
11-785 Team 23

WAME: Waveform Music Editor

A MusicLLM Model for Music Editing

Huankang Chen | Qixiao Zhu | Yuding Wang
Carnegie Mellon University

Abstract

The project, WAME.AI, introduces an advanced AI-driven music editor that enhances music editing capabilities through conversational AI interactions. By leveraging a fine-tuned MusicGen model [1] and a multitrack dataset from the "Mixing Secrets" Free Multitrack Download Library, this project uniquely utilizes LLMs [2] to generate data-driven prompts and employs automatic tag matching for more effective model training. These innovations facilitate deeper musical comprehension and dynamic editing based on complex user prompts. Preliminary results are promising, offering a solid foundation for future advancements in AI-enabled creative music editing.

Code implementations: <https://github.com/YudingWang/GenMusic>

1 Introduction

Music is a form of art that integrates expression and carries the cultures of the human race. Musical instruments are an important medium of turning thoughts into music.

In the last century, electronic music thrived, with the first introduction of analog musical instruments like the theremin around 1920 and later came the first analog synthesizer in the 1950s. Soon after, efforts were made to develop digital systems that mimicked the sounds of the analog instruments. The Yamaha DX7, for example, popularized the digital synthesizer and sparked a new era of electronic music.

Because of the meticulous structure of music, the variety of musical instruments, and the nuanced artistry involved, musical composition typically requires knowledge in Music Theory and an ample amount of time. Early attempts to understand and recreate musical behaviors algorithmically was in the 1950s [3]. Subsequently, more algorithmic systems that incorporated aspects of music theory were developed [4, 5, 6]. As deep learning became the mainstream research of artificial intelligence (AI), earlier music generation models like MelodyRNN [7], WaveNet [8] were developed and recently, MusicLDM [9] and MusicGen [1] became the new state-of-the-art. At the same time, Large Language Models (LLM) like ChatGPT [2] and Llama [10] changed how people interact with the computer, and with the introduction of DALL · E [11], people are able to use everyday language to tell ChatGPT [2] what images they want to draw and later how to edit the image.

We propose a model that leverages music generation models and LLMs to create a system that understands language input and generates music according to the user's needs. Like DALL · E [11], our model can take into account the context of the previous text prompts and let the user input additional information to edit the previously generated music. Our intention is to create an easy and manipulable tool to make music. There is little previous work done on music LLM models as such, and we would like to explore this particular direction.

2 Literature Review

The music generation field has seen significant evolution over recent decades, moving from simple probability models to the sophisticated use of deep learning techniques. Insights from surveys like "A Comprehensive Survey on Deep Music Generation" [12] and "A Survey on Deep Learning for Symbolic Music Generation" [13] provide a broad understanding of the various tasks involved in music creation at different levels.

Previously, computer-based music creation was largely rule-based, attempting to replicate traditional musical theory. However, the advent of deep learning has revolutionized this process, enabling computers to analyze vast amounts of music and generate original compositions. Deep learning models, such as RNNs [14, 15, 16], LSTMs [17], and Transformers [18], have propelled the field forward. For instance, the Transformer model has been effectively used for music infilling tasks, as seen in "Melody Infilling With User-Provided Structural Context." [19] Furthermore, the integration of Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) for music generation, inspired by dance videos in "Dance-conditioned Artistic Music Generation by Creative-GAN," [20] illustrates the creative synergy between data of multiple modalities.

The emergence of the Diffusion model, particularly when combined with the Transformer architecture, marks the latest advancement, offering unprecedented creativity and output fidelity. Notable implementations include "Audiobox: Unified Audio Generation with Natural Language Prompts," [21] showcasing sophisticated audio generation from textual prompts, "MusicGen: Simple and Controllable Music Generation" [1], and "MusicLDM: Enhancing Novelty in Text-to-Music Generation Using Beat-Synchronous Mixup Strategies," [9] which demonstrate innovative text-to-music generation approaches. Additionally, "Noise2Music: Text-conditioned Music Generation with Diffusion Models" [22] leverages a pre-trained language model to interpret prompts, further enhancing the text-conditioned music generation landscape.

Furthermore, we will also draw upon models such as those described in "Symbolic Music Generation with Diffusion Models" [23], which utilize diffusion model for music-to-music generation functionalities, serving as a foundation for our implementation. Although these models often do not incorporate the ability to accept and process prompts based on Large Language Models (LLMs), their methodologies provide crucial insights into the capabilities of diffusion models in music generation contexts.

