

T.R.
GEBZE TECHNICAL UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

GENRE DETECTION FROM LYRICS
USING MACHINE LEARNING

R. HILAL SAYGIN

SUPERVISOR
DR.GÖKHAN KAYA

2024-2025
GEBZE

 <p>GEBZE TECHNICAL UNIVERSITY</p>	<p>GRADUATION PROJECT JURY APPROVAL FORM</p>
--	--

This study has been accepted as an Undergraduate Graduation Project in the Department of Computer Engineering on 15/01/2025 by the following jury.

JURY

Member

(Supervisor) : Dr.Gökhan Kaya

Member : Dr. Gökhan Kaya

Member : Prof.Dr. Ibrahim Soğukpınar

ABSTRACT

This paper focuses on lyrics-based genre classification, leveraging the textual and structural properties of song lyrics to improve classification accuracy. While traditional approaches often treat lyrics as a uniform block of text, this work prioritizes the structural composition of songs, particularly the chorus and the custom features extracted to explore its impact on classification performance. A novel dataset was constructed by combining data from Genius.com and HuggingFace, annotated with section-level labels such as *chorus*, *verse*, and *intro*. The dataset was segmented into *chorus* and *non-chorus* subsets, enabling a comparative analysis of their predictive contributions.

A custom feature extraction system developed, integrating textual, semantic, and stylistic features, including TF-IDF, word embeddings using word vectors, n-grams, rhyme patterns, sentiment analysis, and grammatical properties. These features sets weighted and combined into sparse matrix representations and evaluated using machine learning models such as Random Forest, Multinomial Naive Bayes, and Logistic Regression. Experiments conducted with varying combinations of features sets and included training models on chorus data, non-chorus data, and combined data with weighted contributions from each subset.

The results demonstrated that prioritizing chorus features had positive effect to improve the accuracy. With the best-performing configuration of feature vectors extracted from both chorus and non-chorus sections lyric, where chorus are weighted to have higher effect, achieved around 60-62% accuracy using Random Forest. This work highlights the importance of structural prioritization in lyrics-based classification and provides a scalable framework for integrating structural elements into text-based classification tasks. The findings have implications for music retrieval systems, recommendation engines, and broader text-based classification domains.

Keywords: Machine Learning, Genre Detection from Lyrics, Feature Extraction, Random Forest

ACKNOWLEDGEMENT

I would like to express my gratitude to my supervisor, Dr. Gökhan Kaya, for his guidance and allowing the expression of my ideas. Thanks for all your encouragement.

I also offer my love to my family and my beloved friends who stayed by my side, and my gratitude to all my Professors who have been an example to me with their lives.

R. Hilal Saygin

LIST OF SYMBOLS AND ABBREVIATIONS

Symbol or Abbreviation	:	Explanation
sklearn	:	
nlTK	:	
Naive Bayes, SVM, Random Forest	:	Machine Learning models
tensorflow.keras for LSTM	:	Deep Learning Model tools
Huggingface genius song lyrics	:	Labeled Lyrics Dataset

CONTENTS

Abstract	iv
Acknowledgement	v
List of Symbols and Abbreviations	vi
Contents	viii
List of Figures	ix
1 Introduction and Literature Review	1
1.0.1 Proposal	2
2 Dataset	4
2.1 Initial Dataset Creation	4
2.2 Enhanced Dataset for Structural Analysis	4
2.2.1 Structural Segmentation and Preprocessing Approach	5
3 Experimental Setup	7
3.1 Initial Dataset	7
3.1.1 Preprocessing	7
3.1.2 Features	8
3.1.3 Model Training and Evaluation	8
3.1.4 Discussion of Results	9
3.2 Structured Lyrics Data	9
3.2.1 Preprocessing	9
3.2.2 Feature Sets	9
3.2.3 Discussion of Results	11
3.2.4 Model Training and Evaluation	14
3.2.5 Test Results	14
3.3 Interface	18
4 Results and Conclusions	20
4.0.1 Model Comparison	20
4.1 Conclusion	20
4.1.1 Future Work	20

Bibliography	22
Appendices	22

LIST OF FIGURES

3.1	RF only with other section data, verse,intro,out etc.	12
3.2	Features: tfidf, n-grams. SVD=400. Chorus data features have higher weight.	13
3.3	Features: tfidf, n-grams. SVD=400 reduction. Equal weights for chorus and non-chorus features sets.	13
3.4	RF with all data weighted features, not reduced.	15
3.5	RF with only chorus, unlabeled parts considered.	15
3.6	Chorus labeled only, unlabeled sections excluded, trained with 8k sample set.	16
3.7	RF with all chorus and other section data with additional rhyme features, along with tfidf,n-grams	16
3.8	RF model trained with all chorus and other section data with addition of statistical features to existing feature set	17
3.9	RF with all chorus and other section data with additional rhyme features, along with tfidf,n-grams	17
3.10	RF with combined data, unlabeled parts considered in chorus set, custom features extracted separately and chorus set assigned higher weight.	17
3.11	RF with 10k sample, weighted. All custom features extracted for each chorus and non-chorus, vectorized seperately. After custom vectorization done, features from both subsets reduced to 400 components	18
3.12	Interface for single and multiple songs	19
3.13	Prediction result of single song	19

