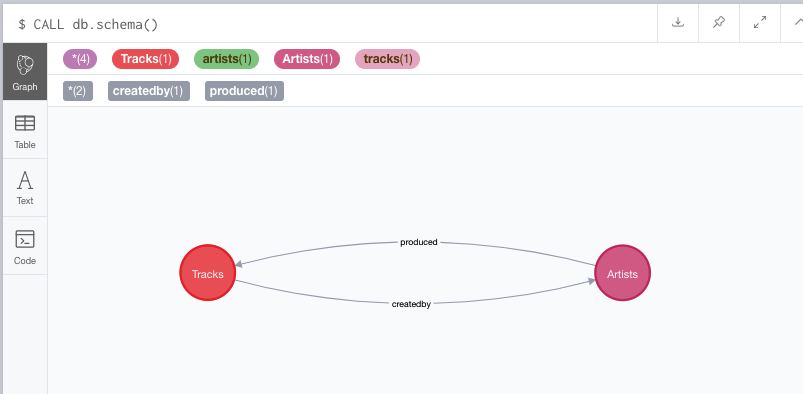


Figure 12 Node Relationships

## Relationship Between Nods Artists and Tracks

The relationships between these two nodes Artists and Tracks can be shown with a sample of25 tracks and is visualized as shown by

MATCH p=()-[r: produced]->() RETURN p LIMIT 25

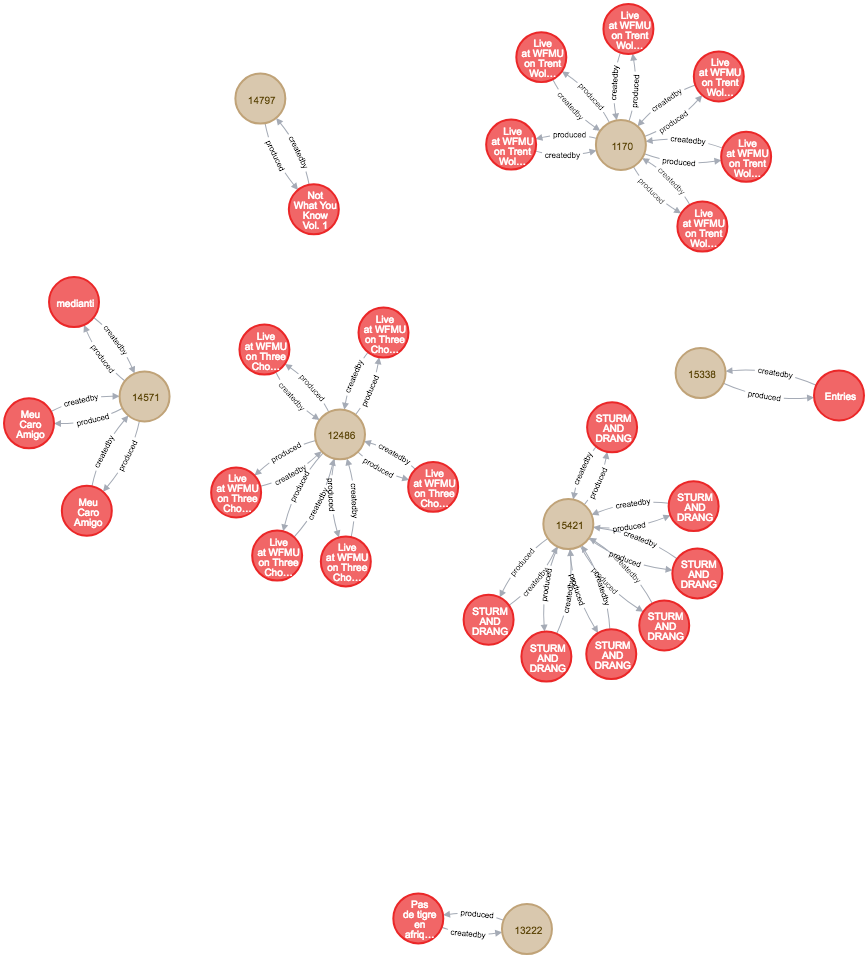


Figure 13 Nodes Relationship Graph

## TOP Tracks listened on system by artist: AWOL

When recommending a track to a user, with details of Track by a particular artist which has highest number of hits and listened over the entire database .

Thus the Artist based recommendation along with the number of listens is the higher most, the hybrid part of recommending the songs can be shown as ,

MATCH (t:Tracks)-[b:createdby]-(a:Artists)

WHERE t.aname = "AWOL"

RETURN t.trackid, t.albumid, t.listens,t.genre,a.artistid

ORDER BY t.listens DESC

LIMIT 10

The computed output can be shown as,

Artist ID: 1 Name: AWOL, Top tracks of AWOL will be,



Figure 14 Cipher Query of Artist "AWOL"

## To fetch all songs of an Album and its details

The Song list of an album can be picked from the repository as shown, This is contextual based filtering which picks from same album or category and the query can be shown as ,

MATCH (a:Artists)-[b:produced]-(t:Tracks)

WHERE t.name = "AWOL - A Way Of Life"

RETURN t.trackid, t.albumid, t.listens,t.genre,a.artistid

ORDER BY t.listens DESC

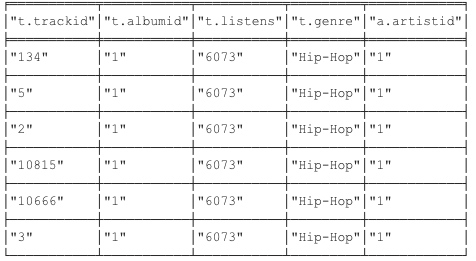


Figure 15 Cipher Query to Compute Hip pop tracks of high listens.

