Analysis of Popularity of Artists on Spotify

Mrugank Jadhav
mjadhav1@binghamton.edu
SUNY Binghamton
Binghamton, New York, USA

Shivani Bhatti
sbhatti1@binghamton.edu
SUNY Binghamton
Binghamton, New York, USA

Akshat Shah
ashah85@binghamton.edu
SUNY Binghamton
Binghamton, New York, USA

Avani Phase
aphase1@binghamton.edu
SUNY Binghamton
Binghamton, New York, USA

Riddhi Jaju
rjaju1@binghamton.edu
SUNY Binghamton
Binghamton, New York, USA

Abstract

This project revolves around an in-depth examination of music-related data housed in a PostgreSQL database, specifically emphasizing the intricate relationship between song features and song popularity. The dataset encompasses diverse information, including artist details, sentiment analysis, Reddit mentions, and various other attributes of songs.

Keywords

Reddit, Spotify, Artists, Music Industry, Songs, Popularity, Albums, top tracks, hate speech recognition, Sentiment Analysis

1 INTRODUCTION

In the realm of music exploration and data analytics, this project delves into the rich tapestry of musical content stored within a PostgreSQL database. Focused on unraveling the complex interplay between song features and popularity, our investigation spans a diverse array of musical attributes. From artist details to sentiment analysis, Reddit mentions, and a myriad of other dimensions, our dataset encapsulates a comprehensive snapshot of the multifaceted nature of contemporary and historical music.

This exploration seeks to decipher the underlying patterns and correlations that contribute to the success and appeal of songs. By leveraging the power of PostgreSQL, a robust relational database management system, we aim to uncover hidden insights and trends within the vast landscape of music-related data. Our project not only scrutinizes the intrinsic characteristics of songs but also delves into the extrinsic factors such as online community discussions and sentiment analysis, offering a holistic perspective on the factors influencing the popularity and reception of music in the digital age.

As we navigate through the intricacies of our dataset, we anticipate shedding light on the nuanced relationships between different musical elements and their impact on audience engagement. Through this endeavor, we aspire to contribute valuable insights to the fields of music analytics, data science, and the broader understanding of the dynamic intersection between artistry and audience preferences in the ever-evolving landscape of the music industry.

2 Datasets

2.1. First Data Source- Spotify

Implementation Details

- The script is scheduled to run every 10 hours during which it starts by fetching artist IDs from tracks within featured playlists. It obtains a list of featured playlists and extracts artist IDs from the tracks in these playlists.
- The get_artists_genres function fetches additional data about artists. It makes API requests to the artists endpoint with a list of artist IDs and collects information such as artist names, music genres, and popularity. This data is inserted into the Spotify _artists table in the database.
- The get_artists_albums function retrieves data about albums by artists. It makes API requests to the artists/id/albums endpoint, gathering details such as album IDs, names, release dates, and available markets. This data is inserted into the artist_albums table.
- The get_artists_top_tracks function collects data on the top tracks of artists. For each artist ID, it makes API requests to the artists/id/ top-tracks endpoint, gathering information such as track IDs, names, release dates, available markets, and track popularity. This data is inserted into the artist_top_tracks table.
- To manage a large number of artist IDs efficiently and prevent rate limiting, the code divides the artist IDs into batches of 50 and processes them separately.

Data Collected

Spotify Artists:

- `artist_id`, `artist_name`, `music_genre`, `artist_popularity`.

Artist Albums:

- `album_id`, `album_name`, `release_date`, `available_markets`, etc.
- Toxicity: `class`, `confidence`.

Artist Top Tracks:

- `track_id`, `track_name`, `release_date`, `available_markets`, etc.
- Toxicity: `class`, `confidence`.

Track Audio Features:

- `track_id`, `acousticness`, `danceability`, `energy`, etc.

2.2. Second Data Source- Reddit

Implementation Details

- The Reddit API has a limitation that allows extracting data from a maximum of 100 posts per API call. Upon thorough analysis, we observed that each subreddit typically had 30-40 recently updated posts. To address this, we decided to retrieve approximately 50 posts using the API's post endpoint.