Despite these advancements, the concept of music-to-music generation guided by human prompts remains an uncharted and challenging area. This innovative approach aims to create a system that takes existing musical pieces as inputs, rather than generating music based on human's orders from scratch, and uses human prompts to guide the re-composition or modification of the music according to specific themes or instructions. Drawing inspiration from seminal works such as "Denoising Diffusion Probabilistic Models," [24] which introduces the diffusion model for generation, and "Attention Is All You Need," [18] which thoroughly elucidates the Transformer model, our project seeks to build upon these pivotal contributions. In particular, we have recently been exploring the capability of AudioSep [25] as a potential addition to our final model for stronger editorial ability, specifically when the model is prompted to adjust the volume or delete specific voices. We will also reference papers on the most relevant and recent applications in this field that align closely with our ideas, including the previously mentioned music generation models [21, 1, 22].

By adopting the principles outlined in these landmark studies and incorporating insights from these cutting-edge applications, we aim to explore the untapped potential of prompt-driven music-to-music generation. This method promises not only to open up a new avenue for interactive musical creativity but also allows users to have a direct impact on the generation process, enabling them to customize the output to reflect their personal tastes or creative intentions.

3 Model

3.1 Baseline Model

As described in our proposal idea, our model intends to have the capability of both generating new music and editing existing music. Thus, our baseline model attempts to combine two primary parts.

The first part uses MusicGen [1], as described in the referenced paper, as the baseline model for our music generation task. MusicGen [1] is a state-of-the-art single-stage language model that generates music based on compressed discrete music representations. It introduces a novel token interleaving pattern that enables the efficient generation of high-quality mono and stereo music samples from textual descriptions or melodic prompts. This approach streamlines the music generation process by eliminating the need for multiple cascading models.

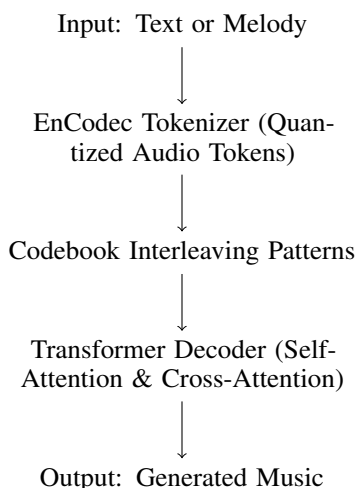
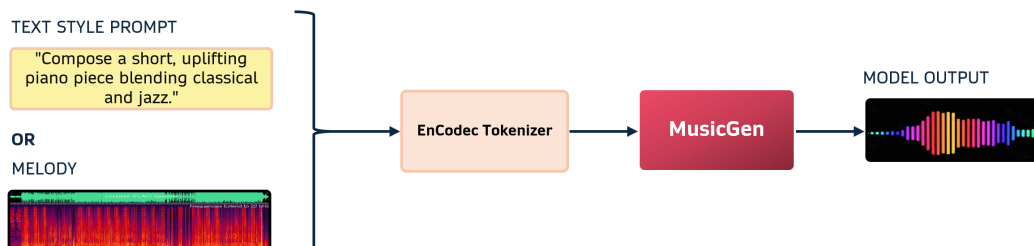


Figure 1: MusicGen Architecture Diagram [1]

The second part of the baseline model features AudioSep [25], a state-of-the-art model that takes in text user query and extracts the specified source from a mixed audio signal, to improve the editing capability of our model when prompted changing or deleting specific voices. We will likely remove the language query in our final model and stick to separating every possible voice in the mixed signal. This model separates existing music by applying a Short Term Fourier Transform (STFT) to convert the input waveform into complex spectrogram in the frequency domain and feed it into an encoder-decoder network. Loss is calculated by comparing the ground truth waveform and the generated separated waveform, both in the time domain.

Ultimately, our final product attempts to combine the two parts mentioned above into a model that contains the best of both worlds.

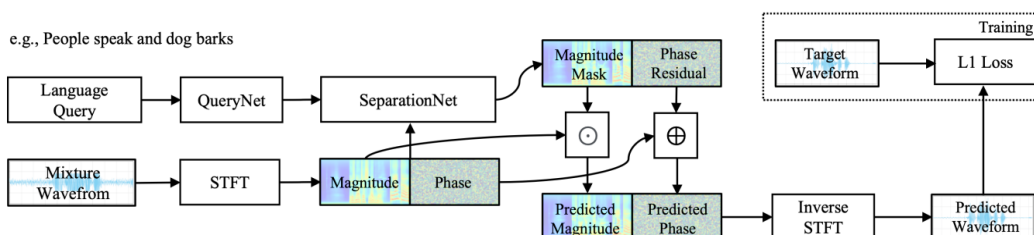


Figure 2: AudioSep Architecture Diagram [25]. "AudioSep contains two key components: a QueryNet and a SeparationNet. The QueryNet is the text encoder of CLIP [26] or CLAP [27] model. The SeparationNet is a frequency-domain ResUNet [28, 29] model." [25]

