# Classifying Songs by Genre Using Lyrics

**Adam Gulick**

## Abstract

This paper searches to determine the abilities of a Naive-Bayes Classifier to be able to determine the genre of a song when given only the lyrics as a list of words. For this work, songs from the genres of Rock, Country, Rap, Pop, and Dance are used for training and testing the model. Two Naive-Bayes Classifiers are trained: the first using every word from every song in the training data and the second using only the 1000 most common non-stop words from the corpus. The better of the two was able to reach an overall accuracy of 39.4%. Also discussed in this paper is the issue of overpredicting that can present itself.

## 1 Related Work

The question that this project stemmed from is not a novel idea. There has been plenty of work done with similar ideas of classifying music into genres. The largest variety in the studies on this topic comes from the models that are trained as well as the number of genres that the songs were classified into.

Even just looking at a few of the recent studies in regards to this question you can see many different models and approaches. Hu et al. (2009) looked into classifying 5,585 songs into 18 different mood categories using the songs lyrical text using Support Vector Machines and determined that Bag of Words were the most successful type. This problem was addressed again by Hu et al. (2017) when they looked at whether or not using a multimodal system made a significant difference in music mood classification.

In the study by Leszczynski and Boonyanit (2021), a LSTM model was trained using GloVe embeddings of the song lyrics, reaching a peak success rate of 68%.

Meenakshi et al. (2020) uses a Bag of Words method for tf-idf calculations and was able to acieve a 63% accuracy.

In a 2020 study, DeMasi (2020) ran this problem on twenty-four different models and found the most success using a Deep Neural Network trained on a tf-idf feature set.

Canicatti (2016) discusses the importance of choosing a proper set of genres to be classified into. For a classifier to have a high accuracy, you need genres to be general enough for each genre's lyrics can be accurately classified but specific enough for each genre to have a unique set of lyrics. A set of 5 distinct genres were established for Canicatti's study.

A similar study was done on Filipino Music by Abisado et al. (2021), in which a Naive Bayes classifier was trained using the scikit-learn library to analyze and classify the moods of more than 200 songs into a binary mood of happy or sad.

A 2017 study by Dammann and Haugh (2017) looked at not only the lyrics of a song on Spotify, but also the song preview and the album artwork when attempt to classify the genre of a song. While the latter two are not relevant to this study, the model used to analyze the lyrics can be recreated for this study. A Naive-Bayes classifier was used specifically for the lyrics. When combining all three types of data, an accuracy of 91.75% was achieved.

Using two more recent language models, BERT and DistilBERT, Akalp et al. (2021) were able to achieve accuracies of 77.63% and 71.29%. A BERT model uses the contextual relationship between words to generate predictions.

Despite there being a wide variety of approaches and training models to this problem. They all follow the same rough pattern. Every study gets a large dataset consisting of songs across many genres. This data is then tokenized and used to train a model. A new set of songs is then given to the model as test data, and the accuracy is recorded.

The largest differences in the various studies comes from the number of genres that the songs were sorted into and the models that were trained. Some variance came from the different ways that the text could be tokenized.

## 2 Methods

The data for this project is rather simple. A set of songs and their lyrics from a wide variety of artists and genres will be needed to train and test the model. In attempts to avoid having to fetch my own data–which would be a long and cumbersome task– I looked at the other studies to see if there was a good pre-existing dataset. There was one particular dataset[1] from Kaggle that was used in multiple studies so that is the dataset that the needed data will be fetched from.

For this project, five genres will be used to classify the songs. These genres are: Rock, Rap, Pop, Country, and Dance. Canicatti (2016) also used five genres. However, for that study, Jazz replaced Dance in the list of genres. The dataset that will be used did not have any songs denoted as Jazz, thus it is omitted and Dance will take its place. For this study, two different models will be trained to classify songs into genre. These two models will then be compared to determine the difference in success.

For the classifiers, some pre-processing of the text will be required. The lyrics will have to be taken from the Kaggle data set, tokenized, and given to the classifier as training data with the correct genre. The test data will consist of 1000 randomly selected songs from each genre for a total of 5000 songs. The training data will be the remaining songs.

### 2.1 Naive Bayes Classifier

The model that will be trained is a Naive-Bayes Classifier using the NLTK module. This is the same model that was trained in Akalp et al. (2021) as well as others. That study was able to reach a 41.67% success rate of correctly classifying the songs. Canicatti (2016) also used a Naive-Bayes classifier and was able to reach a 30.98% success rate across the five genres.

Originally, the set of songs were separated by genre and features were obtained representing every word in each song. However, this led to a very poor accuracy so modifications were made to the training data. The first change was to remove all stop words from the training data. This led to a small improvement but the overall accuracy was still very bad. Thus, a large change was made to the way that features were given to the classifier. Rather than generating features for every single word in the entirety of the training set of songs, a smaller set of only the 1000 most common words in the entire corpus was used. The features consisted of whether or not each of those 1000 words was present or absent in that song.

