Music Words Slap*

Do Words Make Hits?

Cole Gaito†
Applied Comp Sci Student
University of Colorado Boulder
Littleton, Colorado, USA
cole.gaito@colorado.edu

Problem Statement

When it comes to the business of Music, we see a method of communicating with one another. Whether it's the latest cringe Tic-Tok video theme or yet another one of Taylor Swift's break-up songs… music has the ability to pull people in and capture the audience's attention, heart, emotions, and ultimately wallet. Understanding this continuously growing 28.6 billion dollar[1] industry and what captures the attention of this consumer group is the focus of this research project. I am specifically interested in utilizing the Billboard Hot 100 list as the basis for measure. While this list reflects the interests of the American consumer, this group is often the one driving interest and trends worldwide. Most of major music stars arise from the United States, and the typically the US spends the most on music.[2] It also doesn't hurt that a majority of the record labels and technology platforms also reside in or were developed within the borders of the United States.

In this research project in addition to the Billboard Hot 100, would like to do analysis based on Spotify (given the company's pre-eminence within the streaming market and availability to API calls), and lyrical word analysis. Based of the categories available I seek to derive additional data points and see if there is a correlation between the songs as a whole within their genre, album, time and list on the billboard chart.

While ambiguous, I am also interested in seeing if there is a correlation that can be derived to world events and the Billboard's top 100 chart. While I personally highly double the availability of this to measure, I'm interested to see if the songs, by virtue of their words point to influential events in the world. Again due to the highly subjective nature of lyric interpretation, this is unlikely.

As a point of acknowledgement, this project will not survey the rhythmic form of songs, nor its melody. Only the words of the songs themselves and their associated metadata will be analyzed.

Literature Survey

Looking around the research landscape, there are a couple of previous works that I have come across. In *Analysis of Billboard's Top 100 Songs and Lyrics (1964-2015)* by *Elaine Hsu & Hattie Xu* there is a similar analysis of the ranging and the word counts to study the decrease in one-word, two-word, and three-word phrase complexity. There is a generalized perspective that with each passing decade there is an increase in pessimism that grows out of the lyrics. Ultimately some of their hypothesis' held true – more profanity, less complexity, though their idea of a more pessimistic outlook was incorrect. See the reference for additional details.[3]

There is also a Machine Learning Algorithm that was created by GitHub user *yamnihcg.* Here data was used across Spotify and Billboard data pipelines to model if a song would be a future hit.[4]

A Medium author Josh Viner, also used data from Billboard Hot 100 and Spotify's labels identify artists, songs and popularity.[5]

Ultimately though, I could not find anyone who is specifically looking at the text to see if there is a correlation.

Proposed Work

After having hinted at the work proposed, I'll look at outlining what is necessary for understanding and working through the questions stated above. The following numerical values are based on what was theorized in my previous submission.

1. Data Cleaning – Almost None

Having worked with the data so far, this statement remains mostly accurate. The data coming from highly used datasets provides me with little "cleaning". I have found though that there is roughly a ~2%, totaling to 850 items, with an issue between all the unique items. Whether lyrics are not found, names incorrectly identified, or with odd labels for the set there doesn't seem to be a large issue.

2. Data Incorporation – Significant

I knew that this would be the main bulk, capturing the data. As I am pulling the lyrics from a website, I spent ~2 weeks with my machine constantly pinging Genuis' API for Lyrics data. This was also after I managed to get a script written to scrape the data, and then periodically save it into an excel file. The complexity is not necessarily the issue, rather the amount of time to ping, then compile the file was more then I anticipated.

3. Data Preprocessing – Some

Here I recommended having to perform some data manipulate to extract and understand the lyrical text.

Overall, I expect that the interesting part is still to go. I have generally completed the pull of data from the Genuis API scraping the lyrical data from the internet. I want to incorporate additional data point from Spotify and the associated Metadata around the artist, album and song, this is work yet to go.

In addition to the data itself, deriving the most common word in a song or subset and other statical metrics whether by decade, year, month etc. is also outstanding work. I'm not overly concerned about this part though. There are a variety of tools that have and will simplify the process.