1. INTRODUCTION AND LITERATURE REVIEW

Music genre classification has been a challenging task over the course of retrieval of music information. With the rapid expansion of digital music collections, automated systems that can organize and classify music based on genre are vital for applications like music recommendation, playlist creation, and music cataloging. The detection of music genres serves for many practical purposes, such as organizing music libraries into user-friendly categories and enhancing music recommendation systems by incorporating genre-based features. While many researches have focused on audio-based features, lyrics-based genre classification and hybrid usage of audio-lyrics has emerged as promising complementary ways, especially given that lyrics contains semantic and structural elements which are difficult to detect from audio alone. Current music streaming platforms such as Spotify process approximately 60,000 new songs daily, creating an urgent need for automated genre classification systems. This text-based approach has several advantages, including computational efficiency and accessibility, as lyrics are often easier to acquire and process compared to audio data. However, the challenge of lyrics-based genre classification lies in effectively capturing both the semantic meaning and structural patterns within song lyrics. Unlike conventional text, lyrics have unique characteristics including repetition, rhyme schemes, and distinct structural elements like intro, verses and choruses. Which is my final concluded approach in this research. I conducted different experimental setups, extracting different feature sets and trained multiple ML models including

1.0.1. Proposal

The scope of the paper proposes a novel methodology for lyrics-based genre classification that emphasizes the structure aware examination of song lyrics. The primary objective is to investigate how structural prioritization, particularly focusing on choruses patterns, affects the predictive accuracy of genre classification models. The following components outline the proposed methodology:

- **Dataset Construction** The dataset is created by combining lyrics data from multiple sources, including Genius.com and HuggingFace. Each song is annotated with structural labels (e.g., *chorus*, *verse*, *intro*), allowing for the segmentation of lyrics into distinct sections. A curated version ensures uniform genre distribution and includes subsets of *chorus* and *non-chorus* data for comparative analysis.
- **Feature Extraction** A custom feature extraction system is implemented to capture diverse aspects of the lyrics:
 - Statistical features, such as word frequency and lexical diversity.
 - Semantic features, including word embeddings using word2vec, Bagof-Words and TF-IDF vectors transformed into matrices.
 - Structural features, such as rhyme patterns and stylistic metrics (e.g., punctuation ratios and sentiment scores).
 - Combined features from the *chorus* and *non-chorus* subsets, weighted differently to analyze their individual and collective contributions.
- **Experimental Setup** The experimental framework involves training and evaluating classification models on various configurations:
 - Models trained on *chorus* data only.
 - Models trained on *non-chorus* data only.
 - Models trained on combined data with weighted feature contributions.
 - Handling of unlabeled songs by treating them as either *chorus* or *non-chorus* subsets.
 - Models trained on combined data with weighted custom feature sets, results cover both SVD reduction applied and feature reduction not applied.

Machine learning models, including Random Forest, Multinomial Naive Bayes, and Logistic Regression, are employed for classification, with accuracy, precision, recall, and F1-score as evaluation metrics.

- **Expected Contributions** This study aims to:

- Demonstrate the significance of structural([Chorus], [Verse] etc.) prioritization in lyrics-based genre classification.
- Provide a detailed analysis of the impact of feature weighting and preprocessing strategies.
- Present a scalable framework for integrating structural information into text-based classification tasks.

The methodology employed to address the lyrics-based genre classification problem outlined in the following structure; first the dataset section, explaining the process of dataset construction, preprocessing, and organization for experiments. the experimental setup chapter covered, feature extraction techniques and various experimental setup I conducted for this research will explore the test cases limitations.

2. DATASET

In this study, multiple datasets were utilized to analyze the impact of different structural elements of song lyrics on genre classification. The datasets were sourced from Kaggle, Genius.com, and HuggingFace, each offering unique features and annotations suitable for conducting experimental setups.

2.1. Initial Dataset Creation

The first dataset was sourced from Kaggle’s ”Lyrics from 79 Genres,” which comprised two files containing information on artists, songs, and lyrics. These files were merged based on common columns, and only the relevant fields—namely, genre and lyrics—were retained for analysis. While the dataset initially included multilingual lyrics, non-English entries were excluded due to an insufficient number of samples for meaningful training and testing. Similarly, genres with very few song examples were omitted from the study to ensure adequate representation across genres.

I applied comprehensive preprocessing operations on this dataset. Non-alphabetic characters, numbers, and punctuation marks were removed to normalize the textual data. Common English stopwords were eliminated using the stopwords package from the Natural Language Toolkit (NLTK). Additionally, I performed lemmatization using `nltk.stem` to consolidate words with similar roots into a single representative form. This step was specifically chosen to reduce noise in the data and enhance the accuracy of statistical features, such as the counts of most common and unique words etc. The established feature sets will be detailed further.

