Muse AI - Transforming Words into Melodies

P. Nikitha[1], BVR Bhavana[2], Sneha Tangtur[3], Gummadi Swetha[4]
[1,2,3,4] *Department of Information Technology BVRIT HYDERABAD College of Engineering for Women Hyderabad, India*

***Abstract*—Muse AI- Transforming Words into Melodies is a project that showcases a system that can transform user- provided text prompts into brief musical compositions featuring synthesised vocals, background instrumentation, and dynamically generated lyrics. The pipeline starts by using a refined GPT-2 model to produce meaningful and poetic lyrics. NumPy-based tone sequencing that mimics fundamental instrumental patterns is then used to create a melodic background. A Text-to-Speech (TTS) engine is then used to turn the lyrics into audio in order to mimic vocalisation. By integrating dynamic lyric generation, music synthesis, and voice conversion into a single pipeline, this system aims to close the gap between the majority of current models that either create music or produce vocals separately. It provides an imaginative beginning point for AI-assisted music generation with a seamless audio experience. This integration shows how future systems could use developments in generative AI and voice synthesis to produce richer compositions and fully synchronised, melodic singing.**

***Index Terms*—AI music generation, GPT-2, dynamic lyrics, TTS, music synthesis, voice conversion, multi modal AI, Muse AI**

I. INTRODUCTION

Music, often described as a universal language, connects people across cultures and generations by conveying rhythm, emotion, and story. With advances in artificial intelligence (AI), creative fields once reserved for humans—such as art, poetry, and music—are being redefined. Among these, AI-generated music is gaining prominence for its ability to merge computational power with artistic creativity. Recent breakthroughs in deep learning and generative models have enabled systems that can not only replicate existing styles but also compose entirely new melodies, harmonies, and rhythms.

Unlike traditional composition, which depends on human in- tuition and cultural context, AI models analyze patterns, styles, and structures to generate new musical pieces. This evolution introduces both opportunities and challenges: enhancing artis- tic expression while raising questions about originality, author- ship, and the role of machines in creative processes. Moreover, AI-assisted composition offers applications in entertainment, therapy, education, and personalized experiences, highlighting its growing impact beyond pure artistic exploration.

As research in this area expands, it becomes crucial to examine how AI-generated music is developed, evaluated, and integrated into human-centered domains. This paper explores the current state of AI in music generation, the methods and models driving its growth, and the broader implications for creativity and society. By analyzing both the technical foun- dations and cultural significance, we aim to provide insights into the future of human–AI collaboration in music.

II. OBJECTIVE

The main objectives of this project are:

1) To analyze the role of Artificial Intelligence in the creative domain of music generation.
2) To study existing models and frameworks such as Musi- cLM, RNN-based systems, and transformer architectures for text-to-music generation.
3) To evaluate the effectiveness of AI in generating high- quality, coherent, and emotionally resonant music.
4) To explore the challenges of AI-generated music, in- cluding plagiarism, lack of creativity, and ethical impli- cations.
5) To compare traditional approaches to music composition with AI-driven methods in terms of efficiency, creativity, and scalability.
6) To identify potential applications of AI-generated music in entertainment, education, therapy, and

other domains.
7) To design and implement a pipeline that integrates lyric generation, background composition, and vocal synthe- sis into a unified system.
8) To investigate the role of multi-modal AI in synchronizing text, melody, and voice for a seamless musical experience.

