Music Recommendation System Based on Genre

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Computer Science Engineering / Information Technology

by

Pranav Chauhan (191518)

UNDER THE SUPERVISION OF

Dr. Pardeep Garg (Supervisor) &

Dr. Yugal Kumar (Co-Supervisor)

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

CERTIFICATE

Candidate's Declaration

I hereby declare that the work presented in this report entitled "**Music Recommendation System Based on Genre**" in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering/Information Technology submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2022 to May 2023 under the supervision of **Dr. Pardeep Garg (Assistant Professor SG, ECE Department) & Dr. Yugal Kumar (Associate Professor, CS & IT Department)**.

The matter embedded in the report has not been submitted for the award of any other degree or diploma.

(............................)
Pranav Chauhan
(191518)

This is to certify that the above statement made by the candidates is true to the best of my knowledge.

(............................)
Dr. Pardeep Garg
Assistant Professor (SG) ECE

(............................)
Dr. Yugal Kumar
Associate Professor (CSE & IT)
Dated:

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: ....

Type of Document (Tick): PhD Thesis | M.Tech Dissertation/ Report | B.Tech Project Report | Paper

Name: ________________________ __Department: _______________ Enrolment No ________

Contact No. ___________________________E-mail. __________________________________

Name of the Supervisor: ____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): _____________________

____

____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at ....(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) Signature of HOD

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

<table>
<tr><th>Copy Received on</th><th>Excluded</th><th>Similarity Index (%)</th><th colspan="2">Generated Plagiarism Report Details (Title, Abstract & Chapters)</th></tr>
<tr><td></td><td rowspan="4">• All Preliminary Pages
• Bibliography/Images/Quotes
• 14 Words String</td><td rowspan="2"></td><td>Word Counts</td><td></td></tr>
<tr><td>Report Generated on</td><td>Character Counts</td><td></td></tr>
<tr><td rowspan="2"></td><td>Submission ID</td><td>Total Pages Scanned</td><td></td></tr>
<tr><td></td><td>File Size</td><td></td></tr>
</table>

Checked by
Name & Signature Librarian

....

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com

ACKNOWLEDGEMENT

Firstly, I express my heartiest thanks and gratefulness to almighty God for his divine blessing that made it possible for me to complete the project work successfully.

I am grateful and wish my profound indebtedness to Supervisor **Dr. Pardeep Garg (**Department of ECE) & Co-Supervisor **Dr. Yugal Kumar** (Department of CSE&IT) Jaypee University of Information Technology, Waknaghat. Deep Knowledge & keen interest of my supervisors in the field of "**Machine Learning**" helped me to carry out this project. Their endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

I would like to express my heartiest gratitude to **Dr. Pardeep Garg & Dr. Yugal Kumar** , for their kind help to finish my project.

I would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

Pranav Chauhan(191518)

Table of Contents

List of Figures

List of Graphs

List of Equations

ABSTRACT

In today's modern world listening to music is very important to people. As the Internet prices are also very reasonable now, music is always ready at your door steps. There are now millions of songs and music audios available, and this brings the problem of plenty. Listening to music is a relaxing time and people do not want to exercise their mind here. The truth is the confusion can be real i.e what to hear and what to skip. Here, the idea of music recommendation kicks in. The role of a music recommendation system is to recommend people content on the basis of their liking or their past experiences. This saves users a lot of time and confusion searching, as they now get several recommendations similar to his/her choice and user could just tap in to choose to hear anything. This application is also a very integral part of music streaming services which also solves some bigger problems like music piracy and easy of getting music. Well there are several ways to do music recommendation and music recommendation is always evolving for the good. I hereby in this project tried to make a music recommendation model that could be handy and also had done exploratory data analysis that could discover trends in the ever-volving music industry.

CH-1: INTRODUCTION

1.1 Introduction:

- Remember the days when CD collections were the only source of music, and if we wanted to listen to something new, we had to buy new CDs or trade with friends. The effort was arduous and enormous. With music streaming services like Spotify and Apple Music, you can now listen to any song by any artist listed on the Internet. This saves our time, effort, resources as well as needless storage of CD's and cassettes.

- In order for a user to receive pre-buffered music that has been stored minutes or even seconds before the song is played, the streaming service must transfer the data to the streamer in small increments. This contributes to a better music-listening experience for users as it avoids buffering.

- Music Streaming also solves the problem of piracy which indeed is a huge boost for the artists and production houses. Their music gets well deserved recognition and fame, as well as losses from piracy is also reduced over the years due to music streaming. Following are few types of music applications:
  1) Music Streaming
  2) Music Store
  3) Music Storage

- In this project I have built a Music Recommendation System that is used in applications like Spotify, Gaana etc.

- This project will be using the pre-built **Spotify API** as we all know that in today's time Spotify is the most popular music streaming application.

- **Music Streaming Application:** The most popular music streaming apps include those from Soundcloud, Spotify, Gaana, and Apple Music. Playing music that is stored in the service's database is the main goal of the music streaming application. The application's primary components are:

  1) Discovery : The application's availability of music and capacity to identify related genres are its key draws.
  2) Recommendation : We need to put in place a recommendation algorithm that can determine which genre of music a particular user will like based on his past preferences in order to give them the best possible user experience.

- **Music Store :** Applications like Pandora, Apple GarageBand and YouTube Music are the famous Music Stores available. These applications are an upgrade of the music streaming applications as they have more prominent commercial trading elements in their service.
  1) Monetization : The modern features of these applications are now engaging the listener to buy the subscription to unlock the premium features.
  2) Feedback : Since these applications encourage user feedback in the form of comments and reviews that assist other users in

considering their purchases, this area is particularly relevant for the networking component.

- **Music Storage Apps :** Applications like Amazon and Google Music are famous Music storage applications. These applications access internal storage on the device or universal cloud storage. You could create a music player that is entirely user-curated.
  1) Uploading : Apart from music store features such apps also allow external uploads.
  2) Playback : The key element of these applications are the playback capabilities.

- **Music Recommendation :** One of the major parts of these Music applications is music recommendation. There are over a billion of the songs in the whole world so which becomes a problem of plenty for the user. Of Course all the songs will not be matching his/her liking. That's where the music recommendation kicks in. The major role of a music recommendation system is to suggest users music based on their liking. Music recommendation can be done on the basis of many factors. One of the easiest ways is to recommend music on the basis of artists, movies and producers. This is not an effective way to recommend music as it is very simple and basic. One of the more effective ways to recommend music is the basis of genre.

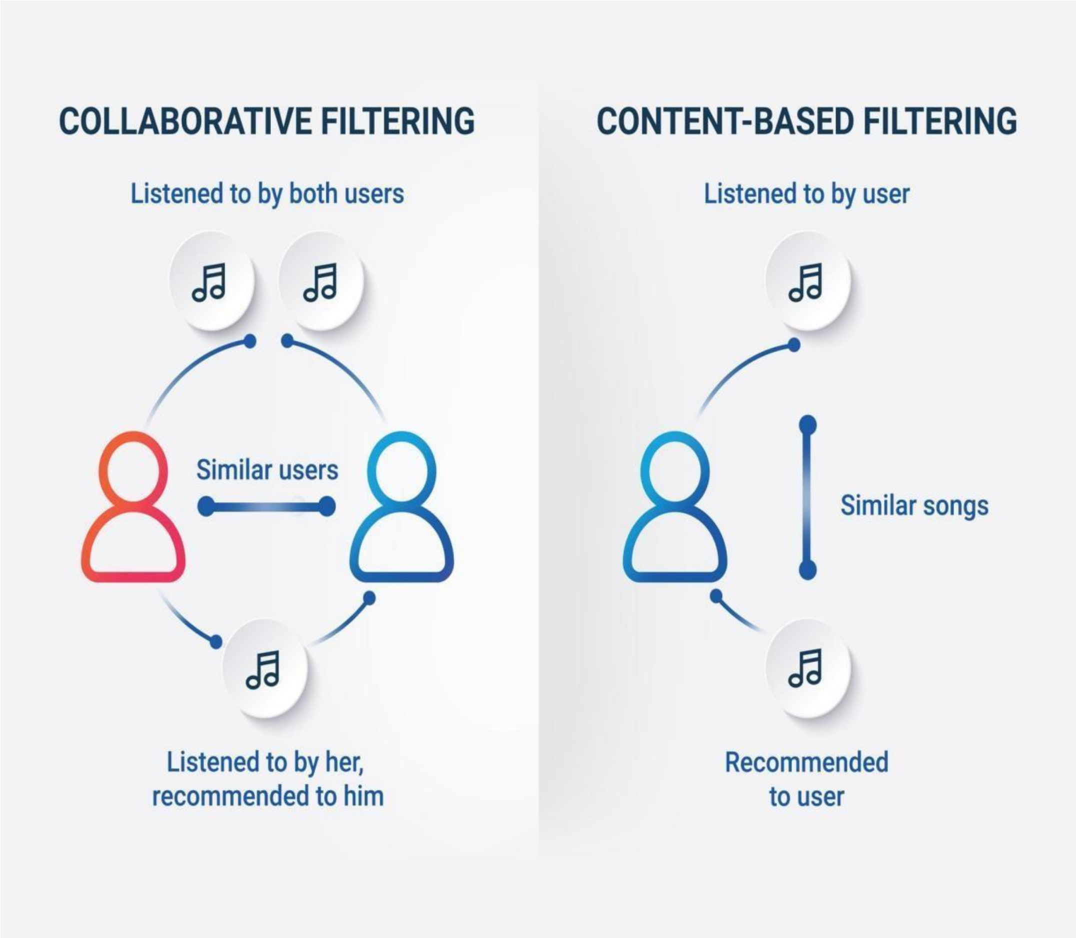


Fig 1.1 (Types of recommendation system)

Types of Recommendation system:

1)Content Based Filtering: This type of system uses similarities in products, services, or content features as well as information accumulated about the user to offer recommendations as shown in Fig 1.1.

2) Collaborative Based Filtering: Collaborative filtering relies on the preferences of similar users to offer recommendations to a particular user as shown in Fig 1.1.

1.2 Problem Statement

- Without having to download files from the internet, streaming music is a way of delivering audio material directly to your device. This technique is used by music streaming services like Spotify, Pandora, and Apple Music to distribute songs to you and combat music piracy. Remember the days when CD collections were the sole source of music, and if we wanted to listen to something new, we had to buy a new CD or trade with our peers. That effort was arduous and enormous. You can now access every song by any artist on the Internet thanks to music apps like Spotify and Apple Music.

- The classification of genres is a crucial subject with numerous practical applications. Tens of thousands of songs were reportedly posted on Spotify each month in 2016, as the amount of music being created everyday continues to soar, especially on internet platforms like SoundCloud and Spotify.

- Any music streaming platform should have the ability to quickly categorize the songs in any given playlist by genre.

- Thus, here we will be making a music recommendation system working on the Spotify dataset. We will be also doing exploratory data analysis on how the music has evolved and changed over the years and what can be the future of modern music.

- We'll also be understanding the working of music streaming apps using Spotify developer kit. Through it we'll get the idea how music streaming works.

1.3 Objectives

- Without having to download files from the internet, streaming music is a way of delivering audio material directly to your device. For it uses a music recommendation system which helps the application suggest selective content to the user.

- For building a music recommendation system, I worked on a Spotify dataset which was made up of five further csv files named (data, data_by_genres, data_by_year, data_w_genres, data_by_artist).

- For understanding how music streaming works, I will be using Spotify for Developers. Spotify Dashboard is the type of Spotify developer kit which is built for all the developers in the world who are interested in knowing how these streaming applications work.

- Spotify for Developers provides a bunch of API's that could be used to give the similar functionality that Spotify has to our own application.

