Natural Language Processing for Music Generation

Girish Kumar Adari, Prerna Chander, Varadh Kaushik
CS 6120 - Fall 2023
Northeastern University

Abstract

The project involves leveraging Natural Language Processing methods and models for music generation. The implemented system utilizes various NLP components, including data collection, sentiment analysis, keyword extraction, topic modeling, and prompt construction to create a comprehensive understanding of song lyrics, themes and musical attributes. Using this, the prompts are fed into text-to-music generation models and evaluated to observe best performing models and comparison between generated music and original music in terms of musical features, to understand the inherent value of natural language preprocessing in music generation. The medium MusicGen model produced good results close to the original music files, showing viability of employing NLP techniques to create musical compositions inspired by the textual content of songs.

I. Introduction

Music generation has witnessed significant advancements with the integration of natural language processing (NLP) techniques. This project aims to explore the synergy between lyrics and music, utilizing NLP to bridge the gap between textual information and musical composition. The system involves the extraction of sentiment, identification of keywords, and topic modeling to provide a nuanced understanding of the lyrical content. By analyzing and understanding the emotional and thematic aspects of lyrics, the system aims to generate music that is not only technically sound but also emotionally resonant.

II. Methods

A. Dataset and Data Pre-Processing:

The dataset for this project is sourced from the Lakh MIDI Dataset, a widely used and comprehensive collection of MIDI files encompassing a vast array of musical genres and styles [1]. The Lakh MIDI Dataset serves as the primary repository for the instrumental aspects of each song. It provides a structured organization of MIDI files, facilitating the extraction of musical elements for each song. The dataset's hierarchical structure allows for the identification of MIDI files corresponding to various songs within different genres and artists. This structured organization aligns with the recursive traversal process utilized during data collection, ensuring efficient and systematic extraction of musical data.

The process involves traversing the specified root directory, parsing MIDI files, and extracting artist and song title information. This information, derived from the

directory and file names, forms the basis for querying the Genius API for lyrical content and the Spotify API for additional metadata [2].

The Genius API is utilized to retrieve song lyrics, providing a diverse range of textual data. Simultaneously, the Spotify API is employed to extract metadata associated with each song, including genres, tempo, and popularity. Figure 1 shows an example song in the dataset. From ~18,000 MIDI files, we restricted the dataset to only ~1830 songs with all the additional data. The integration of these two APIs ensures a comprehensive dataset that not only encompasses the lyrical content but also incorporates valuable information about the musical characteristics of each song.


```
{
  "midi_file_path": "archive-new\\ABBA\\I_Have_a_Dream.mid",
  "artist_band_name": "ABBA",
  "song_title": "I Have a Dream",
  "lyrics": {
    "Verse": [
      "I have a dream, a song to sing",
      "To help me cope with anything",
      "If you see the wonder of a fairy tale",
      "You can take the future, even if you fail",
      "",
      "I have a dream, a fantasy",
      "To help me through reality",
      "And my destination makes it worth the while",
      "Pushing through the darkness; still another mile",
      "",
      "I have a dream, a song to sing",
      "To help me cope with anything",
      "If you see the wonder of a fairy tale",
      "You can take the future even if you fail",
      ""
    ],
    "Chorus": [
      "I believe in angels",
      "Something good in everything I see",
      "I believe in angels",
      "When I know the time is right for me",
      "I'll cross the stream",
      "I have a dream",
      "",
      "I believe in angels",
      "Something good in everything I see",
      "I believe in angels",
      "When I know the time is right for me",
      "I'll cross the stream",
      "I have a dream",
      "I'll cross the stream",
      "I have a dream",
      "You might also like[Interlude]",
      "",
      "I believe in angels",
      "Something good in everything I see",
      "I believe in angels",
      "When I know the time is right for me",
      "I'll cross the stream",
    ],
    "Pre-Chorus": [],
    "Bridge": []
  },
  "spotify_metadata": {
    "energy": 0.374,
    "acousticness": 0.6,
    "danceability": 0.549,
    "instrumentalness": 0,
    "liveness": 0.178,
    "speechiness": 0.0244,
    "loudness": -12.049,
    "tempo": 104.311,
    "time_signature": 4,
    "valence": 0.439,
    "genre": [
      "europop",
      "swedish pop"
    ],
    "popularity": 66
  }
}
```


Figure 1: Example of Final Dataset

B. Exploratory Data Analysis:

While exploring the data, we came across a few interesting results. As seen in Figure 2, the data is distributed mostly across the ‘rock’ genre, with most songs being ‘rock’, ‘soft rock’, ‘classic rock’, ‘album rock’, ‘hard rock’, and ‘heartland rock’. This shows that the dataset is heavily biased towards rock genres.

