Predicting Music Genre using Lyrics with Deep Learning

Jin Young Bang, Christopher Gough, Taesung Yoon
CS 505 – Natural Language Processing, Boston University

Abstract

This research explores the use of neural networks in predicting music genres from song lyrics to enhance recommendation systems. Leveraging the MetroLyrics Dataset, our study emphasizes preprocessing for focused lyric-based analysis. Addressing genre distribution imbalance through categorical cross-entropy, our baseline models, including LSTM, GRU, and HAN, reveal bidirectional variants consistently outperforming non-bidirectional counterparts. The HAN-GRU model stands out, achieving a 59.98% accuracy. Hyperparameter tuning yields modest improvements, underscoring the baseline model's efficacy. Despite resource constraints, this study provides valuable insights into the synergy of lyrics, music genres, and machine learning. Future iterations, with enhanced computational resources, hold promise for further advancements in the dynamic realm of digital music consumption and recommendation systems.

1. Introduction

In the realm of entertainment, the classification of music genres holds immense significance, particularly in shaping the efficacy of recommendation systems. Our research delves into this captivating domain, aiming to harness Neural Networks' potential to create a model proficient in predicting music genres solely from song lyrics. Our pursuit is rooted in the belief that this approach can untangle the intricate web of musical categorization, unlocking invaluable insights for platforms like Spotify. The success of our methodology holds the potential to elevate user experiences by fine-tuning genre recommendations, a fundamental element in today's digitally-driven era of personalized music consumption.

2. Dataset, Preprocessing, and EDA

2.1 Dataset

Our research heavily depended on the MetroLyrics Dataset initially found on Kaggle. However, it was later removed, prompting us to extensively search GitHub for the original data. This dataset was crucial for our study as it contains a vast collection of song lyrics. It formed the foundation of our investigation into predicting music genres based on lyrical content.

Despite having a limited set of features like song name, year, and artist, our focus was primarily on using the complete lyrics and song genres. The dataset included genres such as Rock, Hip-Hop, Electronic, Country, Indie, Jazz, Metal, Pop, R&B, Folk, and Other for our analysis.

2.2 Data Preprocessing

Due to the dataset's collaborative and open-source nature, the dataset exhibited diverse formats for transcribed lyrics, leading to a certain level of disorder. As a result, extensive data cleaning and preprocessing became necessary. Among the contents were instrumental tracks, which required exclusion. Furthermore, the task involved removing most punctuation, non-ASCII characters, and musical annotations like "Chorus" and "Verse" from the lyrics.

Initially, we conducted a comprehensive assessment of the dataset to identify and eliminate any instances of null values, ensuring the integrity of the data for further analysis.

Our data cleaning and preprocessing involved a series of essential steps:

1. Punctuation Removal: We systematically removed punctuation marks from the 'lyrics' column to streamline the text for analysis, enhancing the accuracy of our subsequent processing steps.
2. Elimination of Song-Related Identifiers: Recognizing elements like [Chorus] or [Verse] as identifiers, we eliminated these markers to focus solely on the lyrical content, enabling a more refined analysis.
3. Exclusion of Instrumental Tracks and Non-Lyrical Entries: To maintain the relevance of the dataset, we meticulously filtered out instrumental tracks and songs without discernible lyrics, ensuring the dataset primarily comprised lyrically rich content.
4. Handling Non-ASCII Characters and 'Not Available' Entries: Entries containing corrupted or non-ASCII characters, as well as those marked as 'not available' within the 'genre' column, were systematically filtered out, enhancing the dataset's coherence and reliability.

Finally, acknowledging the presence of non-English language lyrics within our dataset, we employed the 'langdetect' library to identify and subsequently filter out these entries from our cleaned dataframe. This step ensured that our analysis focused exclusively on English language lyrics, optimizing the accuracy of our subsequent modeling and analysis.

By implementing these rigorous preprocessing steps, we aimed to refine the dataset, ensuring a more robust foundation for our subsequent analyses and modeling endeavors.

2.3 Exploratory Data Analysis (EDA)

We decided to perform EDA on our dataset to see what our dataset contained. We were most concerned about the balance of our dataset since an imbalance dataset could potentially cause issues while training our models.