2.2. Enhanced Dataset for Structural Analysis

To evaluate the impact of different structural elements of lyrics on genre classification performance, it was essential to utilize a dataset that provided both lyrics data annotated with structural information (e.g., section labels such as verse, chorus, intro) and genre tags. For the construction process of this enhanced dataset format, I combined data from multiple lyrics sources. The resources for data included:

- **Genius.com:** Lyrics data scraped from Genius.com contained detailed section-level annotations, such as *intro*, *chorus*, and *verse*. These annotations were critical for identifying the structural composition of songs.

- **HuggingFace:** Additional lyrics datasets with similar structural annotations were sourced from HuggingFace to supplement the dataset.

The resulting data had lyrics for 5 main genres (rock, pop, country, rap and rb). To achieve uniformity across genres, the dataset curated comprising around 8,683 entries per genre for testing purposes. Subsequently, larger versions were created with 10,000 and 20,000 entries per genre to evaluate the effect of dataset size on model performance. However, the 20,000-entry version introduced significant noise, making training and testing challenging. As a result, the study proceeded with the 10,000-entry-per-genre version as the optimal balance between data size and model performance.

2.2.1. Structural Segmentation and Preprocessing Approach

To analyze the impact of repetitive and thematically central song components, I divided the complete dataset into two chunks.

- **Chorus Subset:** Extracted lyrics labeled under the *chorus* section, focusing on repetitive and central components of songs.
- **Non-Chorus Subset:** Comprised all other sections, such as *verse*, *intro*, and *outro*.

Moreover, within the scope of the experiments I did with this version of structural dataset, preprocessing step didn't include removal of content. All non-alphabetical characters used as features. I decided to pursue with this option due to the following findings;

- Two models were trained using the chorus and non-chorus subsets to compare preprocessing approaches:
 1. **Model Without Content Removal:** Retained all characters, stopwords, numbers, and punctuation. Instead, custom features were derived from statistical properties of the lyrics, such as the ratios of numbers, punctuation marks, and parts of speech (e.g., verbs and nouns).
 2. **Model With Preprocessing:** Applied the same preprocessing steps as the initial dataset, including removal of non-alphabetical characters, stopwords, and lemmatization.
- The first model, which utilized unprocessed data with custom features, demonstrated superior accuracy compared to the second model. This finding highlighted the predictive value of features derived from raw structural properties of lyrics.

Subsequent experiments involved training various models on the chorus and non-chorus subsets using:

- Weighted contributions from each subset.
- Custom feature extraction like rhyme pattern, word statics, word3vec etc. techniques tailored to the structural properties of lyrics.

These experiments provided valuable insights into the role of structural elements in genre classification and informed the development of the final models used in the study.

3. EXPERIMENTAL SETUP

This section outlines the methodology used to evaluate the performance of genre classification models based on song lyrics. The details of dataset utilized in these experiments is described in first part of Dataset section. Here, we focus on the specific preprocessing techniques, feature extraction methods, and experimental configurations applied to assess the impact of structural elements on classification performance. First case covers the setup and results of unstructured dataset focussing on preprocessing of lyrics data. Second setup focused on the effects of weighted features extracted from structural sections(chorus, verse etc) of lyric data.

3.1. Initial Dataset

The dataset used for the first experimental setup was sourced from Kaggle’s ”Lyrics from 79 Genres.” which the details of the data structure covered in Dataset section.

The final dataset consisted of lyrics labeled across six genres. Figure ?? illustrates the song counts for each genre in the dataset.

3.1.1. Preprocessing

To prepare the data for feature extraction and model training, comprehensive preprocessing steps were applied to normalize the text:

- **Non-Alphabetic Character Removal:** All non-alphabetic characters, numbers, and punctuation marks were removed.
- **Stopword Removal:** Common English stopwords were eliminated using the NLTK *stopwords* package.
- **Lemmatization:** Words were converted to their root forms using *nltk.stem*, grouping different tenses and forms of the same word into a single representative token.

These preprocessing steps reduced noise in the dataset and ensured that the extracted features were meaningful for genre classification.

3.1.2. Features

Following preprocessing, various features were extracted from the lyrics to capture the statistical and contextual properties of the data:

- **Total Word Count:** The total number of words per song after lemmatization.
- **Top 10 Most Common Words:** The most frequently occurring words for each genre were identified as a feature set. Figure ?? visualizes the top words for each genre.
- **N-Grams:** Bi-grams (2-grams) and tri-grams (3-grams) were computed to capture phrase-level patterns unique to each genre.
- **TF-IDF Features:** Term Frequency-Inverse Document Frequency (TF-IDF) vectors were generated to model word relationships and importance within the dataset.

The extracted features were then used as input for machine learning models to evaluate their classification performance.

3.1.3. Model Training and Evaluation

Three machine learning models were employed to classify song lyrics into genres based on the extracted features:

- **Multinomial Naive Bayes (MNB):** A probabilistic model suitable for text data.
- **Decision Tree (DT):** A simple and interpretable model used as a baseline.

