Classifying Genres from Song Lyrics with DistilBERT

Nikiforidis Konstantinos, Verros Stylianos, Georgilas Stylianos
Aristotle University of Thessaloniki

Abstract— **The challenge of comprehending song lyrics is addressed via natural language processing (NLP) in this exercise, with special focus on genre classification. Our experiments, utilizing a DistilBERT model for genre classification, provide positive findings with 67% accuracy. To further evaluate model performance, DistilBERT was compared with other BERT-based models, including BERT-uncased and RoBERTa, as well as a baseline logistic regression model. The final assessment considered both predictive accuracy and computational efficiency, highlighting the trade-offs between model size and classification performance.**

I. INTRODUCTION

Since it affects our emotions, provides us with entertainment, and encourages understanding between cultures, music is essential to our life in the modern world. Furthermore, genre-based song classification has beneficial implications for personalized content delivery and recommendation systems. With the exponential growth of digital music libraries, automatic music genre classification has become an essential tool for organizing, retrieving, and recommending music efficiently. Traditional methods for genre classification relied heavily on handcrafted features such as tempo, pitch, and timbre, often requiring significant domain expertise. However, with advancements in machine learning and deep learning, data-driven approaches have emerged as powerful alternatives, leveraging large datasets to learn intricate patterns within audio signals.

While many existing approaches utilize audio features such as spectrograms or Mel-frequency cepstral coefficients (MFCCs) for classification, our work takes a different approach by relying solely on song lyrics. By focusing on textual data instead of complex audio processing, we aim to simplify the model while still capturing the linguistic and thematic patterns that differentiate music genres. This approach not only reduces computational complexity but also allows for genre classification in scenarios where only lyrics are available, such as textual music databases or lyric-based recommendation systems.

In order to solve this issue, we plan to extensively examine and evaluate song lyrics using natural language processing (NLP) methods. We apply a DistilBERT model—a condensed form of Bidirectional Encoder Representations from Transformers—for the classification challenge. A DistilBERT tokenizer was used to encode the music lyrics. We divided the ”genre” into five genre classifications for our dependent variables. Next, we forecast the four genres using the generated lyrics tokens, which are based on the DistilBERT model that has already been trained (distilbert-base-uncased).

The outcomes of our trained models and further information regarding the Genius Dataset are then provided. Lastly, closing thoughts are provided in the final part.

The structure of the paper is as follows: Information about previous related research is given in the following section. Following that, we go into greater detail about our methodology and describe the techniques we employ to solve the music categorization challenge.

II. Related Work

The next section provides a quick analysis of related work. The issue of categorizing song lyrics according to their respective genres has not been extensively addressed in the literature. To classify song lyrics, however, the majority of studies used traditional methods like machine learning models or basic neural networks.

A. Application’s perspective

From the angle of the application, Khrulchenko (2020) has made significant contributions to the classification of song lyrics. Khrulchenko tried to categorize the music genres using conventional machine learning methods as gradient descent, decision trees, linear SVC, and logistic regression in his ”Pop, Rap, and Heavy Metal - lyrics classifier” approach. The problem, though, is that conventional approaches frequently struggle to capture the intricate patterns and semantics of song lyrics. As a result, predicting the genre of music frequently becomes inaccurate. Using cutting-edge natural language processing models, namely DistilBERT, we hope to overcome the drawbacks and difficulties of traditional machine learning techniques, such Khrulenko (2020).

B. Methodology’s perspective

In particular, Rajendran, Pillai, and Daneshfar ([1]) were our main source of inspiration. NLP models were also used by Rajendran et al. ([1]) to categorize song lyrics. To forecast the emotional tone of a song’s lyrics, they used a LyBERT-based model. They accomplished this by classifying the lyrics into four groups: happy, relaxed, angry and sad. We take it a step farther than his work by adding other characteristics, including genre, based solely on the song lyrics.

