

MUSIC CLASSIFICATION USING LYRICS

Team Members:

Gopika Manvitha Kariyavula – KXG210060

Roshan Rayudu – RXR210122

Suhaas Srinivas Kalisetty – SSK210023

1. Introduction

Our project's goal is to use song lyrics to categorize music into genres. Humans find it difficult to accomplish this task, and since borders are not always obvious, there is frequent discussion regarding where song fits in. Music genres show similarities between tracks, which helps to group music into collections. Songs often fit into more than one genre, indicating that genre isn't always clearly defined. Automating this classification process is highly motivated by technologies such as Spotify, which adds an estimated 60,000 songs to its database every day.

We have merged the data, dropped all non-English songs, and removed duplicate songs. The output genres have been one-hot-encoded. Due to class imbalance, we experimented with oversampling and SMOTE techniques. Our GloVe embeddings were created using the Stanford NLP GitHub repository code, utilizing the corpus generated from our datasets.

2. Data

We worked with two primary datasets:

Dataset 1: Music Dataset: 1950 to 2019

(<https://www.kaggle.com/datasets/saurabhshahane/music-dataset-1950-to-2019?rvi=1>)

Dataset 2: Song lyrics from 79 musical genres

(<https://www.kaggle.com/datasets/neisse/scrapped-lyrics-from-6-genres?rvi=1>)

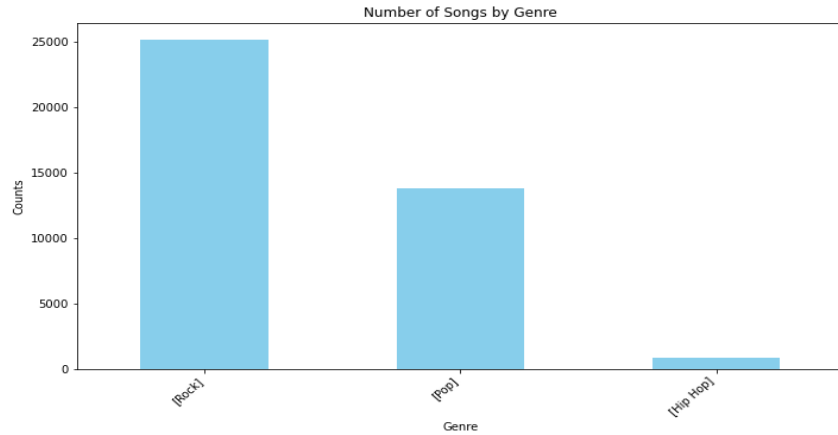


Figure 1.

Bar graph representing the number of songs for each genre.



Figure 2.

Wordcloud (Most frequently used words for Hip Hop) for Hip Hop.

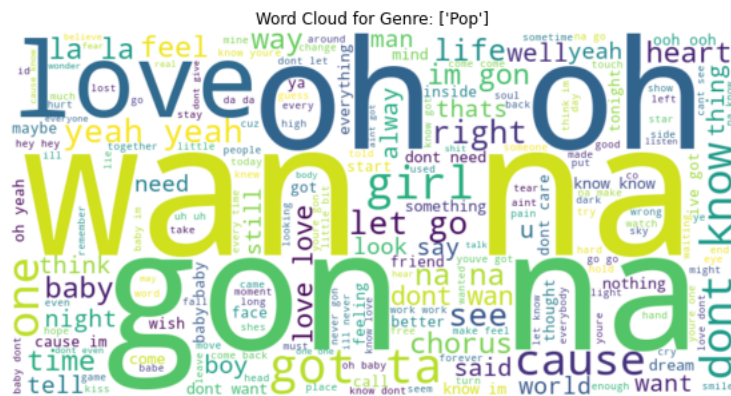


Figure 3.

Wordcloud (Most frequently used words for Hip Hop) for Pop.

5. **Outcome:** The preprocessing yielded a clean and structured dataset suitable for subsequent analyses

3.2. Outputs

MultiLabelBinarizer Encoding for Genre Labels

- a. Encoding Genre Labels: MultiLabelBinarizer is instantiated to encode the 'target' column (genrelabels) into binary form, creating separate columns for each genre label.
- b. Creating Encoded DataFrame: The encoded targets are transformed into a DataFrame with binary columns representing each unique genre label.
- c. The resulting DataFrame contains binary-encoded genre labels, enabling genre-specific analysis and machine learning tasks. The number of unique genre classes and the total count of English songs in the dataset are printed for reference.