Additionally, the Spotify metadata was analyzed [3]. In Figure 3, we can see a negative correlation between danceability and tempo. Danceability is a measure of how suitable a track is for dancing based on a combination of musical elements. A negative correlation with tempo suggests that, in this dataset, faster songs (higher tempo) might be less suitable for dancing, while slower songs (lower tempo) might be more danceable. There is also a strong positive correlation between energy and loudness. Energy is a measure of intensity and activity in the music, while loudness is a measure of the overall volume of the track. A strong positive correlation suggests that energetic songs are likely characterized by higher volume, contributing to a more intense listening experience. There is also a strong positive correlation between danceability and valence. Valence is a measure of the musical positiveness conveyed by a track.

Positive correlation with danceability suggests that more danceable songs are also more likely to convey positive musical emotions.

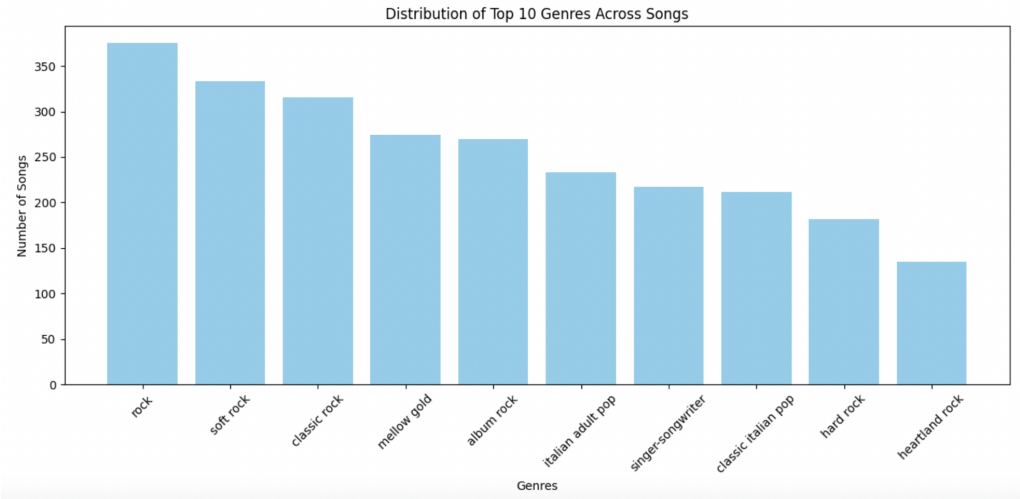

Figure 2: Distribution of Top 10 Genres of Songs

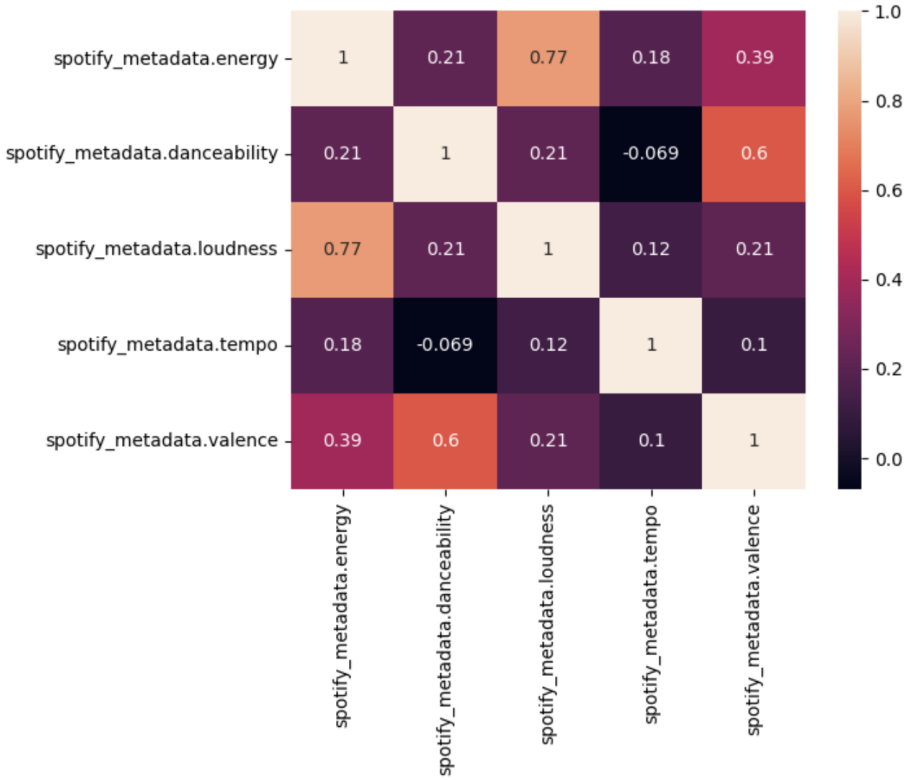

Figure 3: Correlation Matrix of Spotify Metadata

C. Sentiment Analysis and Keyword Extraction:

Once the raw lyrics are obtained, the TextBlob library is employed for sentiment analysis. Sentiment analysis provides a quantitative measure of the emotional