- So using these API's it would be somewhat easy to build any application, like we can use the pre-built USER LOGIN API, this would help us save our time in creating our own database and Spotify Database is more secure, so in terms of security also this would be beneficial.

1.4 Methodology

- As we are building music recommendation system involved in the Music Streaming application, so here we will be intensely making the use of pre-build Spotify API's, that the hard work like maintaining the database would be done by the Spotify itself.

- In this project the major technologies that will be used are **Machine learning, React, Spotify developer kit** for communicating with the Spotify Database.

- Using Machine Learning, we will be building and training a model on the Spotify dataset. Using various libraries of Machine learning like pandas, matplotlib, numpy , I also have performed exploratory data analysis on how the music recommendations have evolved over the years and what is the future scope in it.

- A thin wrapper for the Spotify api is Spotify-web-api-js. It offers support functionalities for all endpoints, such as retrieving user data and metadata (such as searching and looking up artists and albums) (follow users, playlists, and saved tracks).

- The **Spotify for Developers** also provides the **Client Id.** This Client Id is the key to communicate, as it is unique so every developer can use the Spotify API without any risk of leaking the data or any kind of information.

1.5 Organization

Chapter 1: Introduction

Covers the various project related things such as, introduction, Problem statement, motivation and tells us about the reasons behind the choice of this project.

Chapter 2: Literature Survey

Covers the literature surveyed as well as talks about the concepts that have been studied and understood.

Chapter 3: System Development

Covers the tools and technologies that are used. It also talks about System Design- various design diagrams.

Chapter 4: Performance analysis

It covers the implementation of the project and the project snapshots.

Chapter 5: Conclusion

CH-2: Literature Review

- The Internet is a tool for connecting with people and exchanging ideas. Illegal file sharing, especially of digital music files, follows. The music industry has suffered financially from piracy, or the unauthorized copying and sharing of intellectual property; however, with the advent of streaming services like Spotify and Pandora, the sector may be able to recover.

- Streaming provides a means for Internet users and music fans to have pleasure in listening to their preferred musicians without hurting the music business. Despite the fact that music piracy is happening now and unavoidably will in the future, proponents of streaming services assert that these websites have begun to reverse the negative effects of piracy.(Belanin, Thomes, Faughnder, Hruska). However, individuals who are against streaming services offer proof that even though the percentage of music piracy is declining, streaming services may not actually be increasing music revenues and supporting the music industry (Borja, Sparshott, Swanson). Streaming has no longer had a negative effect on the music business, but many who support artists' rights believe that more can be done to protect the industry's future (Swanson, Linshi, Sparshott).

- Both proponents and detractors of music streaming services concur that these platforms are a reaction to the financial harm caused by music piracy (Swanson, Belanin, Faughnder). The first significant music pirate website was called Napster, and it was founded in 1999. It claimed to have over 57 million users at its height.

- The service was discontinued in 2001 due to copyright violations, yet Napster's legacy lives on. Other websites, like The Pirate Bay and LimeWire, have contributed to the illegitimate sharing of music. According to the Recording Industry Association of America (RIAA), music sales in the US have decreased by 47% since the advent of Napster. Additionally, according to a research by Stephen Siwek of the Institute for Policy Innovation, music piracy lowers the U.S. GDP by about $12 billion annually (Swanson).

- The issue of music piracy has an impact on both the American and the global economies. The International Federation of the Phonographic Industry (IFPI) reports that between 2004 and 2010, the global music industry's revenues fell by 31%. (Benlian). According to the IFPI, around a fifth of Internet users worldwide still use sites for unauthorized downloading. According to a MusicWatch research, this comprises roughly twenty million Americans (Faughnder).

- Piracy was fought before streaming services were developed using a number of ineffective legal strategies (Faughnder). Numerous well-known websites, such as LimeWire and Megaupload, have been shut down and fined astronomically. However, piracy websites continue to flourish on a global scale. For instance, even after The Pirate Bay's owners were found guilty of copyright theft in 2009, they still run the site. The website has often been shut down just to reopen (Faughnder).

- While there are still piracy websites, proponents of music streaming websites claim that by giving Internet users an easy and legal option to listen to music and support their favorite artists, these websites are starting to battle piracy (Swanson, Sparshott).

Work Cited :

- Benlian, Alexander, Dörr, Jonathan, Hess, Thomas, and Wagner, Thomas. "Music as a Service as an Alternative to Music Piracy? An Empirical Investigation of the Intention to Use Music Streaming Services." BISE 6 (2013): 383-396.

- Borja, Karla, Daw, Jesse, and Dieringer, Suzanne. "The effect of music streaming services on music piracy among college students." Computers in Human Behavior 45 (2015): 69-76.

- Faughnder, Ryan. "Music piracy is down but still very much in play." LA Times 28 June 2015 <http://www.latimes.com/business/la-et-ct-state-of-stealing-music-20150620-story.html>

- Hruska, Joel. "Streaming music services have killed piracy in Norway." Extreme Tech 27 January 2015.

- Linshi, Jack. "Here's Why Taylor Swift Pulled Her Music From Spotify." Time 3 November 2014. <http://time.com/3554468/why-taylor-swift-spotify/>

- Swanson, Kate. "A Case Study on Spotify: Exploring Perceptions of the Music Streaming Service." MEIEA Journal 13.1 (2013): 207-230.

- Thomes, Tim. "An economic analysis of online streaming music services." Information Economics and Policy 25 (2013): 81–91.

CH-3: SYSTEM DEVELOPMENT

Model Development:

3.1) Analytical

- Let us understand the Spotify model. The Spotify model is a people-driven approach for scaling agile that emphasizes the importance of culture and network. It has helped many of the other organizations and Spotify itself to increase innovation and productivity by focusing on autonomy and quality.

- For making the Music Streaming application successful you need to focus on licensing. Spotify uses two types of licensing for streaming music:
    1) **Sound Recording License** agreements to cover the rights to a particular record.
    2) **Music Composition License Agreements** to cover the people who own the rights to the song.

- **Data Storage** before any other work the key factor is the data storage as before creating the Music Streaming Application like Spotify, Pandora or Apple Music we should take care of data storage. We need the Strong back-end server that supports streaming. For example, we can consider using services as AWS. In this project we are not creating our own data storage system, on the other hand we are using the pre-build data storage system that is provided by Spotify. To access these data storage services we are using the pre-build APIs that are provided by Spotify, like, spotify-web-api-js is one of the API provided by Spotify.

- Let's talk about the technologies that will be used in this project :
    1) React.js **:** I had build the user interface with React.js (UI). Facebook created the open-source React.js framework, a JavaScript framework and library. It is employed to create interactive user interfaces and online applications known as single page applications rapidly and effectively with a lot less code than vanilla JavaScript.
    2) Machine Learning: This is the most important part of the model as model has to be trained using various techniques of machine learning

Streaming Applications works with different technologies in coordination. This can be seen in Fig 3.1

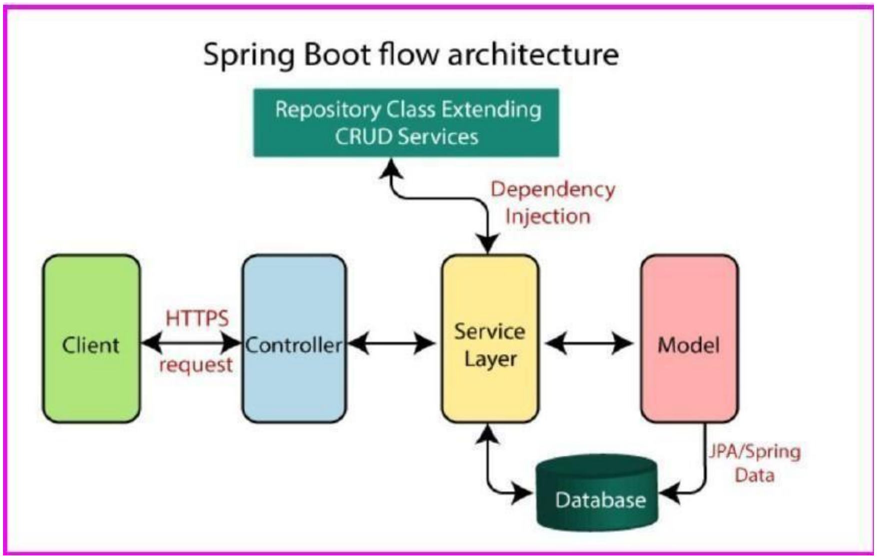


Fig-3.1 (Streaming App Working)

1) Spotify APIs : The RestFull Spotify Web API endpoints return JSON metadata about music artists, albums, and tracks, directly from Spotify Data Catalog. This can be seen in Fig 3.2

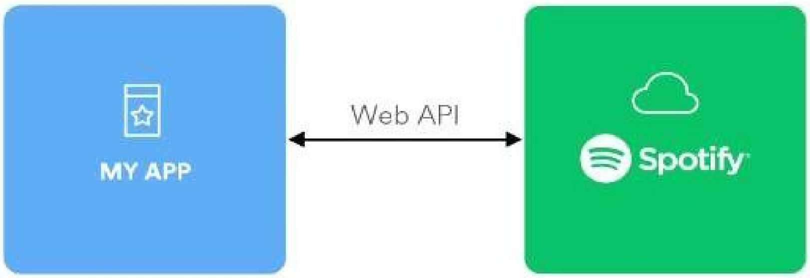


Fig 3.2 (API working)

Access to user data like playlists and songs saved in the user's Your Music library is also made possible using the Web API. Such access is made possible by the user's selected authorization.

The Web API's default address is https://api.spotify.com. A number of endpoints with various paths are offered by the API. An application needs the user's consent to access private data from the Web API, such as user profiles and playlists. Through the Spotify Account Service, authorization is granted.

Requests : REST principles are the foundation of the Spotify Web API. Standard HTTPS queries in UTF-8 format to an API endpoint are used to retrieve data resources.

Conditional Request : The majority of API responses have the necessary cache-control headers set to help with client-side coaching:

Do not make another request for a response that has been cached until it has expired.

Set the If-None-Match request header to the ETag value if an ETag appears in the response.

The Spotify server replies swiftly with 304 Not Notified status if the answer has not changed, indicating that your application should continue to use the cached version.

METHOD	ACTION
GET	Retrieves resources
POST	Creates resources
PUT	Changes and/or replaces resources or collections
DELETE	Deletes resources

These are the majorly used Spotify API endpoints.

Response: JSON objects are typically included in web API responses. Look through the reference documentation to obtain descriptions of typical endpoint answers.

Timestamps: Timestamps are returned in ISO 8601 format as Coordinated Universal Time (UTC) with a zero offset.

Response Schema: Two formats are used by Web API to specify errors:

- Authentication Error Object
- Regular Error Object

Authentication Error Object: Whenever an authentication-related Web API request is made by an application, such as when retrieving or changing an access token.

KEY	VALUE TYPE	VALUE DESCRIPTION
error	string	A high level description of the error as specified in RFC 6749 Section 5.2.
error_description	string	A more detailed description of the error as specified in RFC 6749 Section 4.1.2.1.

Authentication : All requests to Web API require authentication. This is achieved by sending a valid OAuth access token in the request header.

Music Recommendation Model Development:

Procedure-The model has been created and trained using Spotify dataset which is accessible to all. After experimenting using other models, I came to conclusion that the results were best and well matched using K-mean clustering so thus we implemented k-means clustering for this model. I also have done through exploratory data visualization on the dataset using several python libraries such as
Matplotlib , Pandas and NumPy. This notebook is very helpful in understanding different characteristics of different genres , and how these effect liking of users. This also helped us to understand that how music has evolved over the years and what could be future in it

Algorithm:

1)importing data set libraries

2)Reading data

3)Data Understanding by visualization and EDA

4)Understanding Music Over Time

5)Clustering Genres with K-Means

6)Clustering Songs with K-Means

7)evaluating results

Research Areas

Fig 3.3 (Research Areas)

1. Algorithmic Responsibility :

Spotify's algorithmic responsibility research integrates social science and machine learning research to guarantee accurate data decisions and fair algorithmic results. We conduct extensive research, conduct case studies that are product-focused, and create useful tools that teams can really utilize. We achieve this by delving deeply into the peculiarities of large-scale datasets, using machine learning algorithms that integrate different stakeholder goals, and identifying which producers are less accessible in both current recommendations and future modalities like voice.