The models were trained and tested on the processed dataset, with accuracy serving as the primary evaluation metric. Table ?? presents the classification accuracy of each model:

- Multinomial Naive Bayes: **50.6%**
- Decision Tree: **37.6%**

The results, visualized in Figure ??, highlight the relative performance of the models. Naive Bayes performed better compared to Decision Tree model exhibited lower accuracy, likely due to its sensitivity to noisy data and lack of generalization capability.

3.1.4. Discussion of Results

The initial experimental setup revealed that preprocessing and feature extraction significantly influenced model performance. While the overall accuracies were modest, they provided a baseline for subsequent experiments that incorporated structural analysis and advanced feature engineering. The findings also highlighted the limitations of traditional models like Decision Trees in handling complex, high-dimensional text data.

3.2. Structured Lyrics Data

To enhance the performance of genre classification, I focus on prioritizing chorus patterns of song lyrics. This involved constructing a new dataset with section-level annotations and designing multiple test cases to evaluate the impact of features extracted from structural elements on classification accuracy. Varying combinations of subsets of data used to develop custom extraction features and train. Subset to extract feature sets included;

only chorus parts of lyrics,

subset containing data from other non-chorus sections,

and the combination of both with weighted and custom features.

3.2.1. Preprocessing

Initially, tests were conducted on the chorus subset without applying preprocessing steps such as stopword removal, punctuation removal, or lemmatization.

These unprocessed lyrics were retained to explore the contribution of non-alphabetic characters and grammatical structures to classification performance.

In comparison, additional tests were performed on lemmatized and cleaned lyrics, where the removal of non-alphabetic characters and stopwords was applied.

The results demonstrated a decrease in accuracy with lemmatized and processed data, leading to the decision to use unprocessed lyrics for subsequent experiments.

3.2.2. Feature Sets

In the scope of my study I generated a comprehensive set of features that combine word relations, statistical, linguistic, stylistic, rhymes and semantic properties of the lyrics. These features enable a robust representation of the textual data for genre

classification. The key components of the feature set are as follows:

The following features were extracted from the dataset to capture the statistical, linguistic, and structural properties of song lyrics:

- **TF-IDF:** Term Frequency-Inverse Document Frequency vectors were generated, to measure the importance of words in a document relative to a collection of texts. Limited to the top 3,000 features to balance complexity and computational efficiency.
- **N-Grams:** Bi-grams and tri-grams were computed to capture contextual patterns in the text.
- **Custom Features:** To conduct a comprehensive analysis of effects of statistical, linguistic, and structural properties of song lyrics, custom extraction system developed, including;
 - **Statistical features:**
 - * `most_common_count`,
 - * `most_common_ratio`,
 - * `unique_words`,
 - * `total_words`,
 - * `unique_ratio`,
 - * `average sentence length`.
 - **Stylistic Features:** Reflects stylistic tendencies unique to specific genre
 - * Number of exclamation marks.
 - * Number of question marks.
 - * Proportion of uppercase letters.
 - * Proportion of punctuation marks.
 - * Number of repeated lines in the text.
 - **Rhyme Features:** Highlights rhyming aspects of genres with rhyme-heavy lyrics, such as rap.
 - * Analyzes rhyming patterns by grouping words based on their phonetic endings.
 - * Counts distinct rhyming patterns where multiple words share the same rhyme key.
 - * Vectorized as numeric feature indicating the number of rhyming patterns in a text.
 - **Word Embeddings:** It enabled us to have deeper semantic information and word similarities.

- * Word2vec set consist of embeddings represented as dense vectors.
- * Aggregates word embeddings by averaging across tokens in the text.
- * Dense 300-dimensional vector for each single song text.
- **Bag of Words (BoW):** Converts text into a vector of word counts, where each dimension corresponds to a specific word in the vocabulary.
 - * Captures word frequency while ignoring word order.
 - * Sparse matrix where rows represent texts and columns represent unique words, generated using *CountVectorizer*.

All extracted features were converted into sparse matrix representations using Scikit-learn utilities. Feature reduction was performed using Truncated Singular Value Decomposition (TruncatedSVD) with 400 components, which was found to provide a balance between training efficiency and classification performance. It was found that after training models with different component counts, the best classification report obtained from by reducing to components between 400-500 feature size. The features were extract from 10k sample dataset were it contained 10k songs for each of 5 genres which; rock, pop, county, rap, RB.

3.2.3. Discussion of Results

The experimental setup involved the following test cases to assess the impact of structural elements and preprocessing approaches:

1. Train on Chorus Data:

- Training exclusively on the *chorus* subset, focusing on repetitive and thematically central elements of the lyrics.
- Both processed (lemmatized and cleaned) and unprocessed (raw text) versions of the chorus subset were tested. Results indicated that unprocessed data outperformed processed data, suggesting that stylistic elements and non-alphabetic characters contributed significantly to classification.
- Sing lyrics which didnt have sub-section annotation were considered to be in shorus set. Extraction included TF-IDF (3000 features), n-grams (1-3).
- Detailed performance metrics (accuracy, precision, recall, and F1-score) for this test case are shown in Figure 3.5.