3.2 Training Objectives

Our proposed generative model [1] seeks to synthesize music by directly approximating the conditional probability distribution of musical data, X , given a context c , such as textual descriptions or melodic prompts. The training objective is formulated as the minimization of the negative log-likelihood:

$$-\log p(X|c) = -\sum_{i=1}^N \log p(x_i|x_{i-1}, \dots, x_1, c), \quad (1)$$

where x_i denotes the tokens in the sequence of the music output. This objective encourages the model to learn a representation that is faithful to the structure of the music while being able to generate novel compositions when conditioned on new prompts.

During training of the proposed separation model [25], separated voices, s_1 and s_2 , from the training set are mixed into input x while using the loudness augmentation method proposed in [29], where we calculate the energy E_1 and E_2 for both voices using $E = \|s\|_2^2$ and apply a scaling factor $\alpha = \sqrt{D_1/E_2}$ to s_2 such that $x = s_1 + \alpha s_2$. We then minimize the L1 loss between the ground truth waveform s and the separated waveform $\hat{s}$:

$$l = \|s - \hat{s}\|_1$$

3.3 Denoising Diffusion Process

In addition to the token sequence model, our approach incorporates a denoising diffusion process. This process injects noise into the data and tasks the model with recovering the original signal, thereby maximizing the evidence lower bound (ELBO):

$$\text{ELBO} = \mathbb{E}[\log p(X|c)] - D_{KL}(q(X|\tilde{X}, c) \| p(X|c)), \quad (2)$$

where D_{KL} represents the Kullback–Leibler divergence, $\tilde{X}$ is the noised data, and X is the clean music data. Through iterative refinement, the model learns to generate high-fidelity music by reversing the noise process.

3.4 Evaluation Metrics

For the evaluation of our final music generation model, we will employ both established and novel metrics to ensure a comprehensive assessment. We intend to compare our model with MusicGen [1] using the following metrics:

Objective Metrics:

- Fréchet Audio Distance (FAD): To measure the closeness of the distribution of generated audio to that of real audio.
- Kullback-Leibler (KL) Divergence: To quantify the similarity in the distribution of audio features between the generated and real audio.
- Chroma Cosine Similarity: To evaluate the harmonic structure and melody alignment in the generated music.

Subjective Metrics:

- Human Evaluation for Overall Quality (OVL): Where listeners rate the audio quality on a scale from 1 to 100.

- **Relevance to Text Input (REL):** Where listeners rate how well the generated music aligns with the given text description.

Innovative Metric - PAM[30]:

- **Prompting Audio-Language Models (PAM):** A no-reference metric for assessing audio quality by leveraging audio-text paired data and contrasting positive and negative quality descriptors.

Metric Implementation:

All metrics will be implemented as follows:

$$\text{FAD} = \text{function}(\text{generated_audio}, \text{real_audio}), \quad (3)$$

$$\text{KL} = D_{\text{KL}}(P_{\text{real}} || P_{\text{generated}}), \quad (4)$$

$$\text{PAM} = \frac{\exp(\text{similarity_score}_{\text{high_quality}})}{\exp(\text{similarity_score}_{\text{high_quality}}) + \exp(\text{similarity_score}_{\text{low_quality}})}. \quad (5)$$

The chosen metrics allow for a multi-faceted evaluation of the generated music, capturing both technical and perceptual qualities. The inclusion of PAM, in particular, provides a modern approach to audio quality assessment that does not rely on reference audio, thereby offering a fresh perspective on the perceptual quality of generated music.

4 Baseline Model Evaluation and Dataset Collection

As we advanced our project, we implemented and analyzed our baseline model. Our analysis has pinpointed several areas where the baseline model—despite its pioneering approach to music generation—falls short in delivering the nuanced and stylistically faithful musical compositions that users seek. Our project’s ambition is to navigate through these complexities and introduce substantial improvements.