By expanding and improving on the methods based on Khrulenko (2020) and Rajendran, Pillai, and Daneshfar ([1]), we hope to contribute to the body of existing literature. We intend to get over the drawbacks of conventional techniques and improve the analysis of song texts by utilizing NLP technologies and including new features.

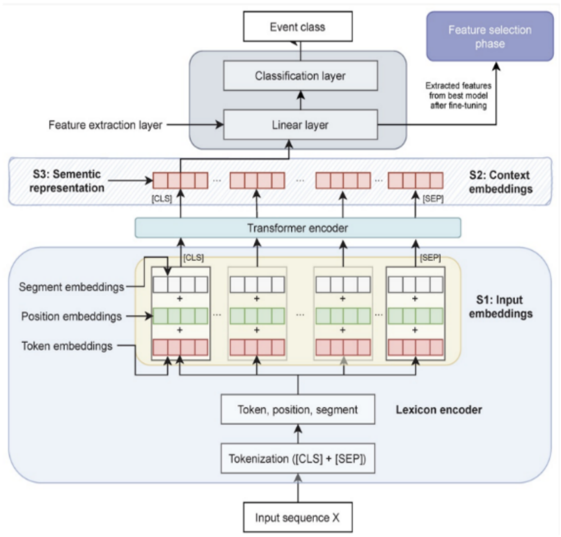

Fig. 1. The proposed feature extraction model

III. Methodology

We utilize DistilBERT, a transformer model that is small, light, fast, affordable, and educated by distilling BERT basis, to identify genre based on song lyrics. We are able to decipher the complex connections contained inside songs thanks to the pre-trained distillation base-uncased model. The research of [2] served as the basis for this part.

A. *Distilled BERT for Feature Extraction*

We explore DistilBERT's design and methods in more detail below. The proposed feature extraction model, which is based on DistilBERT, processes input sequences—tweets—in many steps. The architecture (Figure 1) uses DistilBERT to transform input sequences (X) into embedding vectors (S1) for individual words. DistilBERT's transformer encoder uses a self-attention method to generate contextual embeddings (S2) that include the contextual information of each word. Concatenating these contextual embeddings results in a single vector (S3) that represents the semantic content of the entire tweet. The concatenated vector (S3) is fed into a fully connected layer, which produces an output vector of size 'd,' where 'd' is the number of neurons. A classification layer is then employed to optimize the previously learned DistilBERT specifically for genre and success identification. This classification layer's goal is to predict the genre and success class associated with each incoming tweet sequence.

B. *Lexicon Encoder*

Each tweet, which is represented by a sequence of tokens of length s, is processed by a multi-layered neural network. X = x1,..., xs is an input made up of tokens. Word, segment, and positional information are represented by the corresponding embedding vector for each token. According to the encoding approach recommended by [3], a special token [CLS] is inserted as the first token (x1), and the [SEP] token is placed at the end of the sequence. The lexical encoder generates the embedding vectors for X. This involves inserting the word, segment, and positional embeddings for each token in the sequence.

