Genre-Based Song Lyrics Generation

Gharam Walid, Yousef Hassan

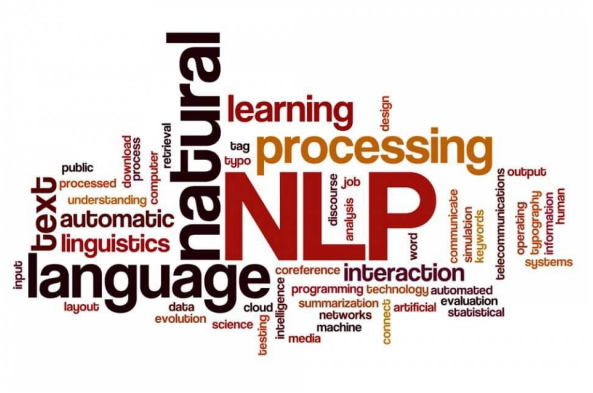
March 14, 2024

1 Introduction

In the era of digital streaming services and personalized playlists, the demand for innovative approaches to music generation and recommendation has surged. Among these, genre-based song lyrics generation stands out as a fascinating avenue, offering a blend of creativity, artificial intelligence, and linguistic analysis. With the advent of advanced machine learning techniques and the abundance of digital music data, researchers and enthusiasts alike are exploring novel methods to automatically generate song lyrics tailored to specific genres. This paper delves into the burgeoning field of genre-based song lyrics generation, discussing its significance, challenges, and recent advancements. By leveraging natural language processing (NLP) algorithms and large-scale datasets, genre-based song lyrics generation holds the potential to revolutionize music creation, empower artists, and enhance listener experiences. Through an exploration of existing methodologies and emerging trends, this paper aims to shed light on the evolving landscape of AI-driven music generation and its implications for the future of music composition and consumption.

2 Literature Review

Neural network-driven language models have demonstrated potential in generating original lengthy written content using minimal initial text, with recent efforts extending to the realm of musical lyric creation. However, crafting lyrics presents a distinct set of challenges not encountered in typical prose. Ensuring appropriate line breaks, mastering stylistic elements such as flow, rhyming, and repetition, and adhering to established lyrical structures like the verse-chorus format are all crucial factors in producing compelling songs. Despite varying degrees of success in these endeavors, it's important to recognize the broad spectrum encompassed by the term "lyrics." Different music genres exhibit unique lyrical styles, characterized by variations in line length, word repetition, semantic nuances, and narrative perspectives. Our research endeavors to explore the capacity of neural networks to generate genre-specific song lyrics while preserving the intrinsic linguistic features characteristic of each musical genre.



LSTM-based lyrical generation has been investigated within defined music genres. In their work "GhostWriter: Using an LSTM for Automatic Rap Lyric Generation," Potash et al. (2015) [Pot15] endeavored to produce lyrics akin to those of a "ghostwriter," crafting content tailored to a particular artist. Nonetheless, their model's scope was constrained to generating lyrics within a singular genre due to its training on a specific artist. Moreover, Potash et al. implemented rhyme schemes by training their model on datasets containing rhyming word pairs.

In [LFM19], the authors introduce iComposer, a system designed to simplify music production by automatically generating melodies and lyrics based on text input. The system is built on a sequence-to-sequence model architecture, with a focus on predicting melody, rhythm, and lyrics. Specifically, the authors utilize LSTM (long short-term memory) networks in their model, which are well-suited for sequence-based tasks due to their ability to capture long-range dependencies in data.

By leveraging LSTM networks, iComposer effectively learns the structure of Chinese popular music and generates melodies and lyrics that align harmoniously. The experimental results demonstrate that iComposer can compose compelling songs at a level comparable to human composers, showcasing the effectiveness of LSTM in capturing the intricate relationships between lyrics and melody. Overall, the integration of LSTM in iComposer enables the system to create music that resonates with listeners, highlighting the potential of LSTM in enhancing automated music composition systems.

