Music Genre Classification Based on Lyrics

Machine Learning & Predictive Analytics Final Project Report

American University of Beirut

May 2023

By:

Lara Baltaji

Haidar Noureddine

Christy Bou Mansour

Hadi Knaiber

Professor: Dr. Weal Khreich

Abstract

With the rise of music digital streaming, the music industry has expanded to become a major part of modern lifestyles in many countries. Due to the expansion of music streaming services worldwide, we have witnessed an increase in the demand for advanced music classification methods, especially for music genre classification. Music genre classification for music information retrieval has become a requirement to keep up with the rise in the music industry. However, accurately pinpointing musical features to classify various genres is still a challenge. Earlier studies have shown various means and methods to classify genres based on linguistic and/or audio features. This report proposes various ways and means to predict the music genre of a song based on the lyrics by using natural language processing techniques to extract relevant features to identify the genre. Throughout the report, we outline the methodology in detail that includes several steps such as data preparation, filtering, encoding, and modeling. After experimenting with various machine learning models, we evaluated the performance of each model based on accuracy, hamming loss, recall, precision, and F1-score. After testing, the results show that BERT and Label Powerset using Random Forest Classifier were the top models for predicting genres based on their hamming loss scores. These models have the lowest hamming loss scores and the highest precision scores in comparison to the other models. Our choice of models proposes promising results and provides a foundation for future studies related to this topic.

1. Introduction:

Over the last couple of years, the music industry has become an integral part of society's lifestyle in several countries, influencing cultures and empowering people's lives. With the rise of digital music streaming and to enhance the user experience, there has become a need to classify the genre of songs based on the lyrics. The goal of music genre classification, which attempts to give more precise and effective methods of classifying music based on its genre, is the focus of substantial study in the field of music information retrieval. There has been an increase in demand for more advanced music classification methods, notably for multi-genre classification, due to the exponential expansion of music streaming services. Despite the advancements made in this field, it is still difficult to pinpoint the various musical features that contribute to the various genres. Lyric-based genre classification, which considers textual data, is one method to improve classification accuracy.

Music genre classification is a significant research topic in the field of music information retrieval. Multi-genre classification is even more challenging than single-genre classification due to the complexity of identifying multiple musical characteristics that contribute to different genres. Lyric-based genre classification is one of the approaches that consider textual information to enhance classification accuracy. Studies have shown that using linguistic features to classify music genres, specifically based on lyrics, often results in lower accuracy compared to classifiers that utilize audio features (Thompson, 2021). This is due to the inherent complexity and subjectivity of lyrics, which may contain metaphors, ambiguous language, and cultural references that are difficult to capture using linguistic features alone. In contrast, audio features, such as tempo, rhythm, and timbre, can provide more objective and concrete information about a song's genre. Thus, the problem we aim to address is the limited capability of using audio features for genre classification to detect song genre accurately, in addition to the inability of using linguistic features. Despite this limitation, there is still potential for linguistic features to contribute to genre classification, particularly when used in combination with other features such as metadata or audio data. The proposed solution is to develop a machine learning model that can process natural language to extract relevant linguistic features from the lyrics and identify the genre of the song. In this report, we explore the means to build a model to predict music genre based on the song's lyrics. It will also address research gaps, and the current work's position in relation to what has been done before. By doing so, our model will be capable of accurately predicting the song's genre according to the song's lyrics.

2. Literature Review:

In this literature review, we explore the existing literature related to genre classification and specifically lyric-based genre classification. It will also address research gaps, and the current work's position in relation to what has been done before.

One of the methods for music genre classification found in the literature uses audio signals to predict music genres. This approach involves processing audio signals by removing unwanted noise in order to extract useful audio features. These features may include musical aspects such as beat, rhythm, timbre and pitch. They may also include the level of abstraction, temporal scope, and signal domain (Chen, Kou, Hou, & Zhou, 2022). These features are then used to classify the audio into different genres. Some of the most commonly used machine learning algorithms for music genre classification based on audio are support vector machines (SVM), k-nearest neighbors (KNN), convolutional neural networks (CNN), recurrent neural networks (RNN) and Naïve Bayes Classifier (Chettiar & Selvakumar, 2021). For example, Chettiar and Selvakumar (2021) extracted Fast Fourier Transformer (FFT) and Mel-frequency Cepstral Coefficients (MFCC) parameters to predict music genre using various Deep Learning models such as CNN, RNN, KNN, Naïve Bayes Classifier and SVM. In their study, Support Vector Machine and the Convolution Neural Network Models performed the best.

Another method is based on music's metadata. Music metadata usually includes previously collected information such as artist name, release date, popularity score, key, energy, acoustic features, liveness, instrumentation and much more. While there has been limited research conducted on music genre classification using metadata alone, many studies have utilized metadata as an additional feature, in combination with audio or lyrics, to improve classification accuracy. This data is not extracted from the music but is collected from the community. For example, a research done by Linn Bergelid (2018) combines lyrics and metadata to classify music genre. The metadata used are the artist name, release year, energy level and valence. Linear Support Vector Machine (LSVM), Multinomial Naïve Bayes (MNB), k-nearest neighbors (KNN) and random forest (RM) classifiers were used to predict the genre, with the RF performing best and the KNN performing worst.