- During each cycle, the scheduler selects one of the 11 subreddits mentioned above and passes it to the script to scrape its posts and comments.
- The function scrapes _posts does the work of extracting hot posts for each subreddit. From the fetched data, the post_id is checked against the already inserted data from the Reddit _posts table to avoid data duplication. The new data is then inserted into the reddit _posts table
- The function scrape _comments is called from function scrape _posts which uses comments endpoint API to fetch comments under each post. From the fetched comment, the comment_id is checked against the already inserted data from the Reddit _comments table to avoid data duplication. The new data is then inserted into the Reddit _comments table
- The function iterate _replies is called from the function scrape _comments which would iterate through the comments JSON to extract every nested comment.

Data Collected

Reddit Posts:

- `post_id`, `title`, `score`, `author`, `date`, `url`, etc.
- Toxicity: `title_class`, `title_confidence`.

Reddit Comments:

- `comment_id`, `post_id`, `score`, `author`, `date`, `body`, etc.
- Toxicity: `body_class`, `body_confidence`.

3 Top 10 Artists for any given year on Spotify and their mentions on Reddit

3.1. Problem Statement

The aim is to investigate and analyze the top 10 artists based on the composite score, considering both the average popularity of their tracks and their artist popularity on Spotify. Additionally, we explore how people talk about these artists online by looking at their mentions in Reddit posts and comments within music-related communities.

3.2. Analysis and plots

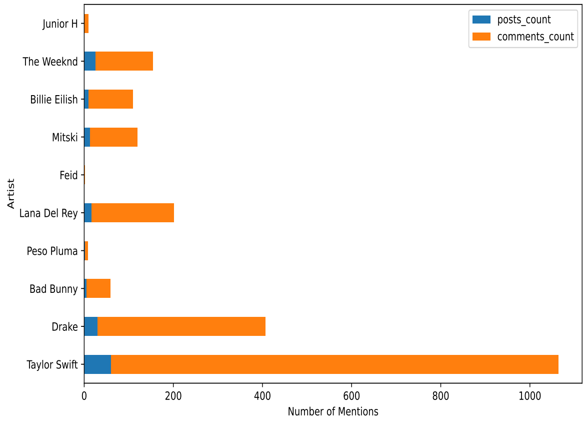


Figure 1: Top 10 Artists on Spotify for the year 2023 and their mentions on Reddit

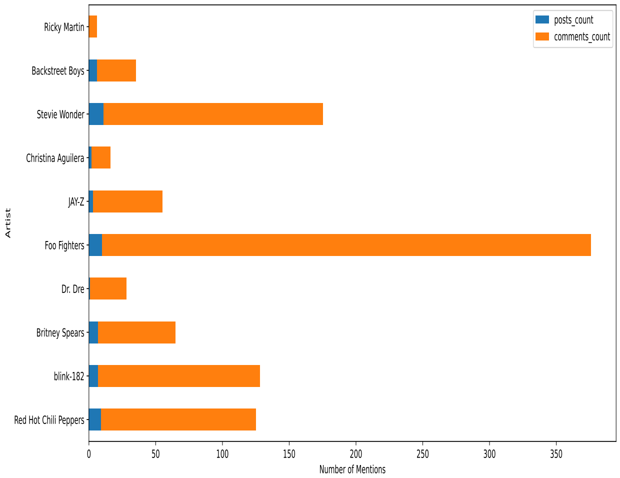


Figure 2: Top 10 Artists on Spotify for the year 1999 and their mentions on Reddit

Figure 1 and Figure 2 gives the top 10 artists and their Reddit data for the year 2023 and 1999 respectively. While there's a general correlation between the popularity and engagement of an artist on Reddit, it's not a strict rule. Some artists with less popularity on Spotify might have higher engagement, suggesting that engagement is influenced by factors beyond popularity. The choice to include specific Reddit subreddits for analysis (e.g.'LetsTalkMusic', 'MusicRecommendations', 'Spotify', 'music') might influence engagement patterns, as different subreddits attract distinct user demographics.

3.3. Workflow/Diagram

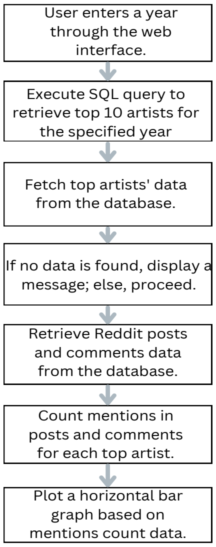

User enters a year through the web interface.
Execute SQL query to retrieve top 10 artists for the specified year
Fetch top artists' data from the database.
If no data is found, display a message; else, proceed.
Retrieve Reddit posts and comments data from the database.
Count mentions in posts and comments for each top artist.
Plot a horizontal bar graph based on mentions count data.