Data Set

The original sources for the data can be located at Kaggle[5] and within the self-maintaining GitHub[7]. The dataset can be found in my public facing GitHub account in an excel file called gaitocole/ Final_Project/ Music_Trends/ *charts.xlsx*[8]. This format is accessible to Pandas, MatplotLib and other coding tools and provides an easy-to-read format that is universally available to both coders and non-coders alike.

Currently there is 3.4 million pieces of data, will additional data to be imported. There are roughly 31.5 thousand unique songs across the dataset starting from August 4th, 1958. As previously mentioned, importing the additional metadata and extracting the metrics is the work to go.

Evaluation Methods

The evaluation methods will be the derivation of the words based in the lyrics, the correlation to the Billboard Hot 100 chart. I assume that this will be along the range of having similarity to the idea of a correlation co-efficient. I would also like to see the demonstration based off a word cloud. I often find myself to be a visual person, so having the visualization is a huge point for me personally.

Tools

The current list of tools are as follows:

GitHub – Data Configuration Management

Git-LFS – Open-Source Large Dataset Git Extension

MS Excel – Basic Data Storage File Handler

Python – programming language v3.13.0

VScode – Integrated Development Environment

Genuis API – Website with API call interface for Lyrics gathering

Spotipy API – Spotify website call interface

Various Python Libraries

Milestones

I assume that this project will continue to be something that evolves through the rest of the semester. I would like to see this project completed in

the next month (by Thanksgiving ’24) as I am going on a scheduled holiday.

Milestone Update 10-28-2024

While I do not intend to submit my paper in this final style format, I am planning on relisting the topics that I touched upon above while incorporating the new details of the data below. I find this will allow the various completed attributes to be discussed in their complexity.

The main milestone that has been completed in the previous two weeks has been the incorporation of Spotify’s metadata. The data has been collected from Spotify’s API called from their python library Spotipy. Utilizing this public facing library, along with creating a developer account via Spotify’s developer website[9] you can create a Spotify sever request from which you can access their main metadata.

Concerning the data extraction I have sought the following categories for interest: *Track Popularity*; *Track Explicit*; *Album*; *Album Release*; *Artist Release*; *Artist Popularity*; *Artist Genre*; *Track ID*; and *Tracks in Album*. These categories are tethered to the Title and Artist that have been extracted from the Billboard Hot One Hundred data that has been collected. Due to the nature of pulling and process the ~31,000 unique songs this process has consumed much of the previous two weeks. In addition, I ran into the first issue of a web-hosted API having a limit. Luckily the developer site is rather flexible and allows the creation of multiple endpoints from which you can request the data from the Spotify Database Servers. In this way I’ve reversed the data and split it between two lists to process a bit more quickly.

At this point in time there are roughly 5.5 million pieces of data when aggregating in the 6 categories created through the Spotify API request. It still wildly blows my mind the sheer volume of data that is generated simply by adding in a few more details surrounding metadata.

Data cleaning for the process was a bit more extensive than I anticipated. The Spotify Library surprisingly lacked the total history of all songs dating back to 1958 when the Billboard Hot 100s were initially started. I find this rather surprising due to the extensive library that both Spotify and Apple maintain. There is definitely an exploratory rabbit hole that could be delved further into on that side of the house.

Overall, I did also notice that the Spotify Data Category of Artist Popularity is wildly subjective. We can see the overall limitations of the dataset being correlated with a time dependent activity. I did not necessarily have an expectation surrounding this topic but I did find it rather interesting that the popularity is dependent on the time of API call request and as such the ability to have general popularity is something that can wain over the weeks and years.

So overall by the time of completing this submission, I would articulate that I have captured the data and am ready to begin thoroughly investigating the data.

Projected Milestones

I was given feedback that the dataset should be utilized for something additional beyond performing basic analysis. While I think that the complexity of the songs presents a simplistic idea there is certainly more complex analysis that can and will be performed as well including but not limited to Machine Learning and Prediction.