2. Audio Intelligence :

The state of the art in understanding music at scale is being advanced through Spotify's Audio Intelligence research to improve how music is generated, recognized, and consumed. By creating machine listening technologies and synthesis algorithms, we create links from raw audio to description, similarity, suggestion, and music composition.

These drive the development of the following wave of distinctive goods and services, blurring the distinction between producers and customers. Information retrieval, source separation, auto tagging, auto mixing, mashups, sound modeling, vocal characterization, and music marketing are a few instances of active study fields in audio intelligence.

3. Evaluation :

Metrics, experimentation, and evaluation—both online and offline—are the foundation of all of Spotify's research and engineering efforts. For administering exams and analyzing outcomes, internet testing research is crucial. Since offline test collections, historical log data, counterfactual analysis, and metrics generated using a mix of quantitative and qualitative methodologies approaches are used by Spotify to understand and validate user behavior, research in offline testing is crucial. Furthermore, Spotify is committed to advancing academic research by disseminating open test collections through challenges like the RecSys Challenge and the WSDM Cup.

4. Human-Computer Interaction :

Metrics, experimentation, and evaluation—both online and offline—are the foundation of all of Spotify's research and engineering efforts. For administering exams and analyzing outcomes, internet testing research is crucial. Since offline test collections, historical log data, counterfactual analysis, and metrics generated using a mix of quantitative and qualitative methodologies approaches are used by Spotify to understand and validate user behavior, research in offline testing is crucial. Furthermore, Spotify is committed to advancing academic research by disseminating open test collections through challenges like the RecSys Challenge and the WSDM Cup.

5. Language Technologies :

Spotify users communicate their wants verbally, whether they are inputting search terms or uttering song titles. Songs and podcasts also contain words that we can decipher, categorize, and match to user preferences. We carry out research on all language technology-related topics that are relevant to audio streaming. Due to our conversational, multilingual, and interactive nature, we aid Spotify in understanding you. We ensure that our methods are scalable and incorporate strategies for creating and maintaining shared vocabularies and ontologies by learning the semantics of audio material and creators from language descriptions, including a knowledge graph object. Our research interests span computational linguistics, voice applications, natural language processing, and machine learning applied to all facets of language.

6. Machine learning :

Every part of Spotify's business is impacted by machine learning. It is utilized to assist listeners in finding content through recommendations and search, to create playlists, to understand voice commands, to serve ads, to develop business metrics and optimization algorithms, to create music with AI-assisted tools, and to extract audio content-rich signals for cataloging and other content-based applications. As we advance the state of the art in machine learning technique and applications, we are dedicated to cultivating expertise in the most recent methods. Approaches in reinforcement learning, approximation inference, graphical models, causal inference, deep learning, times series modeling, and meta-model learning are all of particular interest.

7. Music Creation :

Innovative AI-assisted music composition tools are being developed by Spotify. In both the auditory and symbolic worlds, we contribute to research on music modeling and music creation.

It is utilized to assist listeners in finding content through recommendations and search, to create playlists, to understand voice commands, to serve ads, to develop business metrics and optimization algorithms, to create music with AI-assisted tools, and to extract audio content-rich signals for cataloging and other content-based applications. As we advance the state of the art in machine learning technique and applications, we are dedicated to cultivating expertise in the most recent methods. Approaches in reinforcement learning, approximation inference, graphical models, causal inference, deep learning, times series modeling, and meta-model learning are all of particular interest. Techniques include audio modeling, sequence modeling, such as deep learning audio generation, audio concatenative synthesis, and many others.

8. The Goal :

The goal of Spotify's Search & Recommendations research is to find better ways to give consumers easy access to their preferred audio content, including music and podcasts, and to assist them in developing their tastes. It is utilized to assist listeners in finding content through recommendations and search, to create playlists, to understand voice commands, to serve ads, to develop business metrics and optimization algorithms, to create music with AI-assisted tools, and to extract audio content-rich signals for cataloging and other content-based applications. As we advance the state of the art in machine learning technique and applications, we are dedicated to cultivating expertise in the most recent methods.

9. User Modeling :

It is necessary to build technology that not only recognises user preferences and interests but also takes their contextual preferences into account in order to provide them with pertinent, timely, and inspirational recommendations in order to meet the wide range of needs of millions of users. It is utilized to assist listeners in finding content through recommendations and search, to create playlists, to understand voice commands, to serve ads, to develop business metrics and optimization algorithms, to create music with AI-assisted tools, and to extract audio content-rich signals for cataloging and other content-based applications. As we advance the state of the art in machine learning technique and applications, we are dedicated to cultivating expertise in the most recent methods. Approaches to reinforcement learning, approximation inference, graphical models, causal inference, deep learning, and time series are particularly intriguing.This makes it possible for us to develop datasets and user interaction models based on user interactions and feedback signals to develop engaging and individualized user experiences.

3.2) Experimental modeling

How to build your own "Spotify Model"?

At the beginning of 2011, you were the CTO of Spotify. In a snowy, completely dark Stockholm, you are looking out of a coffee shop window. This year has been amazing. Your company is growing more quickly than before, both domestically and globally. However, Apple and Google are catching up. The question is when, not if, they will launch their own music streaming services. Spotify needs to go global in order to survive that situation. You need to do this while knowing how the real music streaming market works. What do consumers want in reality? Will they pay someone or something? How do you convince someone to stop buying CDs or MP3s.

> "We need to innovate,
> experiment and learn faster than
> the competition."

You must expand your technical crew if you want to accomplish all of this on a worldwide basis. Growing your team from 10 to 100 in the last year has been a significant task. To accomplish this off, others say you might even need to recruit 1,000 more engineers. You experience stress. How do you find the best talent throughout the world with the appropriate mindset?
Managing a team of 100 people is challenging enough. But how can you maintain your agility as you grow even more? How do you keep the start-up mentality that has helped you succeed thus far while avoiding becoming an onerous bureaucracy?

Spotify Organisation Design :

After another two years, the business has 15 million clients. The engineering team now includes 300 members, a threefold increase. With 30 teams, how do we ensure that we create a castle that makes sense to the client instead of a pile of 30 bricks that no one likes? This problem has been on everyone's mind for a while.

The teams have begun testing a scaling model that makes use of Squads, Chapters, Guilds, and Tribes with the goal of implementing "minimum viable bureaucracy" and striking a compromise between high autonomy and high alignment. Fig 3.4 shows a depiction of Spotify organization design.

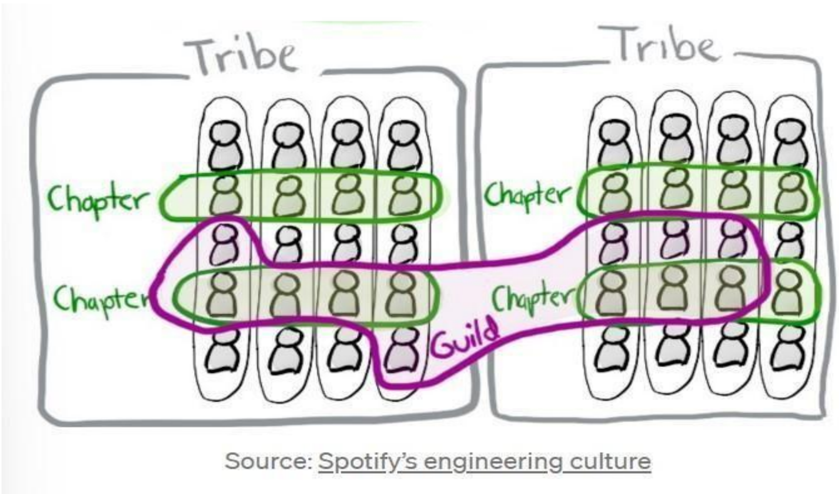


Fig 3.4 (Spotify organizational design)

However, this structure is only one part of the puzzle. Over the course of several workshops, the agile coaches created a set of organizational design guidelines with the autonomous team as the key concept.

The following is an expansion of the agile manifesto, which has been printed on office walls everywhere:

- **Continuous improvement:** My job at Spotify involves looking for ways to advance both personally and within the larger business.
- **Iterative development:**Rapid learning cycles are something we at Spotify value in order to test our theories as soon as is practical.
- **Simplicity:** Spotify's success depends on our capacity to scale. Simplicity should be your guiding for scaling. This applies to both our organizational and operational processes and technical improvements.
- **Trust:** At Spotify, we have faith that our employees and teams will make wise choices on how they work and what they work on.
- **Servant leadership:**At Spotify, managers prioritize coaching, mentoring, and removing obstacles rather than instructing staff what to do..

3.3) Mathematical modeling

Introduction :

After three of the largest firms in the world—Apple, Google, and Amazon—entered the music streaming market, there is a reason why Spotify is still in operation. They can provide me better music recommendations than the other three combined. I initially believed I was being biased because I am a devoted Spotify customer. However, after doing some research, I've discovered that Spotify actually has a system that is far superior to the others.

To make song recommendations, Spotify uses three different methods. Models for natural language processing come first. These compare songs based on the terminology used to describe them in sources like online articles. The following is used to suggest less well-known music. These content-based models listen to the audio in question and utilize similarities to suggest songs that are similar to you. Collaborative filtering is the name of the final style. In essence, it creates a user vector for each user and a song vector for each song. Then it compares them to suggest music that is comparable to each other and music that listeners with similar tastes enjoy. Collaborative Filtering will be covered in detail throughout this essay because it is the foundation of Spotify's most popular music recommendation system, Discover Weekly, and is by far the most widely used. I'll concentrate on this algorithm and, more specifically, the use of linear algebra in it.

It should be mentioned that Last.fm and the Netflix Prize have popularized another strategy that is akin to Spotify's business model. The Collaborative Filtering methodology used by Spotify, which contrasts from Netflix's more explicit feedback approach using ratings and other explicit user data, will be the only one I explore in Nelson 2. It should be noted that this is my understanding of Spotify's Collaborative Filtering based on my research, so it probably isn't exactly accurate.

Collaborative Filtering :

Every week, Spotify creates a playlist for each of its 140 million customers. This playlist is called Discover Weekly. They go through more than 40 million songs for each user to determine which ones are most likely to be favorites and which ones you don't already have in your music library. Collaborative Filtering mimics a buddy telling you about a certain song x after learning that you enjoy song y. This is accomplished using a latent factor model, a machine learning algorithm that transforms unobservable raw data into latent features. An alternating-least-squares method is used to calculate the latent factors while minimizing a cost function. In our case, computing the user-factors and the song-factors alternately is known as alternating-least-squares, and this process continuously causes the cost function to converge.

To determine how far an estimate is from the real value, a cost function is employed. You enter a specific set of characteristics into the equation. Then, using our alternating-least-squares method, you adjust these characteristics until the cost function converges. When it does, it will leave us with a preferences-confidence pair for the users that demonstrates a preference and the degree of confidence we have in that preference.