A third approach for music genre classification is to use lyrics as the input, which this paper will tackle. Numerous methods have been suggested to categorize songs into various genres based on their lyrical content. However, it is generally recognized that any technique relying on lyrics to classify genres must incorporate natural language processing (NLP) as a critical factor. NLP enables the extraction of useful features from the lyrics, such as vocabulary size, word frequencies, and sentiment, which can be used to identify similarities and differences between songs in different genres. Some of the most commonly used machine learning algorithms for lyric-based music genre classification include decision trees, support vector machines (SVM), k-nearest neighbors (KNN), and neural networks.

For example, Junru Yang (2014) uses NLP techniques to classify music genres in two steps. The first step generates a list of the top ten words for each of the seven music genres being analyzed. In the second step, the test data is classified into one of the seven music genres using these top words. Several classifiers are employed in order to assess their accuracy. These classifiers include Naive Bayes, Linear Regression, K-nearest neighbor, decision trees, and sequential minimal optimization (SMO). The Naïve Bayes classifier achieved the highest accuracy of 65.7%.

Another study conducted by Thompson and Warwick (2021) aimed to create hand-craft features which were not used in previous lyrical classifiers. These features include rhyme density, readability, and the frequency of profanity usage. These features were used to train six traditional machine-learning algorithms (Support Vector Machin, Multi-Layer Perceptrons, Naïve Bayes, Random Forest, XGBoost, KNN) to classify lyrics across nine popular music genres. Out of all the models tested, the SVM achieved the highest level of accuracy, with a score of 56.13%.

The present work will use song lyrics in order to perform multi-genre classification using multi-label classification techniques. Let us look at the most popular techniques for multi-label classification techniques used in text. The most commonly used techniques include Binary Relevance, Classifier Chains, Label Powerset.

To start, Binary Relevance is an approach where each label is treated as a separate binary classification task. This involves training a binary classifier, such as logistic regression, on each label separately, to predict whether it is present or absent in the input. For example, in a study that aims to develop and design a new method for multi-label text classification for Arabic texts based on the binary relevance method, the results achieved a good performance with an overall F-measure of 86.8% for the multi-label classification of Arabic texts, in addition to showing an important effect from the used feature selection methods on the classification (J. Pestian, 2007).

The other technique, Classifier Chains, considers dependencies between labels by training a sequence of binary classifiers, in which the input features have the labels that are predicted by classifiers previously in the chain.

Label Powerset, on the other hand, reduces the multi-label problem to a multi-class problem by treating each combination of labels as a separate class. In a study done by Jesse Read (2014), a framework of meta-labels is proposed, which allows for a random projection into a space where nonlinearities can be easily tackled with established linear learning algorithms. The proposed framework is evaluated against high-performing methods and achieves competitive accuracy with less computation. Finally, OneVsRest is a simple method where each label is treated as a separate binary classification task, but the model is trained as a single multi-class classification task with one class for each label.

In the previous work of lyric-based music genre classification, a general pipeline typically involves several stages. The first stage involves data collection and preprocessing of lyrics, which includes tokenization, stop-word removal, and stemming. The next stage is feature extraction, where features such as bag-of-words, n-grams, and TF-IDF are extracted from the pre-processed text. The third stage involves dimensionality reduction and feature selection to reduce the number of features and improve the classification accuracy. The fourth stage is the classification stage, where different machine learning algorithms such as Naive Bayes, Decision Trees, Support Vector Machines, and Neural Networks are applied to classify the lyrics into different genres. Finally, the last stage is the evaluation stage, where the performance of the classification models is assessed using metrics such as precision, recall, and F1-score.

For example, in the study of Yang, J. (2014), the researchers used data from different sources, including Million Song Dataset (MSD), musiXmatch, and Last.fm. They manually labelled the data and divided it into two sets: one for training the machine learning models and the other for testing them. They used Python and SQLite to pre-process the data and identify songs with genre tags. The researchers used different machine learning algorithms to build models that can analyze the lyrics of a song from different perspectives, such as bag-of-words, TF-IDF, Word2Vec, Doc2Vec, and topic modelling. They used various metrics, such as accuracy, precision, recall, F1 score, and AUC, to evaluate the performance of each model. The results showed that the Random Forest classifier with TF-IDF features performed the best, with an accuracy of 0.75.

In conclusion, the use of machine learning in music has the potential to revolutionize the way we understand, create, and enhance musical compositions. It can help us identify and analyse the underlying structures and patterns in different musical genres, generate new pieces of music, and remix existing pieces in novel ways. However, the use of machine learning in lyric-based music genre classification also presents some challenges, including the need for large amounts of high-quality training data and the potential for bias in the training data due to inherent complexity and subjectivity of lyrics, which may include metaphors, ambiguous language, and cultural references. In addition to that, there is also limited research on the effect of cultural and linguistic factors on song genre classification based on lyrics as there is limited research on classifying song genres based on lyrics across different music genres and different languages.

Despite having gaps and question marks in areas in the existing research on lyric-based genre classification, there is high potential in improving genre-classification accuracy by incorporating linguistic features. Our study can contribute to the literature value by addressing this gap and provide insights on the effectiveness of using lyrics for classifying genre. In addition to that, our study can add to the literature through exploring and testing the effectiveness of algorithms related to machine learning models and natural language processing techniques. Our study will also work on several models and compare the performance of different algorithms and this will lead to choosing the most effective model for lyric-based genre classification. Also, our study includes several languages, so this will expand the scope of the training data for machine learning algorithms and this can provide a more diverse and representative sample for the algorithm to learn from and this can help improve accuracy and effectiveness. This will add to the robustness and adaptation of machine learning models since songs from different languages have unique features. Finally, from a cultural point of view, this study can help address cultural bias issues in music genre classification and thus improve the accuracy and the fairness of lyric genre classification.