tone present in each song's lyrics. This information is valuable for understanding the overall mood and sentiment associated with the lyrical content. The RAKE (Rapid Automatic Keyword Extraction) algorithm is then utilized for keyword extraction. By extracting keywords from the pre-processed lyrics, the system identifies prominent themes, subjects, and emotional undertones within each song [4].

D. Topic Modeling:

To uncover latent themes and patterns within the lyrics, topic modeling is employed. The Gensim library is utilized to create a corpus from the pre-processed lyrics, and the Latent Dirichlet Allocation (LDA) model is trained on this corpus. The LDA model assigns topics to the lyrics, providing a structured representation of the underlying themes present in the dataset [5]. This process allows for a nuanced understanding of the recurring topics that characterize the song lyrics.

The combination of sentiment analysis, keyword extraction, and topic modeling creates a comprehensive understanding of the lyrical content. This enriched understanding serves as the foundation for the subsequent stages of music generation.

E. Prompt Construction:

The prompt is constructed by combining information from the sentiment analysis, keyword extraction, Spotify metadata, and identified topics. An example is shown in Table 1 below. The prompt may include statements about the song's sentiment ("has a positive/negative/neutral tone"), thematic keywords ("includes themes like..."), Spotify metadata ("belongs to genres such as... with a tempo of around... BPM"), and identified topics ("The lyrics often reflect topics such as...").

Song Title	Artist Name	Sentiment	Keywords	Spotify Metadata	Topics	Prompt
Caught Up In You	.38 Special	Positive (0.206)	['good love slip away', 'played around enough', 'never wanna get']	{'energy': 0.681, 'acousticness': 0.0229, 'danceability': 0.425, 'instrumentalness': 0.000219, 'liveness': 0.0543, 'speechiness': 0.0316, 'loudness': -8.604, 'tempo': 131.011, 'time_signature': 4, 'valence': 0.933, 'genre': ['album rock', 'classic rock', 'country rock', 'glam metal', 'hard rock'], 'popularity': 65}	Love, Like, Know, Say, Wanna	".38 Special's song 'Caught Up In You' has a positive tone and includes themes like good love slip away, played around enough, and belongs to genres such as album rock, classic rock, with a tempo of around 131.011 BPM. The lyrics often reflect topics such as love, like, know, say, wanna."
Prelude	Billy	Negative	['angry	{'energy': 0.69, 'acousticness':	Little,	"Billy Joel's song 'Prelude

Angry Young Man	Joel	(-3.36E-18)	young man give', 'angry young man', 'angry young man']	0.318, 'danceability': 0.456, 'instrumentalness': 7.79e-05, 'liveness': 0.354, 'speechiness': 0.0479, 'loudness': -10.156, 'tempo': 91.109, 'time_signature': 4, 'valence': 0.706, 'genre': ['album rock', 'classic rock', 'heartland rock', 'mellow gold', 'piano rock', 'rock', 'singer-songwriter', 'soft rock'], 'popularity': 48}	One, Day, Fire, Every	Angry Young Man' has a negative tone and includes themes like angry young man give, angry young man, angry young man and belongs to genres such as album rock, classic rock with a tempo of around 91.109 BPM. The lyrics often reflect topics such as little, one, day, fire, every."

Table 1: Example of Output After Collection, Pre-Processing, Sentiment Analysis, Keyword Extraction & Topic Modeling

F. Using the Facebook MusicGen Models for Music Generation:

MusicGen is a unified Language Model designed with a single-stage transformer LM combined with efficient token interleaving patterns. MusicGen can be conditioned on textual descriptions or melodic features. We used two pre-trained text-to-music models for comparison purposes: *facebook/musicgen-small* (300M model) and *facebook/musicgen-medium* (1.5B model) [6]. We opted for a model size that aligns with our available GPU resources, balancing computational efficiency and performance.

