# CSE 158 ASSIGNMENT 2 REPORT

## 1. INTRO

Music plays a big part in providing a relaxed feeling when we are under stress, a different environment for various types of atmospheres, and can simply provide new outlooks on life. There are many genres of music such as Jazz, Pop, etc., and each of them possesses different kinds of lyrics and tones. The goal of this project is to create a classifier that is able to determine a song's genre from its lyrics regardless of the language of the song. We will be taking a dataset of 290,000 samples of labeled lyrics data to create this classifier.

## 2. DATASET

The data set that we chose is the Multilingual Lyrics dataset on Kaggle [1]. This dataset contains a list of songs with their lyrics, genre, and language. The dataset contains 290,183 entries in total. After dropping the entries with empty lyrics (35 found), we are left with 290,148 entries.

### *2.1.* Data Cleaning

We cleaned the dataset by first removing the columns that will not be used, such as 'Artist' and 'Song Title'. We then encoded each genre by assigning them a specific integer from 1 to 10 (because there are 10 in total): Rock = 1, Metal = 2, Pop = 3, Indie = 4, Folk = 5, Electronic = 6, R&B = 7, Jazz = 8, Hip-Hop = 9, Country = 10. After encoding the genres, we then performed the cleaning process on the lyrics. For each song, we first replaced the newline character '\n' with the space character. We then removed all punctuation and turned each word in the text to lowercase.

### *2.2.* Exploratory Data Analysis

To begin our EDA, we first looked at the distribution of songs in each genre. We found that Rock is the most popular genre, having 121,391 total songs under this genre. The second most popular genre was Pop, having 108,693 songs. The rest of the genres had a significantly lower number of songs (all under 21,000) with the lowest of these being Country, having only 1,890 songs. This may be noticeable when we actually try to predict the genre of a certain song, given that most of the dataset falls under either Rock or Pop.

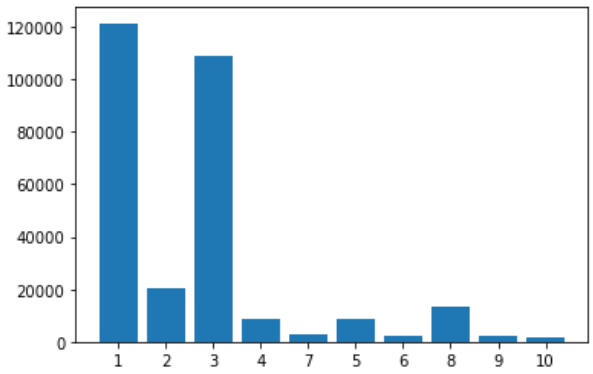


*Figure 1. Rock (1) and Pop (3) are the most popular genre in the dataset*

Next, we looked at the top 10 most popular words across all genres, removing stop-words while doing so. We found that the three most popular words were "I'm", "don't", and "love". The word "I'm" had significantly more occurrences at 575,383, while the rest of the words fell below the 404,000 mark. We then

wanted to explore how these word counts changed across different genres. Across different genres, the two most popular words were “I’m” and “love”, with Rock, Metal, Pop, Indie, Electronic, and Hiphop having “I’m” as the most common word. Genres Folk, R&B, Jazz, and Country had “love” as the most common word. Next, looking at the average amount of words per song for a specific genre, we found that Hiphop actually had the highest average amount of words at around 500 words per song. The next highest average word genre was Pop having around 300 words per song. The rest of the genres had an average number of words lower than 250, with the distribution of these counts being relatively the same across these genres.

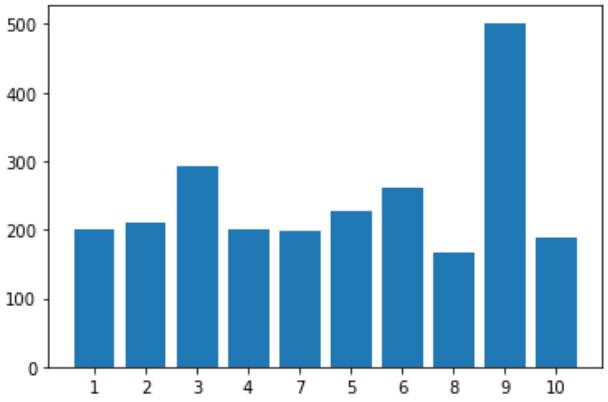


*Figure 2. Hiphop (9) has the highest average word count*

This fact was surprising to us; we originally thought a genre like R&B or Country would have the highest average number of words per song. Finally, we looked at common words that all genres share. This facet was not too surprising, considering the shared words are common in everyday language. These words were “don’t”, “I’m”, “know”, and “like”.