Cost Function :

The cost function Spotify uses is below along with its explanation.

$$min(x,y)\sum_{u,i}^{k} c_{ui}\left(p_{ui} - x_u^T y_i\right)^2 + \lambda\left(\sum_{u}^{n} \|x_u\|^2 + \sum_{i}^{m} \|y_i\|^2\right)$$

Equation 3.1 (Spotify Cost Function)

We have two matrices made up of user and item (song) vectors to begin with. It is a single vector that is a user vector. We must first introduce our raw data variable before we can explain the preference and confidence variables. This indicates how many times a user has listened to a certain song. It is what we prefer. A binary variable, the preference is where:

$$p_{ui} = \begin{cases} 1 & r_{ui} > 0 \\ 0 & r_{ui} = 0 \end{cases}$$

In other words, a song only has a preference if it has been listened to; otherwise, it has none.Preference alone is insufficient. Other than the fact that they don't like the song, there are many other reasons why a user might not have listened to a certain song, as well as several other reasons why they might have done so. We introduce our confidence variable for this reason. Again, there are several approaches to determining how certain you are about a preference. We'll pick to employ:

$$c_{ui} = 1 + \alpha r_{ui}$$

Equation 3.2 (Preference calculation)

This allows us to give songs that have been listened to more than once more importance. The rate of growth remains constant. Although I am unsure of the exact constant Spotify Nelson 4 is using, I have discovered that it is likely around 40. The cost function's second half:

$$\lambda\left(\sum_{u}^{n} \|x_u\|^2 + \sum_{i}^{m} \|y_i\|^2\right)$$

Equation 3.3 (Cost Function’s second part)

is applied to regularize it and prevent it from fitting too tightly. This implies that we shall make use of it to guarantee cost function convergence.

Computing Features :

We must determine how to use the alternating-least-squares approach to reduce the cost function now that we have a better understanding of it. First, we initialize and. The user vector is then calculated by:

$$x_u = (Y^T C^u Y + \lambda I)^{-1} Y^T C^u p(u)$$

Equation 3.4 (User Vector Calculation)

This is followed by computing the item vector with

$$y_i = (X^T C^i X + \lambda I)^{-1} X^T C^i p(i)$$

Equation 3.5 (Item Vector Calculation)

Until the cost function converges, we alternately insert each of these new values. The eigenvalue depends on the

After minimizing the cost function, we may rank the songs using the preference-confidence pair. Again, there are several approaches to accomplishing this. One approach would be to concentrate on the songs that are more well-known, as there is a reason why they are so. Another would be to narrow the selection to songs in that genre, making the playlist as a whole more similar. Finally, you could just rank the songs and add the top thirty to the playlist. Nelson 5 Weekly might then suggest various songs by keeping track of those that have already been suggested. Due to the fact that I assume Spotify keeps it a secret, this topic didn't come up frequently during my study.

Issues and Optimizations :

Every recommendation system has drawbacks. What is referred to as a "cold start problem" is the major drawback with collaborative filtering. You might enjoy a song if you hear it, but since it hasn't been heard by many people, it is unlikely to be recommended to you if it is uploaded by an unknown artist. Another problem with implicit data is that it is impossible to tell if a user dislikes a track—only whether they like it. Because the model cannot distinguish between a song you simply haven't listened to and a song you dislike, it may cause problems if you are recommended a song you dislike that other users like.

Collaborative filtering optimizations are not the focus of this paper, therefore I won't go into great detail about them. I will say that there has to be improvement in both speed and quality. I'll list a few and explain them. Spotify shortens the length of time instead of using the number of times a user has listened to a whole song as the raw data. This is due to the possibility of skipping a song's ending that you still enjoy. They also apply a logarithmic scale to the initial preference matrix in an effort to minimize the cold start issue.

Conclusion:

to provide a summary of the issues raised in this work. I think that any music streaming service's Discover Weekly recommendation engine is the finest. It is made by gathering all user-song pair matrices. To rate the songs for each user based on criteria like what other users who share similar interests listen to and what other songs are comparable to the ones they listen to, the playlist employs collaborative filtering to convert the pairs into Nelson 6 preference-confidence pairs. I covered how a machine learning cost function is powered by linear algebra in this work and how you can use an alternating-least-squares model to minimize the cost function to find the ideal playlist. The only way to create such a playlist for 140 million users on a service with 40 million songs is through Linear Algebra.

3.4) Statistical Model :

According to subscriber count, Spotify is the largest music streaming service in the world. One of the biggest music collections in history, along with podcasts and other audio content, are all accessible to users of the service who simply register.

It uses a freemium business model. Free Spotify access has less-than-stellar audio quality, is internet-required, and has ads. Those who purchase Spotify Premium have access to high-quality recordings without interruption and the ability to download songs for offline listening.

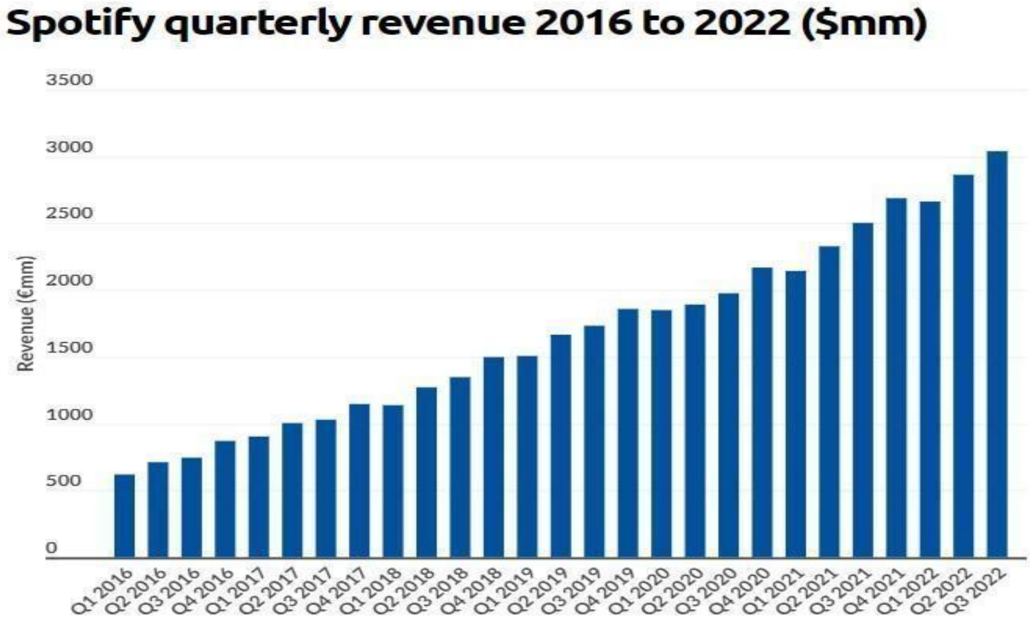
In Stockholm, Sweden, Daniel Ek and Martin Lorentzon founded Spotify. In the early 2000s, the two sought to address the growing problem of online music piracy by developing a legitimate digital music platform.

Spotify was introduced in 2008 after eventually persuading record companies to contribute content in exchange for a combined 20 percent ownership. It gained popularity quickly thanks to a relationship with Facebook and became an immediate success. After surviving the shift to mobile, Spotify went public in April 2018 and opened for business with a market cap of $26.5 billion.

Spotify Key Statistics :

1. Spotify generated €9.66 billion revenue in 2021, a 22% increase year-on-year
2. Spotify has never published a net profit. In 2021, it posted a €39 million loss

3. 422 million people use Spotify once a month, 182 million are subscribers
70 million songs are available on Spotify and 2.9 million podcasts



Graph 3.1 (Spotify Quarterly Revenue)

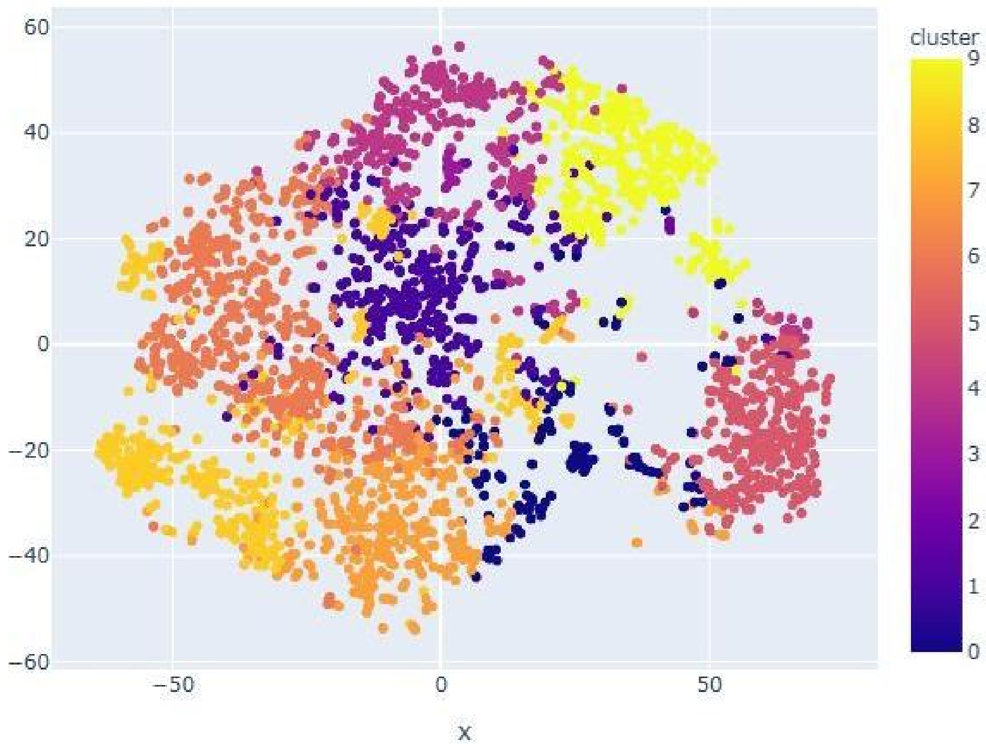
CH-4: PERFORMANCE ANALYSIS

This Chapter will give you the brief about our project working and performance.

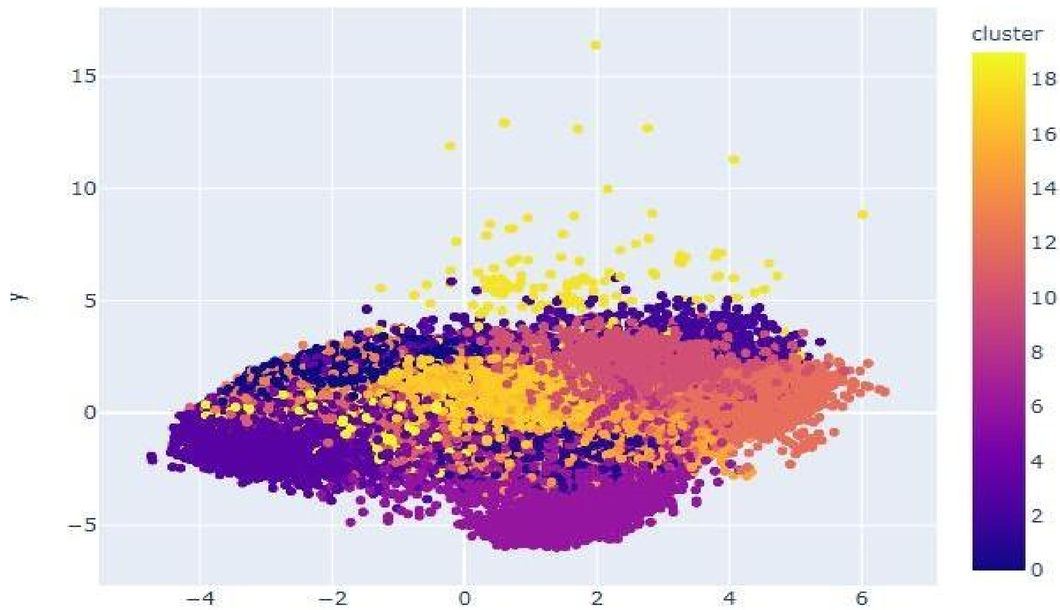
I tried to built a music recommendation system, which would recommend songs.