Taking advantage of pandas, we converted our cleaned and preprocessed dataset into a dataframe and plotted out the value_counts of our genre column.

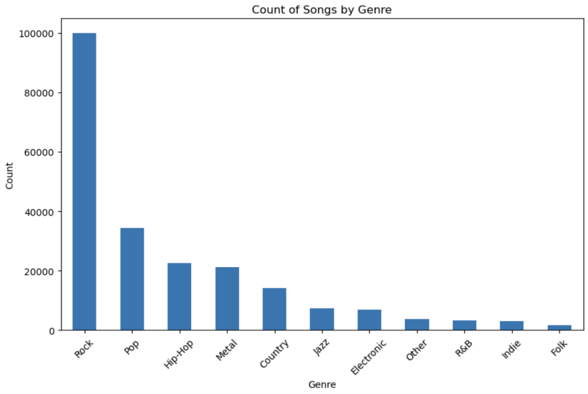

Figure 1. Count of Songs by Genres

Upon analyzing the figure, it became evident that there is a significant imbalance in our dataset. The “Rock” genre overwhelmingly dominates, with the count of songs being notably higher than in other genres. This imbalance could lead to a model bias towards the “Rock” genre, impacting the performance and accuracy of predictions for other genres.

To mitigate this issue and enhance the model’s ability to generalize across all genres, we will employ categorical cross-entropy as our loss function during model training. Categorical cross-entropy is particularly adept at handling classification problems where classes are imbalanced and ensures that the model learns effectively from each class by penalizing misclassifications proportionally to their severity.

This approach will help us to train a more balanced and robust model, capable of accurately classifying songs across a diverse range of genres. By doing so, we aim to ensure that our model's predictions are not unduly influenced by the over-represented "Rock" genre, but instead reflect the true characteristics of each song's genre.

3. Baseline Models

Given the substantial size of our dataset, particularly due to the extensive word count within lyrics, we opted to train our baseline models using a subset of 25,000 randomly selected data points from our preprocessed dataset. This decision stemmed from challenges related to RAM limitations and the considerable time required for model training.

3.1 Long Short-Term Memory (LSTM) Model

We explored multiple baseline models in this study. One of our chosen baselines featured a standard recurrent neural network, specifically a vanilla LSTM cell comprising 128 hidden units.

LSTMs can be effective for predicting genres from music lyrics due to their ability to capture sequential information and handle temporal dependencies within the text data.

3.1.1 Vanilla LSTM Model

The consistent accuracy of 45.95% throughout the training process suggests a potential issue with insufficient model complexity. It is likely that the model lacks the capacity to capture the intricate patterns within the data adequately. One possible improvement to address this limitation is to make the model bidirectional, allowing it to consider contextual information from both past and future sequences.

3.1.2 Bidirectional LSTM Model

By enabling bidirectional processing, the model gains a more comprehensive understanding of the input data, which may enhance its ability to learn and generalize effectively, potentially leading to improved performance on the task at hand.

The bidirectional model shows signs of overfitting, as seen in the significant increase in training accuracy compared to validation accuracy. While training accuracy steadily improves to 87.20%, the gap between training and validation accuracy suggests the model is fitting too closely to the training data and struggling to generalize to new samples.

3.2 Gated Recurrent Unit (GRU) Model

GRU (Gated Recurrent Unit) models, much like LSTMs, are a type of recurrent neural network designed to handle sequential data. Their ability to handle sequential data and capture dependencies makes them a viable alternative to LSTMs for this task.

3.2.1 Vanilla GRU Model

Similarly, we opted to train a simple vanilla GRU model to understand how the model worked. By adding three layers (embedded, GRU, and linear), we were able to achieve an accuracy of 49%, with a steady drop of training loss. However, after 10 epochs, the model showed no increase in accuracy, likely due to its simplicity. With this in mind, we concluded that introducing complexity to the model would be the next best step.

