Music Genre Classification Using Song Lyrics

Parsa Keyvani
keyvanip@union.edu

1 Background and Motivation

The purpose of this research is to develop an English music genre classification model using song lyrics. The project aims to explore the use of natural language processing and machine learning algorithms to accurately predict the genre of a song based on its lyrics. The model will be trained on a large dataset of song lyrics and corresponding genre labels, and will be evaluated on its ability to correctly classify new song lyrics into their respective genres. The project aims to contribute to the field of music classification and provide methods for automatically categorizing music based on its lyrical content. The results of this research could have implications for music recommendation systems, music industry professionals, and music fans looking for new artists and genres to explore.

2 Prior Work

Numerous research studies have explored music genre classification by utilizing features such as rhythm, pitch, and lyrics. Nevertheless, fewer studies have concentrated on music genre classification based solely on the lyrics of a song.

One notable study is the work of Sturm et al. (2017), who developed a genre classification model using both lyrics and audio features. They extracted features from the lyrics using various natural language processing techniques and trained a support vector machine classifier. The model achieved an accuracy of 74% on a dataset of 188,000 songs (Sturm et al., 2017).

Another study by Chen et al. (2018) used deep learning techniques to classify music genres based solely on lyrics. They used a convolutional neural network to extract features from the lyrics and trained a multi-class classification model. The model achieved an accuracy of 70% on a dataset of 5,000 songs (Chen et al., 2018).

In another study, Vellinga et al. (2019) explored the use of word embeddings to classify music genres based on lyrics. They used a combination of convolutional and recurrent neural networks to extract features and trained a multi-class classification model. The model achieved an accuracy of 58% on a dataset of 10,000 songs (Vellinga et al., 2019).

These studies provide a foundation for our research on music genre classification using song lyrics. However, we aim to extend these studies by exploring the effectiveness of different feature extraction techniques and machine learning algorithms in accurately classifying music genres based solely on lyrics.

3 Data and Software

Our Kaggle dataset contains lyrics and genres of 160,000 songs from genres across several languages with majority of them being English (Neisse, 2020). The dataset is presented in two separate tables, one listing artists and their genres and the other listing songs, artists, and lyrics. It is worth mentioning that the dataset includes songs that are classified under multiple genres. We do not need any additional software. The accessibility of the dataset has been verified and works seamlessly with our programming language and software (Spyder and RStudio).

4 Data Preprocessing

The dataset used for this project consisted of 160,000 songs across 72 different genres and multiple languages. However, a challenge arose because genres were assigned to artists rather than lyrics. As some artists had multiple genres, it was difficult to classify lyrics under a single genre accurately. To overcome this, the decision was made to only use artists with a single assigned genre, ensuring more accurate genre classification for the lyrics.

To achieve this, the two tables in the Kaggle dataset - one containing artist-genre pairs and the other containing song-artist-lyric information - were merged. Non-English songs were removed, leading to the exclusion of 3 out of the 6 original genres, and duplicate songs were eliminated.

To prepare the lyrics for GloVe embedding, punctuation was tokenized so that each punctuation mark was recognized as its own word during the embedding process. The code from Stanford NLP's GitHub was utilized to generate the GloVe embeddings for the lyric corpus.

To train the model, the dataset was split into training, validation, and test sets, using an 80-20-20 split. Finally, to balance the distribution of genres, pop and hip-hop were oversampled to generate new datasets, ensuring that the three genres had a similar number of lyrics. The output genres were then one-hot-encoded to facilitate the model training process.

The lyrics in the dataset are tokenized using the Tokenizer class from the Keras library. The maximum number of words to keep is set to 5000, and an out-of-vocabulary (OOV) token is used for words not in the vocabulary. The tokenizer is fit on the lyrics, and the resulting sequences are padded to have the same length using the pad$_sequencesmethodfromKeras$.

5 Approach

Load the pre-trained GloVe embeddings: Pre-trained GloVe embeddings are loaded from a text file containing word vectors for words in the vocabulary.

Create the embedding matrix: An embedding matrix is created for the words in the vocabulary using the pre-trained GloVe embeddings. The dimension of the embedding vectors is set to 100, and the size of the vocabulary is set to the minimum of the number of words in the vocabulary and 5000. For each word in the vocabulary, if the word is not in the pre-trained GloVe embeddings, the corresponding row in the embedding matrix is set to all zeros.