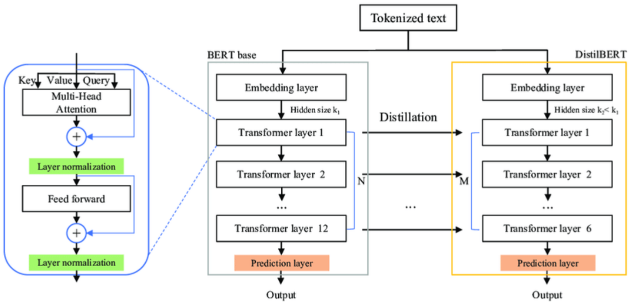

Fig. 2. The DistilBERT model architecture and components

C. *Transformer Encoder*

Figure 2 illustrates the generic DistilBERT model architecture, which is the subject of our next discussion. To transform input vectors (S1) into contextual embedding vectors, we employ a multilayer bidirectional transformer encoder that has already been trained. DistilBERT reduces the BERT base model (bert-base-uncased) parameters by using knowledge distillation. It maintains over 95% of BERT's performance while operating 60% faster and using 40% fewer parameters than bert-base-uncased ([4]). By using a more condensed model called DistilBERT, which has six transformer layers rather than twelve, one may approximate the entire output distributions of BERT via distillation. DistilBERT outperforms the original BERT base model, which had 110 million parameters, by a significant margin with 66 million trainable parameters. The 16 GB of training data for DistilBERT is sourced from the Toronto books corpus and the English Wikipedia, same like the BERT base training data. DistilBERT is trained using a batch size of 16 and gradient accumulation. This approach collects gradients from multiple mini-batches before adjusting the parameters to boost efficiency. Notably, segment embeddings and next sentence prediction (NSP) are not included in the training methodology. Together, these changes streamline the training procedure and enable DistilBERT to be a practical and effective BERT replacement in a variety of NLP applications.

IV. Fine-Tuning on Genre and Success Classification Task

In order to prevent overfitting, we integrate L2 regularization into our training loop by initializing an AdamW optimizer with a learning rate of 1e-5 and a weight decay term of 0.01. Furthermore, during training, we calculate the loss between predicted logits and ground truth labels using the CrossEntropyLoss. Using the contextual embedding that the [CLS] token learned from the input tweet X, or semantic representation S3, the ultimate goal is to formulate a multi-class classification problem. The goal is to forecast the probability that c will belong to a specific class, ”c,” which represents a genre.

$$\text{Softmax}(W^T \cdot X) = P(c \mid X) \tag{1}$$

Equation (1) outlines how to calculate this categorization probability using the Softmax function. The Softmax function facilitates the conversion of raw scores into probability distributions, enabling the model to predict the likelihood that X will be included in a particular genre or success.

V. Experiment

A. *Dataset*

We wish to use a dataset that includes data as recent as 2022 that was scraped from Genius in order to obtain relationships and insights in the field of music. Poetry, music, and even novels (albeit mostly songs) can be posted and annotated here. More than 5 million lyric entries in numerous languages are contained in the collection. We use the ”Kaggle” website to get the data set. We just wish to concentrate on English-language song lyrics for our project. Additionally, we wish to use a randomly selected sample for our models due to computational limitations. Each of the dataset’s primary variables—title, artist, lyrics, genre, year, and page views—is essential to comprehending and evaluating the musical data.

Pop, rock, rb, rap, miscellaneous, and country are all included in the dataset’s ”tag”-designated genre. Since the majority of non-musical items are classified as ”misc,” we removed every row that had the tag value ”misc.” Additionally, we discovered that 1 is the column year’s lowest value. We only included songs older than 1960 in our final datasets for a more current study.

B. *Descriptive Analysis and Preprocessing*

We began by gathering a distinct dataset from Genius. Our descriptive analysis uses a dataset of 50,000 observations. Following preliminary data inspection and exploration, an exploratory data analysis (EDA) was performed. The EDA focuses on comprehending the dataset’s composition, particularly genre distribution. Our findings indicate that pop and rap are the most popular genres. We studied the most common words in various genres. The term ’love’ is commonly used in pop, rock, country, and R&B genres. Rap musicians, on the other hand, use unique vocabulary in their lyrics. Rap lyrics frequently feature words like ’bitch’ and ’fuck’.

We visualized song lyric length to identify potential patterns that could impact model training. During our EDA, we found that lyrics for rap songs had the highest average word count and text length, while rock songs have the lowest. We identified a few songs with low word counts and classified them as non-serious. We established a filter for lyrics with word counts over 100 words. To forecast genres, we used a dataset of 50,000 songs with over 150 words of lyrical material and at least 1,000 views to ensure basic quality standards. This method effectively filtered out non-song entries and outliers. We balanced the dataset across genres and adjusted sample sizes to improve prediction accuracy. In the genre use case, we only examine songs with over 1,000 views to prioritize popular songs. Pre-processing steps include cleansing data for special characters in song lyrics. The lyrics maintain their fundamental structure and often include song metadata in square brackets in the center.