3.2.2 Bidirectional GRU Model

Switching to a bidirectional model showed improvements over its vanilla counterpart with a higher accuracy. A dropout layer was added after the model showed signs of overfitting, resulting in a lower discrepancy between the model's training and validation accuracy.

The performances of the bidirectional LSTM and GRU models consistently surpassed that of their non-bidirectional counterparts, concluding that they are effectively better at capturing the intricate patterns of musical lyrics.

3.3 Hierarchical Attention Networks (HAN) Model

The main idea behind HAN[1] is to catch the relationships between different levels of this hierarchy. For instance, in document classification, a document can be seen as a hierarchy of sentences, and each sentence is made up of words. HAN uses attention mechanisms to focus on specific parts of this hierarchy, helping the model decide what's important at each level.

This architecture was first proposed by Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, and Alex Smola, a joint group of Carnegie Mellon and Microsoft researchers.

Using the specifications as described in the research, we integrated attention mechanisms at both the word and sentence levels, enabling our model to selectively attend to specific segments of musical lyrics.

As our baseline GRU model exhibited superior performance compared to the LSTM baseline, we introduced a regular HAN-GRU model and modified HAN-GRU model, inspired by Tsaptsinos[2]. Given Tsaptsinos' model producing promising results on 20 genres, we anticipated similar outcomes with our MetroLyrics dataset, which features a reduced genre set. This exploration aims to ascertain the efficacy of these models in capturing hierarchical document dependencies, with a specific focus on their performance.

3.3.1 Vanilla HAN-GRU Model

The first iteration of our HAN model took a longer time to train when compared with our baseline LSTM and GRU models. Due to time constraints, we reduced the number of epochs from 10 to 5. Despite limiting the training to a mere five epochs, the HAN model immediately exhibited signs of overfitting, showing a 95.61% training accuracy and a sharp decline in training loss. This discrepancy raised concerns about the generalization capability of the model. Validating our model showed a testing accuracy of 51.22%, showcasing a potential for improvement. It became evident that fine-tuning parameters and optimizing hyperparameters were essential steps in improving our new model.

3.3.2 Modified HAN-GRU Model

The second model employs a unified attention mechanism class for both word and sentence levels, utilizing trainable parameters for attention score computation. Additionally, the second model incorporates dropout for regularization.

Results showed significantly less overfitting and a steady increase in accuracy and steady decrease in training loss. One notable improvement was the smaller discrepancy between the training and testing accuracy. With a bigger sample size and more hyperparameter tuning, we can further enhance generalization performance, reduce overfitting, and potentially achieve improved accuracy.

3.4 Baseline Model Evaluation

The table below showcases the accuracies obtained from our validation dataset post-training of our models. This served as a guide, helping us identify the most promising model for subsequent hyperparameter tuning and further refinement.

Model	Accuracy
Vanilla-LSTM	45.95%
Bidirectional-LSTM	45.05%
Vanilla-GRU	49.00%
Bidirectional-GRU	50.40%

HAN	51.22%
HAN-GRU	51.30%

4. Results

4.1 Hyperparameter Tuning

After evaluating our baseline models, we opted to conduct hyperparameter tuning specifically for the HAN-GRU model, this time utilizing the entire dataset instead of the previously sampled 25,000 entries.

In our initial attempt, we maintained the existing hyperparameters while training on the full dataset, curious to observe the impact of increased data volume on accuracy. As anticipated, with a larger dataset, the model achieved an accuracy of 59.98%.

For our subsequent experiment, we aimed to enhance the model's performance by modifying specific hyperparameters. We increased the attention size from 100 to 200 to enable a more comprehensive capture of intricate lyrical relationships. However, this adjustment posed a potential risk of overfitting. To counter this, we raised the dropout rate from 0.3 to 0.4, mitigating the risk of overfitting by introducing more regularization. Additionally, we reduced the hidden size to encourage the model to generalize better on unseen data. This adjustment aimed to facilitate the model's ability to discern underlying patterns within the lyrics, contributing to improved generalization capabilities. This model had an accuracy of 55.79%.

In our third experiment, our aim was to enhance the complexity of our model by adding an additional layer to our GRU cell. Despite this augmentation, the results mirrored those of the initially tuned model, yielding an accuracy of 59.54%.