Train logistic regression model: A logistic regression model is trained on the padded sequence data and the encoded genre labels using the LogisticRegression class from the sklearn library.

Make predictions on testing set: The trained logistic regression model is used to make predictions on the testing set.

Evaluate model performance: The performance of the logistic regression model is evaluated using the F1 score, accuracy, precision, and recall.

Train LSTM model: An LSTM model is created using the Sequential class from keras. The first layer is an embedding layer with the embedding matrix created in step 2. The second layer is an LSTM layer with 128 units, a dropout rate of 0.2, and a recurrent dropout rate of 0.2. The final layer is a dense layer with 10 units and a softmax activation function. The model is compiled with the sparse categorical cross-entropy loss function, the Adam optimizer, and the accuracy metric. The model is trained on the training set and validated on the testing set for 10 epochs with a batch size of 64.

6 Visualizations

1. Average Lyric Length by Genre: This visualization represents a table that documents the average length of lyrics for each genre in the dataset. The code first computes the length of each lyric and adds it as a new column to the dataset. Then, the dataset is grouped by genre, and the mean of the lyric lengths is computed for each group. The resulting table shows the average length of lyrics for each genre in the dataset.

2. Top 10 Unique Words by Genre: This visualization represents a table that shows the top 10 unique words mentioned in each genre the most that are different in other genres. The code loops through each genre in the dataset and gets all the words for the current genre and the other genres. The occurrences of each word in the current genre are counted, and the top 10 unique words for the current genre are printed. The resulting table shows the top 10 unique words for each genre.

3. Confusion Matrix for Logistic Regression Model: This visualization represents a confusion matrix for the logistic regression model using average GloVe embeddings. The rows represent the true genre, while the columns represent the predicted genre. The code computes the confusion

matrix using the confusion matrix function from the sklearn library and displays it using the ConfusionMatrixDisplay class from the sklearn library. The resulting plot shows the confusion matrix for the logistic regression model

7 Results

The logistic regression model achieved an accuracy of 0.645 and an F1 score of 0.605. On the other hand, the LSTM model achieved an accuracy of 0.694 and an F1 score of 0.681. This shows that the LSTM model outperformed the logistic regression model in terms of accuracy and F1 score.

The confusion matrix for the logistic regression model showed that the model had difficulty in accurately predicting certain genres, such as hip-hop and jazz. In contrast, the LSTM model showed improved performance in predicting these genres. Additionally, the bidirectional LSTM model outperformed both the logistic regression model and the LSTM model with an accuracy of 0.692 and F1 score of 0.684 .

Overall, the LSTM model with bidirectional LSTM is the better-performing model for this task of music genre classification using song lyrics. However, it is important to note that the evaluation metrics for the models could be influenced by the dataset used and the model hyperparameters chosen.

8 Discussion

The bidirectional LSTM model has performed better than the logistic regression and the standard LSTM models because it is better equipped to handle the sequential nature of text data.

The bidirectional LSTM model processes the input sequence both forward and backward, allowing it to capture both the past and future context of each word. This enables the model to better understand the meaning and context of the text data and make more accurate predictions.

Furthermore, the bidirectional LSTM model uses pre-trained GloVe embeddings, which are a type of word embedding that captures the semantic relationships between words based on their co-occurrence in a large corpus of text. These embeddings can improve the model's understanding of the meaning and context of words, which is crucial in music genre classification, where different genres may use similar words or phrases. In addition, the bidirectional LSTM model is trained on longer sequences of text data than the logistic regression model, allowing it to capture more information about the lyrics and their underlying meaning.

Overall, the bidirectional LSTM model is better suited for the task of music genre classification using song lyrics due to its ability to handle sequential data, use of pre-trained embeddings, and ability to process longer sequences of text data.