## 3. PREDICTIVE TASK

Using this dataset, our task is to accurately and precisely classify the genre of a song given its lyrics. This classifier can be useful when building a song recommender system. For this multi-class classification problem, we will be using logistic regression and another model that incorporates language to predict the genre. We will evaluate these models using different metrics such as accuracy and precision. We will then compare the metrics of those models to our baseline model, which will predict the genre of a song based on whether or not it contains what we think is the most popular word in that genre.

In order to perform this predictive task, we will be using the words from the lyrics as the features, and the genres will be the labels. During the cleaning and exploratory process, we extracted the lyrics from the dataset and split them into individual words. This way, we are able to keep track of individual word count, word popularity, and potentially transform them into feature vectors.

## 4. MODEL

### *4.1.* Baseline

Our baseline model will predict the genre of a song based on whether or not it contains what we think is the most popular word in that genre. We will conduct research to see which word is popular in a certain genre, and then we analyze the song to see if its most popular word matches. If it does not contain any occurrences of that word, then the model will predict using the most popular genre in general.


*Figure 3. Diagram showing the most popular words from each genre of music*

We chose this model as our baseline because it performs the basic prediction task by using the 'average' (or the most popular) genre. This model is, however, weak due to the fact that it only predicts the most popular. It does not take in account any other features such as word popularity, language, etc.

### *4.2.* Language-based model

This model predicts the genre of a song based on the most popular genre in that language. In other words, instead of looking at the lyric of a song, this model looks at the language in which the lyric was written in and then predicts based on the most popular genre in that language.

We chose this model because we thought that a certain region (with a specific language) can have a certain type of song that is more popular than the rest, which can influence the distribution of genres.

### *4.3.* Logistic regression

This model predicts the genre of a song based on its lyric. The input will be in the form of a matrix of feature vectors where each vector represents the lyric of a particular song. This model computes the prediction by first taking the softmax of the linear transformation of the input and a weight matrix. Then, it outputs the class with the highest probability (since softmax will return a probability distribution of the classes).

We chose this model because logistic regression is one of the working models for multi-class classification tasks. In addition, with this model, we are able to take the lyrics into consideration when predicting the genre.

We used the 80/20 split and fitted our training set using a logistic regression model. We used 2 different ways to form the input vectors before feeding them into the model, and then we compared the results at the end. Below are the 2 methods.

#### 4.3.1. Most frequently used words

In this method, we used the cleaned lyrics (no stop words or punctuation) and ranked them from the most to least frequently used. We then chose the top k (k = 2000) words that we want to use as a way to form the feature vectors. We then fitted Sklearn's Logistic Regression model using the formed feature vectors with C = 0.001 and max_iter = 800. We optimized the C value by training the data on different C values (0.001, 0.01, 0.1, 1, 10) and picked the best one.

#### 4.3.2. TF-IDF

In this method, the process of forming the feature vectors is similar to the above. However, instead of using the frequency of words, we calculated the tf-idf score for each word. In

addition, our feature vectors were based only on the first k=1000 words here due to limitations in computing power. We then fitted Sklearn's Logistic Regression model using the formed feature vectors with C = 10 and max_iter = 800. We optimized the C value by training the data on different C values (0.001, 0.01, 0.1, 1, 10) and picked the best one.

## 5. LITERATURE

There has been substantial research on the connection between the lyrics and genre of songs. The multilingual lyrics for genre dataset [1] that we used was collected by faculty at the University of Bucharest with data pulled from the Sparktech 2018 Textract Hackathon and three other Kaggle datasets. We were unable to find the work that those at the University of Bucharest completed, but the main parts of the data were extracted from a Hackathon event in 2018, where contestants competed to predict song genre based on lyrics. The work that we have done is similar to what those in the hackathon competed to complete, but the dataset we used was also supplemented by 3 other Kaggle datasets, which gave us a much larger pool of songs.