An interesting topic would be two different avenues of machine learning – determining common hit words as a prediction method for further future prediction and the use of generalized chart metrics to determine how long a particular song may last on the Billboard Hot 100 based on word choice. In both of these propositions, I am seeking more of a concrete understanding and use of the songs. I have yet to see a method for converting the rhythms of the songs into a computer readable and interpretable format, although I could see the chords/melodies charted on sheet music and then determining the probability of success. There is something though that cannot be captured that exists in each person’s vocal skills. I think to the example of having Adele or Cher sing a Sting song. Music tends to be focused and popularized by the individuals who sing it. Another

means might be trying to slice and dice the data across the weekly timeframe.

Method 1: Determining Hit Words

Extracting the relevant data from the now captured lyrics, I plan on pulling the most common words bucketed within each of the songs while excluding the articles of speech (i.e. the, a, an, etc.). This will occur on a per-song basis. In addition, I would like to analyze to see if there is a general theme within either the longevity on the charts or particular brackets. Think of hit songs numbers one through ten, again eleven through twenty and so on. I find that this topic can also be interesting to understand considering the category and genre that the songs fall into. Do particular themes within particular genres do better than others?

In such a way the main words can be seen as a method for understanding how themes chart based on genre and thus how long that theme may last. A difficulty may be dealing with synonyms rather than exactly the same word. I don't think that other than the artist would typically use the same word over and over again. Hence bucketing the various hits into 10 brackets may turn out to be the most reasonable way of accomplishing this task.

Method 2: Slice and dice of Weekly Data

In this approach being able to align the weeks and see the linear progression rather than a single flat file could allow the user or analyst to understand the progression over time. In this way genres and other metadata categories can be utilized to track the predicted popularity. The incorporation and work done for Method 1 could also be useful as a means of folding in additional datapoints to make the algorithm more accurate.

From a personal perspective this will certainly be more challenging as creating the data cubes discussed in class will become necessary. However, challenging this could certainly make the end result more fascinating and provide a greater experience trying to digest and discern what is occurring within the data itself.

Potential Pitfalls

With the methods listed above there are several significant assumptions that may prove to be costly in the pursuit of trying to capture and predict the longevity of a song on a chart and ability to predict future hits. I have taken the time to break out the issues into the categories below:

Insufficient Word Diversity

This issue may arise from a lack of repeated words in both the songs in-of-themselves and between the songs as they are placed into their respective casts. It would be interesting to see if there exists an application which can identify words and synonyms but I find this unlikely. Even if such an application or API exists into a database trying to find all the synonyms that exist for each word in the song would be feasibly impractical.

Arrangement of Data

I see there arising the inability for pandas, the current data manager to handle the complexity of the dataset that I'm trying to present to it. While this issue certainly is not as severe, due to the general time constraints I can foresee the becoming an unrulily question that I simply do not have the skillset to manage. As a simple solution, I want to explore working with TensorFlow or some other data management tools. I am sure that this problem is not novel and as such finding an appropriate tool may be the most difficult part of this task.

Lack of Correlation

While this is not necessarily a problem, it would be unfortunate to find that the data lacks a correlation between any of the categories. What could be worse is the problem of having strange items identified as the driving cause of a song's popularity, like the spelling of an Artist's name or something akin to that issue. Again, while it is not necessarily an outright issues, it could become a headache if there is nothing interesting found.

Complexity between Repeated Items

Here another issue that may end up simply requiring the analysis to be performed twice is the situation of having two similar but different datasets. The dataset which lists the weeks on the charts and thus has the single long list of every time a song appeared on the Billboard Hot 100 and the Unique Song List Data Set that I have created. The later was an ends to speeding up the API requests, but may end up being useful both in the need to have a smaller data pool, and the need to expedite learning. From my previous courses learning how to effectively manage the resources to train AI is necessary. While not an explicit issue, I wonder if analysis on both sets would yield the same results. Theoretically it should if I have structured my analysis correctly, but if not I could merely be opening the door to future questions about the work done here.