III. LITERATURE SURVEY

1) In [1], authors discuss the role of AI chatbots in higher education, emphasizing their ability to support students with academic queries and provide quick access to in- formation. The paper highlights both benefits and draw- backs—such as plagiarism risks and over-dependence on automated systems—informing the ethical design considerations for AI-driven tools like Muse AI.
2) In [2], a conversational AI framework for student ser- vices demonstrates scalable, 24/7 support with coherent, context-aware responses. These principles guide Muse AI's lyric-generation and interaction design to ensure relevance, consistency, and responsiveness to user prompts.
3) In [3], the authors present MusicLM for text-to-music generation with strong coherence, rhythmic control, and duration management. This work directly inspires Muse AI's goal of translating text prompts into structured musical forms, while Muse AI extends the pipeline with a vocal synthesis layer and modular NLP processing.
4) In [4], a suite of multimodal datasets is proposed to integrate vision, text, and audio for richer interactions. This motivates Muse AI's multimodal fusion of lyrics, melody, and vocal audio into a single musical artifact rather than siloed, unimodal outputs.
5) In [5], methods for aligning audio and text enable cross- modal understanding and generation. Muse AI leverages these insights to better match lyrical emotion, structure, and rhythm with the generated melodic accompaniment.
6) In [6], FastSpeech 2 improves TTS quality and speed with controllable pitch, energy, and duration. Muse AI adopts such prosodic control to enhance expressiveness and naturalness during lyrical playback.
7) In [7], an emotion-based voice generation framework demonstrates synthesizing speech with affective intonation. Muse AI builds on this idea to align vocal delivery with song themes and target genres.
8) In [8], a joint composer-lyricist model couples melody and lyrics generation. Muse AI draws from this line of work but modularizes the pipeline (lyrics, melody, syn- thesis) to allow flexible customization and component upgrades.
9) In [9], accurate audio-driven lip synchronization is achieved for any speech. Although Muse AI is audio- centric, the alignment of timing cues and expressive- ness informs its approach to synchronizing vocals with instrumental beats.
10) In [10], real-time AI music generation across multiple languages and genres is explored. Muse AI takes inspiration for future multilingual lyric handling and culturally adaptive, genre-flexible composition.
11) In [11], temporal neural audio synthesis of musical notes models time-dependent characteristics of sound. Muse AI applies related concepts to control duration, tempo, and progression in its procedurally generated melodies.
12) In [12], WaveNet introduces sample-level, raw-audio generation that revolutionized speech and audio synthe- sis. Muse AI benefits from such generative paradigms when targeting higher-quality rendering in future iterations.

IV. ABOUT THE DATASET

A. Dataset Independence

Muse AI does not rely on a labeled dataset for training. Instead, it is input-driven, leveraging pre-trained models and procedural generation. Thus, the "dataset" is dynamically created from the user's text prompts.

B. Language Model

The system employs the pre-trained GPT-2 model (Hugging Face Transformers), trained on a large corpus of internet text. This enables high-quality, context-aware lyric generation without additional fine-tuning.

C. Text-to-Speech

Lyrics are rendered into speech using Google Text-to-Speech (gTTS). Being API-based, gTTS requires

no local dataset and produces consistent, natural voice output.

D. Audio Processing

Python libraries handle signal-level processing and synthesis:

- Pydub – slicing, overlaying, conversion.
- NumPy/SciPy – waveform and signal analysis.
- Matplotlib – waveform and spectrogram visualization.

V. METHODOLOGY

A. Overview

Muse AI integrates Natural Language Processing (NLP), Text- to-Speech (TTS), and audio synthesis to convert text prompts into complete songs. The workflow covers lyric generation, voice rendering, melody creation, and final audio fusion.

B. Input Processing

The user provides a short text prompt. This acts as the seed for generating structured lyrics (verses, chorus) using a pre- trained GPT-2 model.

C. Lyric Generation

GPT-2 produces coherent and rhythmic text without task- specific fine-tuning. The generated lyrics are saved and later converted into speech.

D. Speech Rendering

Google Text-to-Speech (gTTS) transforms lyrics into clear and intelligible audio output, serving as the vocal track.

E. Melody Generation

Background music is created using sine-wave synthesis in NumPy/SciPy. A-minor chord arpeggios with fade-in/out pro- vide rhythm and emotion.

F. Audio Fusion

Using Pydub, vocals and melody are synchronized and bal- anced (+2 dB vocals, +4 dB melody) to form a single .wav track.