Once the data that was needed for this study was extracted from the Kaggle dataset, it had to be tokenized and stored as a list of strings. When it was fully pre-processed, the breakdown of the data was as follows:

| | Training | Test | Total |
|---|---|---|---|
| Country | 9630 | 1000 | 9730 |
| Dance | 10895 | 1000 | 11895 |
| Pop | 52858 | 1000 | 53825 |
| Rap | 16114 | 1000 | 17114 |
| Rock | 93992 | 1000 | 94992 |
| Total | 183489 | 5000 | 188489 |

Table 1: Number of Training, Test, and Total Songs per genre

Once the training and testing sets had been created, that data could be used to train the classifier and the accuracy could be measured.

## 3 Results

For the initial classifier[2], the features for a song consist of every word that is contained in that song. The overall accuracy for this method was 20.12%. However, when you calculated the accuracy for each individual genre rather than the entire test set, an interesting result was clear.

| Genre | Accuracy |
|---|---|
| Country | .5 % |
| Dance | 0 % |
| Pop | 0 % |
| Rap | 100 % |
| Rock | .1 % |
| Overall | 20.12 % |

Clearly, Classifier 1 is very good at predicting Rap songs and very bad for any other genre. A look at the confusion matrix for this classifier (Table

[1] https://www.kaggle.com/datasets/neisse/scrapped-lyrics-from-6-genres?resource=download

[2] For simplicity, call this Classifier 1.

2) shows that this was happening because almost every song was predicted to be a rap song. Out of the 5000 songs in the test data, 4976 (99.52%) of them were labeled as Rap songs by the classifier.

| | Cntry | Dance | Pop | Rap | Rock |
|---|---|---|---|---|---|
| Country | **5** | 0 | 0 | 994 | 1 |
| Dance | 0 | **0** | 0 | 1000 | 0 |
| Pop | 3 | 0 | **0** | 997 | 0 |
| Rap | 0 | 0 | 0 | **1000** | 0 |
| Rock | 12 | 0 | 2 | 985 | **1** |

Table 2: The Confusion Matrix for Classifier 1. Rows represent the actual genre. Columns represent the prediction.

To determine if this extreme overprediction was caused by the way the classifier was trained, the rap train and test data sets were removed and the accuracies were calculated again. When Rap songs were not included, the overall accuracy increased to 39.4% and the per genre breakdown was as follows:

| Genre | Accuracy |
|---|---|
| Country | 57 % |
| Dance | 37.9 % |
| Pop | 56.7 % |
| Rock | 6 % |
| Overall | 39.4 % |

These results show that there is no error in the way the training data is being given to Classifier 1 but rather that it is simply overpredicting rap songs.

The reasons for this occurrence will be discussed more in depth in Section 4 but it is very clear that something needs to be changed about Classifier 1.

This led to a new classifier[3] being created by modifying the features that are used to train the classifier. Because of the overwhelming number of distinct features in the corpus, only the thousand most common words will be used for Classifier 2. After creating the training data, the thousand most common non-stop words[4] from the entire corpus were calculated and used to create the features. This means that each song will have exactly 1000 features, one for each word. The feature will denote whether or not that specific word was present in that song. The difference between Classifier 1 and Classifier 2 is that Classifier 2 will not only be trained according to what words are present in each song in a certain genre, but also what words are absent in each song, whereas Classifier 1 only looked at which words were included.

Classifier 2, while still not having an amazing overall accuracy, was significantly better than Classifier 1. Classifier 2 had an overall success rate of 44.22% and did not have an overprediction of any genre. The per genre accuracy and confusion matrix are shown below:

| Genre | Accuracy |
|---|---|
| Country | 23.9 % |
| Dance | 17.5 % |
| Pop | 30.2 % |
| Rap | 80.9 % |
| Rock | 68.6 % |
| Overall | 44.22 % |

| | Cntry | Dance | Pop | Rap | Rock |
|---|---|---|---|---|---|
| Country | **239** | 22 | 209 | 24 | 506 |
| Dance | 13 | **175** | 339 | 72 | 401 |
| Pop | 36 | 107 | **302** | 58 | 497 |
| Rap | 4 | 54 | 70 | **809** | 63 |
| Rock | 50 | 31 | 202 | 31 | **686** |

Table 3: The Confusion Matrix for Classifier 2 using 1000 tokens per song. Rows represent the actual genre. Columns represent the prediction.

Clearly, Classifier 2 predicts more than just one genre and has an overall accuracy roughly double that of Classifier 1. The accuracy for each genre increases substantially with the exception of Rap, but this is to be expected as previously the accuracy was 100%. One thing of note is that Classifier 2 has a similar overall accuracy as Classifier 1 without rap songs, and actually does worse for three of the four remaining genres. However, because Classifier 2 is able to perform with the same accuracy on all five genres, this is a better classifier.