4.1 Addressing Inherited Challenges

1. **Randomness and Rhythm Consistency:** The baseline MusicGen [1] model occasionally produces outputs that seem abruptly random or exhibit chaotic rhythms, detracting from the user’s experience by making the AI-generated nature of the music overtly apparent. Our modifications aim to introduce a more structured and consistent approach to randomness and rhythm.
2. **Enhanced Music Understanding:** The original model’s capacity to interpret and generate music based on abstract or nuanced stylistic instructions is limited. For instance, while capable of generating music aligned with broad genre categories, it struggles with more refined or specific stylistic imitations, such as accurately emulating the nuances of Michael Jackson’s style. Our work seeks to deepen the model’s understanding of musical styles beyond mere genre classifications.
3. **Broadening Functional Capabilities:** Beyond addressing the constraint to melody and the need for incorporating randomness, our focus extends to refining how the model interprets musical genres and elements like instrument types. The challenge of maintaining the original melody after multiple rounds of feedback and the model’s handling of negative prompts (e.g., instructions on what not to add) are areas we aim to improve.

4.2 Leveraging AudioSep [25] for Instrumental Insight

A key strategy in our approach involves integrating the AudioSep [25] model to enhance the baseline MusicGen’s [1] comprehension of music creation at a granular level. This model’s adeptness at separating different instruments from a mixed track will inform and refine MusicGen’s [1] generative process, aiming to rectify the issue of AI-signature sounds in the output and foster a more human-like composition quality. We aim to enrich the model’s understanding of music by providing more detailed, track-separated musical components combined with appropriate labeling of multi-track files. This

labeling effort will be further enhanced through the implementation of a model-driven approach, and utilizing ChatGPT’s [2] API to expand on synonyms for labels, thus achieving a model-training-model effect. Rather than producing audio content mixed with abrupt random sounds, our ultimate goal for this phase is to deepen the model’s comprehension of the various musical components, aiming to generate more aesthetically pleasing music that aligns with human tastes.

4.3 Enhanced Dataset for Targeted Fine-tuning

Our project’s stride towards achieving a nuanced music generation capability is significantly bolstered by incorporating the Cambridge Music Technology [31] dataset, a meticulously curated collection of 597 music projects specifically designed to support mixing practice and educational purposes.

The library offers an extensive array of multitrack projects available for free download, each project presented as ZIP archives containing uncompressed WAV files. These files stand out for their high fidelity, with resolutions spanning 24-bit or 16-bit and a standard sample rate of 44.1kHz. A hallmark of this dataset is the contributors’ commitment to providing audio in its "raw" form—unprocessed and devoid of additional effects beyond essential treatments applied during tracking or editing. This practice ensures the maximum mixdown flexibility and authenticity for educational and practice purposes, which ensures the simulation of human music creation.

Our project leverages this rich dataset, annotated for detailed musical characteristics such as genre, instrument types, and style, to enable more precise fine-tuning of the MusicGen [1] model. By immersing our model in the depth and diversity of the Cambridge Music Technology dataset, we aim to refine its generative capabilities, ensuring the production of music that resonates with human creativity and stylistic accuracy.

4.4 Labeling Engineering

Utilizing Large Language Models (LLMs) like ChatGPT [2] for data labeling has revolutionized the efficiency and cost-effectiveness of processing tasks. This method eliminates the need for extensive manual labeling and model training, offering a streamlined approach that can rival human performance. In our project, we propose leveraging this advanced technique to create a dataset for fine-tuning the model.

According to Data Labeling using GPT APIs [32], the process of data labeling with LLMs typically involves the following procedures: initially manually labeling a small dataset, then employing a few-shot labeling prompt for preliminary labeling, and refining this prompt based on review outcomes to enhance accuracy. This process is repeated as needed. Finally, using the final prompt to label the rest of the dataset.

This innovative approach, integrated with models like AudioSep [25], enhances MusicGen’s [1] music creation process by refining instrument separation and labeling, aiming for a more human-like composition quality and aligning with our goal of generating music that resonates with public tastes.

4.5 Future Directions

Our project is committed to refining MusicGen [1] through targeted enhancements, addressing the inherited limitations to produce music that not only meets the diverse stylistic expectations of users but does so with a level of quality and creativity that blurs the line between AI-generated and human-composed music. In the pursuit of these objectives, we aim to innovatively integrate the diffusion model with Large Language Models (LLMs), to enhance the model’s understanding of music and its ability to interpret human instructions more accurately. This integration seeks not only to enable the generation of high-quality music but also to facilitate the editing of music based on prompts provided by users. By tackling these challenges head-on, we aim to set a new standard for AI in music generation.