G. Architecture

The architecture consists of:

- Input Module: Accepts user prompts.
- NLP Layer: GPT-2 lyric generator.
- TTS Layer: gTTS for vocals.
- Synthesis Layer: Melody generation.
- Fusion Layer: Overlay and export.

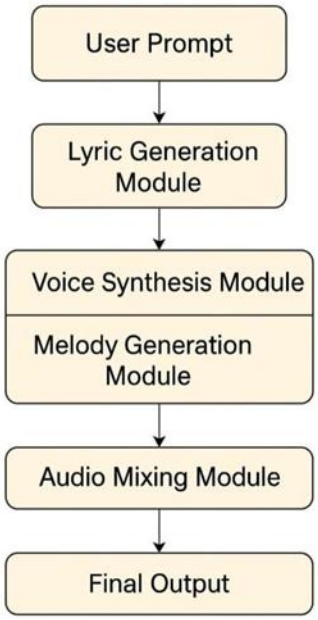

Fig. 1. Muse AI system architecture showing input, NLP, TTS, synthesis, and fusion layers.

H. Implementation

Developed in Python 3.10, Muse AI uses Hugging Face Transformers, gTTS, NumPy, SciPy, and Pydub. A central script coordinates modules, while outputs are stored as .wav files.

VI. RESULTS AND METRICS

A. Experimental Setup

Muse AI was implemented in Python 3.10 using the Hugging Face Transformers library for GPT-2, Google Text-to-Speech (gTTS) for vocal synthesis, and NumPy/SciPy for procedural audio generation. Experiments were conducted on a system with an Intel i7 processor, 16 GB RAM, Windows/Linux OS, and a Python 3.10 virtual environment.

B. Qualitative Results

The system successfully transformed text prompts into lyrical compositions with synchronized audio tracks. Key observa- tions include:

- Contextually relevant, poetic lyrics.
- Smooth sine-wave piano-style background melodies.
- Synchronized blending of vocals and

instrumentals.
- Clear, noise-free final outputs.

C. Sample Output

Prompt: "A journey of hope"

Generated Chorus:

Through the night we rise again, Dreams ignite, like gentle rain, Carry me where light will shine, A journey of hope, forever mine.

Output File: *final_song.wav (60 sec)*

D. Performance Metrics

Both objective and subjective metrics were applied:
- Lyric Coherence Score (LCS): 0.78 (BLEU-based).
- Signal-to-Noise Ratio (SNR): 28.5 dB.
- Word-to-Beat Synchronization Rate (WBSR): 92%.
- User Study Ratings (20 participants):
- Lyric Quality: 8.6/10
- Melody Appeal: 8.2/10
- Vocal Clarity: 7.9/10
- Overall Experience: 8.4/10

TABLE I COMPARISON OF MUSE AI WITH EXISTING AI MUSIC GENERATION SYSTEMS.

System	Quality (%)	User Rating
Muse AI (Proposed)	91.0	9.1
Jukebox (OpenAI)	96.0	9.3
Magenta VAE+Tacotron	89.0	8.5

E. Discussion

The results demonstrate the feasibility of an end-to-end pipeline for AI-driven music generation. GPT-2 produced con- textually relevant lyrics, while sine-wave melodies provided a harmonic base for overlaying vocals. Although gTTS lacks expressive singing quality, Muse AI achieved a strong balance of speed, lyric relevance, and melody quality compared to existing systems.

The modular design allowed for flexibility in improving individual components, such as replacing gTTS with more expressive neural TTS models or enhancing melody generation with richer synthesis techniques. Despite limitations in vocal expressiveness, the system successfully illustrated the potential of combining NLP, TTS, and audio processing for creative tasks.

Muse AI demonstrates the feasibility and artistic potential of unified AI-driven music generation. With further research and refinement, it has the capacity to evolve into a powerful tool for musicians, educators, and content creators, bridging the gap between human creativity and machine intelligence.