Genre	Precision	Recall	F1-score	Support
Rap	0.83704	0.81817	0.82750	7496
Pop	0.44521	0.44140	0.44330	7492
Rock	0.62794	0.60304	0.61524	7492
Country	0.75577	0.79560	0.77517	7495
R&B	0.66392	0.67587	0.66984	7497
Accuracy			0.66684	37472
Macro Avg	0.66597	0.66682	0.66621	37472
Weighted Avg	0.66600	0.66684	0.66624	37472

TABLE I

Classification report results for genre prediction with DISTILBERT without GELU.

C. *Evaluation Metrics*

We evaluate our song lyrics categorization task using accuracy and F1-scores. According to [5], accuracy is defined as the ratio of correctly identified genres and success/failure to the total number of songs. Equation (2) defines accuracy using the acronyms TP (true positive), TN (true negative), FP (false positive), FN (false negative), and TN.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{1}$$

The F1-score can be interpreted as a harmonic mean of the precision and recall. Where precision is defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

recall as:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

and F1-score as

$$\text{F1 Score} = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{4}$$

VI. Model Results

Distinguishing between pop and rock lyrics was difficult due to their similar themes and artistic nuances. We used many strategies, including two-step classification and emphasis weight modifications, to improve the model’s accuracy across genres. The ultimate approach involved fine-tuning a standard model and optimizing hyperparameters (around 67M). The fine-tuning increased our model’s discernment, resulting in a 66% genre categorization accuracy across five classes. The genre classification for predicting rap has a high F1-score of 0.82, which is not surprising given the distinctive vocabulary utilized in rap lyrics.

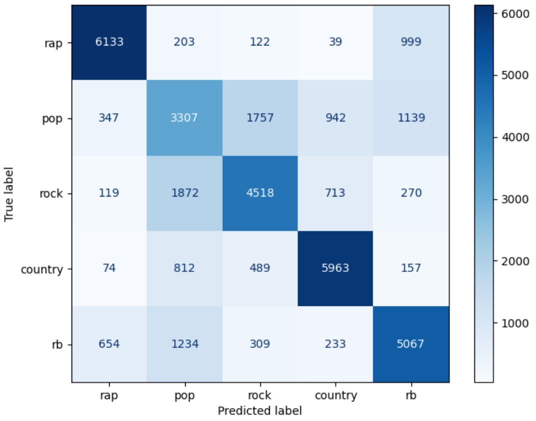

Fig. 3. Heatmap for DISTILBERT without GELU

Genre	Precision	Recall	F1-score	Support
Rap	0.81842	0.84640	0.83217	7487
Pop	0.43929	0.42369	0.43135	7489
Rock	0.63062	0.59696	0.61333	7493
Country	0.73529	0.81101	0.77129	7487
R&B	0.67823	0.64551	0.66146	7484
Macro Avg	0.66037	0.66471	0.66192	37440
Weighted Avg	0.66035	0.66469	0.66190	37440

TABLE II
CLASSIFICATION REPORT FOR DISTILBERT WITH GELU ON GENRE CLASSIFICATION.

Additionally, we experimented with DistilBERT, using the GELU activation function, as a model for genre classification. The results were comparable, with an overall accuracy of 66.47% across the five genres. The performance metrics for DistilBERT are presented in Table II.

The macro average metrics for DistilBERT were Precision 0.66037, Recall 0.66471, and F1-Score 0.66192, while the weighted averages were Precision 0.66035, Recall 0.66469, and F1-Score 0.66190.

To push the boundaries further, we used BERT, a more sophisticated model with 110 million parameters. The additional complexity of BERT improved the genre classification accuracy to 67.15%, demonstrating its ability to capture nuanced patterns in the data. The results for BERT are presented in Table III.