References

- [1] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. *arXiv:2306.05284*, 2023.
- [2] OpenAI. ChatGPT: Openai’s language model. <https://www.openai.com/research/chatgpt>, 2024. Feb 18.
- [3] Lejaren Arthur Hiller and Leonard M Isaacson. *Experimental Music: Composition with an electronic computer*. McGraw-Hill Book Co, 1959.
- [4] Steven R. Holtzman. Using generative grammars for music composition. *Computer Music Journal*, 1981.
- [5] Jon McCormack et al. Grammar based music composition. *Complex systems*, 1996.
- [6] Walter Schulze. *A formal language theory approach to music generation*. MS Thesis, University of Stellenbosch, 2009.
- [7] Elliot Waite. Generating long-term structure in songs and stories. *Google Blog*, 2016.
- [8] Sicong Huang, Qiyang Li, Cem Anil, Xuchan Bao, Sageev Oore, and Roger B Grosse. TimbreTron: A WaveNet(CycleGAN(CQT(Audio))) pipeline for musical timbre transfer. In *ICLR*, 2019.
- [9] Ke Chen, Yusong Wu, Haohe Liu, Marianna Nezhurina, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. MusicLDM: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies. *arXiv:2308.01546*, 2023.
- [10] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *arXiv:2302.13971*, 2023.
- [11] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv:2102.12092*, 2021.
- [12] Shulei Ji, Jing Luo, and Xinyu Yang. A comprehensive survey on deep music generation: Multi-level representations, algorithms, evaluations, and future directions. *arXiv:2011.06801*, 2011.
- [13] Shulei Ji, Jing Luo, and Xinyu Yang. A survey on deep learning for symbolic music generation: Representations, algorithms, evaluations, and challenges, 2023.
- [14] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature volume 323, pages533–536 (1986)*, 1986.
- [15] J.L. Elman. Finding structure in time. *Cognitive Science Volume 14, Issue 2, April–June 1990, Pages 179-211*, 1990.
- [16] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation (1997) 9 (8): 1735–1780*, 1997.
- [17] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. *arXiv:1409.3215*, 2014.
- [18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.
- [19] Chih-Pin Tan, Alvin W.Y. Su, and Yi-Hsuan Yang. Melody infilling with user-provided structural context. *arXiv:2210.02829*, 2022.
- [20] Jiang HUANG, Xianglin HUANG, Lifang YANG, and Zhulin TAO. Dance-conditioned artistic music generation by creative-gan. *2023EAP1059*, 2023.
- [21] Apoorv Vyas, Bowen Shi, Matthew Le, Andros Tjandra, Yi-Chiao Wu, Baishan Guo, Jiemin Zhang, Xinyue Zhang, Robert Adkins, William Ngan, Jeff Wang, Ivan Cruz, Bapi Akula, Akinniyi Akinyemi, Brian Ellis, Rashel Moritz, Yael Yungster, Alice Rakotoarison, Liang Tan, Chris Summers, Carleigh Wood, Joshua Lane, Mary Williamson, and Wei-Ning Hsu. Audiobox: Unified audio generation with natural language prompts. *arXiv:2312.15821*, 2023.

- [22] Qingqing Huang, Daniel S. Park, Tao Wang, Timo I. Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, Jesse Engel, Quoc V. Le, William Chan, Zhifeng Chen, and Wei Han. Noise2music: Text-conditioned music generation with diffusion models. *arXiv:2302.03917*, 2023.
- [23] Gautam Mittal, Jesse Engel, Curtis Hawthorne, and Ian Simon. Symbolic music generation with diffusion models. *arXiv:2103.16091*, 2021.
- [24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv:2006.11239*, 2020.
- [25] Xubo Liu, Qiuqiang Kong, Yan Zhao, Haohe Liu, Yi Yuan, Yuzhuo Liu, Rui Xia, Yuxuan Wang, Mark D. Plumbley, and Wenwu Wang. Separate anything you describe. *arXiv:2308.05037*, 2023.
- [26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021.
- [27] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.
- [28] Q. Kong, Y. Cao, H. Liu, K. Choi, and Y. Wang. Decoupling magnitude and phase estimation with deep resunet for music source separation. In *ISMIR*, 2021.
- [29] Q. Kong, K. Chen, H. Liu, X. Du, T. Berg-Kirkpatrick, S. Dubnov, and M. D. Plumbley. Universal source separation with weakly labelled data. *arXiv:2305.07447*, 2023.
- [30] Soham Deshmukh, Dareen Alharthi, Benjamin Elizalde, Hannes Gamper, Mahmoud Al Ismail, Rita Singh, Bhiksha Raj, and Huaming Wang. Pam: Prompting audio-language models for audio quality assessment. *arXiv:2402.00282*, 2024.
- [31] Mike Senior. Cambridge music technology. <https://cambridge-mt.com/ms/mtk/>, 2024. Accessed: March 20, 2024.
- [32] Harry Wang. Data labeling using gpt apis. <https://harrywang.me/gpt-labeling>, 2024.