I used K-mean clustering to cluster genres and songs. Clustering models can be evaluated by performance perimeters like Within Cluster Similarity.We found quite encouraging results .

Following are some results we observed post model implementation



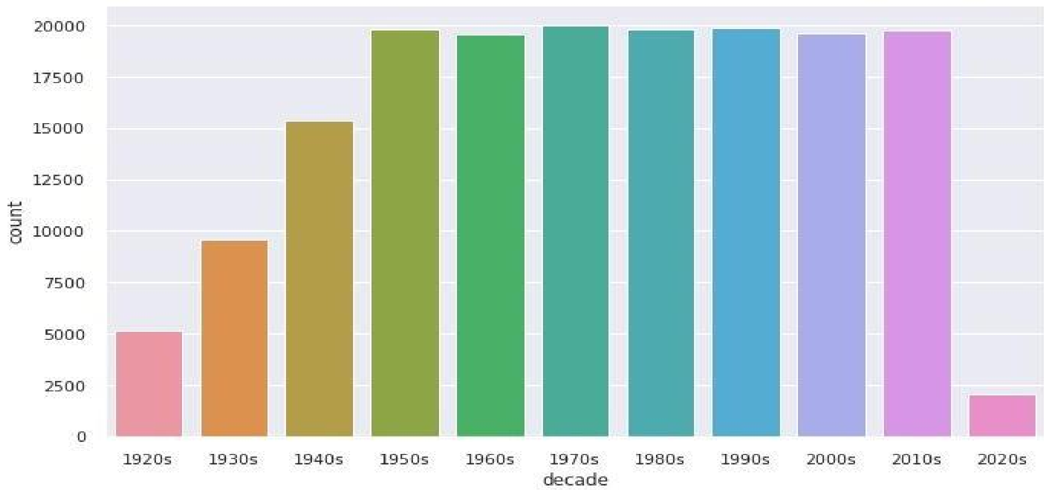
Graph 4.1 (Clustering of Genres)



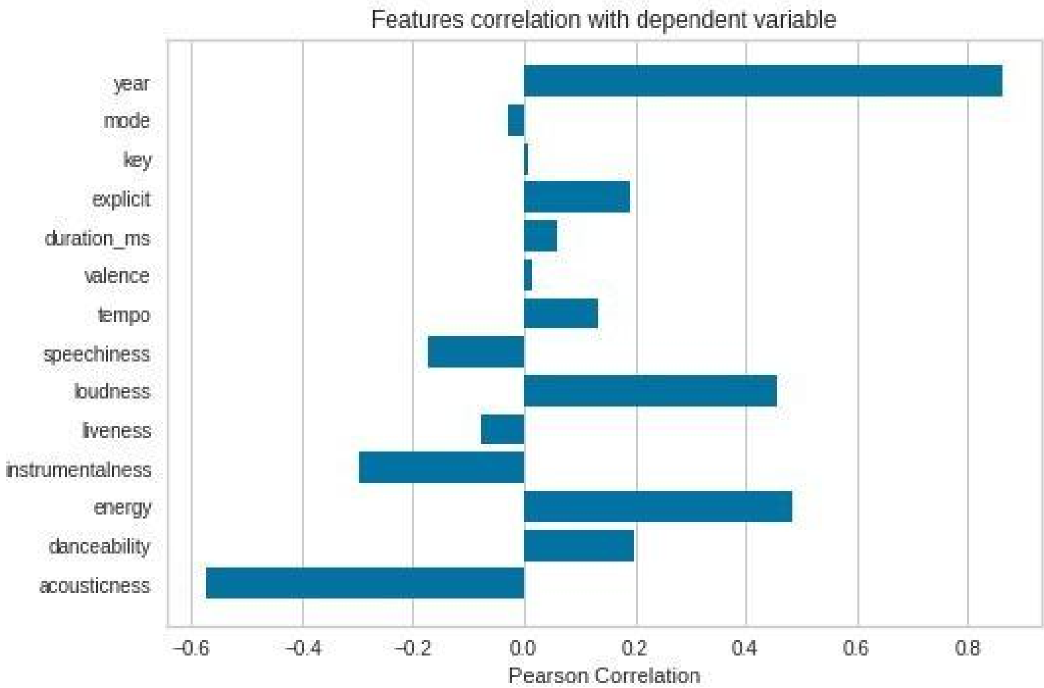
Graph 4.2 Clustering of songs

The EDA notebook which was made to discover hidden trends and other important information. Here are some of the results which we got according to the data.

Following are some of the results derived from exploratory data analysis shown in Graph 4.3 and Graph 4.4.



Graph 4.3 (Distribution of Songs over year)



Graph 4.4 (Characteristics of Different Genres)

So, we are using the Spotify Developer Dashboard to understand the working of streaming application services and the name of our application will be **Tune-Tide**. See Fig 4.1

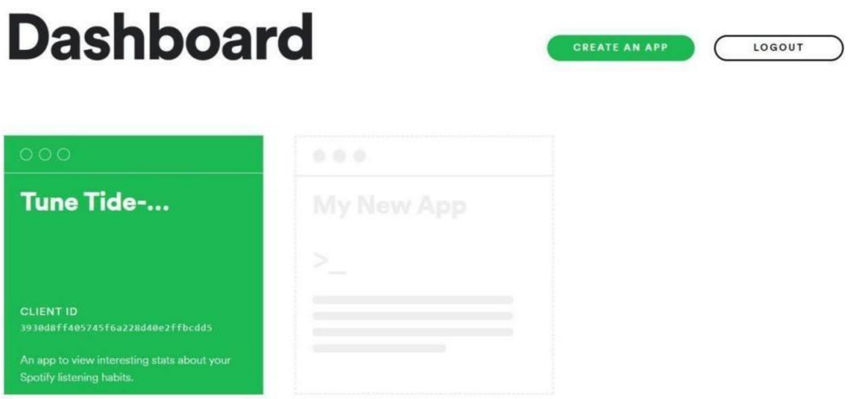


Fig 4.1 (Spotify Dashboard)

Spotify Developer Dashboard provides the Client Id which is unique for all the developers.

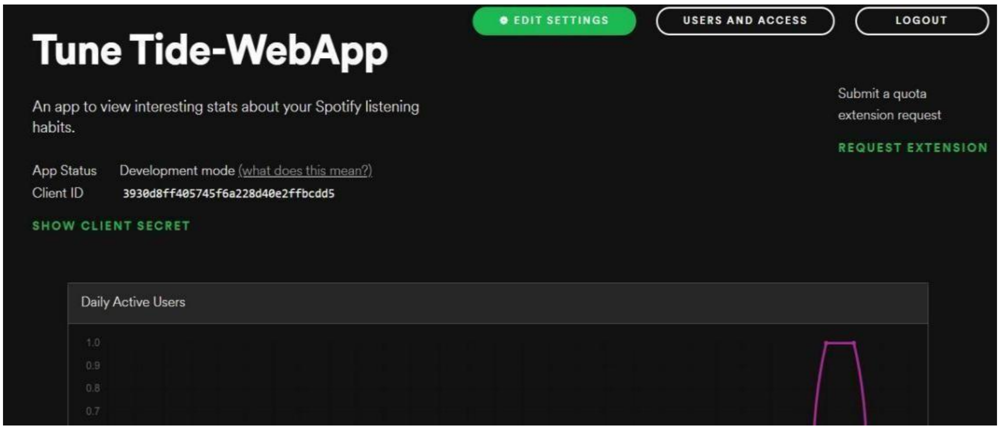


Fig 4.2(Client ID Info)

The Fig 4.2 that our web app is in development mode with the Client Id 3930d8ff405745f6a228d40e2ffbcdd5 .

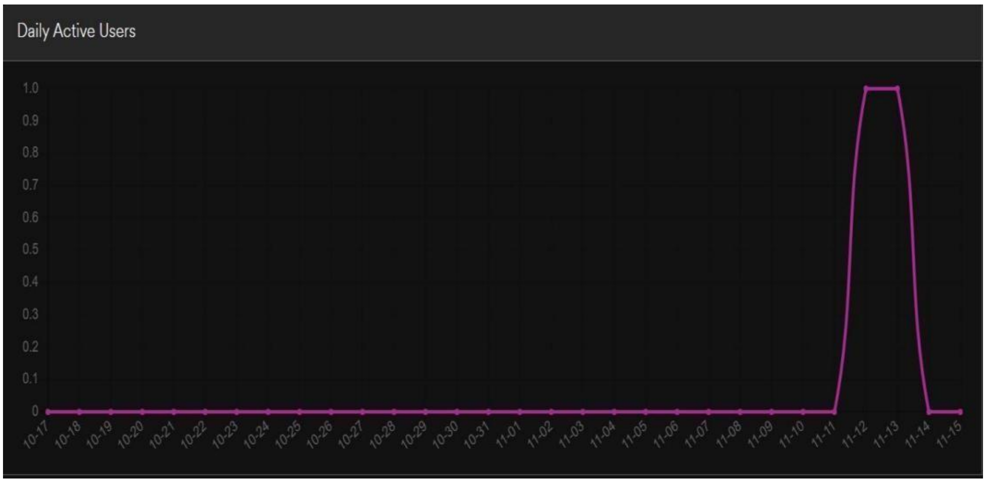


Fig 4.3 (Daily Active Users Information)

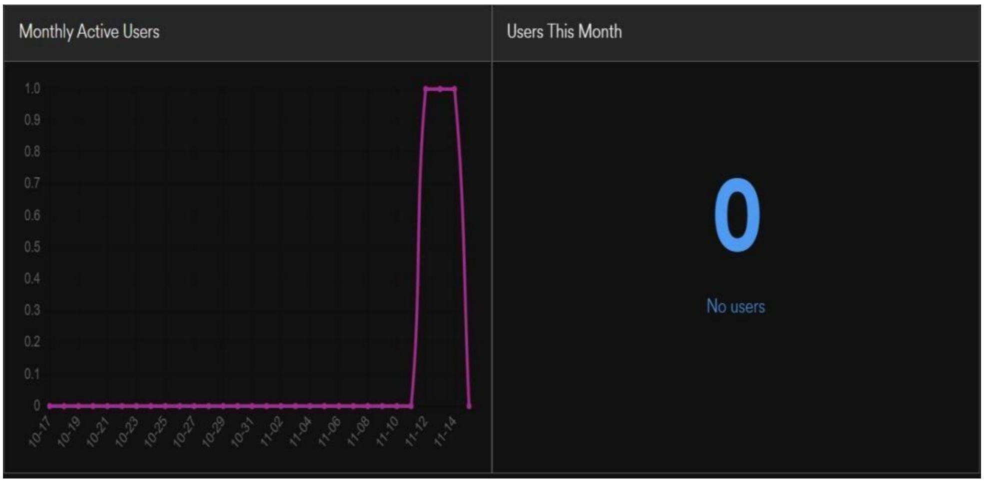


Fig 4.4(Monthly Active Users Information)

This Fig 4.3 and Fig 4.4 shows the active users that have tried my application till date.

Functionality :

When I launch my programme, it would then reroute to my client-side application, where, upon selecting Login, it would reroute to Spotify, which would then handle all user authentication, before rerouting back to our application. The Spotify web API will be used for this. A thin wrapper for the Spotify API is Spotify-web-api-js. It offers support functionalities for all endpoints, such as retrieving user data and metadata (such as searching and looking up artists and albums) (follow users, playlists, and saved tracks).

Spotify APIs : Directly from the Spotify Data Catalog, the Rest Full Spotify Web API APIs return JSON metadata about musical artists, albums, and tracks. We can find music and podcasts, manage our Spotify libraries, manage audio playing, and much more with the help of Spotify's Web API. On smaller displays, use the navigation bar at the top of this page to browse the Web API endpoints we currently offer.