In [GLM20] , the paper "Deep Learning in Musical Lyric Generation" explores the application of neural network-based language models, specifically LSTM, in generating original song lyrics across different genres. The study focuses on key linguistic characteristics such as average line length, word variation, point-of-view analysis, and word repetition to capture the essence of various music genres. By training models on specific genres like jazz, country, metal, pop, rap, and rock, the researchers generated lyrics based on 16-word prompts, showcasing the ability of the models to mimic genre-specific lyrical styles.
Despite challenges in capturing certain metrics like point-of-view in rock lyrics, the models demonstrated a high degree of semantic relevance and stylistic accuracy. The research not only highlights the potential of deep learning in music composition but also opens avenues for personalized text generation tailored to specific audience preferences within distinct writing styles.

The following figure is from [GLM20] which shows the percentage changes from original to generated lyrics for each metric and genre. This figure provides a detailed comparison of how the generated lyrics differ from the original ones in terms of various linguistic metrics, offering insights into the model's performance and the extent to which it captures the nuances of different music genres.

Genre	*Average Line Length*	*Song Word Variation*	*Genre Word Variation*	*I vs. You*	*Word Repetition*
Metal	-54.3	-25.1	-13.5	82.7	88.2
Rock	-38.4	-5.0	-5.9	77.0	34.9
Rap	-52.3	-64.6	-99.0	75.5	-49.6
Pop	-28.0	-16.7	-44.6	94.6	36.5
Country	-74.1	-36.6	-23.0	-229.6	75.2
Jazz	-24.1	-38.7	-78.5	73.7	94.6

Figure 1: Percentage Changes from original to generated lyrics for each metric and genre

In [FS14] , the paper presents a novel approach for analyzing and classifying song lyrics. The authors experiment with n-gram models and more sophisticated features to model different dimensions of song texts, such as vocabulary, style, semantics, orientation towards the world, and song structure. They demonstrate the effectiveness of these features in three classification tasks: genre detection, distinguishing the best and worst songs, and determining the approximate publication time of a song. The study draws on earlier work on text classification, poetry analysis, and lyrics-based music classification.
The authors collected their own dataset of English song lyrics and used metadata such as genre information, quality ratings, and publication years for their experiments. They designed 13 feature classes grouped into five abstract sets, and conducted three experiments to classify songs by genre, quality, and publication time. The results show that lyrics-based statistical models can be employed to perform different music classification tasks, and the extended feature set consistently improves classification performance. The study provides insights into the potential benefits of lyrics-based music classification for music retrieval, recommendation systems, and musicology research.

In [GSB], the paper investigates the effectiveness of different RNN-based language modeling architectures for music lyrics classification and generation. The custom dataset contains 80k song lyrics of the genres Rock, Pop, Rap, and Country. The classification performance of Vanilla RNN, GRU, and various LSTM variations is evaluated, with the GRU achieving the best performance. For lyrics generation, LSTM consistently produced higher quality lyrics. The paper also explores the use of initial cell-state and hidden-state to generate lyrics in specific genres. The results show that LSTM and GRU architectures are well-suited for classification and generation tasks. Additionally, the paper discusses the challenges in genre classification and the potential for future work using transformer-based models.

3 Dataset

In this dataset manipulation process, we first loaded a CSV dataset file using the pandas library in Python. We attempted to read the dataset into a DataFrame, ensuring proper encoding and handling potential parsing errors. Subsequently, we displayed the first few rows of the dataset and retrieved information about its structure, including data types and missing values. After handling missing values and removing duplicate rows, we proceeded to tokenize the text data using the NLTK library, breaking down each artist's name into individual words. Next, we cleaned the tokenized text by converting words to lowercase and removing punctuation and stopwords. Finally, we saved the modified DataFrame to a new CSV file and provided a link for downloading the processed dataset. This comprehensive process allowed us to prepare the dataset for further analysis or modeling tasks.