9 Conclusion

Our main goal was to most accurately identify the genre of a song based off its lyrics. After preprocessing our data and creating GloVe embeddings, we were able to reach impressive performance with logistic regression baselines. Then, by adding memory and word order to our model in the form of an LSTM, we were able to achieve equal accuracy. Finally, by adding both backward memory in addition to forward in a bidirectional LSTM, we were surprised to see a drop in accuracy overall. With additional time, we could investigate why a bidirectional LSTM would have worse performance. Also because our dataset has inaccuracies, it's hard to trust the accuracy scores of our model. If we had time to go through the dataset and fix the mistakes or use a different dataset entirely, it would be interesting to run our models again to see if some of the models have bumps in performance. In the future, we also hope to try running the experiments on already trained GloVe embeddings on a Wikipedia dataset to see if our pre-trained embeddings made a difference

10 Reflection

In this project, my role included cleaning and preprocessing the data, which involved text normalization, removal of stopwords, and other text processing techniques. Additionally, I implemented the code to tokenize the punctuation in the word embeddings, which is a crucial step in natural language processing and text data analysis. I also played a key role in training and testing the machine learning models used for music genre classification, helping to optimize their performance and accuracy. Furthermore, I took on the responsibility of creating the presentation slides for the project, which required synthesizing complex technical information into a clear and concise format. My groupmate Sandra made significant contributions to the project by attempting to concatenate the embeddings and balancing the data, which would have improved the performance of the models. Although we were not able to complete this task, her efforts and insights were invaluable in identifying areas for further research and improvement. Furthermore, Sandra provided valuable insights into the paper that we tried to emulate, which helped guide the implementation of the project.

Several NLP concepts and skills were relevant to this project, including:
Text preprocessing: The project required cleaning and preprocessing of the dataset, including text normalization, removal of stopwords, and tokenization.
Word embeddings: The project utilized pre-trained GloVe word embeddings, which are a type of word representation that captures the semantic relationships between words based on their co-occurrence in a large corpus of text.
Sequence modeling: The project involved training machine learning models, such as logistic regression and LSTM, to classify music genres based on the lyrics. Sequence modeling techniques, such as LSTM, are particularly useful for modeling sequential data such as text.
Natural language understanding (NLU): The project involved understanding and interpreting the meaning of song lyrics, which requires NLU skills such as syntax analysis, semantic analysis, and discourse analysis.

One thing I learned from this project is the importance of data preprocessing and feature engineering in natural language processing tasks. The quality of the input data and the features used to represent it can have a significant impact on the performance and accuracy of machine learning models. Through this project, I gained a deeper understanding of the various techniques used for text data preprocessing, such as text normalization, removal of stopwords, and tokenization, and how they can affect the quality of the data. Additionally, I learned how to use pre-trained word embeddings, such as GloVe, to represent text data, and how to train and optimize machine learning models, such as logistic regression and LSTM, for music genre classification. Overall, this project provided valuable hands-on experience in applying NLP techniques to real-world problems and reinforced the importance of thoughtful and deliberate data preprocessing and feature engineering in achieving accurate and effective machine learning models.

If I were to do the project over again, I would approach the data cleaning and preprocessing step more systematically and rigorously. While we did perform several preprocessing steps, I believe that we could have explored additional techniques to improve the quality of the data and reduce noise in the dataset. Additionally, I would experiment with different machine learning models and hyperparameters to identify the best approach for music genre classification, and potentially explore deep learning models, such as transformers, which have become increasingly popular in NLP tasks.

One thing that I would do the same is to maintain good communication and collaboration with my team members throughout the project. I believe that our team was successful in working together to achieve our objectives, and that was largely due to our frequent communication, openness to feedback, and willingness to support one another. Furthermore, I would continue to use a project management tool, such as Trello or Asana, to keep track of tasks and deadlines, as it helped us stay organized and on track throughout the project.

References

Peng Chen, Yifan Wu, Yu Hu, Xue Chen, and Xiao Wang. 2018. A deep learning approach to music genre classification based on lyrics. In *Proceedings of the 9th International Conference on Audio, Language and Image Processing*, pages 250–255. IEEE.

Lucas Neisse. 2020. Scrapped lyrics from 6 genres [data set]. Kaggle. https://www.kaggle.com/datasets/neisse/scrapped-lyrics-from-6-genres?select=lyrics-data.csv.

Bob L Sturm, Raul Santos-Rodriguez, Emmanouil Benetos, and Simon Dixon. 2017. An analysis of the reliability of genre recognition using mir datasets. In *Proceedings of the 18th International Society for Music Information Retrieval Conference*, pages 241–247.

Aryan Vellinga, Frans Wiering, and Remco C Veltkamp. 2019. Learning from lyrics: Classifying music genres with distributed representations of lyrics and audio. *Journal of New Music Research*, 48(4):327–343.