Project 1 - M2: Classifying Music Genre by Song Lyrics

Data Devils: Michael Hijduk (leader), Laura Espuna, Anthony Jiang

Executive Summary:
This project develops a machine learning model to classify music genres based solely on their song lyrics. Data collected through the LyricsGenius and LastFM APIs undergo TF-IDF encoding with n-grams to capture linguistic patterns. The analysis plan outlines preprocessing, modeling, and evaluation steps with the goal of achieving a logistic regression model with at least an 80% classification accuracy.

Hypothesis: Since song lyrics exhibit genre-specific patterns in word frequency, semantics, and syntax, song lyrics can be used to construct a logistic regression model that can be trained to classify songs by genre with an expected accuracy of around 80%. [1]

Modeling Approach: The plan is to aggregate song genre and lyrics from the LastFM and LyricsGenius APIs, respectively, and represent them using TF-IDF encoding with unigrams, bigrams, and potentially trigrams to capture both word frequency and order [2]. The task is to use logistic regression as the baseline model to perform classification. Logistic regression is robust to correlated features, offers interpretable coefficients that highlight influential words, and provides simple linear decision boundaries; therefore, it makes a strong starting point for analysis.

Dataset Establishment Details:

Data Summary:
The dataset includes 2000 songs across 8 music genres (pop, rock, electronic, country, metal, hip-hop, jazz, and R&B). The dataset contains the top 250 songs from each genre from the year 2025 to ensure that the data is uniform and bias can be minimized in the model training. The following features are collected from each song: title, artist, genre, and lyrics. The dataset is stored as a JSON file on GitHub, which makes it publicly accessible through the repository's file system. Anyone with the link to the file can view it directly in their browser or download it by clicking the "Download raw file" button or using the raw file URL. This allows easy access for analysis in Python (ex. Using json.load), R, or other tools that can read JSON.

Provenance:
The dataset was created by programmatically gathering music data from two public APIs: Last.fm and Genius. Using the Last.fm API, the top 250 tracks were retrieved for each of the eight genres, resulting in a diverse pool of songs. For each track, metadata such as title, artist, and genre were stored in a JSON structure. Then, the Genius API was used to query the corresponding lyrics by song title and artist, with preprocessing steps to remove section headers and handle retries for failed requests. Each track entry was enriched with lyrics when available, or left blank if lyrics could not be retrieved. This process produced a curated dataset that combines

structured metadata from Last.fm with lyrical content from Genius, allowing for exploratory analysis of music across genres.

License:

All original code in this project is released under the MIT license, allowing individuals to freely use, modify, and distribute it with proper attribution. The dataset metadata was collected from Last.fm API may be used and shared under the Last.fm API Terms of Service, which require proper attribution. However, the song lyrics retrieved via the Genius API are copyrighted by the respective music publishers and are not covered by the MIT license. While the repository contains lyrics for analysis purposes, redistribution of these lyrics without permission would violate copyright law and Genius' API terms. Users of this repository may use the lyrics locally for personal analysis but must not publicly redistribute them. In summary, the MIT license applies only to the code and any sharable metadata, while the lyrics remain protected intellectual property outside the scope of this license.

Data Dictionary:

Column	**Description**	**Potential Response**
name	Title of the song	Expresso
artist	Individual who performs/sings the original song	Sabrina Carpenter
genre	Musical genre or style of the song (pop, rock, electronic, country, metal, hip hop, jazz, and R&B)	pop
lyrics	The words/context of the song	"Now he's thinkin' 'bout me every night, oh\nIs it that sweet? I guess so\nSay you can't sleep, baby, I know\nThat's that me espresso\nMove it up, down, left, right, oh\nSwitch it up like Nintendo\nSay you can't sleep, baby, I know\nThat's that me espresso..."

Ethical Statement:

The song lyrics included in this dataset are copyrighted material owned by their respective artists, songwriters, and music labels. They are provided here solely for educational, research, or non-commercial purposes, such as text analysis or natural language processing projects. Users must comply with the Terms of Service of the data sources, such as the Genius API, and should not redistribute lyrics for commercial purposes. Proper attribution to the original artists and sources is expected when using this dataset. Any usage beyond academic or research contexts may require additional permissions from copyright holders.