The reason that 1000 words was chosen was that any larger and too much RAM would be required. A larger number of features per song would be more accurate. However, 1000 features per song is the highest amount that can be handled. In an attempt to increase the number of words and decrease the require RAM, only the features that marked a word as present were included, and the features that denoted which words were absent were ignored. However, upon using this method, the same problem that appeared for Classifier 1 occurred again and almost every song was predicted to be Rap. This shows that when it comes to creating a successful classifier, it is just as important to make

[3] Naturally, call this Classifier 2.

[4] The list of English stop words was imported from the nltk corpus.

note of what features are absent as making note of features that are present.

## 4 The Overprediction of Rap Songs

The largest problem of Classifier 1 and the need for Classifier 2 was the overprediction of Rap songs. As stated, 99.52% of the 5000 songs in the test data set were predicted to be Rap songs by Classifier 1.

Canicatti (2016) talked about running into a similar issue in his study. The Naive Bayes classifier that was trained in that study performed 4 percentage points better when rap songs were not included in the training and test data sets. In that study, it was determined that rap songs had the largest amount of unreliable and inconsistent data which led to more inaccurate results that other genres, hence an improved classifier once removed.

While that was not the exact same as the problem with Classifier 1, rap songs posed a large issue for the classifier. Because rap has the most distinctive word-presence features, the classifier predicted many songs to be rap that were not actually rap. After training Classifier 1, the 100 most informative features were calculated. All 100 were labeled as features of rap songs.

While the exact reason for the overprediction cannot be isolated and removed, it is likely caused by the fact that the features of rap songs are highly more informative than those of other genres. This makes sense with what one would predict because there is typically a large set of words that are unique to rap songs. Many explicit language, slang, and other niche words appear uniquely in rap songs.

One thing to note is that country was the only other genre with more than one song to be correctly predicted by Classifier 1. Country is the genre that arguably has the next most unique set of lyrics, second to rap. Many words such as southern slang, farming words, or religion vocabulary appear much more frequently in country songs than any other genres. Thus, the classifier would have a greater chance of still being able to correctly identify a song as country.

By making the changes in Classifier 2, the model is not only trained on which words are present in the lyrics, but also which words are absent. Because the lyrics of rap songs are very similar to other songs in the genre, they will likely all contain the most distinct words. Thus, if a song does not contain any of the three most informative rap features, it is likely not a rap song and Classifier 2 will have a greater chance of predicting the correct genre than Classifier 1 would with only knowing which words are present.

## 5 Conclusions

A majority of the work for this project was in the preprocessing of the lyrics and then in transforming Classifier 1 to Classifier 2 as an attempt to solve the overprediction that was present in Classifier 1. By minimizing the number of features given to the classifier and training it on the features that were absent as well as the ones that are present, the overall accuracy was able to increase twofold. This shows that it is equally important for a classifier to know what is absent as knowing what is present.

If future work were to be done on this study, an attempt to be able to increase the number of features given per song would be a natural place to start. By increasing the number of features, the classifier should become more accurate. Alternate approaches could include training the classifier on bigrams, trigrams, or n-grams rather than just the individual words. Using punctuation or the structure of the songs could also help the classifier become more accurate. Being able to look at the number of unique tokens in a song might also help a classifier predict a genre as some genres have much more or less repetition of words than others.

## References

Mideth Abisado, Mardyon Yongson, and Ma.Ian De Los Trinos. 2021. *Towards the Development of Music Mood Classification of Original Pilipino Music (OPM) Songs Based on Audio and Lyrics Keyword*, page 87–90. Association for Computing Machinery, New York, NY, USA.

Hasan Akalp, Enes Furkan Cigdem, Seyma Yilmaz, Necva Bölücü, and Burcu Can. 2021. Language representation models for music genre classification using lyrics. *2021 International Symposium on Electrical, Electronics and Information Engineering*.

Anthony Canicatti. 2016. Song genre classification via lyric text mining.

Tyler Dammann and Kevin P. Haugh. 2017. Genre classification of spotify songs using lyrics , audio previews , and album artwork.

Nick DeMasi. 2020. Can you hear me now? predicting song genre from song lyrics using deep learning.

Xiao Hu, Kahyun Choi, and J. S. Downie. 2017. A framework for evaluating multimodal music mood classification. *Journal of the Association for Information Science and Technology*, 68.

Xiao Hu, J. S. Downie, and Andreas F. Ehmann. 2009. Lyric text mining in music mood classification. In *ISMIR*.

Megan Leszczynski and Anna Boonyanit. 2021. Music genre classification using song lyrics.

K. Meenakshi, M. Safa, G. Geetha, G. Saranya, and J. SundaraKanchana. 2020. Music genre classification using lyric mining based ontf-idf.