3. Methodology:

The methodology section explores and outlines the steps taken to develop the machine learning model that aims to use song lyrics to predict the genre. In order to develop the model, we began by importing the necessary packages in the Python notebook followed by uploading two dataset, which are the “Artist Dataset” and the “Lyrics Dataset”. The “Artist Dataset” includes the artist name of the song (Artist), the genre of the song (Genres), the song count per artist (Songs), the popularity scale of the song (Popularity) and the link (Link). On the other hand, the “Lyrics Dataset” includes the Alink (the link of the artist to connect the 2 datasets), the SName which is the song name, the SLink which is the song link, the lyrics of the song (Lyric) and the language (Language).

Data Preparation

The first step in the data preparation phase involved checking the unique values of the genre column in order to identify the languages of the genres in the Lyrics dataset. Next, a language map dictionary was created in order to map the language abbreviations to their full names; for example, “en”: “English”. We then used this dictionary created to map the language abbreviation column to each language’s respective name. To further process the dataset, we dropped unnecessary columns for each dataset. From the Artist, we dropped the song count and the popularity, and from the Lyrics dataset, we dropped the SLink column. This is because they are not required features as we only need the lyrics and the genre features. Following that, we merged the 2 datasets and we dropped the Alink (since we merged the 2 datasets), the song names, the artist and the link. We applied the count function to check the songs’ languages, and the top 5 languages were English (191,387), Portuguese (156,941), Spanish (9,916), Kinyarwanda (1,679), and Italian (1,426).

To ensure consistency in terms of the songs’ language, we filtered our dataset to include only English songs. This was done by creating a dataframe that only includes the English language (using .loc function to filter language). Afterwards, the null values were checked and found in the Genre column and were later removed. Also, we checked for duplicate values to ensure that there are no duplicates and dropped them. Additionally, we dropped the duplicates in the “Lyric” feature to ensure that the duplicates are absolutely removed.

After language exploration, we explored the multiple genres assigned to each song in order to better understand the distribution of the genres. Then, we started preprocessing the Genres column as a preparation step for the machine learning model. To do so, encoding of the Genres column is required, so we used the MultiLabelBinarizer in order to transform the Genres feature to a binary form. Note that MultiLabelBinarizer is an important function that changes the column values to separate columns represented by a binary value (0 or 1).

We created an instance of the MultiLabelBinarizer and then applied the fitting and transforming to the Genres column in order to encode it. The resulting binary format is then saved in a new dataframe labeled “y”. We then used the concat function to merge the english dataset and the newly encoded data frame “y”. The resulting dataset was labeled “df_eng”. Also, we looked for null values and removed them. We also dropped the “Genres” column. Now, we have a dataset of the lyrics, the language “English” and each column with a genre encoded with either 0 or 1 for each respective song. We then fixed the datatypes of the remaining columns and changed them to integer type, this ensures the data is appropriate and in a proper format for subsequent steps in the project.

To explore different genres and analyze their distribution in the dataset, we extracted the column names from the df_eng dataset and we calculated the number of songs under each genre label, storing the result in a new dataframe called “df_stats”. Using a bar chart, we plotted the top 10 genres with the highest number of songs. The result showed rock being the #1 genre followed by pop, then pop/rock. Then, we calculated the total number of genres and counted the number of songs in each value of the genre column. The resulting data frame is sorted in an ascending order according to the number of genres for every song and then plotted on a bar chart. Following, we plotted a word cloud that shows the most used words for the top 9 genres and then we kept only the top 5 genres for dimensionality reduction purposes ('Lyric','Rock','Pop','Pop/Rock','Heavy Metal','Hip Hop'). Next, we reduce the number of genres to the top 5 genres and the number of songs per genre is then recalculated and plotted using a bar chart. Finally, we apply what we did in the word cloud previously but only with the number of songs that have multiple genres within the top 5 genres and plot it.

Machine Learning:

Machine learning models are powerful tools for making predictions and analyzing complex data, thus generating valuable insights. In this section, we will explore the means and approaches for building our model.

First, a sample of 5,000 rows is taken and then preprocessing the Lyric column is applied (to reduce processing time). In the preprocessing step, we first explored the Lyric column and looked for noise using “noise_scan()” and “extract_stopwords()” using the “nt” TextFrame and TextExtractor which explore the text for noise. For the preprocessing, we define a preprocessing function that replaces newline \n characters with spaces, convert all text to lowercase, remove punctuation and special characters, tokenize the text into individual words, remove stop words and stem or lemmatize the remaining words to their root forms. We then deploy the function to the Lyric column.

Feature Engineering:

We proceed with the feature engineering as this step is crucial in machine learning; it includes transforming features into meaningful ones that can improve the performance of the predictive models. We begin with using the Term Frequency-Inverse Document Frequency (TF-IDF) which calculates the importance of words. We initialize the TfidfVectorizer and fit it to the corpus. Following that, we transform the preprocessed lyrics to a matrix of the TF-IDF features and then convert it to an array. Following, we split the data using the train_test_split function from scikit-learn. Regarding Multi-Label problems, there are 3 solutions which should be taken into account and these are: Problem transformation, Adapted algorithm, and Ensemble approaches. We will be using the problem transformation which refers to transforming the multi-label problem into a single-label problem by using 3 methods which are:

- Binary Relevance: a technique used in multi-label classification in order to treat each label as a separate binary classification task. This approach is beneficial when labels are independent of each other. For example, regarding the genres “Rock” and “Pop”, the model will train separate classifiers for each label in which each classifier predicts whether a song belongs to a particular genre or not.