New Tools:

Jupyter Notebook – Data Science Code Workbook

Tensor Flow – Initial Machine Learning Library

Pandas – Data Framework for Data Manipulation

MatplotLib – Graphing Tool

Results so Far

With the data ingested into the Jupyter Notebook and running simple metrics on it I found the following metrics of particular interest:

When analyzing the number of weeks that a song remains on the board, the following details are extracted: Mean: 9.320, Standard Deviation: 7.920, Min: 1, Max: 91 with 4, 7, 13 being the quartiles at 25, 50, 75 percents respectively. Overall, I am extremely interested in the way that data is skewed throughout this process. While not totally surprising, I wonder how the entrance of particular means of increasing popularity and the delivery platform have changed the way that particular songs peak. Taylor Swift was accused of re-releasing albums to suppress competitor musicians by Billie Elish, perhaps there is truth to her statement.

Additional Deliverables and Work To Go

I foresee delivering along with the code, presentation, etc. a Jupyter Notebook. Within this space I would like the ability to develop and provide an interactive graph and chart file. As I have navigated through the various courses, I personally have utilized the ability to interact with graphs and data as validation that I understand and have correctly processed the code which I developed. In this case I find that providing the user another means to interact with the data will allow them to engage further with their own coding and explore additional resources that may not be listed or were explored.

For this particular item I plan on placing the Jupyter Notebook here in my public facing GitHub account: called gaitocole/ Final_Project/ Music_Trends/ *DataEvaluation.xlsx*[10].

In addition to the various charts mentioned and maintained by the Jupyter Notebook it may be beneficial to host the AI model outside of the workbook. I have not extensively worked with Tensor Flow or other AI learning algorithms and libraries, but that will be part of the fun in the forward work.

Overall I am still quite pleased with what has occurred thus far and believe that the outlined work will keep me both sufficiently busy, but also be manageable moving forward. I look forward to continuing to explore the questions of understanding and predicting future Billboard Hot 100 songs by means of Lyric Word Analysis and overall exploration of the metadata collected by Spotify.

REFERENCES

[1] **Statista.** 2023. Global revenue of the music industry from 2002 to 2023. *Statista.* Available: https://www.statista.com/statistics/272305/global-revenue-of-the-music-industry/#:~:text=In%202023%2C%20the%20total%20revenue,compared%20to%20the%20previous%20year.

[2] **Wikipedia.** 2023. List of largest recorded music markets. *Wikipedia, The Free Encyclopedia*. Available: https://en.wikipedia.org/wiki/List_of_largest_recorded_music_markets.

[3] **Brown University Department of Computer Science.** 2011. CS100 Student Projects - Project 11. *Brown University*. Available: https://cs.brown.edu/courses/cs100/students/project11/.

[4] **Yamnihcg.** 2023.Billboard100. *GitHub Repository*. Available: https://github.com/yamnihcg/Billboard100.

[5] **J. D. Viner.** 2023. What makes a hit song? Analyzing data from the Billboard Hot 100 chart. *Medium*. Available: https://joshdviner.medium.com/what-makes-a-hit-song-analyzing-data-from-the-billboard-hot-100-chart-74c1d5ad3fa3.

[6] **D. Dave.** 2023. Billboard - The Hot 100 Songs. *Kaggle Datasets*. Available: https://www.kaggle.com/datasets/dhruvildave/billboard-the-hot-100-songs.

[7] **UT Data.** 2023. rwd-billboard-data. *GitHub Repository*. Available: https://github.com/utdata/rwd-billboard-data.

[8] **C. Gaito.** 2023. Music Trends - Final Project. *GitHub Repository*. Available: https://github.com/gaitocole/Final_Project/tree/main/Music_Trends.

[9] **Spotify.** 2024. *Spotify for Developers*. Retrieved October 27, 2024 Available: https://developer.spotify.com/

[10] **C. Gaito.** 2023. Music Trends - Final Project. *GitHub Repository*.

Available: https://github.com/gaitocole/Final_Project/tree/main/Music_Trends