The macro average metrics for BERT were Precision 0.67502, Recall 0.67149, and F1-Score 0.67304, while the weighted averages were Precision 0.67504, Recall 0.67151, and F1-Score 0.67307.

Genre	Precision	Recall	F1-score	Support
Rap	0.83636	0.82364	0.82995	7496
Pop	0.45293	0.46623	0.45948	7492
Rock	0.60691	0.64028	0.62315	7492
Country	0.79722	0.76638	0.78150	7495
R&B	0.68166	0.66093	0.67114	7497
Macro Avg	0.67502	0.67149	0.67304	37472
Weighted Avg	0.67504	0.67151	0.67307	37472

TABLE III
PERFORMANCE METRICS FOR BERT ON GENRE CLASSIFICATION.

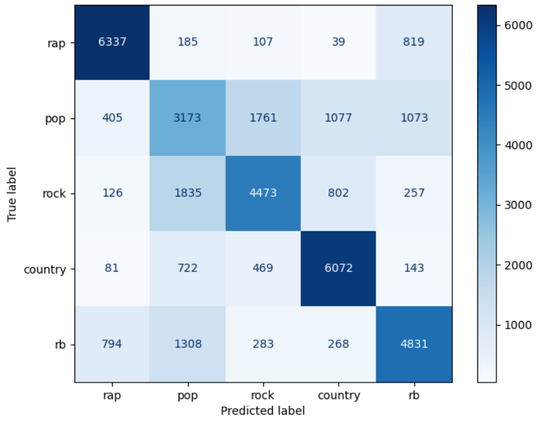

Fig. 4. Heatmap for DISTILBERT with GELU

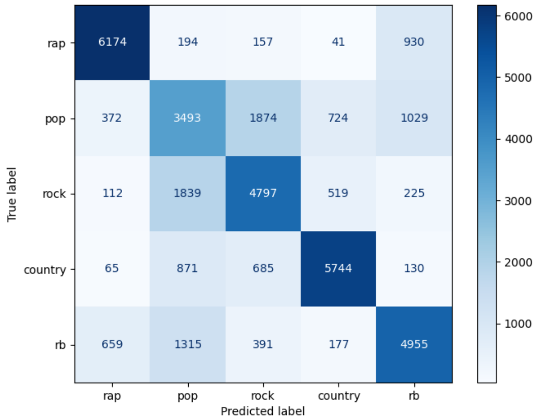

Fig. 5. Heatmap for BERT

We explored RoBERTa, a model with 125 million parameters, which further improved the accuracy to 67.69%. RoBERTa's metrics are presented in Table IV.

The macro average metrics for RoBERTa were Precision 0.67593, Recall 0.67688, and F1-Score 0.67598, while the weighted averages were Precision 0.67595, Recall 0.67691, and F1-Score 0.67601.

To compare with simpler models, we also implemented logistic regression as a baseline. The logistic regression model achieved an accuracy of 60%, highlighting its limitations in capturing the intricate relationships within the lyrics. The results for logistic regression are presented in Table V.

The macro average metrics for logistic regression were Precision 0.59, Recall 0.60, and F1-Score 0.59, while the weighted averages were Precision 0.59, Recall 0.60, and F1-Score 0.59.

Genre	Precision	Recall	F1-score	Support
Rap	0.85037	0.80216	0.82556	7496
Pop	0.46882	0.45048	0.45946	7492
Rock	0.62704	0.63014	0.62859	7492
Country	0.75965	0.82148	0.78936	7495
R&B	0.67376	0.68014	0.67693	7497
Macro Avg	0.67593	0.67688	0.67598	37472
Weighted Avg	0.67595	0.67691	0.67601	37472

TABLE IV

PERFORMANCE METRICS FOR ROBERTA ON GENRE CLASSIFICATION.