Another interesting work of research was done by students at Stanford University. This report is titled *Music Genre Classification using Song Lyrics* [2]. In this report, the researchers used a LTSM model to classify songs using just the lyrics. As a result, they found their model to be 68% accurate, which is slightly better than our best models. One major difference between their work and ours is that they only studied 3 different genres, while we looked at 10; this may be the reason that their accuracy was slightly better than ours. Overall, their methods and results were comparable to the research that we conducted, indicating that our work was carried out adequately.

## 6. RESULT

### 6.1. Baseline Result

We tested this baseline model by splitting up our dataset into train and test sets using an 80/20 split. The result of this baseline model was as we had expected. The accuracy for this model in predicting the genre was quite low sitting at around 0.33. It also had a low average precision at around 0.145. We believe that our other models will perform a lot better than our baseline model at predicting the genre of music.

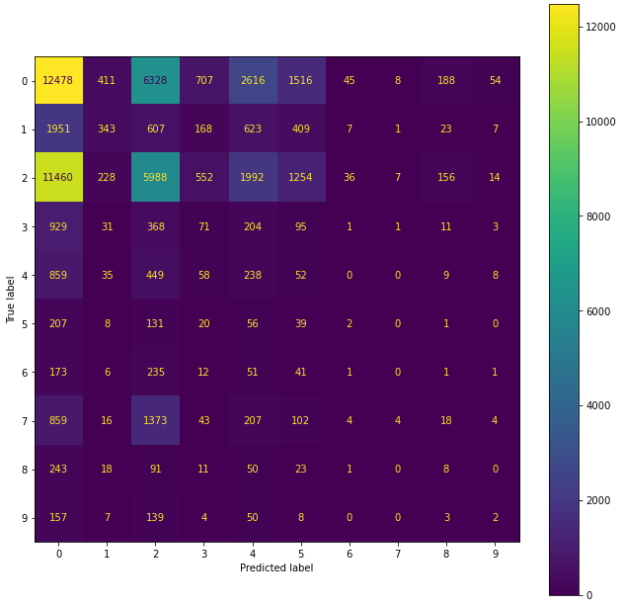


*Figure 4. Confusion matrix for the baseline model*

### *6.2.* Language-based result

For our next model, we decided to predict a given song's genre by looking at the most popular genre of that song's language of origin. We tested this model using the same train/test splits and found the most popular genre for each of the possible languages (33 total languages) from the training dataset. The accuracy for this language-based genre predictor was quite a bit

higher than the baseline, with an accuracy of 0.43. Also, the precision was almost 4x higher than the baseline, with an average precision of 0.574. As we expected, this model performed much better than the baseline, which was a relatively naive approach. However, it should be noted that much of this accuracy can likely be attributed to the fact that across almost all languages, Rock remained the most popular genre, having far more samples than other genres.

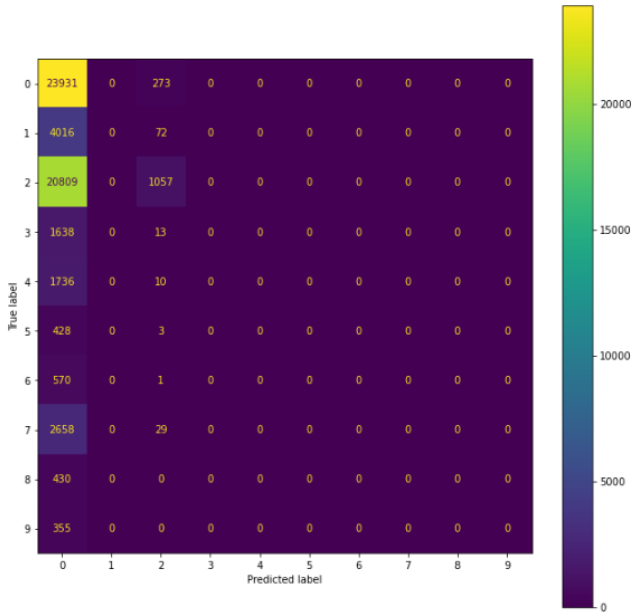


*Figure 5. Confusion matrix for the language & popularity-based model*

### *6.3.* Logistic regression result

After fitting our model using 2 different methods of forming the inputs, we learned that the lyrics are better at predicting the genre than the other features we used. This might have been due to the fact that each genre of music has different styles. The models, however, did not provide an impressive performance because we chose a very small proportion of words to form our feature vectors (1000 or 2000 out of 357,000 total unique words). Below are the results from each method