Figure 3: Workflow to find top 10 Artists on Spotify and their mentions on Reddit

Figure 3 provides the workflow for analyzing and visualizing data related to the top 10 artists for a specified year. The process begins by querying a PostgreSQL database to retrieve information about artists from Spotify, including their tracks and artist popularity. A composite score is calculated based on the average popularity of tracks and individual artist popularity. If no artist has released a track in that year, then a message will be displayed saying there is no data available. Subsequently, the code retrieves data from Reddit to determine the number of mentions for each artist within relevant music-related subreddits. Finally, a horizontal bar graph is generated using Matplotlib like Figure 1 and Figure 2

4 Sentiment Analysis on Reddit Data

4.1. Problem Statement

The aim is to justify the popularity of an artist or a track using sentiment analysis on Reddit data.

4.2. Analysis and plots

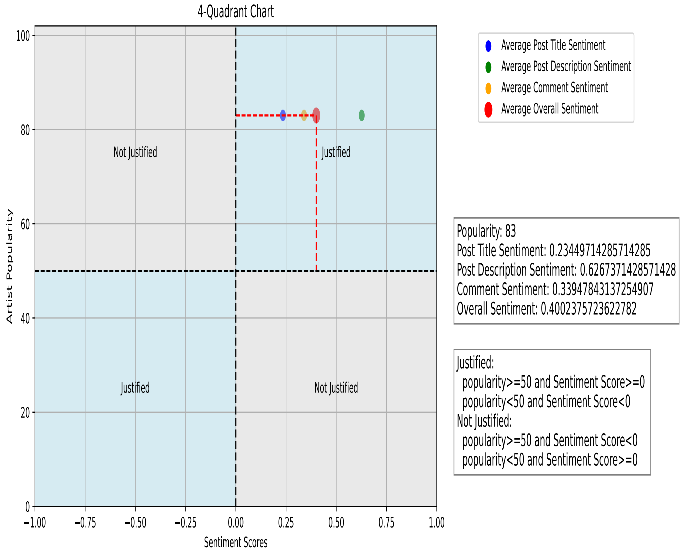


Figure 4: Justification of popularity using reddit sentiments

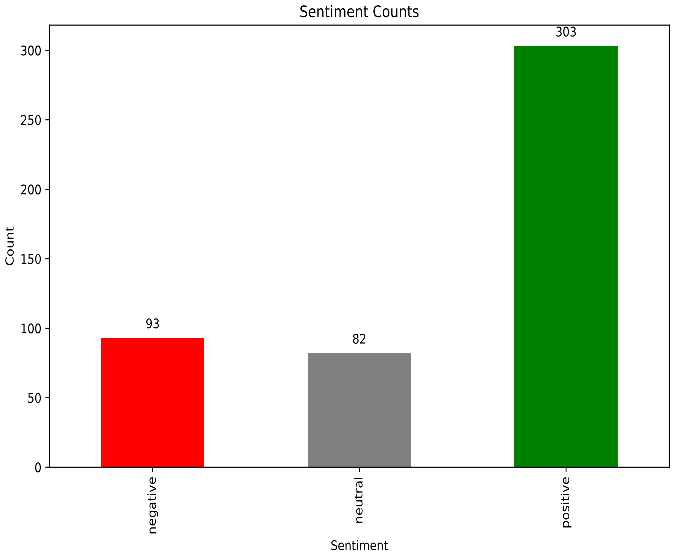


Figure 5: Count of sentiment data taken into consideration for justification

In Figure 4, the correlation between popularity and the average sentiment score of Reddit posts and comments is presented, offering insights into the rationale behind the popularity of a specific artist or track. Meanwhile, Figure 5 illustrates the quantity of sentiment data considered in establishing this rationale. Although, in general, a higher popularity score of an artist or track on Spotify aligns with a greater number of positive sentiments on Reddit, certain instances highlight underrated artists or tracks on Spotify. These cases reveal that despite having a low popularity score, these artists or tracks boast a notably higher record of positive sentiment on Reddit.