- Classifier Chains: in this multi-label classification method, a sequence of binary classifiers is built, and the first classifier is trained on the input data. However, the following classifiers are trained on the input space and all the prior classifiers in the chain. The process continues until all the labels are predicted.

- Label Powerset: this method aims to treat each unique label combination in a separate class. Once the label combinations are converted into classes. We can train a multi-class classifier on the transformed dataset.

We proceeded with applying the multi-label classification methods (Binary relevance, Classifier chains and Label powerset) on six different machine learning model, Multinomial Naive Bayes, Gaussian Naive Bayes, Random Forest Classifier, K- Nearest Neighbors, Logistic Regression and Extreme Gradient Boosting (XGboost).

The metrics used to evaluate each of the above models were the accuracy score and the hamming loss score. The accuracy score detects the fraction of predictions that the model got right while the hamming loss score is a performance metric that evaluates multi-label classification methods. On the contrary, the hamming loss measures the fraction of misclassified labels (the predictions that are inaccurately predicted). The lower the hamming loss score, the better the model performs since the hamming loss is a measure or error.

Deep learning:

BERT

For deep learning models, BERT is used and we will be using Hugging Face Transformers library to load a pre-trained BERT model and its associated tokenizer. The tokenizer is used to preprocess text inputs into a format that can be analyzed by the BERT model, which is necessary for tasks, for example the text classification or question answering. By using the BertTokenizer class, Hugging Face automatically downloads the necessary tokenizer files from their model hub if they haven't already been cached. We proceed with defining the corpus (the lyric preprocessed column) and rearrange the columns. We then set the hyperparameters followed by the target list and the tokenizer. We then test the BertTokenizer by Hugging Transformers and then create the dataset class. We then split the data and deploy the model before finally calculating the accuracy, hamming loss, precision, recall, and F1-score.

Multi-Layer Perceptron Classifier (MLP)

We define the MLP model and train it using the fit method. We then make predictions on the test data and evaluate the metrics mentioned above for this model.

4. Results:

After performing the necessary preprocessing, feature engineering, and machine learning models, the results of the classification of music genres can be analyzed. For all the 20 models tested, the metrics of: Accuracy, Hamming loss, Precision, Recall, and F1 score are calculated. A table summary of the results (found below) shows the metric scores.

	Model	Accuracy	Hamming Loss	Precision	Recall	F1 Score
18	BERT	0.341000	0.221900	0.604900	0.442700	0.511200
14	Label Powerset using RFC	0.382000	0.228267	0.578177	0.446328	0.503768
2	Binary Relevance using RFC	0.281333	0.229467	0.604824	0.334874	0.431074
8	Classifier Chains using RFC	0.279333	0.229600	0.605042	0.332820	0.429423
4	Binary Relevance using LR	0.255333	0.234533	0.591262	0.312789	0.409137
16	Label Powerset using LR	0.366000	0.236267	0.557227	0.437596	0.490219
5	Binary Relevance using XGB	0.261333	0.237467	0.565149	0.369800	0.447066
17	Label Powerset using XGB	0.369333	0.239867	0.545623	0.454545	0.495937
6	Classifier Chains using MNB	0.234667	0.241733	0.567404	0.289676	0.383543
0	Binary Relevance using MNB	0.227333	0.243200	0.562564	0.284027	0.377474
12	Label Powerset using MNB	0.324667	0.255333	0.510667	0.393426	0.444444
10	Classifier Chains using LR	0.328667	0.256000	0.508761	0.402671	0.449541
11	Classifier Chains using XGB	0.325333	0.260933	0.496970	0.421161	0.455936
19	MLP	0.241000	0.267000	0.481200	0.388300	0.429800
3	Binary Relevance using KNN	0.226000	0.271867	0.466521	0.329224	0.386028
9	Classifier Chains using KNN	0.272667	0.294400	0.422827	0.367232	0.393073
15	Label Powerset using KNN	0.254667	0.298800	0.414634	0.366718	0.389207
13	Label Powerset using GNB	0.262000	0.300800	0.409278	0.357987	0.381918
7	Classifier Chains using GNB	0.176000	0.326933	0.380502	0.412943	0.396059
1	Binary Relevance using GNB	0.174000	0.346400	0.359152	0.426297	0.389854

Since the dataset is imbalanced (for example, the “Rock” genre as stated previously has the highest count among the genres), the accuracy score alone cannot be considered. Therefore, we relied on other accuracy scores such as the hamming loss, precision, F1-score, and recall deciding which model would be the best fit. After analysis of the performance metrics, we decided that among the best performing models are BERT, Label Powerset using RFC, Binary Relevance using LR, Binary Relevance using XGB and the worst performing models are Label Powerset using GNB, Classifier Chains using GNB and Binary Relevance using GNB. The top 2 best performing models we decided on were: BERT and Label Powerset using RFC. This is because they are among the best accuracy scores, they have the lowest hamming loss, among the highest recall scores, among the highest precision scores, and among the top F1-scores. Therefore, we believe that these models are the chosen candidates for accurately and precisely predicting the appropriate labels.