3.3. Classification

- 1. Baseline Models - The baseline models were initially trained and evaluated on the imbalanced dataset to assess their performance in the natural data distribution. - At the outset of the analysis, Logistic Regression, Random Forest, and XGBoost were selected as baseline classification models for music genre prediction based on lyrics data. The dataset exhibited a notable class imbalance, with 'Pop' being heavily represented, followed by 'Rock' and 'Hip Hop' with fewer instances.

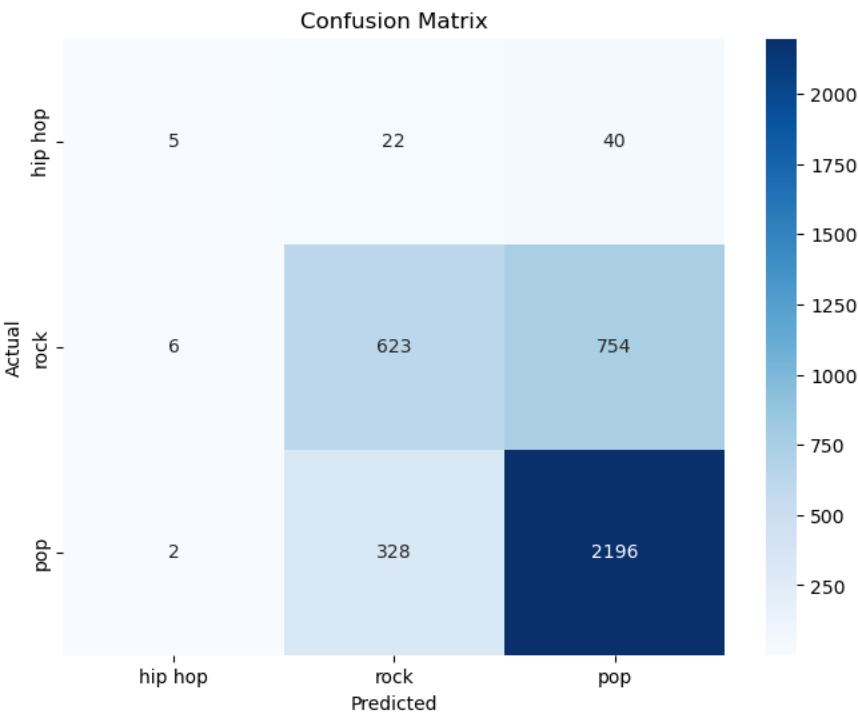


Figure 5.
cm_without_class_balance.

2. Handling Class Imbalance through the implementation of SMOTE oversampling, the project successfully managed class imbalance and improved the robustness of machine learning models for music genre classification based on lyrics data.

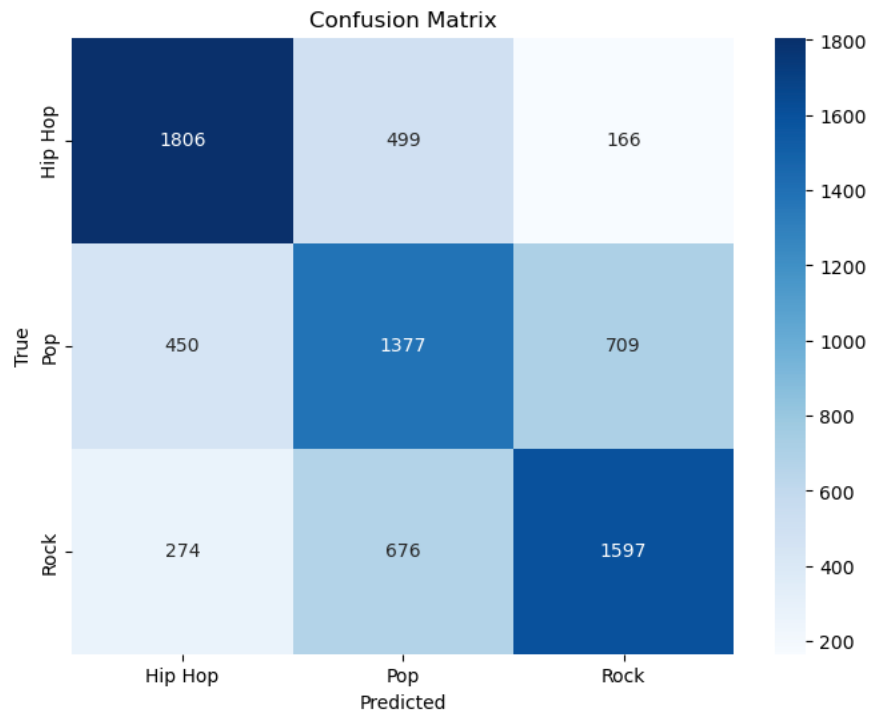


Figure 6.
logistic_after_balance

3. Best Model - Random Forest's combination of ensemble learning principles, robustness, scalability, interpretability, and feature importance analysis makes it the best model for music genre classification tasks based on lyrics data. Its ability to handle complex relationships, generalize to new data, and provide valuable insights into genre characteristics positions Random Forest as a top-performing and reliable choice for this domain.