2. Train on Non-Chorus Data:

- Models were trained exclusively on the *non-chorus* subset, containing sections such as *verse*, *intro*, and *outro*.

- **Preprocessing:** As with the chorus subset, both processed and unprocessed versions of the data were tested. Results showed a decrease in accuracy compared to the chorus subset, highlighting the lower predictive power of non-chorus sections.
- **Feature Set:** The same feature set was applied as in the chorus data tests.

Random Forest Model with Tfidf5k - Not reduced
RESULTS WITH NON-CHORUS

Accuracy: 59.48

Classification Report:

	precision	recall	f1-score	support
country	0.55	0.41	0.47	3223
pop	0.68	0.15	0.25	1910
rap	0.67	0.85	0.75	7311
rb	0.53	0.66	0.59	5585
rock	0.51	0.29	0.37	2602
accuracy			0.59	20631
macro avg	0.59	0.47	0.48	20631
weighted avg	0.59	0.59	0.57	20631

Figure 3.1: RF only with other section data, verse,intro,out etc.

3. Combined Data with Weighted Features:

- Models were trained on a combined dataset, integrating both *chorus* and *non-chorus* features. The contribution of each subset was adjusted through feature weighting.
- **Weighting Strategy:**
 - Case 1: Chorus features assigned higher weights than non-chorus features.
 - Case 2: Non-chorus features assigned higher weights than chorus features.
- **Results:** Assigning higher weights to chorus features consistently improved accuracy, reaffirming their importance in genre classification. The better-performing configuration achieved 61.84% accuracy.

```

Accuracy: 0.618422908731838
Classification Report:

```

	precision	recall	f1-score	support
country	0.60	0.54	0.57	1316
pop	0.78	0.29	0.43	914
rap	0.70	0.80	0.75	1901
rb	0.54	0.75	0.62	1831
rock	0.64	0.45	0.53	1127
accuracy			0.62	7089
macro avg	0.65	0.57	0.58	7089
weighted avg	0.64	0.62	0.61	7089

Figure 3.2: Features: tfidf, n-grams. SVD=400. Chorus data features have higher weight.

```

Accuracy: 0.6126548376297288
Classification Report:

```

	precision	recall	f1-score	support
country	0.58	0.55	0.56	3334
pop	0.73	0.27	0.40	2288
rap	0.68	0.79	0.73	4607
rb	0.55	0.75	0.63	4664
rock	0.63	0.46	0.53	3029
accuracy			0.61	17922
macro avg	0.63	0.56	0.57	17922
weighted avg	0.63	0.61	0.60	17922

Figure 3.3: Features: tfidf, n-grams. SVD=400 reduction. Equal weights for chorus and non-chorus features sets.

4. Handling Unlabeled Songs:

- Songs without structural annotations were alternately treated as part of the *chorus* subset or to be excluded. These tests evaluated the system's ability to generalize when faced with incomplete or unstructured data.
- Results:
 - When unlabeled songs were treated as *chorus*, the accuracy shifted from 83% to 72%. This accuracies obtained from Random forest model trained on 8k sample set.
- Implications: Including unlabeled songs increased the model's generalizability but introduced noise, slightly reducing classification performance compared to the chorus only dataset. But unlabeled song, that didnt have chorus related information cannot be excluded. Thus iin the further test all chorus subset included the unlabeled parts as well.

Including unlabeled songs reduced accuracy compared to models trained solely on chorus sections. However, this approach expanded the system's applicability by including songs with complete but unlabeled lyrics, ensuring no data was skipped.

When chorus features were assigned higher weights, the models demonstrated

improved accuracy, underscoring the significance of thematic and repetitive elements in the chorus. Random Forest emerged as the best-performing model across these cases, outperforming Multinomial Naive Bayes, Logistic Regression, and Decision Trees. Consequently, subsequent experiments prioritized Random Forest for further testing.

3.2.4. Model Training and Evaluation

The following machine learning models were implemented to evaluate the extracted features:

- **Multinomial Naive Bayes (MNB):** A probabilistic model well-suited for text data.
- **Random Forest (RF):** An ensemble learning method that combines multiple decision trees for improved performance.
- **Logistic Regression (LR):** A linear model that performs well on high-dimensional data.
- **Decision Trees (DT):** A simple, interpretable baseline model.

Among these models, Random Forest consistently demonstrated superior accuracy and was selected for subsequent experiments. Accuracy served as the primary evaluation metric, with additional metrics like training time and computational efficiency considered for feature reduction experiments.

3.2.5. Test Results

The concluded results and some reports belonging to models trained with varying data and feature sets can be found as;

- The inclusion of raw text features (unprocessed data) resulted in higher accuracy compared to lemmatized and stopwords-removed data.
- Weighted combinations of chorus and non-chorus features improved classification performance, with higher weights for chorus features.

Details displayed in Figure 3.4 for weights assigned.