Authorization using Web API :

The process of allowing a user or application access permissions to Spotify data and features is known as authorization. The OAuth 2.0 authorization framework is used by Spotify:

Fig 4.5 (Client-User Flow)

Where:

- *The Spotify user is the end user. The protected resources are made accessible by the End User (e.g. playlists, personal information, etc.)*
- *The client requesting access to the restricted resources is my app (e.g. a mobile or web app).*
- *a server that hosts the protected resources and offers authentication and authorisation using OAuth*

One or more scopes determine who has access to the protected resources. Scopes allow your application to interact with a user's library, read a playlist, or just stream content. The access rights that the user is prompted to provide depend on the set of scopes that you specify during authorization. The scopes guide contains comprehensive information regarding scopes.

Valid client credentials, including a client ID and client secret, are required for the authorisation procedure. To understand how to create them, refer to the App settings tutorial.

The authorization server then issues an access token to the user or application, who can then use it to call APIs on their behalf.

The OAuth2 standard defines four grant types (or flows) to request and get an access token. Spotify implements the following ones:

- Authorization code + PKCE extension
- Client credentials
- Implicit grant

The following table summarizes the flows' behaviors:

FLOW	ACCESS USER RESOURCES	REQUIRES SECRET KEY (SERVER-SIDE)	ACCESS TOKEN REFRESH
Authorization code	Yes	Yes	Yes
Authorization code with PKCE	Yes	No	Yes
Client credentials	No	Yes	No
Implicit grant	Yes	No	No

Authorization Code Flow :

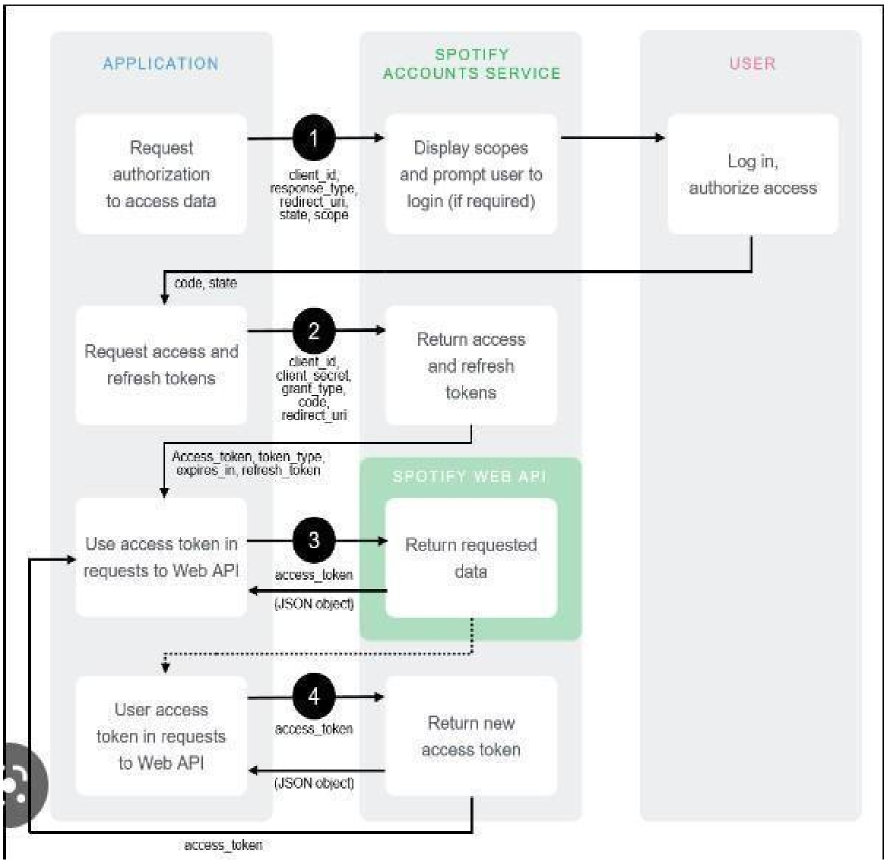


Fig 4.6(Authorization Code Flow)

Authorization Scopes:

You must learn about scopes if you want to use the Spotify Platform. With scopes, Spotify customers who utilize third-party apps can feel secure knowing that only the information they chose to share will be shared. This can be observed in Fig 4.6

Overview :

- Images
  - ugc-image-upload
- Spotify Connect
  - user-read-playback-state
  - user-modify-playback-state
  - user-read-currently-playing
- Playback
  - app-remote-control
  - streaming
- Playlists
  - playlist-read-private
  - playlist-read-collaborative
  - playlist-modify-private
  - playlist-modify-public
- Follow
  - user-follow-modify
  - user-follow-read
- Listening History
  - user-read-playback-position
  - user-top-read
  - user-read-recently-played
- Library
  - user-library-modify
  - user-library-read
- Users
  - user-read-email
  - user-read-private

Implicit Grant Flow:

The client side executes the implicit grant flow, which does not use any secret keys. As a result, you can utilize it without any server-side code. There is no refresh token provided for the issued access tokens, hence they expire quickly.

The following diagram shows how the *Implicit Grant Flow* works:

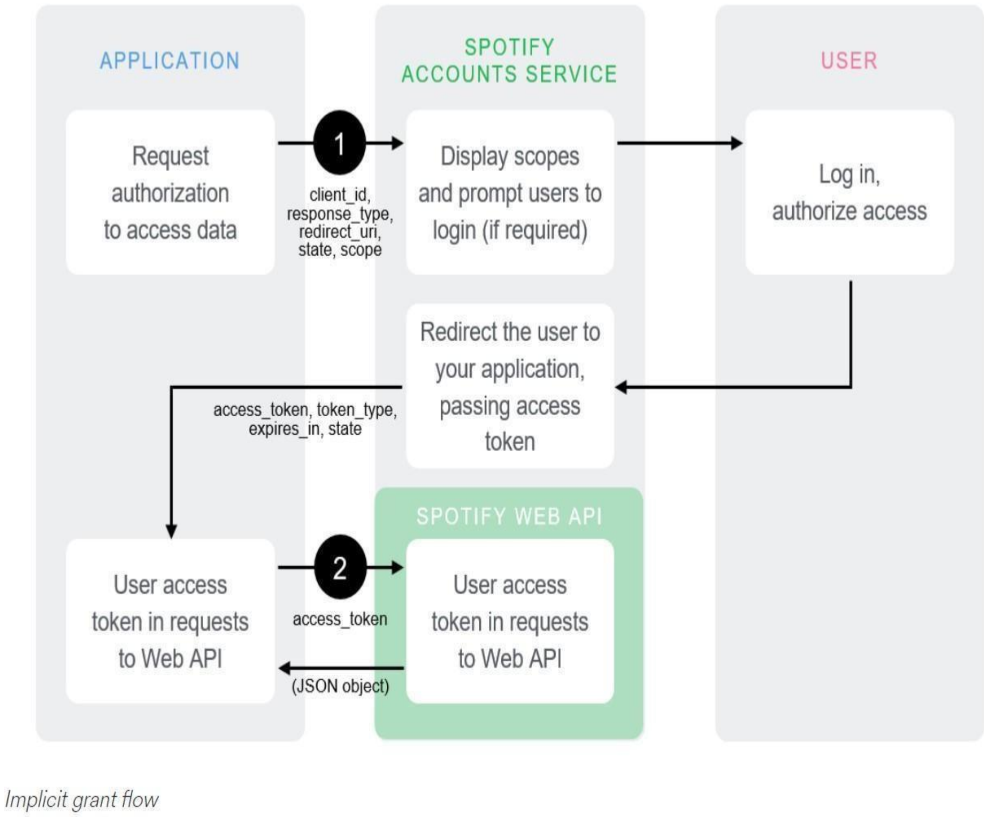


Fig 4.7 (Application overall working)

Client Credential Flow :

The Client Credentials flow is used in server-to-server authentication. Since this flow does not include authorization, only endpoints that do not access user information can be accessed.

The following diagram shows how the *Client Credentials Flow* works:

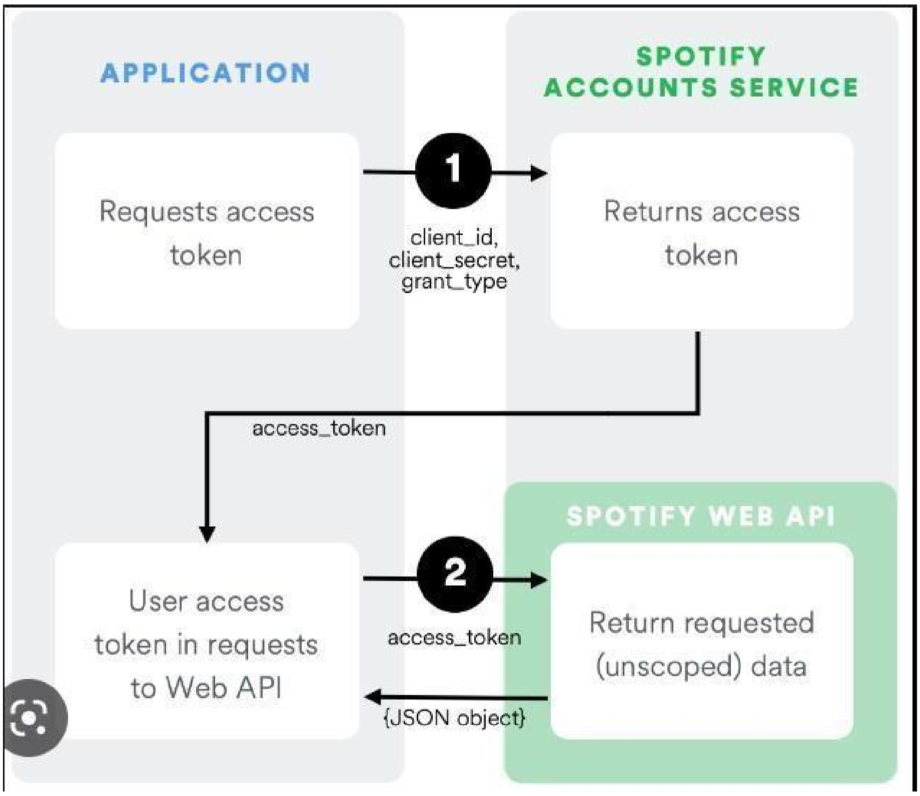


Fig 4.8 (Client-Credit Flow)

Application Working :

After running the application we will be redirected to the User Login page.