In our final phase, we explored diverse learning rates for our HAN-GRU model. Throughout training, we observed instances where the loss decreased sharply while the accuracy increased dramatically, hinting at possible overfitting to the training data. To address this, we experimented with variations in the learning rate to explore its impact.

As anticipated, increasing the learning rate from 0.001 to 0.01 resulted in a steadier drop in loss during training. Conversely, decreasing the rate from 0.001 to 0.0001 led to a more substantial decrease in loss. These variations in learning rates provided insights into the model's behavior during training, indicating the sensitivity of the training process to different learning rate regimes.

4.2 Evaluation

Outlined below are the accuracies obtained from our series of hyperparameter experiments:

HAN-GRU Model	IPYNB Name	Accuracy
Baseline	tuning-han-gru-1	59.98%
Attention Size Increase; Hidden Size Decrease	tuning-han-gru-2	55.79%
Added complexity (num_layers=2)	tuning-han-gru-3	59.54%
Learning Rate of 0.0001	han-lr-0.0001	58.51%
Learning Rate of 0.01	han-lr-0.01	59.64%

The results underscore the substantial improvement in model accuracy achieved by incorporating the entire dataset during training. Despite our endeavors in hyperparameter tuning, the initial baseline architecture for our HAN-GRU model emerged as the most effective, delivering an accuracy of 59.98%. While this accuracy may appear modest, considering the sole reliance on lyrics for genre classification, it stands as a reasonable achievement.

Evidently, the model demonstrates a commendable ability to classify distinct genres at a reasonable level.

Introducing additional predictors or variables could potentially enhance the model's performance even further, offering opportunities for improved classification accuracy.

5. Other Considerations

In retrospect, there are key areas we would address if revisiting this project. One significant consideration involves leveraging more powerful computing resources to delve deeper into experimentation. During our initial phase, we encountered constraints in training our models due to limitations in available resources. Google Colab, for instance, struggled to handle the volume of data within its RAM, necessitating us to work with a smaller subset. Additionally, training times were extended significantly.

Another area for potential improvement involves extending the number of training epochs. Our experiments were capped at 5 to 10 epochs, potentially restricting the model's learning capacity. Despite the time-consuming nature of training, delving deeper into this aspect might have enriched our outcomes.

Despite these challenges, our project yielded commendable conclusions and results. However, allocating additional time for extensive experimentation might have potentially yielded even more refined outcomes. In future iterations, investing in powerful computing infrastructure and allowing for prolonged experimentation could enhance the depth and accuracy of our analyses, potentially yielding more nuanced and robust results.

6. Code and Findings

The code for our project is accessible via the following link:

https://github.com/jinyoungbang/CS505-Final-Project

For guidance on navigating through the codebase, detailed instructions are provided in the `README.md` file within the repository.

7. Team Contribution

Jin Young Bang

- Led the initial investigation into available datasets, identifying and securing the MetroLyrics Dataset.
- Implemented and fine-tuned baseline GRU and bidirectional GRU models
- Collaborated on the hyperparameter tuning experiments, exploring variations to enhance model performance

Taesung Yoon

- Conducted exploratory data analysis, identifying and addressing genre distribution imbalances
- Implemented and fine-tuned the HAN-GRU models, contributing to the final model selection
- Collaborated on the hyperparameter tuning experiments, exploring variations to enhance model performance

Christopher Gough

- Led data preprocessing efforts, ensuring the quality and focus of the lyric-based dataset
- Implemented and fine-tuned the baseline LSTM and bidirectional LSTM models, contributing to the baseline model selection
- Collaborated on the hyperparameter tuning experiments, exploring variations to enhance model performance

8. References

[1] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). Hierarchical attention networks for document classification. https://www.cs.cmu.edu/~./hovy/papers/16HLT-hierarchical-attention-networks.pdf

[2] Tsaptsinos, A., & Wang, G. (2017). A Hierarchical Attention Model for Improved Music Transcription. https://ccrma.stanford.edu/groups/meri/assets/pdf/tsaptsinos2017preprint.pdf