III. Results and Analysis

In this project, we conducted an analysis and comparison of audio files generated using the medium model with their corresponding original `.wav` files. The focus was on assessing the fidelity and accuracy of the generated audio in terms of tempo, spectral features, and overall fluidity. An example of the output generated using Bach Johann Sebastian's 'Musette BWV Anh.126' is shown below in Figures 4, 5, 6. The prompt generated was: “Bach Johann Sebastian's song 'Musette BWV Anh.126' has a neutral tone and belongs to genres such as baroque, classical with a tempo of around 170.949 BPM”.

A. Tempo Analysis: The tempo of the audio generated by the medium model was closely aligned with the required tempo. This was evident from the tempo extraction where the beats per minute (BPM) of the generated audio matched more precisely with the target BPM in the original `.wav` files. This indicates a high degree of temporal accuracy in the model's audio generation.

B. Waveform and Fluidity Comparison: Visual waveform comparison between the original and generated audio files showed a high degree of similarity. Notably, the

pseudo-waveform representation of MIDI files, converted to audio by the medium model, demonstrated a remarkable fluidity. The transitions between notes were smoother, and the overall waveform had fewer abrupt changes, indicating a more natural and fluid sound.

C. **Subjective Listening Tests**: Subjective assessments through listening tests further corroborated these findings. The audio generated by the medium model was perceived to be more coherent, with smoother transitions and a more natural flow, enhancing the listening experience.