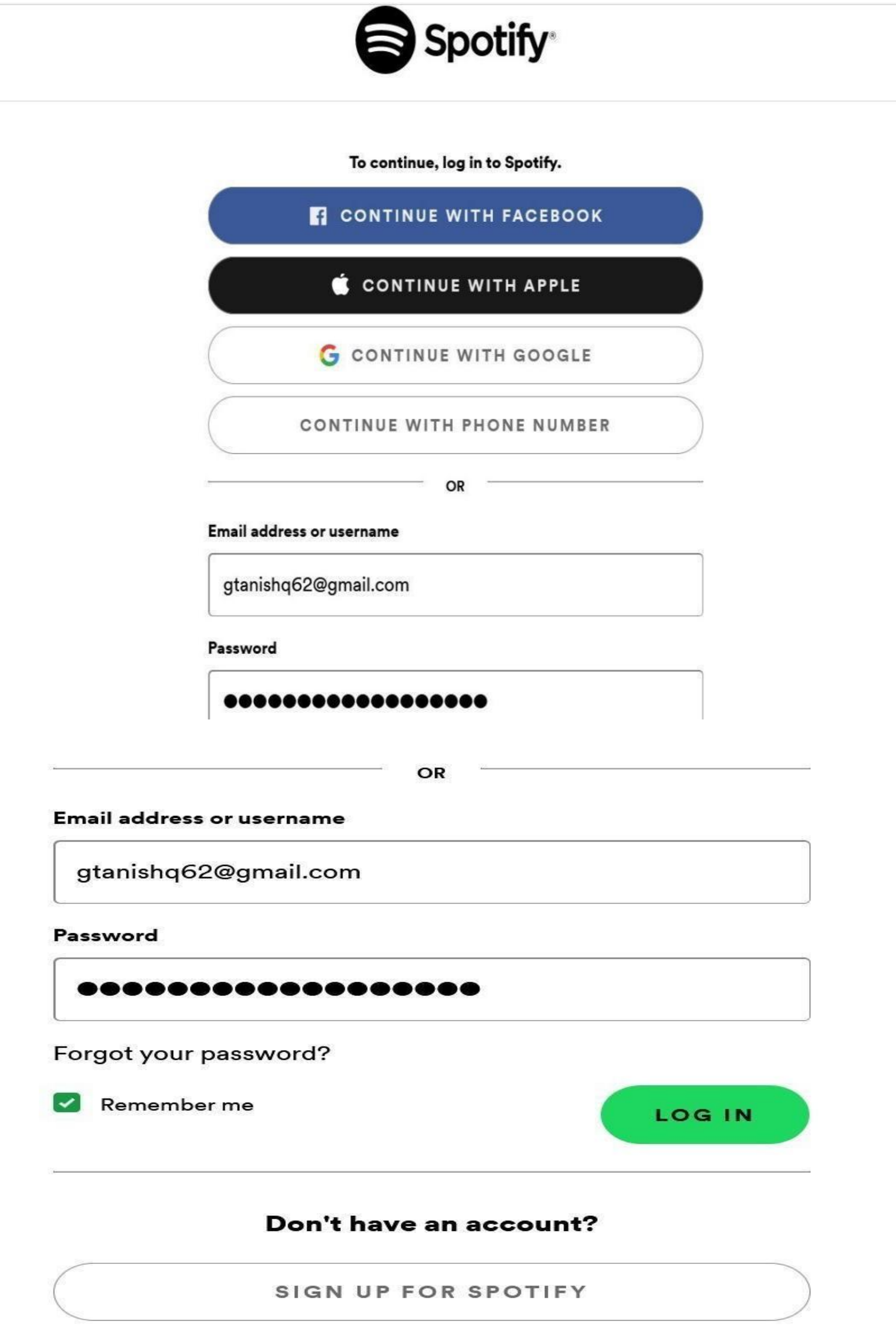
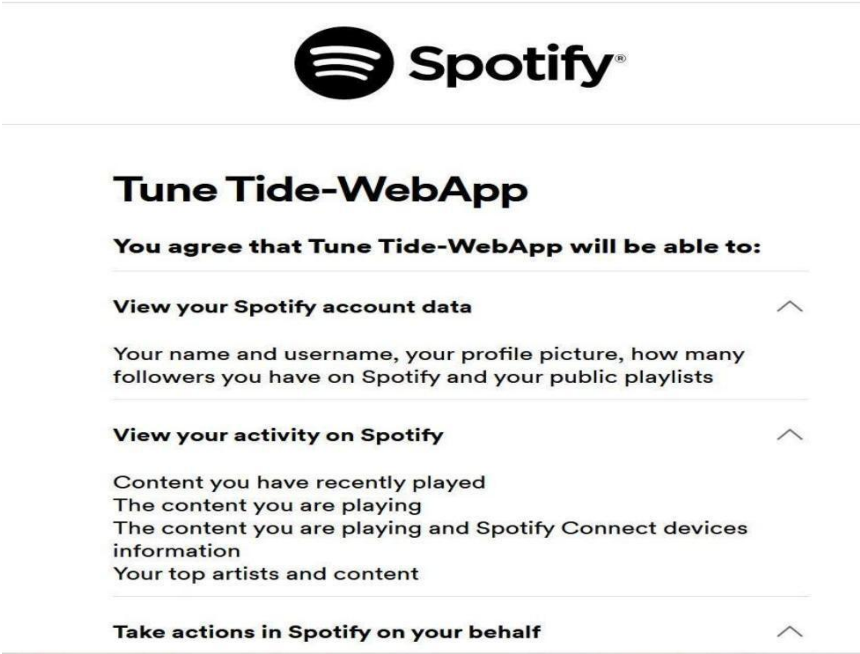


Fig 4.9(User Login Portal)

After Logging in :



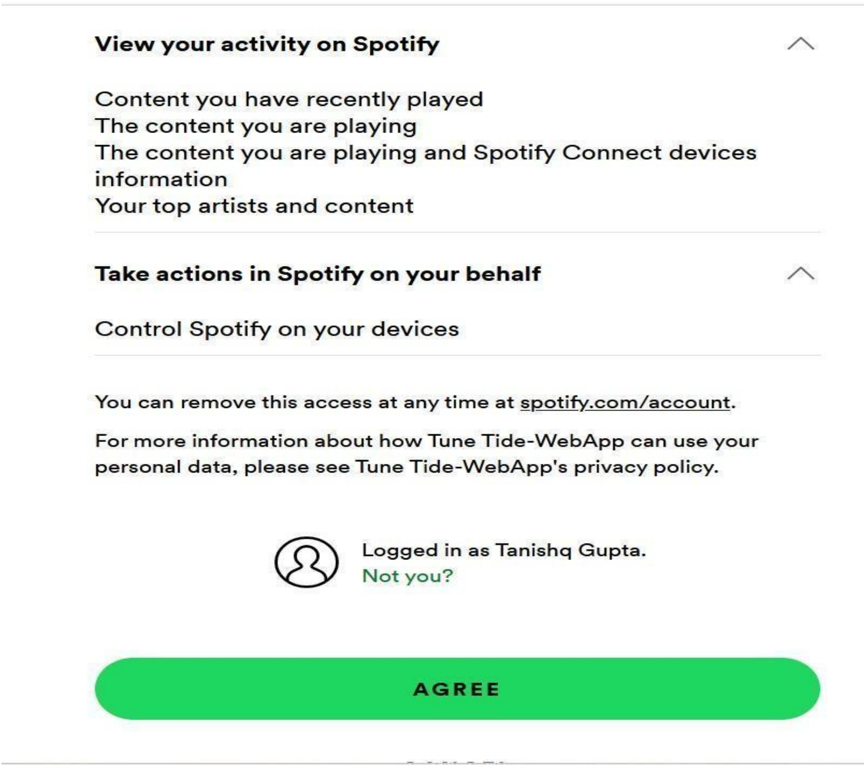


Fig 4.10 (After Login Portal)

So as you can see that after login through spotify it had redirected to our Tune-Tide Web Application.

Datasets Used :

1) The Million Playlist Dataset : The Spotify Million Playlist Dataset Challenge includes an evaluation and dataset to support music recommendation research. The RecSys Challenge 2018, which lasted from January to July 2018, is continuing with this project. Between January 2010 and October 2017, users of the Spotify platform generated 1,000,000 playlists, complete with playlist and track titles. Given a seed playlist title and/or an initial set of tracks in a playlist, the assessment job is to predict the succeeding tracks in that playlist. No awards will be given for this open-ended competition, which aims to promote music recommendation research (other than bragging rights).

2) We funded the RecSys Challenge 2018 in 2018, which brought the Million Playlist Dataset (MPD) to the research community, to enable this kind of research at scale. This dataset of 1 million playlists, which was drawn from the over 4 billion public playlists on Spotify, is the largest collection of publicly available music playlists ever and includes over 2 million distinct recordings by about 300,000 artists. Between January 2010 and November 2017, US Spotify users produced public playlists that are included in the dataset. 2018's competition saw 1,467 entries from 410 teams between January and July. The ACM Transactions on Intelligent Systems and Technology released an overview of the contest and the top-scoring entries.

3) In September 2020, we re-released the dataset as an open-ended challenge on AIcrowd.com. The dataset can now be downloaded by registered participants from the Resources page.

Task :

The challenge's objective is to create a system for automatically continuing playlists. Participants' systems shall produce a list of suggested music that can be added to a given playlist given a set of playlist attributes, therefore "continuing" the playlist. We explicitly define the task as follows:

Input

- A user-created playlist, represented by:
  - Playlist metadata (see the dataset README)
  - K seed tracks: a list of K tracks in the playlist, where K can equal 0, 1, 5, 10, 25, or 100.

Output

- A list of 500 recommended candidate tracks, ordered by relevance in decreasing order.

Note that the system should also be able to cope with playlists for which no initial seed tracks are given! To assess the performance of a submission, the output track predictions are compared to the ground truth tracks ("reference set") from the original playlist.

Evaluation Parameter :

The ensuing metrics will be used to assess submissions. Both the track level (precise track match) and the artist level evaluation of all metrics will take place (any track by the same artist is a match).

The ground truth set of tracks by G and the ordered list of suggested tracks by R are indicated in the sections that follow. |.| indicates the size of a set or list, and we use from:to-subscripts to index a list. In the event of a tie on any one of the individual measures, earlier entries take precedence.

R-Precision :

R-precision is the number of retrieved relevant tracks divided by the number of known relevant tracks (i.e., the number of withheld tracks):

$$\text{R-precision} = \frac{\left|G \cap R_{1:|G|}\right|}{|G|}$$

Equation 4.1(R-precision)

The metric is averaged across all playlists in the challenge set. This metric rewards the total number of retrieved relevant tracks (regardless of order).

Normalized Discounted Cumulative Gain (NDCG) :

The ranking quality of the suggested tracks is measured by Discounted Cumulative Gain (DCG), which rises as relevant tracks move up the list. Calculating the DCG and dividing it by the ideal DCG, in which the suggested tracks are ideally ranked, yields the normalized DCG (NDCG):

$$DCG = rel_1 + \sum_{i=2}^{|R|} \frac{rel_i}{\log_2 i}$$

Equation 4.2 (Discounted Cumulative Gain)

The ideal DCG or IDCG is, in our case, equal to:

$$IDCG = 1 + \sum_{i=2}^{|G \cap R|} \frac{1}{\log_2 i}$$

$$NDCG = \frac{DCG}{IDCG}$$

Equation 4.3 (Normalized Discounted Cumulative Gain)

Recommended Songs :

A Spotify tool called Recommended Songs suggests 10 songs to add to a playlist based on a set of music in the playlist. Ten additional tracks can be created by updating the list. The number of refreshes required before finding a relevant tune is the number of Recommended Songs clicks. The formula is as follows.:

$$\text{clicks} = \left\lfloor \frac{\arg\min_i \{R_i : R_i \in G|\} - 1}{10} \right\rfloor$$

Equation 4.4 (Recommended Songs Clicks)

If the metric does not exist (i.e. if there are no relevant tracks in R1, a value of 51 is picked (which is 1 greater than the maximum number of clicks possible).

Spotify Dataset :

We'll be using Spotify Dataset. This is an open dataset accessible to all.This dataset provides us vast information including different genres, different characteristics value of different genres, songs by different artists and produced in different year. The dataset is divided into five further csv files i.e data.csv, data_by_artist, data_by_year, data_by_genres, data_w_genres.

Spotify Podcast Dataset :

Podcasts are an audio-only media that is quickly expanding. They entail novel usage patterns and communicative standards, and they inspire research in a variety of novel areas. We offer the Spotify Podcast Dataset, which contains information in both English and Portuguese, to aid in such study.

This dataset consists of more than 100,000 episodes from various podcast shows on Spotify, with 100,000 episodes each in English and Portuguese. The dataset is accessible for analysis.

