**MeloMatch**

Project Proposal

**DSCI-D590 Intro To NLP for Data Science**

Professor Olga Scrivner

October 23, 2023

**Submitted by:**

Anay Gangal

Rohan Bankala

Trevor Speich

**Objectives**:

We are going to create a music recommendation engine wherein we take a song title as an input, and we use sentiment analysis to output similar songs. The problem we will solve is we are going to provide an efficient way to discover new music based on the user’s music taste. The user does not need to have any additional knowledge than the song title in order to be provided with similar music.

**Usefulness**:

With currently over 600 million music streaming subscribers in the world today, a common problem for most users is identifying new music that fits their taste (Duarte). Faced with countless choices, it can be overwhelming to find new songs. Our service allows users to simply provide a single song title that they enjoy and be presented with a curated list of similar songs that align with the user’s music preferences.

Most popular streaming services such as Spotify, or Apple Music, provide a recommendation engine where users can discover similar music. However, these algorithms require the user to spend a lot of time listening to different songs before they can provide capable suggestions whereas our service only requires the user to provide a singular song title which does not take up a lot of effort or time commitment.

Our service is going to be free whereas all the other streaming platforms usually have a monthly subscription plan. Lastly, our service is web-based which implies that the user does not really have to create any account to get recommendations and their information is not stored or used anywhere.

Our target group is people who prefer simplicity, security, and efficiency when it comes to discovering new music.

**Data:**

To develop a music recommendation engine based on the title, we will be using the Kaggle dataset "Song lyrics from 79 musical genres." This dataset was developed by Anderson Neisse through web scraping the Brazilian lyric website Vagalume (vagalume.com.br). Web scraping is a technique for extracting data from web pages. In this case, Neissie used R to parse the HTML code of Vagalume pages and extract specific nodes. Some preprocessing was required, such as removing line breaks, but no text content was preprocessed overall. The data was gathered and updated about two years ago, on October 20, 2021. Neissie originally gathered the data to create and train an LSTM (Long Short-Term Memory) network to compose song lyrics. LSTMs are used in a variety of everyday machine learning systems, from translation engines to recommendation/prediction engines. They require a lot of data to train, so the Kaggle dataset contains two CSV data files: one for artists and one for lyrics, with 4,239 and 379,893 rows, respectively.

The artist data file contains the following columns: Artists, Genres, Songs (quantity), Popularity (number of times accessed on the website), and URL link. We will not use this data file as our main dataset, but it will be helpful for trimming down our recommendations based on genre and popularity. The lyrics data file contains the following columns: Artist, Song Name, Lyrics, and Lyrics Language. Due to the nature of our system, we will restrict our recommendations to English-only songs, resulting in a dataset of just over 100,000 songs. There are no NA values in the dataset except for in the Lyrics Language column, which is used to indicate instrumental songs (e.g., classical). We will handle this by filtering on the Lyrics Language column.

The only data cleaning required is for the actual text component of the Lyrics column. Beyond spaces and punctuation, we will need to remove elements between square brackets ([ ]), as these are used to highlight key points of the song (e.g., intro, verses, choruses) or to indicate who is singing which part of the song if it is a duet. There are also soft spoken or background lyrics that exist between parentheses that need to be kept but do represent a different element of the song. Though unknown at this point we do believe that we will also have to add our own stop words due to the nature of the music industry. Artists tend to use non-tradition English words to express ideas, these could clutter our NLP logic (e.g. ya or uah uah). We expect preprocessing to be a very iterative process given the size of the dataset.

**Functionalities:**

**NLP Functions:**

**Lyric Extraction:**

Obtain the lyrics of the song title provided by the user. This may involve integration with a song lyrics API or database.

**Sentiment Analysis:**

Process the extracted lyrics through a sentiment analysis model to gauge the overall emotion or sentiment (e.g., happy, sad, angry, etc.).

**Latent Dirichlet Allocation (LDA):**

(LDA), a topic modeling technique, can greatly enhance the music recommendation engine focused on song lyrics. LDA can unveil underlying themes in lyrics, such as love, heartbreak, or rebellion. When paired with sentiment analysis, this offers a nuanced lens: while both songs might share a 'sad' sentiment, one could revolve around lost love and another around life challenges.

**Song Matching:**

Match the analyzed sentiment with other songs in the database using LDA and those that have a similar sentiment profile.

**Lyric Similarity:**

Beyond sentiment and LDA, another layer can be added that matches lyrics based on topics and word usage. This can be enhanced through parts of speech (POS) tagging allowing the ability to analyze similar structural and grammatical elements between song lyrics. POS can allow for a more refined song recommendation based on semantic similarities.

**Song Recommendation Output:**

Return a list of song titles (and possibly artists) that match the sentiment of the user's input.

**User Interaction Functions:**

**Search Input:**

Provide a user-friendly text box where users can type or paste the song title of their choice.

**Search Button:**

A simple button adjacent to the text box to initiate the recommendation process once the user has entered their song title.

**Result Display:**

Display the list of recommended songs in an organized, scrollable, and visually appealing manner. The list could have clickable song titles that lead to external links (e.g., music platforms) or provide more details.

**Communication and sharing:**

Our preferred communication method is discord. We will be using discord extensively to meet and discuss regarding the final project. We are also utilizing a Github repository to store our files and track any and all changes that are made. The link to the public Github repository is provided below:

<https://github.com/aagangal/NLP_Group_7.git>

The repository also includes a README.md file that includes our project description as well as a weblink to the previously mentioned Kaggle dataset we are using for this final project. Lastly, in addition to discord and Github, we have also exchanged phone numbers and have a group text chat for when responses are required more urgently.

**Citations:**

Duarte, F. (2023, October 10). Music Streaming Services Stats (2023). *Exploding Topics*. [h](https://explodingtopics.com/blog/music-streaming-stats) ttps://explodingtopics.com/blog/music-streaming-stats

**Personal Contribution Statements:**

|  |  |  |
| --- | --- | --- |
| Team Member | Team Project Contribution | Individual Project Contribution |
| Anay | Assisted with the objectives and usefulness sections. Created the team discord server and Github repository and wrote the communication and sharing sections. Helped with deciding on a name for our project. Assisted with document formatting. Designated team lead for submissions. | I hope to utilize the techniques we have learned to preprocess our dataset and get it ready for NLP. I also hope to assist with setting up the web interface as well as the actual NLP implementation. I also hope to continue to effectively use all of the communication and sharing tools and submit all of the assignments in a timely manner. |
| Rohan | Assisted with objectives, usefulness sections. Facilitated with figuring out all the required functionalities for the project. Helped with deciding on a name for our project. | I will assist with the front-end aspect of the project. This will include setting up the website and making sure the UI is simple and efficient to use. Along with that I will also be assisting with the NLP aspects of the project and help with the data cleaning and NLP implementation. |
| Trevor | Assisted with the creation of the objectives and usefulness sections. Facilitated the selection of the dataset and wrote the dataset section. Collaborated with Rohan to develop NLP functionalities. Assisted with formatting of paper. | Through my experience in Applied Machine learning I look to assist the preprocessing of the data and its iterative approach, along with developing and implementing our NLP functionalities. Preprocessing of the data will start immediately and will be the primary focus of my time. Once that is complete within the next week I will be able to shift my focus to the NLP portion. |