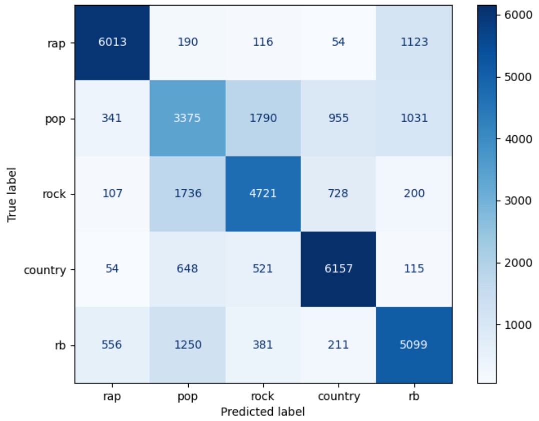

Fig. 6. Heatmap for RoBERTa

Genre	Precision	Recall	F1-score	Support
Country	0.66	0.71	0.68	7500
Pop	0.37	0.31	0.34	7500
Rap	0.79	0.79	0.79	7500
R&B	0.59	0.59	0.59	7500
Rock	0.54	0.58	0.56	7500
Macro Avg	0.59	0.60	0.59	37500
Weighted Avg	0.59	0.60	0.59	37500

TABLE V

PERFORMANCE METRICS FOR LOGISTIC REGRESSION ON GENRE CLASSIFICATION.

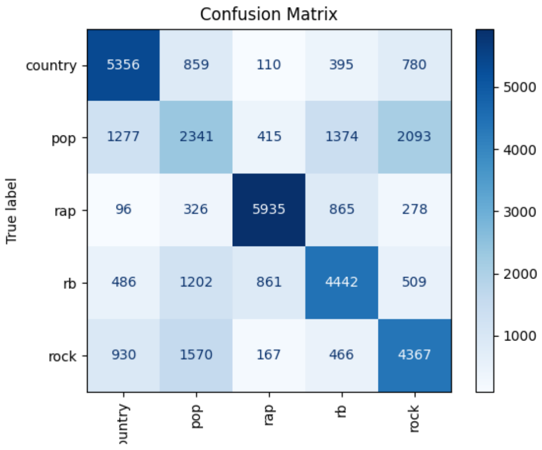

Fig. 7. Heatmap for Logistic Regression

Model	Parameters (M)	Size (MB)
DistilBERT	67.0	267.829
RoBERTa	125.0	500.960
BERT	110.0	440.307

TABLE VI

MODEL PARAMETERS AND ESTIMATED SIZE

CONCLUSION

In conclusion, our exploration of genre classification through various models highlighted the challenges and opportunities in analyzing lyrical data. Logistic regression provided a straightforward baseline, but its limitations in capturing intricate patterns were evident in its modest performance. Transitioning to transformer-based models, such as DistilBERT, BERT, and RoBERTa, revealed the transformative potential of deep learning in this domain. Each model's increasing parameter count and architectural sophistication brought incremental improvements, with RoBERTa achieving the highest accuracy of 67.69%. However, distillBERT is easier to implement, faster to train with little loss on its accuracy compared to the other models, making it the best option for our case. Notably, the distinctiveness of rap lyrics contributed to consistently high performance across all models, whereas the overlap in themes between pop and rock posed persistent challenges. These findings underscore the importance of model selection and fine-tuning in genre classification tasks.

REFERENCES

[1] R. V. Rajendran, A. S. Pillai, and F. Daneshfar, “Lybert: Multi-class classification of lyrics using bidirectional encoder representations from transformers (bert),” 2022.

[2] H. Adel, A. Dahou, A. Mabrouk, M. Abd Elaziz, M. Kayed, I. M. El-Henawy, S. Alshathri, and A. Amin Ali, “Improving crisis events detection using distilbert with hunger games search algorithm,” *Mathematics*, vol. 10, no. 3, p. 447, 2022.

[3] J. Devlin, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[4] V. Sanh, “Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter,” *arXiv preprint arXiv:1910.01108*, 2019.

[5] A. Baratloo, M. Hosseini, A. Negida, and G. El Ashal, “Part 1: simple definition and calculation of accuracy, sensitivity and specificity,” 2015.