Milestone 1
Songs Information and Lyrics

Afonso Caiado, Diogo Martins, José Maçães

up201806789@up.pt, up201806280@up.pt, up201806622@up.pt

FEUP — MEIC — Processamento e Recuperação de Informação

November 18, 2021

Abstract

Nowadays, with the increasing use of streaming services, listening to music is just a click away. With the huge amount of variety that is available, what we choose to hear can say a lot about us, our own personality and our feelings.

For this reason, taking the information available on the work of the most popular artists of the last decade, we consider it interesting to gather and organize it.

Thus, in addition to the presentation of the lyrics, genre and album to which a song by one of these singers or bands belongs to, it will be possible to explore the same data in order to find specific words in them.

Keywords: Music, Songs, Lyrics, Data Processing, Pipelines.

1 Introduction

With this report, our goal is to provide information as detailed as possible about the project that we are developing that consists of an information system that presents data about songs such as its name, artist, genre, lyrics and more. In the first Milestone, we will focus on details about the dataset we prepared, like its its characterization and extraction, the correspondent data processing pipeline and data visualization.

2 The Datasets and Data Cleaning

The data we are using comes in a .csv format, which is easy to manipulate with Pandas and Python. We found a lot of datasets in Google Dataset Search, but only the following presented the type of data that we wanted.

2.1 Song Lyrics

1. *Origin:* The Song Lyrics dataset is a dataset by Deep Shah, being the source of it Genius, a digital media company and website that allows users to provide annotations and interpretation to song lyrics. This dataset was fetched from Kaggle.

2. *Description:* This dataset is composed of 21 csv files, each one about a different one of the top artists from the last decade. These files range from 100 entries to about 1000 entries, having 6 columns each.

3. *Treatment:* We decided to keep all 21 of the csv files and combine them, as the data in each file is organized the exact same way, the major difference being that the songs in a file are all from the same popular artist. We removed one column from the files when we combined them, the one related to the year of the release of the song, since it was redundant because we already have that information in the column "date" in the format "YYYY-MM-DD" that refers to the year, month and day. We consider that this dataset was a good choice since it gathers all the songs by the most popular artists in the last decade, so the use of the information system we are developing would be more relevant to the general population that listens to music.

2.2 Spotify Past Decades Songs Attributes and Spotify - All Time Top 2000s Mega Dataset

1. *Origin:* The biggest audio streaming and media services provider right now, Spotify, has an open API, with the type of information that we desire available. Thus, The Spotify Past Decades Songs Attributes dataset is a dataset by Nicolas Carbone, and Spotify - All Time Top 2000s Mega Dataset is a dataset by Sumat Singh, both finding its source in Playlist Machinery. Playlist Machinery is a website that allows users to organize songs on Spotify by many parameters, and, in the case of both datasets, the parameter used to gather the songs was popularity, so, in an indirect way, its source is Spotify. Both were also fetched from Kaggle.

2. *Description:* The first dataset was composed of 7

files with 15 columns each. Each file corresponding to a decade of music. The second dataset is composed of only 1 file with 15 columns about the 2000 most popular songs from the 2000s and 2010s decade. These files contain data about the most popular songs of the respective data, including genre, that was missing in the first one.

3. *Treatment:* First of all, we decided to keep only 2 files out of the 7 files from the first dataset of this section: those from the last 2 decades. This decision was made because our main dataset (from the previous subsection) is composed by the most popular artitists from the last decade, so there was no use in having 5 more files containing songs from artists that weren't going to match with those on our main dataset since they only released songs in the 2000s and 2010s. We also decided to reduce all the files from the 2 datasets in this subsection to the 3 columns we needed: Title, Artist and Genre. These datasets were fetched purely with the intent of knowing the Genre of the songs in the Song Lyrics dataset.

3 Data Mining

After the selection and refining of the datasets, we ended up with two major datasets. They were obtained with the Pandas tool and Python Scripts.

3.1 Combined Lyrics

This is our main dataset and it contains 5 columns after refinement:

- *Artist:* this is the name of the singer or band to whom the authorship of this song belongs to
- *Title:* this is the title of the song
- *Album:* this is the name of the album to which the song belongs to
- *Date:* this is the date when the song was released
- *Lyric:* these are the lyrics of the song

3.2 Song Genres

This dataset informs us on the genre of songs and has the following information:

- *Artist:* this is the name of the singer or band to whom the authorship of this song belongs to
- *Title:* this is the title of the song
- *Genre:* this is the genre of the song

4 Pipeline

To process the described data, the following pipeline was used:

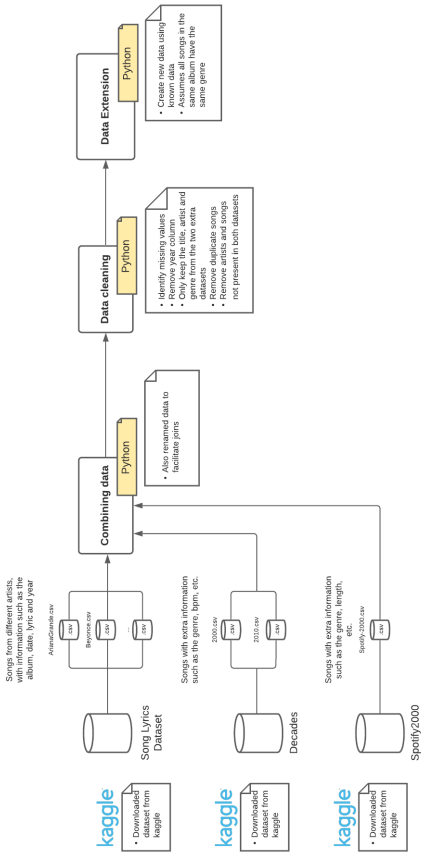

Figure 1: Data Processing Pipeline

As we can see, we start by downloading 3 datasets, both from kaggle: the Song Lyrics Datatset, Spotify Past Decades Songs Attributes (named Decades in our diagram) and the Spotify - All Time Top2000s Mega Dataset (named Spotify2000).

With Python scripts, we combine the data merging the files and renaming data to facilitate joins. Our next step corresponds to Data Cleaning, where the data was refined by identifying missing values, removing the "Year" column that came from the first dataset due to reasons previously explained. We only kept the title, artist and genre from the other two datasets, so columns like bpm, loudness and energy were removed since they don't seem to have a purpose for now in our project. Duplicate songs and the songs that didn't had a thing in common comparing datasets (regarding artist, name of the song or album) were also removed.

Finally, we proceeded to Data Extension, also using Python scripts, by creating new data using known data, assuming all songs in the same album have the same genre.

5 Domain Conceptual Model

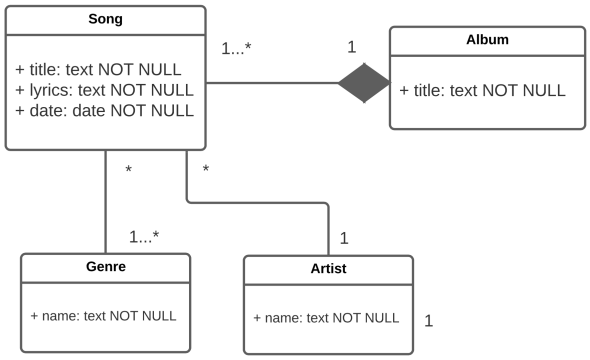

Figure 2: Domain Conceptual Diagram

The domain conceptual diagram consists of very few classes since we are only focusing on some attributes of a song for now. Our main class is exactly the *Song* one and it has the following attributes:

- "title" is the name of a song. Each song has a title but it is not necessarily unique.
- "lyrics" corresponds to the lyrics of a song.
- "date" corresponds to the date the song was released and it is not unique.

The other classes help to complement our main class with additional information that can be needed to the data retrieval part of our project. This other classes are: *Artist*, which has only one attribute:

- "name" corresponds to the name of the artist.

Our dataset Song Lyrics is organized by artist and each song has a single artist associated with it, the "main artist" in case the song is a collaboration, so we represented it like that in our UML.

Album , which has only one attribute:

- "title" corresponds to the name of the album.

Since a song corresponds to a single album (the one it was originally released on) we represented that the song could only be present in a single album and not more than that one. Finally, we have the class *Genre*, which has also only one attribute:

- "name" corresponds to the name of the genre.

We assumed that a song could have more than one genre (for example, *Pop, Dance Pop* or *Hip Hop, RB.*

6 Data Characterization

In the next figures, we can see the most common words in lyrics of songs by Ed Sheeran, Beyoncé and Justin Bieber respectively (these specific graphs were provided by Deep Shah, Kaggle). We hope that, in the future, this same information has value in a way that we can explore the data. As we can see, for example, in Figure 5, the most common words in Ed Sheeran songs' lyrics are "time", "like", "go", "love" and "never". The letters that are bigger are the ones that are more common and the size of the others decreases as they are less common

Figure 3: Most common words in Justin Bieber songs' lyrics.

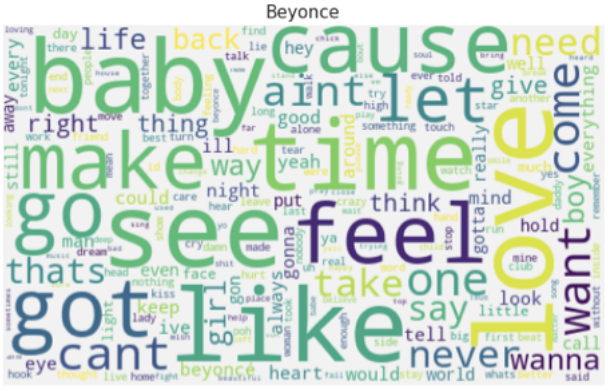

Figure 4: Most common words in Beyoncé songs' lyrics.