- Random Forest consistently outperformed other models, achieving the highest accuracy of **63%** when all data used. But the accuracy decreased when reduction applied on the extracted features reduced using SVD to **61%**. But without reduction, the prediction time was high thus SDV must be used.

Figure 3.10 shows model results when SVD reduction applied.

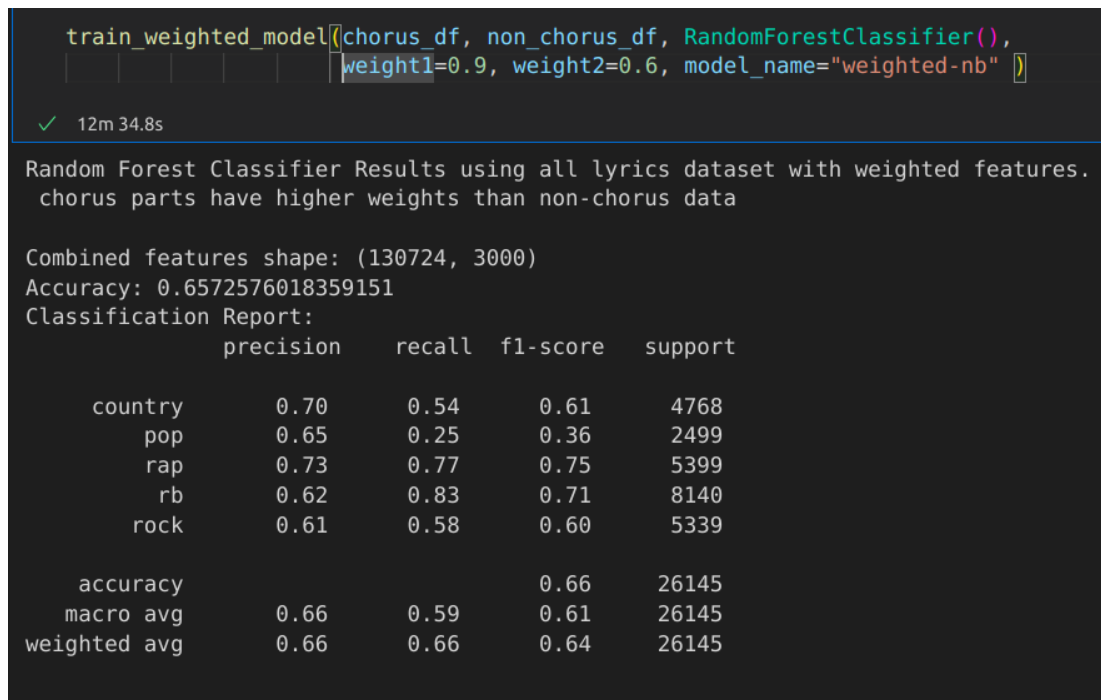


Figure 3.4: RF with all data weighted features, not reduced.

- Including unlabeled songs reduced accuracy compared to models trained solely on chorus sections but enhanced the system's generalizability by retaining songs without structural annotations. Figure 3.6
- Training on chorus and unlabeled part data gave the highest accuracy with **72%** for Random Forest. Reference Figure 3.5

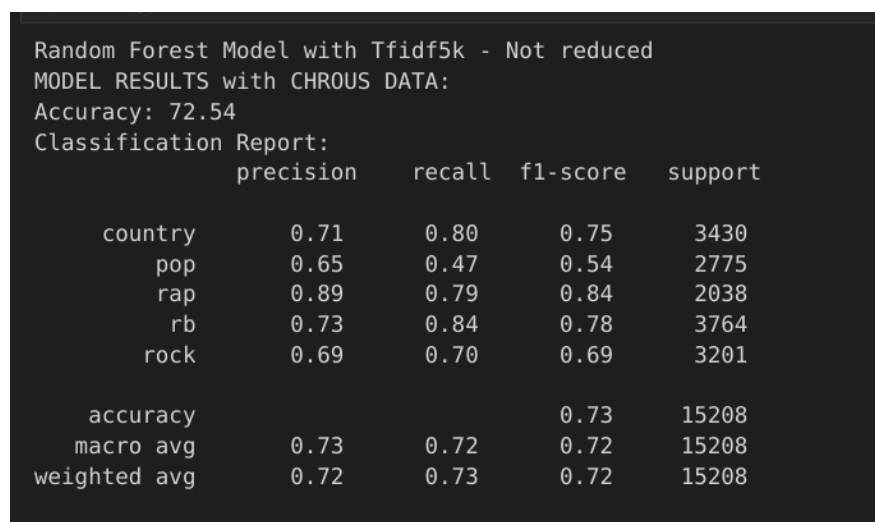


Figure 3.5: RF with only chorus, unlabeled parts considered.