For every new user, there wouldn’t be any new track record or history. Therefore, by default Top 10 listened Songs in and around the demographic area and its respective suggestions are fetched and recommended to the user. Thus, the Cipher query of the database can be given as,

MATCH (a:Artists)-[b:produced]-(t:Tracks)

WHERE t.listens > "50000"

RETURN t.trackid, t.albumid, t.listens,t.genre,a.artistid

ORDER BY t.listens DESC

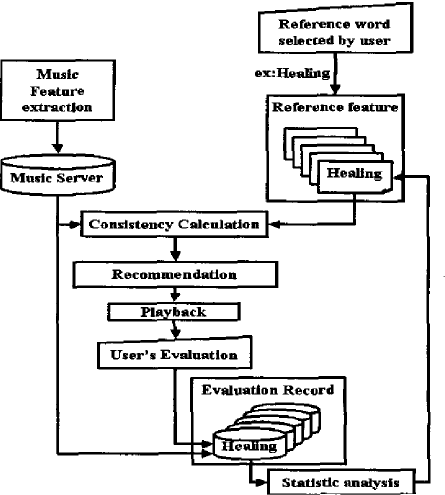
LIMIT 10

The o/p of the top 10 songs of the entire database, which the recommender system delivers or suggests to a newly logged in user can be given as,



Figure 16 Query to Compute Electronic Genre Songs with Highest Listens.

Thus based on the number of Times the track is listened ,tags of the track , there is an evaluations made with the user. Thus based on both user feedback and the no of time the song recommended , the model is built and songs has been pushed onto the user based on his feature selection. Thus this make sure that the song in the recommendation listed has passed through both content based filtering (tags) as well as collaborative filtering forming an Hybrid Recommender system.



Categorizing can be done based on the user needs, i.e. No of Listens >50000; genre should be Electronic and of a particular artist. The song list can be given as,



Figure 17 Songs with Most listen Count

## FUTURISTIC APPROACH

With suggested approach, end users will get better recommendations but with advancements in future powered by AI and advanced machine learning will help in getting some advanced recommendation systems.

# CONCLUSION

Music and AI when hold together, its more than merely determining playlist suggestions. The next advancements would eventually be online music composing platforms like Google’s Nsynth Super. This will create a great platform to create music and generate different kinds of items, publicize them and provide it to the other users. There was a period, where people where figuring out that music industry will disappear. But with the Advancements in Machine learning and AI provides new advancements provides new advancements in the field of recommender systems in the forthcoming years.

# REFERENCES

Benjamin Mayo 2018, *Apple Music now incorporates lyrics from Genius, Genius adds embedded Apple Music player,* 9TO5Mac, viewed 8 June 2019, <<https://9to5mac.com/2018/10/11/genius-apple-music/>>

Brain Tan 2017, *5 Things I Learned From My Comparative Study of Spotify, Apple Music, and Deezer,* Hacernoon, viewed 7 June 2019, < <https://hackernoon.com/5-things-i-learned-from-my-comparative-study-of-spotify-apple-music-and-deezer-fe8512022ae1>>

Sarah Perez 2018, *Spotify is falling behind on lyrics and voice*, Techchruch, viewed 8 June 2019, < <https://techcrunch.com/2018/08/15/spotify-is-falling-behind-on-lyrics-and-voice/>>

Daniil Korbut 2017, *Recommendation System Algorithms*, Blog, Stats and Bots, viewed 7 June 2019, < <https://blog.statsbot.co/recommendation-system-algorithms-ba67f39ac9a3>>

Giuliano Giacaglia 2019, *Spotify’s Recommendation Engine*, Data Driven Investor, viewed 8 June 2019, < <https://medium.com/datadriveninvestor/behind-spotify-recommendation-engine-a9b5a27a935>>

Kevin Liao 2018, *Prototyping a Recommender System Step by Step Part 1: KNN Item-Based Collaborative Filtering*, Towards Data Science, viewed 8 June 2019, < <https://towardsdatascience.com/prototyping-a-recommender-system-step-by-step-part-1-knn-item-based-collaborative-filtering-637969614ea>>

Brian Srebrenik 2018, *Machine Learning and Music Classification: A Content-Based Filtering Approach*, Towards Data Science, viewed 7 June 2019, < <https://towardsdatascience.com/machine-learning-and-music-classification-a-content-based-filtering-approach-f2c4eb13bade>>

Elena, S. 2017, ‘Collaborative filtering for music recommender system’, *IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, St. Petersburg, 2017, pp. 548-550.