#### 6.3.1. Most frequently used words

For this method, we got an accuracy of 0.6 and an average precision of 0.511.

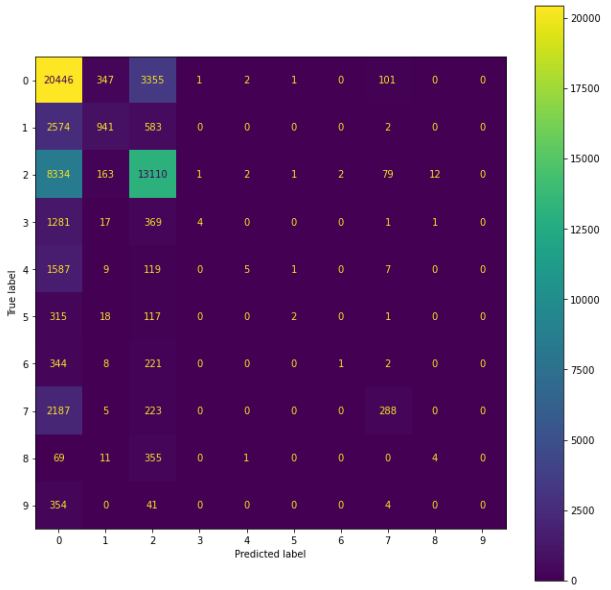


*Figure 6. Confusion matrix for the logistic regression model using most frequently used words as inputs*

Although the accuracy and precision are still somewhat low, this is a significant boost from the baseline and the language-based models.

#### 6.3.2. TF-IDF

For this method, we also got an accuracy of 0.6 and an average precision of 0.34.

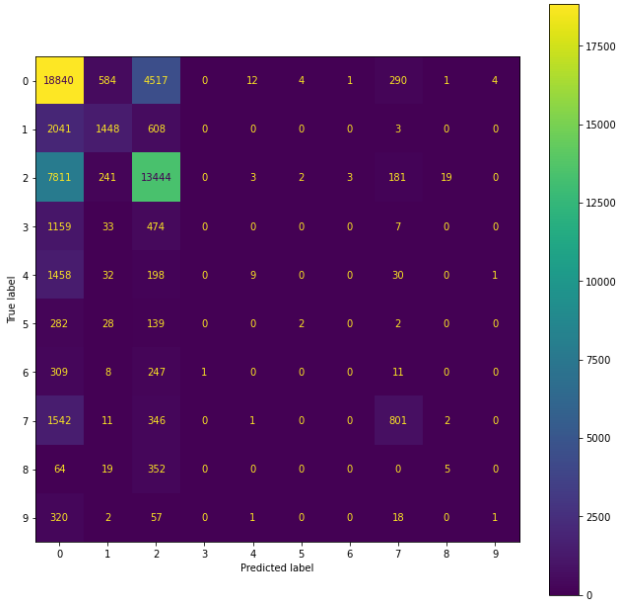


*Figure 7. Confusion matrix for the logistic regression model using tf-idf*

The accuracy did not improve from the most-frequently-used method, and the average precision dropped significantly. We suspect that this is because we chose a smaller dictionary size when forming our input vectors (k=1000). In addition, we might have needed a larger max_iter since our model did not converge.

**7. References**

[1] Bejan, M. (2021, January 8). *Multi-lingual lyrics for genre classification*. Kaggle. https://www.kaggle.com/datasets/mateibejan/multilingual-lyrics-for-genre-classification

[2] Anna Boonyanit and Andrea Dahl. *Music Genre Classification using Song Lyrics.* https://web.stanford.edu/class/cs224n/reports/final_reports/report003.pdf