- Addition of rhyme features increased accuracy considerably better than statistical

features. Figure 3.9 show results for tfidf,n-grams and rhyme features with chorus data results.

```

Accuracy: 82.99
Classification Report:

```

	precision	recall	f1-score	support
country	0.92	0.72	0.80	1869
pop	0.97	0.57	0.72	456
rap	0.98	0.73	0.84	737
rb	0.79	0.95	0.86	3452
rock	0.80	0.83	0.81	2258
accuracy			0.83	8772
macro avg	0.89	0.76	0.81	8772
weighted avg	0.84	0.83	0.83	8772

Figure 3.6: Chorus labeled only, unlabeled sections excluded, trained with 8k sample set.

```

Accuracy: 0.66472883041383
Classification Report:

```

	precision	recall	f1-score	support
0	0.66	0.62	0.64	2586
1	0.53	0.38	0.45	2016
2	0.87	0.65	0.75	1323
3	0.67	0.87	0.75	4037
4	0.67	0.63	0.65	3111
accuracy			0.66	13073
macro avg	0.68	0.63	0.65	13073
weighted avg	0.66	0.66	0.66	13073

Figure 3.7: RF with all chorus and other section data with additional rhyme features, along with tfidf,n-grams

Figure 3.8 shows when model trained on combination of chorus and non-chorus data with custom extracted statistical features along with tfidf.


```

Random Forest Results - with custom features

Accuracy: 0.5885911318129575
Classification Report:

```

	precision	recall	f1-score	support
0	0.63	0.38	0.47	5822
1	0.50	0.31	0.38	2669
2	0.70	0.72	0.71	7733
3	0.53	0.79	0.64	10261
4	0.59	0.42	0.49	6577
accuracy			0.59	33062
macro avg	0.59	0.52	0.54	33062
weighted avg	0.60	0.59	0.57	33062

Figure 3.8: RF model trained with all chorus and other section data with addition of statistical features to existing feature set

```

Accuracy: 0.66472883041383
Classification Report:

```

	precision	recall	f1-score	support
0	0.66	0.62	0.64	2586
1	0.53	0.38	0.45	2016
2	0.87	0.65	0.75	1323
3	0.67	0.87	0.75	4037
4	0.67	0.63	0.65	3111
accuracy			0.66	13073
macro avg	0.68	0.63	0.65	13073
weighted avg	0.66	0.66	0.66	13073

Figure 3.9: RF with all chorus and other section data with additional rhyme features, along with tfidf,n-grams

```

Accuracy: 0.6126548376297288
Classification Report:

```

	precision	recall	f1-score	support
country	0.58	0.55	0.56	3334
pop	0.73	0.27	0.40	2288
rap	0.68	0.79	0.73	4607
rb	0.55	0.75	0.63	4664
rock	0.63	0.46	0.53	3029
accuracy			0.61	17922
macro avg	0.63	0.56	0.57	17922
weighted avg	0.63	0.61	0.60	17922

Figure 3.10: RF with combined data, unlabeled parts considered in chorus set, custom features extracted separately and chorus set assigned higher weight.

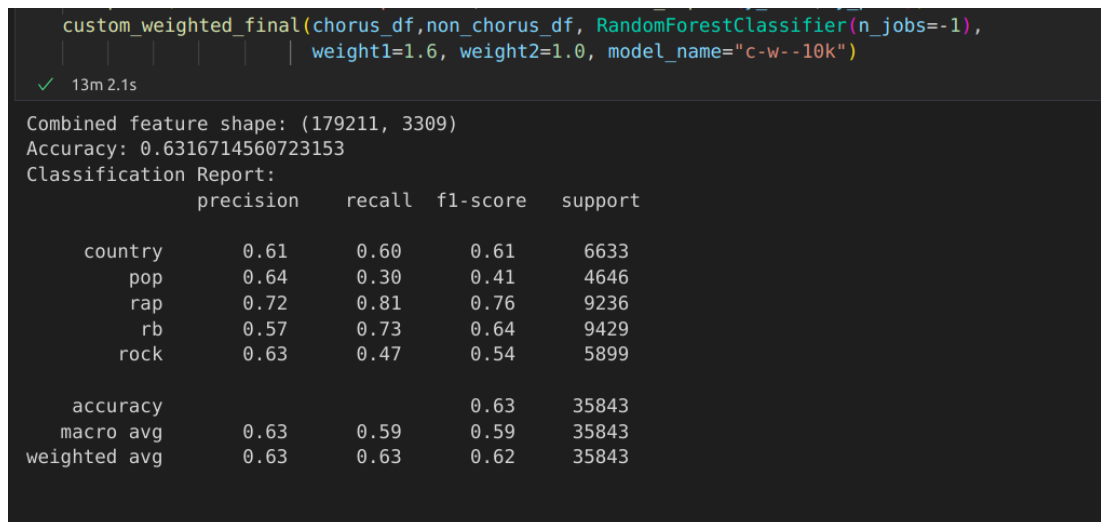


Figure 3.11: RF with 10k sample, weighted. All custom features extracted for each chorus and non-chorus, vectorized separately. After custom vectorization done, features from both subsets reduced to 400 components

3.3. Interface

The interface developed for the genre detection system provides an user-friendly way to test and evaluate the model's performance. It has two main functionalities: (1) Single Input Prediction, where user input song lyrics into a text field and obtain the predicted genre in real time; and (2) Batch Evaluation, which allows users to upload a CSV file containing lyrics and their associated true genres for bulk testing and classification report displayed. The results are evaluated against the ground truth to provide performance metrics. The interface supports seamless interaction, enabling both individual predictions and large-scale evaluation for thorough performance analysis. The clean layout and clear instructions ensure accessibility for users of all technical backgrounds.