4.3. Workflow/Diagram

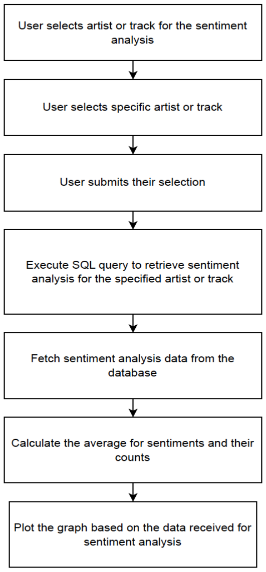

User selects artist or track for the sentiment analysis
User selects specific artist or track
User submits their selection
Execute SQL query to retrieve sentiment analysis for the specified artist or track
Fetch sentiment analysis data from the database
Calculate the average for sentiments and their counts
Plot the graph based on the data received for sentiment analysis

Figure 6: Workflow for Sentiment Analysis of a track or an artist

Figure 6 provides the workflow for analyzing the sentiment of an artist or a track selected by a user. The process begins by querying a PostgreSQL database to retrieve information about an artist or a track from Reddit. An average score is calculated based on the sentiments and their counts. Finally, a graph is generated using Matplotlib for sentiment analysis.

5 Research Question

5.1. Problem Statement

The aim is to derive a relation between track popularity using track features from Spotify data.

5.2. Analysis and plots

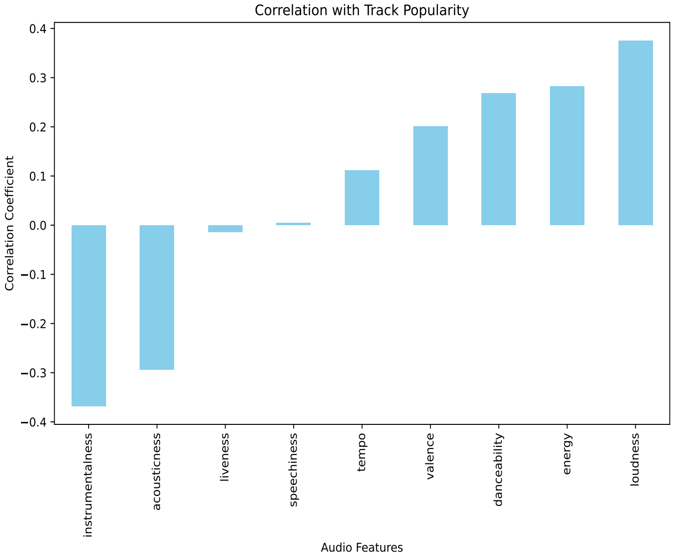


Figure 7: Correlation of track popularity with song features

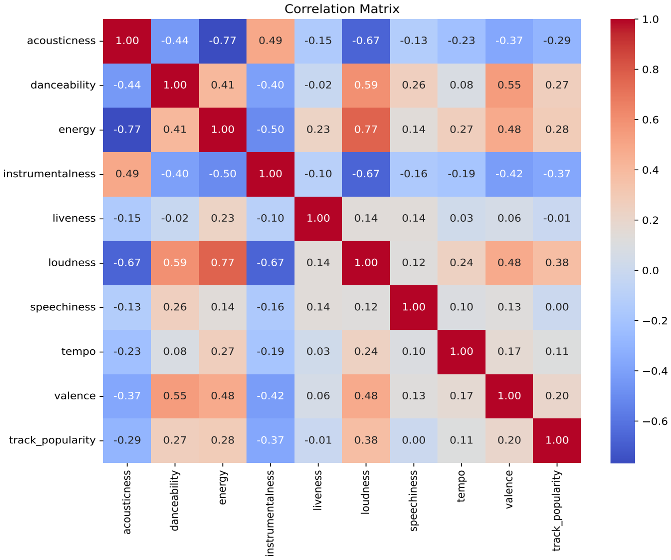


Figure 8: Visualization of the correlation data using heatmap

In Figure 7, the correlation between the popularity of tracks and their features is presented. Figure 8 helps visualize the correlation matrix generated while calculating the correlation. We can see that factors like loudness, energy, and danceability have a positive impact on popularity whereas instrumentals and acoustics have an overall negative impact.