5. Conclusions and Recommendations:

In conclusion, music genre classification is an important research topic in the field of music information retrieval as the world is exploring a rise in the industry of digital music streaming. This led to applications and streaming websites to develop hunger for accurate music genre classification. Despite using audio features for classifying music genre has shown success in the past, the subjectivity and the complexity of the lyrics can sometimes be challenging to accurately classify music genres. So, there is a need to explore the potential of using natural language processing techniques to analyze the lyrics and extract relevant features to identify the genre of the song.

Throughout this course of study, we developed and tested machine learning models that used natural language processing methods to classify the genres of music according to lyrics. We explored various methods and means to build the model. Throughout the process, we followed the methodology that involved data exploration, preprocessing, and preparation. The dataset we used was based on the English language lyrics.

Despite our models achieving relatively good accuracy compared to the previous literature, there remains room for improvement. Further research and trials are needed to expand this field of study and achieve high accuracy models that can correctly predict music genres. A limitation we faced in this study is imbalance data which affected our accuracy scores. In addition, the use of only 5000 rows for faster processing has been a limitation. Future studies should use a larger subset of the data. In addition, the fact that our labels are imbalanced might have affected the performance of the models. Future studies should also try using stratified splitting and tuning the parameters of the models that are related to the data imbalance.

We also recommend digital streaming platforms to develop an interactive web-based application based on classifying music genres to make the model accessible to a wider audience. This will not only enhance the user experience but also increase the practical applications of the model in the music industry. Another recommendation is to consider incorporating audio features and metadata (such as tempo, rhythm, and timbre) to achieve better performance results. Moreover, improving the quality of the lyrics dataset is important to achieve better results. Digital streaming platforms should generate lyrics datasets that are of high-quality, well-structured, free of error, and containing a diverse variety of genres and songs as this can significantly impact the accuracy of the model.

As a final note, natural language processing methods can be a valuable addition to the field of music genre classification, and with further research and future studies, it definitely has the potential to achieve greater results. With the continued development of machine learning techniques, we look forward to the potential of building more accurate and efficient models to classify music genres.