Genre Detection Demo


1. Predict Genre for a Single Lyrics Input

Enter your lyrics here:

Predict Genre

2. Evaluate on a Test Set (CSV)

Upload a CSV file with two columns: 'lyrics' and 'tag'

 Drag and drop file here
Limit 200MB per file • CSV

Browse files


 samp.csv 0.7MB ×

Figure 3.12: Interface for single and multiple songs

1. Predict Genre for a Single Lyrics Input

Enter your lyrics here:

I never could
Baby come here love hasn't disappeared you're just feeling low
And let me tell you my darling it's a feeling I know
It don't mean too much we just got out of touch well that's easy to change
Because loving you isn't too hard to arrange and anyway
[Chorus]
I never could never would never will ever kill what's between us...",country

Predict Genre

Predicted Genre: country

Figure 3.13: Prediction result of single song

4. RESULTS AND CONCLUSIONS

This section presents the findings from the experimental setups, focusing on the impact of structural prioritization, feature extraction, and preprocessing techniques on genre classification performance. The performance metrics include accuracy, precision, recall, and F1-score, as reported for each test case.

4.0.1. Model Comparison

The Random Forest model trained outperformed in most cases compared to performance of Gaussian, Multinomial Naive Bayes, Logistic Regression and Decision Trees across all test cases. Its ability to handle high-dimensional and sparse features, such as TF-IDF and BoW, contributed to its superior performance. A summary of model performance metrics is provided in Table ??.

4.1. Conclusion

Key findings include:

- **Structural Prioritization:** Chorus sections were found to be more predictive than non-chorus sections, with models trained exclusively on chorus data achieving the highest accuracy.
- **Preprocessing:** The use of unprocessed lyrics retained stylistic and structural features, contributing to better performance than lemmatized and cleaned data.
- **Feature Integration:** Combining structural, semantic, and linguistic features in a weighted manner enhanced the classification results, particularly when chorus features were prioritized.
- **Model Performance:** Random Forest emerged as the most effective model, %.

The study also explored the inclusion of unlabeled songs, demonstrating the trade-off between generalizability and classification performance. While incorporating unlabeled data slightly reduced accuracy, it allowed for a broader application of the system to incomplete datasets.

4.1.1. Future Work

Expanding the scope of the research for future work can be:

- Expanding the dataset to include additional genres and more diverse structural annotations. However larger dataset will require larger memory due to computational consumption needed to generate custom features in format of matrices.
- Incorporating deep learning models, such as LSTM,CNN to capture contextual relationships in lyrics.
- Developing interpretability methods to create combinations of subsets to better understand the contributions of structural elements and specific features to classification decisions.

The findings of this study provide a robust framework for integrating structural information into text-based classification tasks, with potential applications in music recommendation systems, music retrieval, and other domains requiring textual analysis.

APPENDICES

1. Wang, I. (2023). Music feature extraction and classification algorithm based on deep learning. *IEEE Access*, 10, 56923–56930. <https://doi.org/10.1109/ACCESS.2023.10059542>
2. CoreStrata Insights. (2023). Comparative study on BERT and RoBERTa-based sentiment analysis. Retrieved from <https://www.corestratai.com/post/comparative-study-on-bert-and-roberta-based-sentiment-analysis>
3. Kaggle. (2012). Million Song Dataset Challenge data. Retrieved from <https://www.kaggle.com/c/msdchallenge/data>
4. Towards Data Science. (2023). Fine-tune smaller transformer models for text classification. Retrieved from <https://towardsdatascience.com/fine-tune-smaller-transformer-models-text-classification-77cbbd3bf02b>
5. Hugging Face. (2023). NLP with transformers. Retrieved from <https://github.com/nlp-with-transformers>
6. Zhang, D. (2021). Music feature extraction and classification algorithm based on deep learning. *Scientific Programming*, 2021. <https://doi.org/10.1155/2021/5671384>
7. Mayerl, M., Brandl, S., Specht, G., Schedl, M., Zangerle, E. (2022). Verse versus Chorus: Structure-aware Feature Extraction for Lyrics-based Genre Recognition. *International Society for Music Information Retrieval Conference*.
8. Chai, W., Vercoe, B. (2003). Structural analysis of musical signals for indexing and thumbnailing. In *Proceedings of the 5th International Conference on Multimedia and Expo (ICME)* (Vol. 1, pp. 401–404). IEEE. Retrieved from <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=e48ca0b7a1796747cd3e22ef0586bded9faaf4ae>
9. CS224N, S., Leszczynski, M., Boonyanit, A., Dahl, A. (2021). Music Genre Classification using Song Lyrics.