3.1 Data Analysis

In the given dataset, the analysis focused on determining the most frequent word present in the 'artist' column. After processing the text data by joining the 'artist' column entries into a single string and then splitting them into individual words, a dictionary was created to store the frequency count of each word. Through iteration over the words, their frequencies were tallied in the dictionary. Subsequently, the word with the highest frequency was identified using the max() function, which returned the word 'John'. The frequency of occurrence of this word was found to be 968 times within the 'artist' column. This outcome suggests that 'John' appears most frequently among all the artist names recorded in the dataset. Such analysis provides valuable insights into the distribution of artist names and the prominence of specific terms within the dataset.

The analysis focused on determining the most frequent word present in the 'song' column. The text data from the 'song' column was combined into a single string, and then split into individual words. A dictionary was created to store the frequency count of each word. Iterating over the words, their frequencies were tallied in the dictionary. Subsequently, the word with the highest frequency was identified using the max() function. In this case, the word 'The' was found to be the most frequent, appearing 7933 times within the 'song' column. This outcome suggests that 'The' is the most commonly occurring word among all the song titles recorded in the dataset. Such analysis provides valuable insights into the distribution of song titles and the prominence of specific terms within the dataset.

In this analysis of the dataset, the objective was to identify the most frequent word in the 'text' column. Initially, the textual data from the 'text' column was concatenated into a single string. This string was then split into individual words. A dictionary was employed to track the frequency count of each word encountered. Through iteration over the words, their frequencies were recorded in the dictionary. Subsequently, utilizing the max() function, the word with the highest frequency was determined. In this instance, the word 'the' emerged as the most frequently occurring word in the 'text' column, with a frequency count of 446,872. This finding underscores the prevalence of the word 'the' within the textual content of the dataset, offering valuable insights into common linguistic patterns present in the dataset.

We aimed to identify the largest elements within each column of our dataset. By iterating through the columns, the code computed the maximum length of each element within a column and stored the largest element along with its size. Upon execution, the output revealed the largest elements along with their corresponding sizes for each column in the dataset. For instance, in the 'artist' column, the largest element identified was 'Joseph And The Amazing Technicolor Dreamcoat', with a size of 44 characters. Similarly, the 'song' column featured the largest element 'A Simple Desultory Philippic (Or How I Was Robert McNamara'd Into Submission)' with a size of 77 characters. Notably, the 'text' column exhibited the largest element, a lengthy lyrical excerpt, spanning 3997 characters. These findings underscore the considerable variability in element sizes across different columns, reflecting the diverse nature of the textual data contained within the dataset.

3.2 Limitations

In spite of the richness of the dataset, several limitations warrant consideration. One notable limitation is the absence of null values or duplicate entries within the dataset. While the absence of null values and duplicates ensures data integrity and reliability, it also contributes to the dataset's substantial

size. As a consequence, managing and processing such a voluminous dataset may pose challenges in terms of computational resources and time efficiency.

We obtained an empty dataframe after performing correlation analysis, it typically means that there is no linear relationship between the variables in the dataset.

It's essential to acknowledge another limitation observed within the dataset: the presence of identical song names attributed to different artists. This phenomenon introduces ambiguity and potential confusion, as song titles may lack the specificity required to uniquely identify a musical composition

References

[FS14] Michael Fell and Caroline Sporleder. Lyrics-based analysis and classification of music. pages 620–631, 2014.

[GLM20] Harrison Gill, Daniel Lee, and Nick Marwell. Deep learning in musical lyric generation: an lstm-based approach. *The Yale Undergraduate Research Journal*, 1(1):1, 2020.

[GSB] Daniel Gordin, Arnab Sen Sharma, and Jaydeep Borkar. Music lyrics classification & generation using rnns.

[LFM19] Hsin-Pei Lee, Jhih-Sheng Fang, and Wei-Yun Ma. icomposer: An automatic songwriting system for chinese popular music. pages 84–88, 2019.

[Pot15] Romanov A. Rumshisky A. Potash, P. Ghostwriter: Using an lstm for automatic rap lyric generation. *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing.*, 2015.