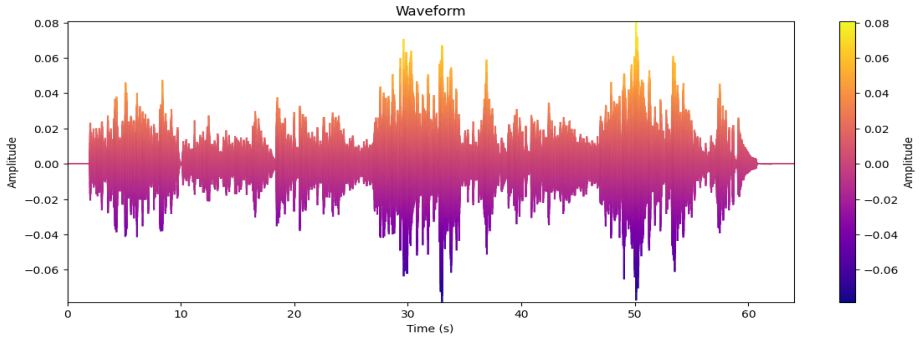

Figure 4: Original Piece, Tempo: 170.95 BPM

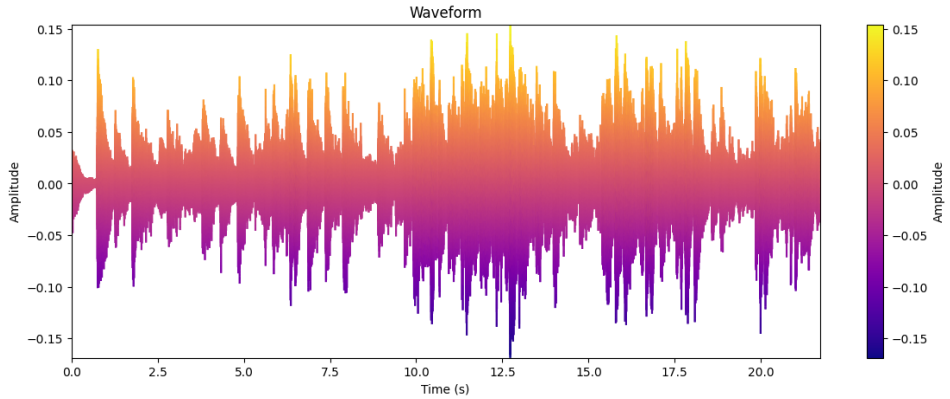

Figure 5: *musicgen-small* Model, Tempo: 117.45 BPM

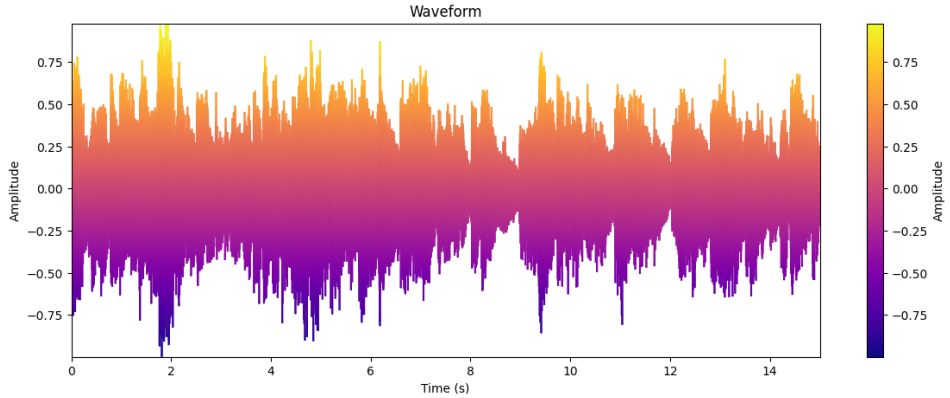

Figure 6: *musicgen-medium* Model, Tempo: 152.00 BPM

IV. Future Work

A. Fine-Tuning the Facebook MusicGen Models for Better Results:

By using the prompts along with the midi data, we can further tune the pre-trained MusicGen models to specialize them for a particular domain or genre that may not have been adequately covered in the original training data. This helps to improve performance by avoiding overfitting and enhancing the model's ability to generate diverse and high-quality music.

In the course of the project, We delved into the fine-tuning process and provided the initial code we worked on. We parsed the midi files and combined the data with the prompt tokens, split the data into train (80%) and test (20%) and provided the input to the *facebook/musicgen-small* model. However, due to complexity and dynamic nature of fine-tuning, additional time and extended exploration will contribute valuable insights and advancements to the overall success of the project.

B. Potential for Larger Pre-Trained MusicGen Models:

The current model, while effective within these constraints, is limited in its capacity to capture more intricate patterns and nuances inherent in complex music compositions. Given additional GPU resources, we foresee the opportunity to delve into more expansive model architectures, such as *facebook/musicgen-large* (3.3B model) [6]. The larger models are equipped to handle a broader range of musical intricacies, potentially leading to more nuanced and sophisticated music generation.

V. Conclusions

This project has showcased the potential of leveraging NLP techniques to enhance the creative process. The adoption of NLP components, including sentiment analysis, keyword extraction, and topic modeling, has proven to be a strategic approach in constructing prompts for the text-to-music generation models.

The systematic extraction of emotional and thematic nuances from song lyrics has enriched the prompt construction process, enabling the generation of music that not only aligns with technical specifications but also resonates with the underlying sentiments and themes of the original textual content.

The results of our experiments with the Facebook MusicGen models, particularly the medium-sized model, have demonstrated success in terms of tempo accuracy, spectral features, and overall fluidity.

Looking ahead, the potential for fine-tuning the MusicGen models based on these NLP-informed prompts opens avenues for specialization in various domains and genres. Additionally, the scalability of model performance relative to size, as evidenced by the success of the medium-sized model, underscores the adaptability and efficiency of such models in music generation tasks.

In essence, this project not only establishes the viability of employing NLP techniques for prompt construction but also emphasizes the promising outlook for the synergy between NLP and music generation. As we continue to explore the potential of larger models and fine-tuning strategies, the harmonious collaboration between language processing and musical creativity offers exciting prospects for the future of AI-driven music composition.

VI. References

[1] Gupta, S. (2021, June 21). *Lakh midi dataset clean*. Kaggle. https://www.kaggle.com/datasets/imsparsh/lakh-midi-clean

[2] Khan, M. (2021, November 21). *How to leverage Spotify API + genius lyrics for data science tasks in Python*. Medium. https://medium.com/swlh/how-to-leverage-spotify-api-genius-lyrics-for-data-science-tasks-in-python-c36cdfb55cf3

[3] Trench, A. (2022, February 23). *Mining Spotify metadata to track the mood of a nation*. Medium. https://medium.com/@andrewtrench/mining-spotify-metadata-to-track-the-mood-of-a-nation-7b55d29cf045

[4] Sanyal, S. (2021, October 26). *Rapid keyword extraction (rake) algorithm in Natural Language Processing*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/10/rapid-keyword-extraction-rake-algorithm-in-natural-language-processing/

[5] *Gensim: Topic modelling for humans*. LDA Model - gensim. (2022, December 21). https://radimrehurek.com/gensim/auto_examples/tutorials/run_lda.html

[6] *MusicGen: Simple and controllable music generation*. GitHub. (n.d.). https://github.com/facebookresearch/audiocraft/blob/main/docs/MUSICGEN.md

VII. Link to GitHub

GitHub repository contains dataset, pre-processed outputs, all python notebooks used for the project (including the fine-tuning future work), along with a ReadME to explain them: https://github.com/agirishkumar/NLP-Project-lyrics-to-music

VIII. Link to Presentation Recording

https://youtu.be/_5vtP_i_Huo