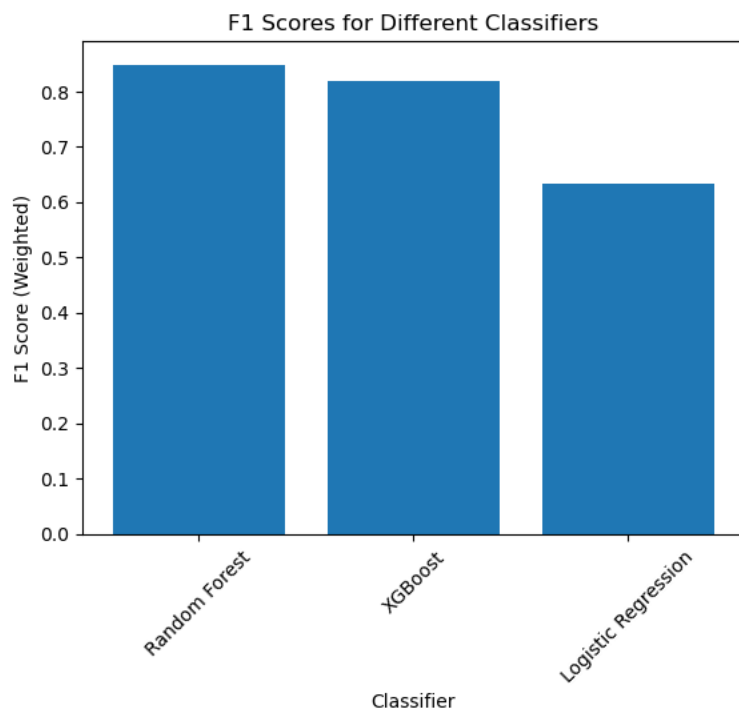


Figure 7.
F1 scores

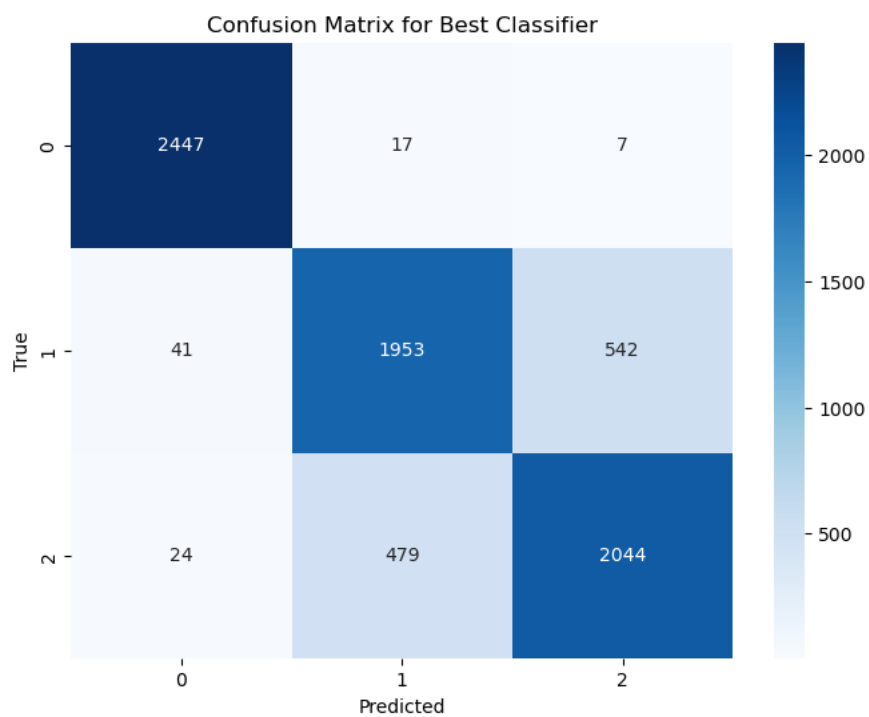
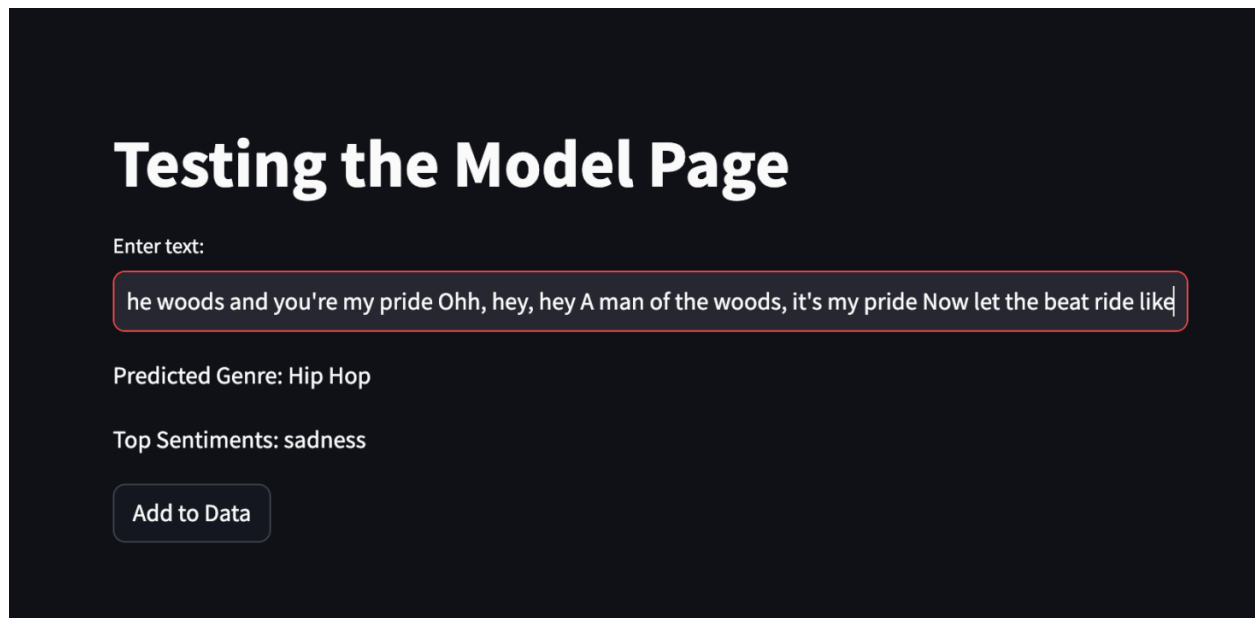


Figure 8.
Best model.

3.4. Sentiment Analysis

We have performed Sentiment Analysis on our lyrics dataset using a Bert based model (j-hartmann/emotion-english-distilroberta-base). We chose this model because Bert models use attention mechanisms to better capture the context and dependencies between words and are better at handling long sequences. This model has been trained on 6 different datasets and has been finetuned to predict scores for 6 different emotions and a neutral class. We use this model to get scores for 7 classes which are anger, disgust, fear, joy, neutral, sadness and surprise. Once we get emotion scores for song, we assign top three emotions to that song using 0.2 as a threshold (we assign at most three different emotions to a song, only assign an emotion to a song from the top 3 emotion scores if score is greater than 0.2).

3.5. Results



Testing the Model Page

Enter text:

he woods and you're my pride Ohh, hey, hey A man of the woods, it's my pride Now let the beat ride like

Predicted Genre: Hip Hop