5.3. Workflow/Diagram

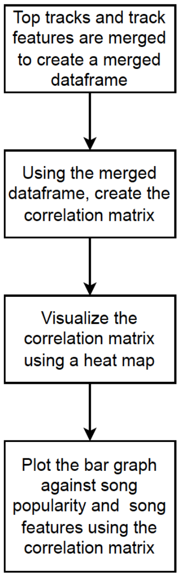

Top tracks and track features are merged to create a merged dataframe
Using the merged dataframe, create the correlation matrix
Visualize the correlation matrix using a heat map
Plot the bar graph against song popularity and song features using the correlation matrix

Figure 9: Workflow for track popularity and track features correlation

Figure 9 provides the workflow for the correlation derived between a track's popularity and track features. Finally, a bar graph is generated using Matplotlib for the correlation.

6 Web dashboard/ Flow chart

6.1. Tools and Technology

(1) **HTML:** Standard markup language for creating the structure of the web page.
(2) **CSS:** Used for styling and layout to enhance the visual presentation.
(3) **Python:** Backend scripting language.
(4) **Flask:** A micro web framework for Python used to create web applications and RESTful APIs.
(5) **Natural Language ToolKit (NLTK):** NLTK is a suite of libraries and programs in Python for symbolic and statistical natural language processing for English written in the Python programming language. It supports classification, tokenization, stemming, tagging, parsing, and semantic reasoning functionalities.
(6) **matplotlib:** A plotting library for the Python programming language and its numerical mathematics extension NumPy.
(7) **pandas:** A powerful data manipulation and analysis library for Python.
(8) **psycopg2-binary:** A PostgreSQL adapter for Python, providing a fast and efficient way to interact with PostgreSQL databases.
(9) **requests:** A simple HTTP library for making requests to web services.
(10) **seaborn:** A statistical data visualization library based on Matplotlib, providing a high-level interface for drawing attractive and informative statistical graphics.

6.2. Workflow/Diagram

We have used python Flask for the web app development and call our function from our python script to dynamically give us the data, which we will then use to generate the plots. We have used matplotlib for plotting, and psycopg2 to dynamically fetch data from PostgresSQL when the function is called. We then use html to show the image files generated by the python code. Our webapp then displays these images on localhost.

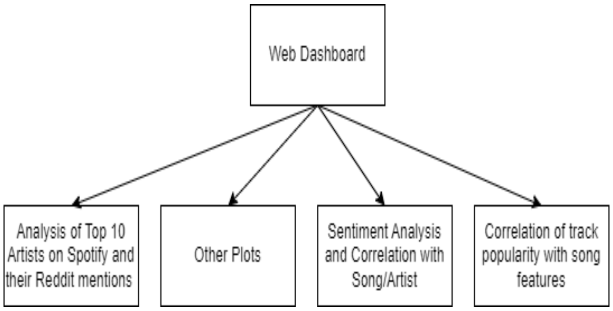
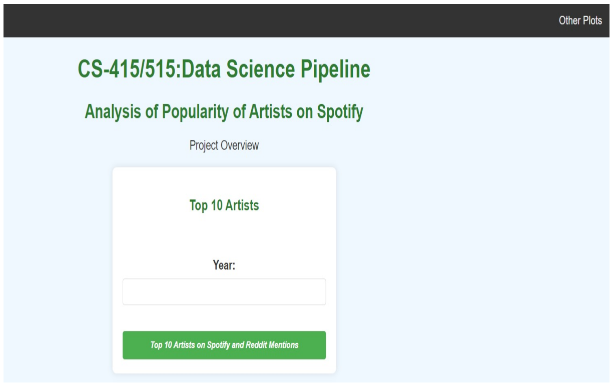

Web Dashboard
Analysis of Top 10 Artists on Spotify and their Reddit mentions
Other Plots
Sentiment Analysis and Correlation with Song/Artist
Correlation of track popularity with song features

Figure 10: Workflow for Web Dashboard

Figure 10 illustrates the workflow of an interface designed to access various analyses. Users can navigate through the interface to perform tasks such as top artists and their reddit mentions,other plots,sentiment analysis and exploration of correlations between song features and popularity.

6.3. Screenshots



CS-415/515:Data Science Pipeline

Analysis of Popularity of Artists on Spotify

Project Overview

Top 10 Artists

Year:

Figure 11: Screenshot of web dashboard for Top 10 Artists on Spotify and their mentions on Reddit

In Figure 11 the interface includes a tab where users can input a specific year. By clicking the corresponding button, the system generates a graph (Figure 1 and Figure 2) plotting the top 10 artists of that year alongside the respective counts of post and comment mentions on Reddit.