The TREC Podcasts Track shared tasks were the original purpose for which the English-language dataset was developed. Two tasks focused on comprehending podcast content and improving podcast search capabilities were given to participants.

We are making this dataset more freely available to encourage research on podcasts from the perspectives of linguistics, information retrieval, voice and audio technology, and natural language processing. The dataset includes more than 1 billion transcribed words and over 100,000 hours of audio. The episodes range in length, subject matter, style, and quality.

CH-5: CONCLUSION

Music Streaming is a very popular area which will definitely be in trend for the upcoming decade. Though it will grow leaps and bounds. Music streaming apps are a great way of generating revenue and for this attaining maximum users to use your service is the target. A great user experience is the best way to attract users. Any user will buy any premium service only when he/she is assured of something new and unique. In today's time competition is constantly increasing rapidly and only the applications which are coming up with something new and unique
will stay in command, others will fade away.

Our target is to create a great environment for people to enjoy music and content along with some other services which would be unique in itself. Spotify is the leading platform in music streaming.We intend to take its essence but also intend to try other things which would be handy in better user experience.

Some of the ideas which our work wish to encourage:

- **User interactive space**- We wish to create an interactive and dynamic space where our enthusiastic users can also participate and share their own content i.e(short music videos,poetries and ghazals) made by themselves. We would also allow other users to see their content and give their feedback. This would give people a platform where along with having fun, they can also share their talent and have honest feedback.

- Mood recognition- This is an area which is new and still undiscovered. Usually people listen to music which goes along with their mood. If we can identify what mood our users are experiencing and then suggest them music and content according to that it will present a great experience. Though it will be a complex step to take but if made possible it will give an edge over other brilliant powers in this field.Mood recognition could be done by reading facial expressions or noting their general experience in general.

- News Podcasts- News is an area which is very important. News is such a topic which can give useful knowledge of the current happenings which is very essential for any common user. Though in today's time we are seeing a trend of fake news and even hate mongering. News has become a platform of spreading agendas of their own. The main observation here is that getting unbiased and true news without filtering in today's time has become such an impossible task. We wish our platform to stay away from these false and biased news and have independent and unbiased news content.

- Customized Playlists-Playlists is an easy yet most convincing way of listening to music. We wish to create playlists for different moods and different occasions which would be most suitable and likable for a particular user. This could be done by noting their past choices.

- Podcasts-Instead of spending the entire time reading or viewing a video, podcast audio content allows listeners to delve deeper into a subject. Podcasts also provide information in more manageable bits, making them ideal for busy people or everyday commuters. Podcasts offer a wonderful experience and are the way of the future since they have the ability to challenge conventional wisdom.

Future scope:

Music Streaming isn’t just a system and format, it is a large business model. It has a large scope for expansion as music will always be in demand. Music streaming accounts for over 85% of the revenue generated by the music industry all over the globe. Those days are gone when we would run for a CD/DVD of an album of our favorite artist.

The Recording Industry of America reported that the revenue generated by paid streaming services, and radio streaming grew by 12%. Even Goldman Sachs predicted that this industry would be about a 30 billion dollar industry by 2030 this means this has no slowing down.

Ultimately new technologies would take recorded streaming to the next level.

Can Social Media be the face of Music Streaming in the near future?

In covid times the ability of artists to engage with their fans at concerts has become low. This opens up an opportunity for potential new ways to come and take over advantage.Without being performing at concerts ,artists have taken to social media platforms to engage with their audience.

The changing face of music streaming

The 'one size fits all' design of these platforms, which is the final complaint of music streaming services as they currently exist, does not support all musical genres. One of music streaming major advantages is its capacity to put a vast musical repertoire at users' fingertips, yet some musical genres can perceive this as a drawback.

For instance, jazz and classical music, which typically have longer play times, have been thought to suffer from the popularity of playlists with a range of shorter songs with obvious hooks.

This might result in new options for streaming providers to cater to users with multiple subscription tiers. Amazon Music, which offers users multiple different levels ranging from a free tier to a $14.99 HD membership, already uses this tactic. Personalized features and services allowing users to connect more closely with their favorite artists may also be available in the future.

For a range of prices, streaming services may provide specialized catalogs for followers of particular genres or niches, the option to buy one-time "day passes" to access playlists for an occasion like a party or barbecue, or higher "VIP" subscriptions for super-fans who may be given access to exclusive content, early access to tour dates, etc.

This would allow the industry to cater to more casual listeners who may not want to pay for a full subscription and also offer perks for avid music listeners who want to access more content. Music is varied and personal, so why should its streaming be any different?

How could Music Streaming be evolved for better?

1) **Mood recognition-** This is an area which is new and still undiscovered. Usually people listen to music which goes along with their mood. If we can identify what mood our users are experiencing and then suggest them music and content according to that it will present a great experience. Though it will be a complex step to take but if made possible it will give an edge over other brilliant powers in this field. Mood recognition could

be done by reading facial expressions or noting their general experience in general.

2) **User interactive space**- More the person could participate in something, more deeply involved he/she will become. Imagine there would be a streaming platform where you could share your content being completely anonymous and have an honest feedback about it from other people. Wouldn't it be good. Similarly we wish to create an interactive and dynamic space where our enthusiastic users can also participate and share their own content i.e (short music videos, poetries and ghazals) made by themselves. We would also allow other users to see their content and give their feedback. This would give people a platform where along with having fun, they can also share their talent and have honest feedback.

3) **Podcast**-Podcasts is the way ahead to give useful knowledge . In this fast paced world, no one has the time to read long books or to study about things whereas podcasts are something you could listen to peacefully with your regular schedule. If you choose the right podcasts made by the right people, these can educate you in a short span of time. Instead of spending the entire time reading or viewing a video, podcast audio content allows listeners to delve deeper into a subject. Podcasts also provide information in more manageable bits, making them ideal for busy people or everyday commuters. Podcasts offer a fantastic experience and are the way of the future since they have the ability to challenge conventional wisdom.

Application Contributions:

From the beginning our focus is on to improve lively user experience. We know that the high quality music streaming is already being provided by apps like Spotify but our target is to provide something unique which would be the way ahead.

Some features which our work would encourage applications better experience:

a) **User interactive space**- We wish to create an interactive and dynamic space where our enthusiastic users can also participate and share their own content i.e(short music videos, poetries and ghazals) made by themselves. We would also allow other users to see their content and give their feedback. This would give people a platform where along with having fun, they can also share their talent and have honest feedback.

b) **Mood recognition**- This is an area which is new and still undiscovered. Usually people listen to music which goes along with their mood. If we can identify what mood our users are experiencing and then suggest them music and content according to that it will present a great experience. Though it will be a complex step to take but if made possible it will give an edge over other brilliant powers in this field.Mood recognition could be done by reading facial expressions or noting their general experience in general.

c) **Folk Music**- We know that India is a place of wide heritage and culture.Folk Music is a part of our heritage.The point is that we don't get access to our real folk music that often.Our application intends to give these a music a platform and when users can enjoy ou real cultural music.Wouldn't that be great.It would be an unique experience.

d) **News Podcasts**- News is an area which is very important. News is such a topic which can give useful knowledge of the current happenings which is very essential for any common user. Though in today's time we are seeing a trend of fake news and even hate mongering. News has become a platform of spreading agendas of their own. The main observation here is that getting unbiased and true news without filtering in today's time has become such an impossible task. We wish our platform to stay away from these false and biased news and have independent and unbiased news content.

References

- Benlian, Alexander, Dörr, Jonathan, Hess, Thomas, and Wagner, Thomas. "Music as a Service as an Alternative to Music Piracy? An Empirical Investigation of the Intention to Use Music Streaming Services." *BISE* 6 (2013): 383-396. Online Database.
- Borja, Karla, Daw, Jesse, and Dieringer, Suzanne. "The effect of music streaming services on music piracy among college students." *Computers in Human Behavior* 45 (2015): 69-76. Online Database.
- Faughnder, Ryan. "Music piracy is down but still very much in play." *LA Times* 28 June 2015 <http://www.latimes.com/business/la-et-ct-state-of-stealing-music-20150620-story.html>
- Hruska, Joel. "Streaming music services have killed piracy in Norway." *Extreme Tech* 27 January 2015. Online Database.
- Linshi, Jack. "Here's Why Taylor Swift Pulled Her Music From Spotify." *Time* 3 November 2014. <http://time.com/3554468/why-taylor-swift-spotify/>
- Sparshott, Jeffrey. "Spotify, Downloads and Piracy: How Streaming Affects Music Sales." *Wall Street Journal* 30 October 2015.<http://blogs.wsj.com/economics/2015/10/30/spotify-downloads-and-piracy-how-streaming-affects-music-sales/>
- Swanson, Kate. "A Case Study on Spotify: Exploring Perceptions of the Music Streaming Service." *MEIEA Journal* 13.1 (2013): 207-230. Online Database.
- Thomes, Tim. "An economic analysis of online streaming music services." *Information Economics and Policy* 25 (2013): 81–91. Online Database

Datasets

1. Kaggle dataset- This is an open dataset provided by Kaggle with several csv files containing information about different.

2. "An Analysis of Approaches Taken in the ACM RecSys Challenge 2018 for Automatic Music Playlist Continuation" by H. Zamani, M. Schedl, P. Lamere, C.W. Chen.

3. Details on each of the top submissions, including papers, slides, and code, can be found on the RecSys Challenge 2018 website, and in the Proceedings of the ACM Recommender Systems Challenge 2018.

Plag_rep

ORIGINALITY REPORT

18%	16%	3%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	research.spotify.com Internet Source	3%
2	www.aicrowd.com Internet Source	2%
3	sites.dartmouth.edu Internet Source	2%
4	developer.spotify.com Internet Source	2%
5	medium.com Internet Source	1%
6	beta.developer.spotify.com Internet Source	1%
7	Submitted to University of Central Lancashire Student Paper	1%
8	Submitted to Napier University Student Paper	1%
9	Submitted to Sunway Education Group Student Paper	1%

10	www.cleveroad.com Internet Source	1 %
11	Submitted to Jaypee University of Information Technology Student Paper	<1 %
12	Submitted to BIMM Group Student Paper	<1 %
13	Submitted to University of Nottingham Student Paper	<1 %
14	research.atspotify.com Internet Source	<1 %
15	Submitted to International University - VNUHCM Student Paper	<1 %
16	Submitted to University of Sheffield Student Paper	<1 %
17	Submitted to Instituto de Empress S.L. Student Paper	<1 %
18	Submitted to National College of Ireland Student Paper	<1 %
19	elizabethsloans.co.uk Internet Source	<1 %
20	Submitted to Florida Institute of Technology Student Paper	<1 %

21	Submitted to Loreto College Coorparoo Student Paper	<1%
22	Vasiliy Rubtsov, Mikhail Kamenshchikov, Ilya Valyaev, Vasiliy Leksin, Dmitry I. Ignatov. "A hybrid two-stage recommender system for automatic playlist continuation", Proceedings of the ACM Recommender Systems Challenge 2018 on - RecSys Challenge '18, 2018 Publication	<1%
23	Submitted to University of Portsmouth Student Paper	<1%
24	www.businessofapps.com Internet Source	<1%
25	Submitted to Birkbeck College Student Paper	<1%
26	Submitted to University of Wales, Bangor Student Paper	<1%
27	Submitted to British University of Vietnam Student Paper	<1%
28	Submitted to Purdue University Student Paper	<1%
29	explore.openaire.eu Internet Source	<1%
30	harvest.usask.ca Internet Source	<1%

31	reliefweb.int Internet Source	<1%
32	umkmnews.com Internet Source	<1%
33	www.peninsulaclarion.com Internet Source	<1%
34	www.cell.com Internet Source	<1%
35	www.section.io Internet Source	<1%

Exclude quotes	On	Exclude matches	< 7 words
Exclude bibliography	On		