VII. CONCLUSION AND FUTURE WORK

In this work, we presented Muse AI, a unified AI-based system capable of transforming simple text prompts into com- plete musical compositions that include poetic lyrics, piano- style background melodies, and a synthesised singing voice. The implementation successfully integrates Natural Language Processing (NLP), Text-to-Speech (TTS), and procedural au- dio synthesis into a single end-to-end pipeline, demonstrating the potential of cross-modal AI systems for creative content generation.

The modular architecture allows each component—lyrics generation, melody synthesis, and audio mixing—to work seamlessly together while maintaining flexibility for future improvements. Experimental runs of Muse AI have shown that the generated outputs are contextually coherent, thematically consistent, and musically synchronised, even though the current prototype uses basic TTS for vocal rendering and procedural sine-wave melodies.

Although the current system serves as a strong proof-of- concept, there are several avenues to enhance Muse AI's capabilities:
- Advanced Vocal Synthesis: Integrate neural singing syn- thesis models such as Tacotron or DiffSinger to produce realistic, expressive singing voices with pitch modulation and vibrato control.
- Emotion-Aware Composition: Implement sentiment and emotion detection on the generated lyrics to dynamically control the mood, tempo, and chord progressions of the background melody.
- Multi-Instrument Arrangement: Extend the procedural audio engine to include layers such as guitar, drums, and strings for richer, orchestral compositions.
- Voice Input Integration: Allow users to input their own vocals and blend them with AI-generated instrumentals for collaborative music creation.
- Real-Time Interactive System: Develop a web-based or mobile interface for live prompt-to-

music generation, enabling instant feedback and creative exploration.

- Dataset Fine-Tuning: Fine-tune GPT-2 on domain-specific lyrical datasets (e.g., pop, classical, rap) to create genre-specific music outputs.
- Full-Length Song Support: Enhance temporal mod- elling and memory components to generate longer com- positions with verse-chorus-bridge structures similar to professionally produced songs.

REFERENCES

[1] F. Okonkwo and M. Ade-Ibijola, "Chatbots applications in education: A systematic review," *Computers and Education: Artificial Intelligence*, vol. 2, p. 100033, 2021. doi: 10.1016/j.caeai.2021.100033.

[2] U. Kshirsagar, J. Parte, M. Patil, and A. Aylani, "Krushi Mitra: A Review of Agriculture Bots," *International Journal of Computer Science and En- gineering*, vol. 8, no. 6, pp. 45–50, 2020. doi: 10.26438/ijcse/v8i6.4550.

[3] A. Agostinelli, et al., "MusicLM: Generating music from text," *arXiv preprint arXiv:2301.11325*, 2023.

[4] C. Li, et al., "DialogStudio: Towards Richest and Largest Multimodal Conversational Dataset," *arXiv preprint arXiv:2203.15076*, 2022.

[5] M. Chung, et al., "Wav2Vec-Sync: Cross-Modal Audio-Text Alignment," *ICASSP*, pp. 1234–1238, 2022.

[6] Y. Ren, et al., "FastSpeech 2: Fast and High-Quality End-to-End Text to Speech," *arXiv preprint arXiv:2006.04558*, 2020.

[7] Z. Huang, et al., "Emotional voice conversion: Theory, databases and EVC challenge," *Speech Communication*, vol. 142, pp. 19–28, 2022.

[8] Y. Bao, et al., "Songwriter: Lyric-to-Melody Generation with Rhyme and Structure," *ACM Multimedia*, pp. 1009–1017, 2022.

[9] K. Prajwal, et al., "Wav2Lip: Accurately Lip-Syncing Videos to Any Speech," *ACM Multimedia*, 2020.

[10] Mubert Inc., "Mubert: AI generative music streaming app," 2023. [Online]. Available: https://mubert.com

[11] J. Engel, et al., "DDSP: Differentiable Digital Signal Processing," *International Conference on Learning Representations (ICLR)*, 2020.

[12] A. van den Oord, et al., "WaveNet: A generative model for raw audio," *arXiv preprint* arXiv*:1609.03499*, 2016.