Question(s) explored about the dataset:

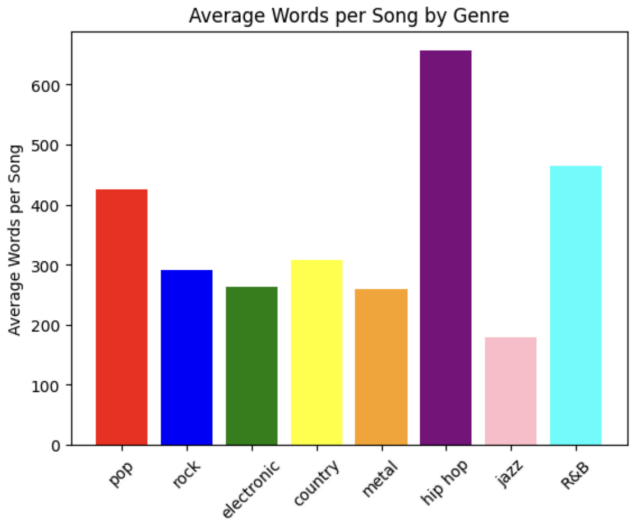
- What is the distribution of our data based on genre?
    - Equal distribution of songs from different genres was achieved
- What is the average number of words per song for each genre?
    - It was observed that hip-hop has the most words per song on average, while jazz has the least words per song on average
- What words are the most popular for each genre?
    - It was observed that across most of the genres, words such as “love” and “i’m” are popular

Current Unknown(s):

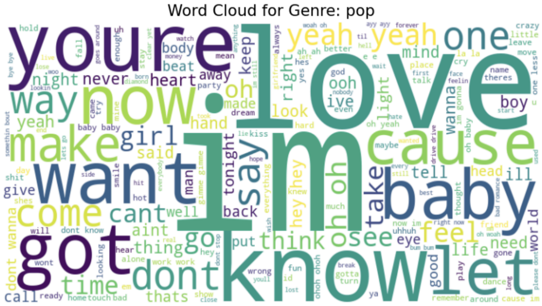
It is currently unclear whether including trigrams (sequences of three words) in our TF-IDF vectorization will improve genre classification. Adding trigrams could capture more context from the lyrics, but it would also increase the dimensionality and complexity of the analysis. The plan is to first evaluate the model using unigrams and bigrams, and if the performance does not meet our expectations, the inclusion of trigrams to potentially enhance accuracy may be reconsidered.

Exploratory Plots:

1. Average number of words per song per genre
2. Word clouds for each genre (visualization of word frequencies)

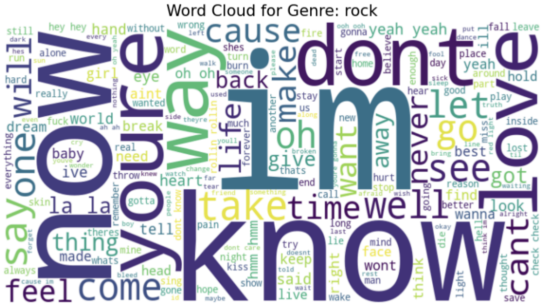


Word Cloud for Genre: pop

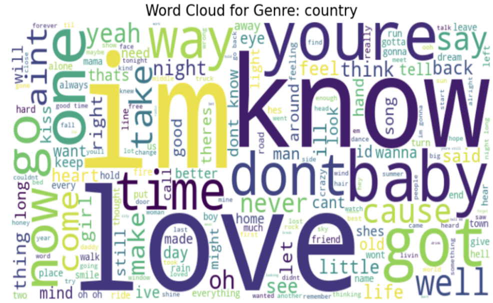


Word Cloud for Genre: electronic

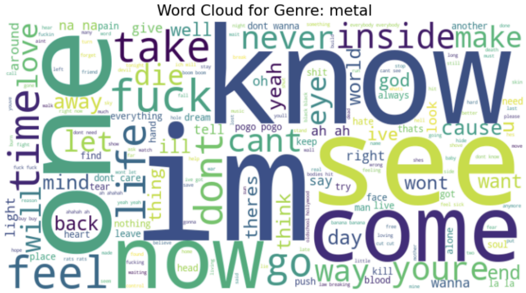
Word Cloud for Genre: rock



Word Cloud for Genre: country

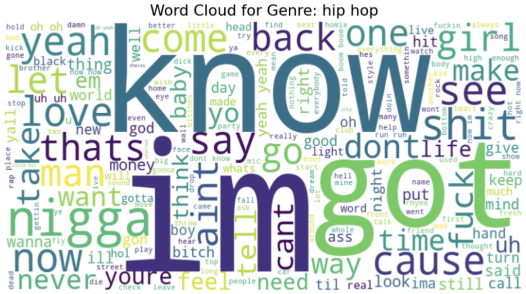


Word Cloud for Genre: metal



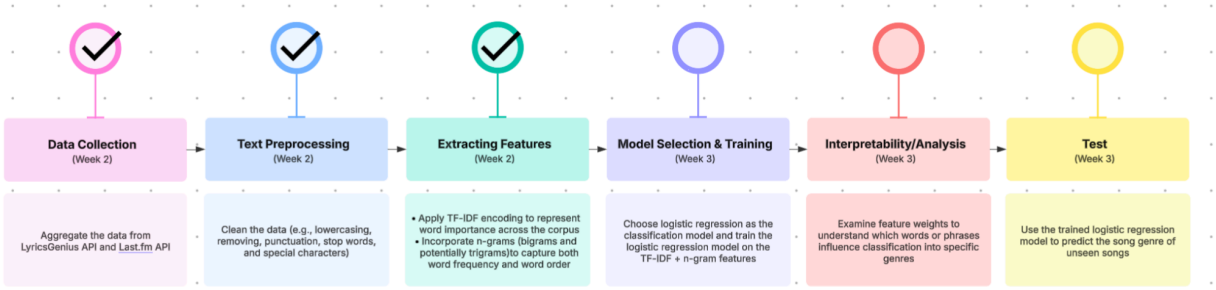
Word Cloud for Genre: jazz

Word Cloud for Genre: hip hop



Word Cloud for Genre: R&B

Analysis Plan:



The first step of the analysis plan is data collection. The song lyrics and metadata from Genius was aggregated using the LyricGenius API and song.fm API. This targets a balanced dataset across eight genres (pop, rock, electronic, country, metal, hip hop, jazz, and R&B). Data is initially retrieved and stored in JSON format, which allows for the preservation of raw fields such as title, artist, lyrics, genre, and duration. For analysis, these JSONs are transformed into a tidy DataFrame with one row per song. The goal is to reach at least 1,000 songs (100 per genre) for MI3, with a stretch goal of 10,000 depending on API throughput.

Once collected, the text is preprocessed and the lyrics are standardized through a cleaning pipeline that lowercases text, removes punctuation, special characters, stop words, and bracketed stage directions. Duplicate rows, like same artist or title or identical cleaned lyrics, are removed, and entries missing lyrics or genre labels are dropped. Explicit genre tokens are stripped to prevent data leakage. The result is a reliable text column that can be consistently vectorized for downstream modeling.

The next step is to extract the features from the preprocessed data. The cleaned lyrics are represented using a TF-IDF vectorizer applied to unigrams and bigrams, capturing both individual word frequencies and common word pairs. The vectorizer applies thresholds to eliminate overly rare or overly common terms and reduce noise. Genre names and other leakage terms are excluded from the vocabulary. The fitted vectorizer is saved so that the same transformation is applied across cross-validation folds and the final test set. This ensures reproducibility and comparability of results.

Once the features are extracted, the correct model will be selected and the training will be completed on it. The baseline model is multinomial logistic regression, chosen for its interpretability and strong performance on text classification. The dataset is split into an 80/20 train/test partition and 5-fold cross-validation is done on the training set to evaluate regularization strengths. To address any residual imbalance, class weights will be balanced if needed. The majority-class accuracy is recorded as a sanity check baseline, ensuring that the logistic regression model demonstrates meaningful predictive power beyond simple guessing.

In order to understand the data, the model will be evaluated for interpretability. After training, the top weighted coefficients for each genre will be examined to identify the words and phrases most influential in classification.

These features are validated against intuition and domain knowledge to ensure they are meaningful and not artifacts of data leakage. The per-genre confusion matrices are also analyzed to highlight which genres are most related and harder to differentiate. This interpretability step links model performance back to linguistic patterns, offering insights into how lyrics shape genre identity.

To understand how effective the model is, the final trained model is evaluated on the held-out 20% test set. The metrics reported will be overall accuracy, macro-F1, balanced accuracy, and per-genre precision/recall/F1, along with a confusion matrix. Success is defined as achieving ≥80% accuracy and ≥0.70 macro-F1 on the test set, with no single class falling below 0.60 in F1. Meeting these criteria represents the project finish line and signals that experimentation is done and the team can move on to preparing the presentation and final report.

References:

[1] R. Mayer, R. Neumayer, and A. Rauber, "Rhyme and style features for musical genre classification by song lyrics," in *Proc. 9th Int. Conf. Music Information Retrieval (ISMIR)*, Philadelphia, USA, 2008, pp. 337–342. [Online]. Available: https://www.researchgate.net/publication/220723682_Rhyme_and_Style_Features_for_Musical_Genre_Classification_by_Song_Lyrics. [Accessed Sept. 11, 2025].

[2] M. Fell, C. Sporleder, “Lyrics-based Analysis and Classification of Music,” in Proc. 25th Int. Conf. Computational Linguistics, Aug. 2014, pp. 620-631. [Online]. Available: https://aclanthology.org/C14-1059.pdf. [Accessed Sept. 11, 2025].