Top Sentiments: sadness

Add to Data

Figure 9.
HipHop

Testing the Model Page

Enter text:

o to you, hello world (I am a loser) Maybe tomorrow, maybe today Who knows who will win (I am a loser)

Predicted Genre: Pop

Top Sentiments: sadness,joy

Add to Data

Figure 10.
Pop

Testing the Model Page

Enter text:

realize you're not dead Take a look at an open book Do what you like, that's what I said Do what you like

Predicted Genre: Rock

Top Sentiments: neutral,anger

Add to Data

Figure 11.
Rock

References

- Music Genre Classification using Song Lyrics (https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1214/reports/final_reports/report003.pdf)
- Train GloVe Embeddings using Stanford NLP code (<https://stackoverflow.com/questions/48962171/how-to-train-glove-algorithm-on-my-own-corpus>)
- Over 60,000 tracks are now uploaded to Spotify every day. that's nearly one per second (<https://www.musicbusinessworldwide.com/over-60000-tracks-are-now-uploaded-to-spotify-daily-thats-nearly-one-per-second/>)
- Stanford NLP GitHub repository (<https://github.com/stanfordnlp/GloVe/tree/master/eval>)
- Hugging face.co (<https://huggingface.co/j-hartmann/emotion-english-distilroberta-base>)