6. Appendix:

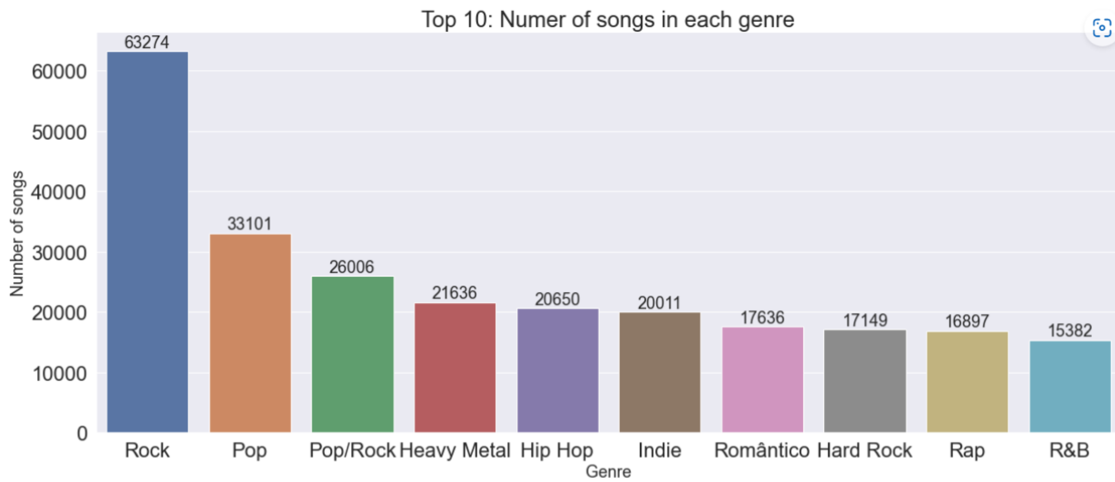


Figure 1: The number of songs per genre for the original dataset

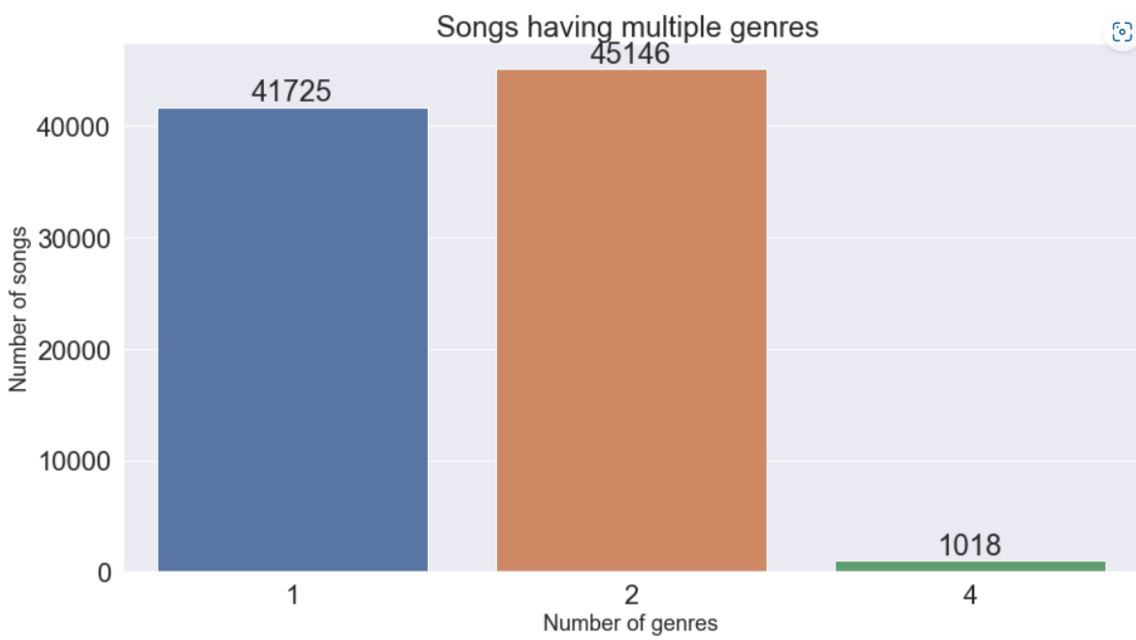


Figure 2: The number of genres per song for the original dataset

Figure 3: Word cloud for the top 9 genres for the original dataset

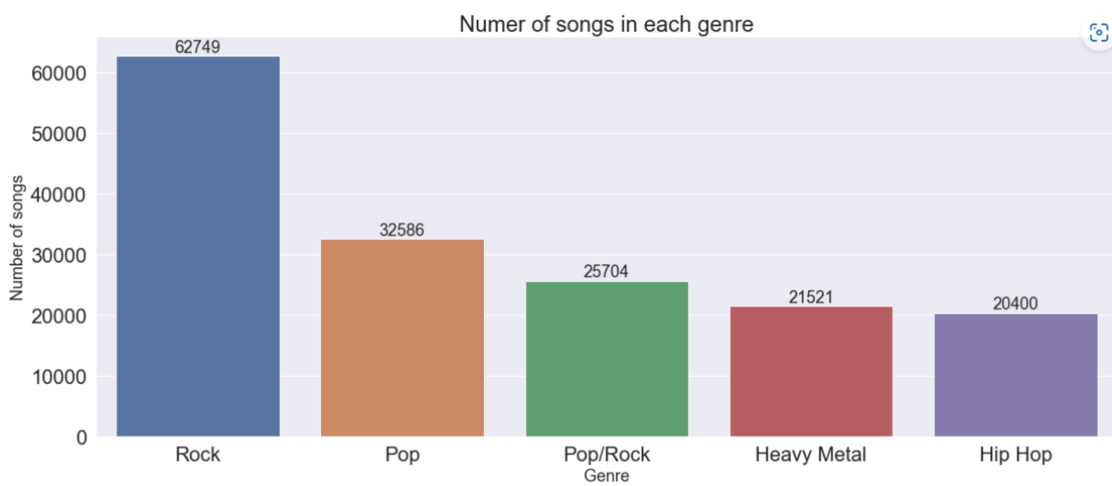


Figure 4: Number of songs per genre from the new dataset (containing only the top 5 genres)

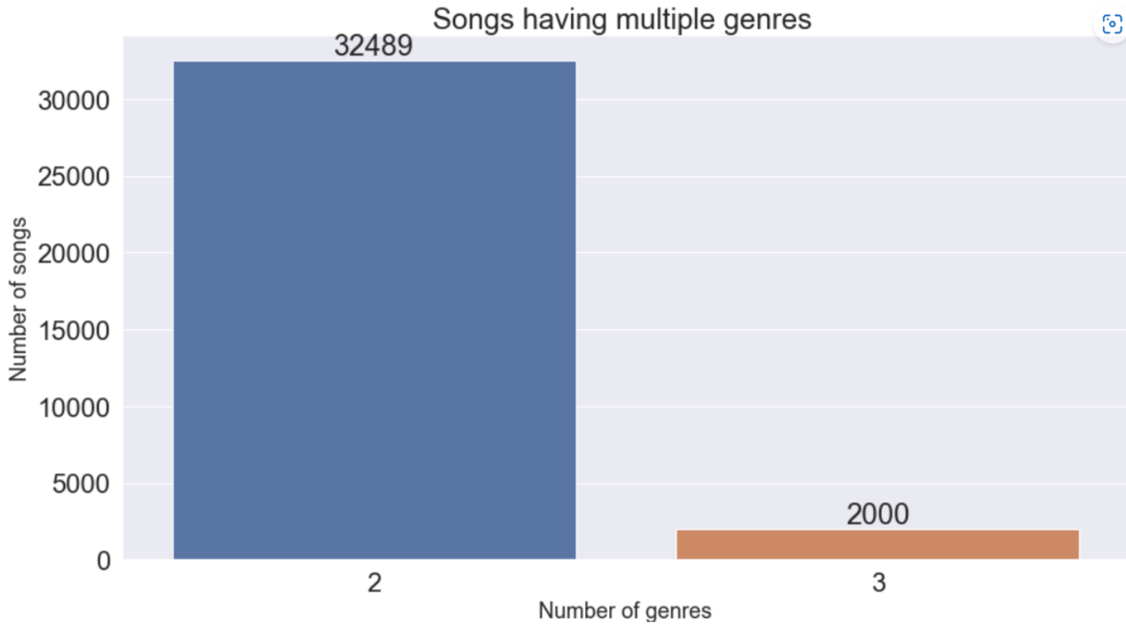


Figure 5: Number of genres per song in the new dataset

Figure 6: Word cloud for the new dataset

Multi-Class and Multi-Label Text Classification:

Tagged.S01.720p.AMZN.WEBRip.DDP2.0.x264-NYH[rartv] IMDb TV
Thriller IMDB: 7.3/10

StarBeam.S02.1080p.NF.WEBRip.DDP5.1.x264-NTb[rartv] IMDb TV
Animation, Action, Adventure, Comedy, Family, Fantasy, Sci-Fi

Multi-Class

X	Class
X1	Positive
X2	Negativ e
X3	Neutral

Multi-Label

X	Clas s1	Clas s2	Clas s3
X1	0	0	1
X2	0	0	0
X3	1	0	1

Problem Transformation

Adapted Algorithms (KNN,RF,etc)

Ensemble Methods

Figure 7: Multi-Class and Multi-Label Text Classification

MultiLabel Text Classification: Binary Relevance

X	Clas s1	Clas s2	Clas s3
X1	0	0	1
X2	0	0	0
X3	1	0	1

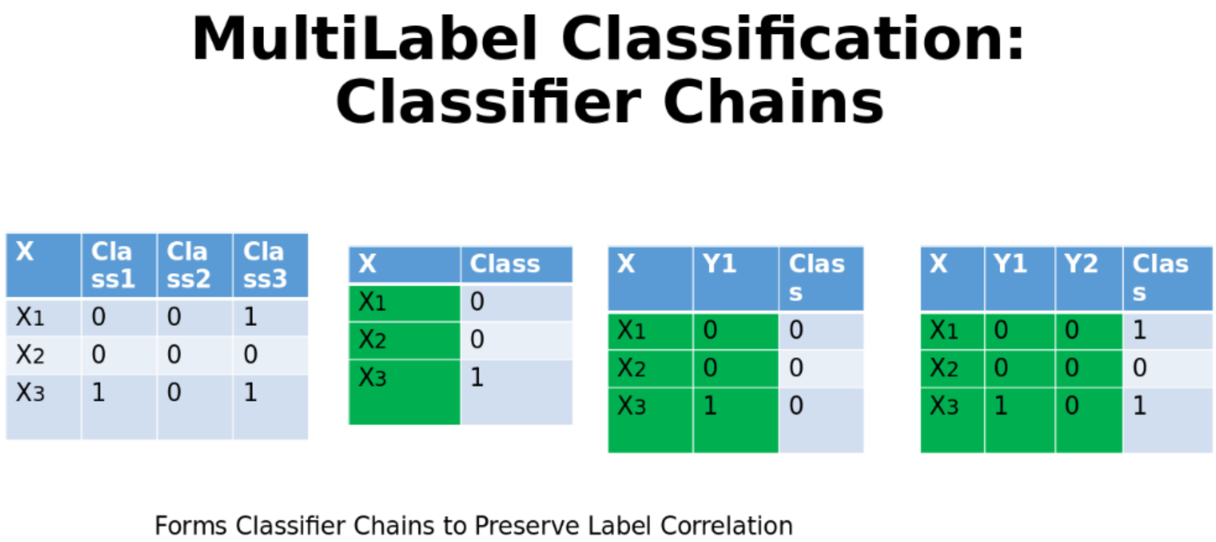
X	Class1
X1	0
X2	0
X3	1

X	Class2
X1	0
X2	0
X3	0

X	Class3
X1	1
X2	0
X3	1

Treats each label as a separate single classification problem
Label Correlation may be lost