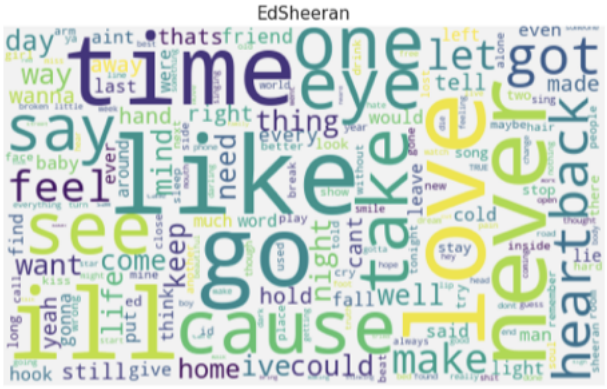

Figure 5: Most common words in Ed Sheeran songs' lyrics.

In the next figure, we can compare the amount of songs available in the dataset per artist in a graph we elaborated. As we can see, Eminem, Drake and Taylor Swift are the ones with more songs available and Khalid, Cardi B and Charlie Puth are the ones with less ones, which makes sense since they started their carreer way earlier than the others.

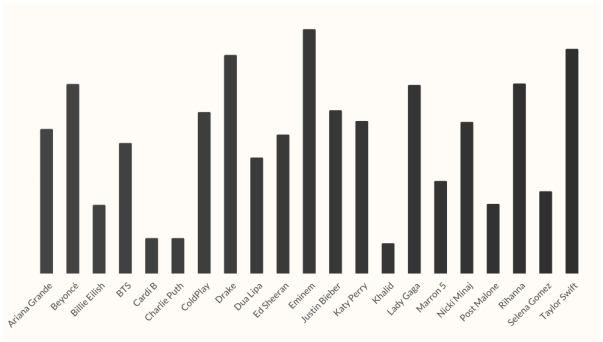

Figure 6: Graphic with the amount of songs per artist in the dataset.

7 Possible Queries

In order to extract information from the data, we thought of some queries that would be interesting to present in the future:

- Search songs by artist
- Search songs by date
- Search songs by genre
- Search songs by album
- Search songs by name
- Search songs by words and quotes
- Search songs by the feelings and emotions expressed in it

Searching songs by artist, date, genre, album, and of course, name, are definitely crucial, but searching them by words or quotes that are present in the lyrics is something that we hope to develop since it is one of the biggest focus of our project and it's from there that we hope we can explore the data.

To provide you some concrete examples on what the above mentioned queries would look like, the User would input something like: *Songs about love* or even *Dance Pop Songs*. Our algorithm would then search for songs with the word love in it, or search for songs with a Dance Pop genre.

8 Conclusion

In this first phase of the project, we found and chose the data that felt relevant to our purpose. We combined it, refined it and came with an organized dataset that contains the information that we desire and that we intend to work with. The Data Model and the Pipeline were also crucial points to this project that were developed and that helped us during the elaboration of the work until now since they allowed a better organization regarding tasks and work that needed to be done. In this report, we covered the first milestone and mentioned some information related to what we pretend to do in the next one, so, as a future work, information retrieval and the search of songs are intended to be developed by us.

9 References

[1] - "Genius (website)" Wikipedia, Wikipedia Foundation, ⟨https://pt.wikipedia.org/wiki/Genius_(website)⟩

[2] - "Spotify" Wikipedia, Wikipedia Foundation, ⟨https://en.wikipedia.org/wiki/Spotify⟩

[3] - "Song Lyrics Dataset" Deep Shah, Kaggle, ⟨https://www.kaggle.com/deepshah16/song-lyrics-dataset⟩

[4] - "Web API", Spotify for Developers ⟨https://developer.spotify.com/documentation/web-api/⟩

[5] - "Spotify Past Decades Songs Attributes", Nicolas Carbone, Kaggle ⟨https://www.kaggle.com/cnic92/spotify-past-decades-songs-50s10s⟩

[6] - "Spotify - All Time Top 2000s Mega Dataset", Sumat Singh, Kaggle ⟨https://www.kaggle.com/iamsumat/spotify-top-2000s-mega-dataset⟩

[7] - "Spotify reveals the decade's most-streamed songs, from Ariana Grande to Drake" Mark Savage, BBC, ⟨https://www.bbc.com/news/entertainment-arts-50642141⟩

[8] - "Psychology of music preference", Wikipedia, Wikipedia Foundation, ⟨https://en.wikipedia.org/wiki/Psychology_of_music_preference⟩

[9] - "Playlist Machinery", Playlist Machinery, Playlist Machinery, ⟨http://www.playlistmachinery.com⟩

[10] - "Applying Data Mining for Sentiment Analysis in Music", Lucía Martín Gómez, María Navarro Cáceres (2017)