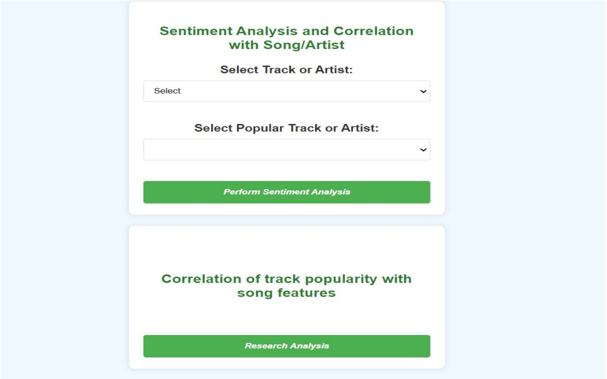

Sentiment Analysis and Correlation with Song/Artist
Select Track or Artist:
Select
Select Popular Track or Artist:
Perform Sentiment Analysis
Correlation of track popularity with song features
Research Analysis

Figure 12: Screenshot of web dashboard for Sentiment Analysis and Correlation of track popularity with song features

In Figure 12 The interface features a dedicated tab for sentiment analysis, prompting users to choose either a specific track or artist. Upon making the initial selection, users can then choose a popular track or artist based on their first choice. Clicking the designated button initiates sentiment analysis, displaying a graph plot (Figure 4 and Figure 5) that assesses whether the popularity is justified.

Furthermore, the correlation between track popularity and song features is explored. This involves examining how various parameters within the song's audio features influence its popularity. The resulting graph (Figure 7 and Figure 8) provides insights into the relationships between these song attributes and overall popularity.

7 Other plots

7.1. Analysis and plots

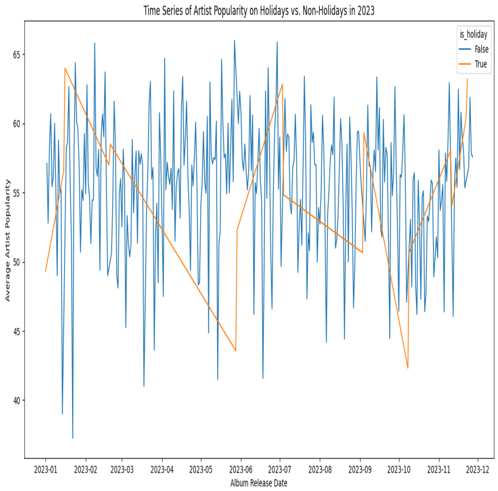


Figure 13: Albums and tracks popularity and holidays correlation Analysis

Figure 13 depicts how the popularity of albums and tracks by artists is affected during different periods, specifically distinguishing between holiday and non-holiday times. It showcases how the success or reception of their music varies depending on these distinct time frames

7.2. ModerateHateSpeech: Albums' and tracks' toxicity on Reddit

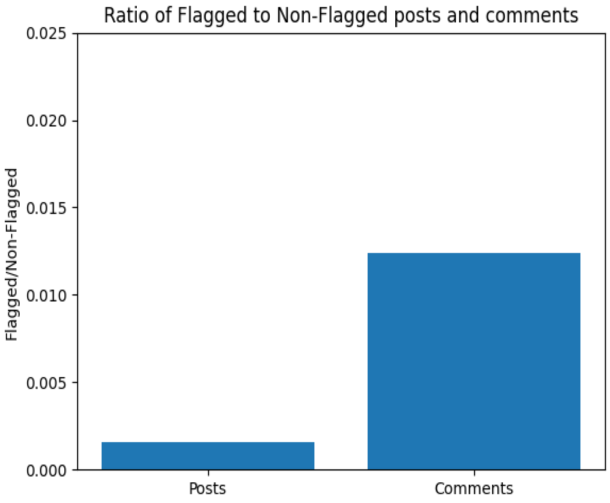


Figure 14: ModerateHateSpeech: Flagged/Non-flagged posts and comments on Reddit

Figure 14 showcases the correlation between posts and comments that have been flagged versus those that haven't on Reddit, utilizing the ModerateHateSpeech API.