Figure 8: MultiLabel Text Classification: Binary Relevance



MultiLabel Classification: Classifier Chains

X	Class1	Class2	Class3
X1	0	0	1
X2	0	0	0
X3	1	0	1

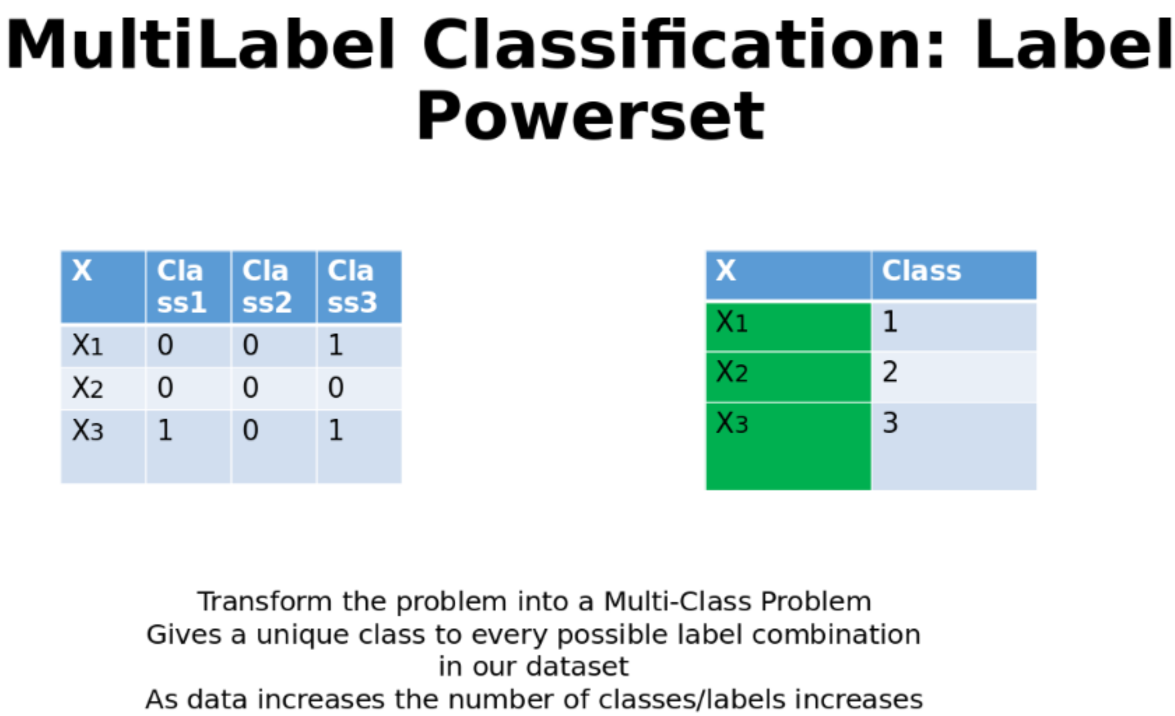
X	Class
X1	0
X2	0
X3	1

X	Y1	Class
X1	0	0
X2	0	0
X3	1	0

X	Y1	Y2	Class
X1	0	0	1
X2	0	0	0
X3	1	0	1

Forms Classifier Chains to Preserve Label Correlation

Figure 9: MultiLabel Classification: Classifier Chains



MultiLabel Classification: Label Powerset

X	Class1	Class2	Class3
X1	0	0	1
X2	0	0	0
X3	1	0	1

X	Class
X1	1
X2	2
X3	3

Transform the problem into a Multi-Class Problem
Gives a unique class to every possible label combination in our dataset
As data increases the number of classes/labels increases

Figure 10: MultiLabel Classification: Label Powerset

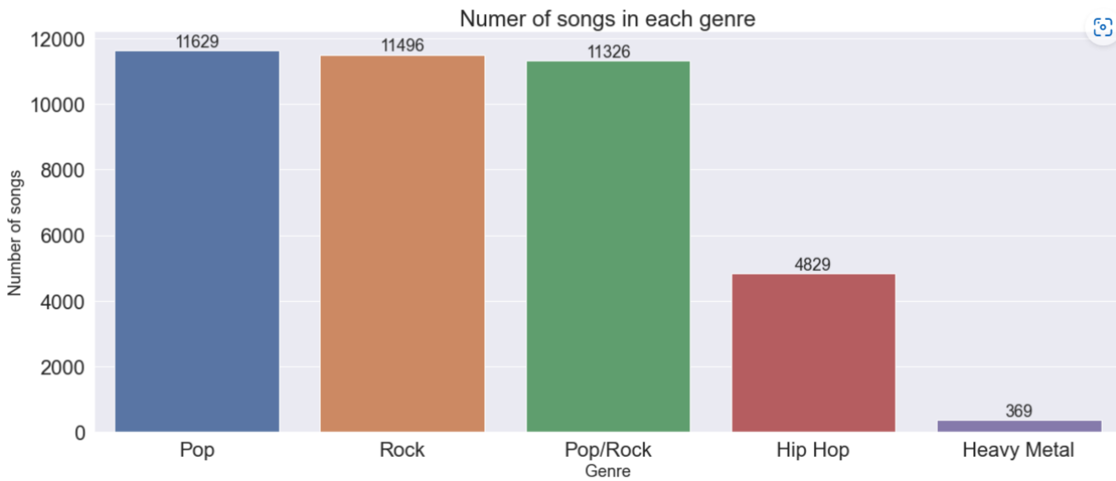


Figure 11: Number of songs in each genre

7. References:

Cai, L., Song, Y., Liu, T., & Zhang, K. (2020). A Hybrid BERT Model That Incorporates Label Semantics via Adjustive Attention for Multi-Label Text Classification. IEEE Access, 8, 152183-152192. doi: 10.1109/ACCESS.2020.3017382.

Bergelid, L. (2018). Classification of explicit music content using lyrics and music metadata. KTH Royal Institute of Technology.

Chen, B., Kou, H., Hou, B., & Zhou, Y. (2022, April 18). Music Feature Extraction Method Based on Internet of Things Technology and Its Application. Computational Intelligence and Neuroscience.

Chettiar, G., & Selvakumar, K. (2021, November 19). Music Genre Classification Techniques.

Thompson, C. (2021). Lyric-Based Classification of Music Genres Using Hand-Crafted Features. an International Journal of Undergraduate Research.

Tsaptsinos, A. (2017). Lyrical-Based Music Genre Classification Using a Hierarchical Attention Network. 18th International Society for Music Information Retrieval Conference. Suzhou, China.

Yang, J. (2014). Lyric-Based Music Genre Classification. University of Victoria.

Pestian, J., Brew, C., Matykiewicz, P., Hovermale, D. J., Johnson, N., Cohen, K. B., & Duch, W. (2007, June). A shared task involving multi-label classification of clinical free text. In *Biological, translational, and clinical language processing* (pp. 97-104).

Read, J., Puurula, A., & Bifet, A. (2014). Multi-label classification with meta-labels. In 2014 IEEE International Conference on Data Mining (pp. 941-946). IEEE. doi: 10.1109/ICDM.2014.38.

Raissi, T., Tibo, A., & Bientinesi, P. (2018, April). Extended pipeline for content-based feature engineering in music genre recognition. In 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP) (pp. 2661-2665). IEEE.

Shah, M., Pujara, N., Mangaroliya, K., Gohil, L., Vyas, T., & Degadwala, S. (2022). Music Genre Classification using Deep Learning. In 2022 6th International Conference on Computing Methodologies and Communication (ICCMC) (pp. 974-978). Erode, India: IEEE. doi: 10.1109/ICCMC53470.2022.9753953.