7.3. Reddit and spotify data collection

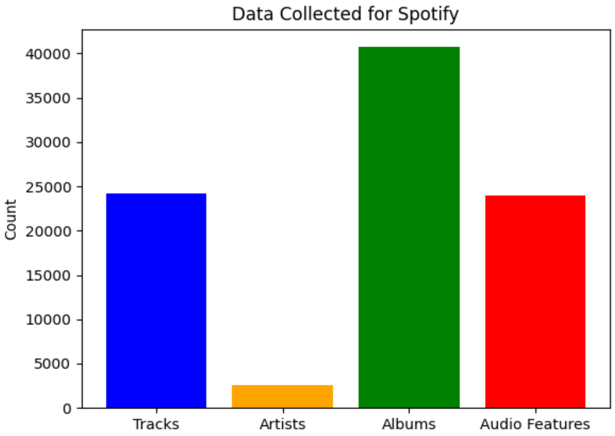


Figure 15: Data collected for Spotify

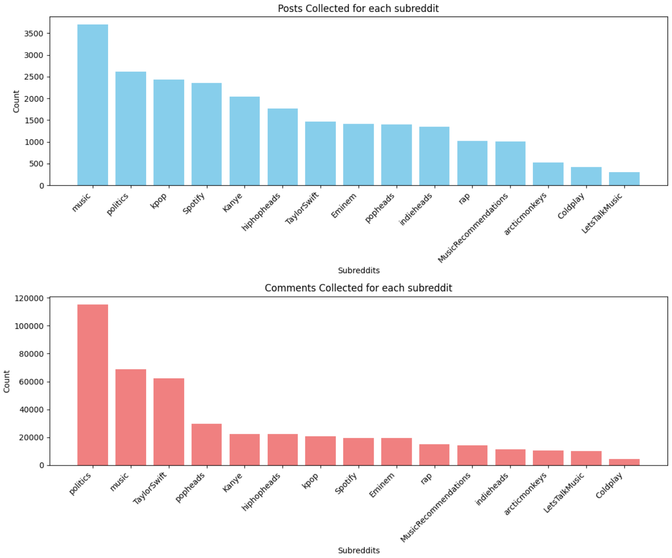


Figure 16: Posts and Comments collected for each subreddit

Figure 15 shows data counts of tracks, artists, albums, and audio features for spotify.Figure 16 shows data counts of posts and comments for each subreddit on reddit.

7.4. Politics submissions

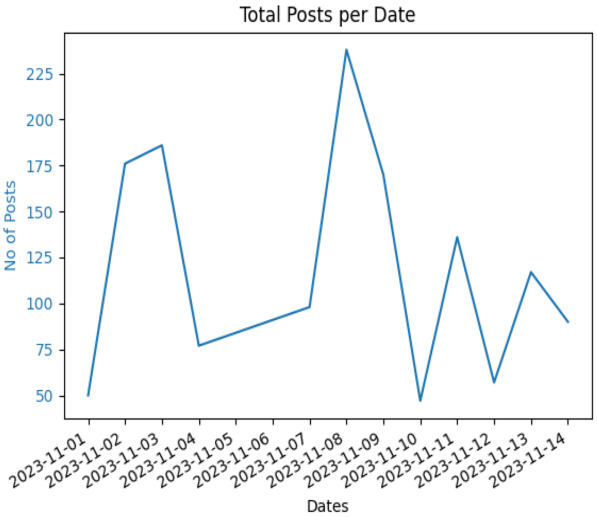


Figure 17: Total posts per day on reddit

Figure 17 illustrates the total submissions per day on the Politics

8 Conclusion

This research delves into Spotify and Reddit data, uncovering how song features, Reddit sentiment, and artist engagement relate to music popularity. Factors like loudness, energy, and danceability impact track success. Reddit sentiment doesn't always align with Spotify popularity, showcased via a user-friendly web dashboard. This study offers insights into the complex dynamics shaping music popularity in the digital era.In future analysis, exploring evolving trends in music consumption patterns could enhance this study.

References

[1] Nugroho, A., Manongga, D., Purnomo, H. D., & Hendry, H. (2023). *Analysis Of Spotify Top Songs During Covid-19 Pandemic. International Journal of Marketing and Digital Creative*, 1(2), 1-14. https://www.researchgate.net/publication/374686086_Analysis_Of_Spotify_Top_Songs_During_Covid-19_Pandemic

[2] Li, J. (2022). *Analysis of The Trend of Spotify. BCP Business & Management*, 34, 919-926. https://www.researchgate.net/publication/366295843_Analysis_of_The_Trend_of_Spotify