GEORGE BROWN PRESS
JOURNALS, BOOKS & PROCEEDINGS

Journal of Computer, Signal, and System Research

Article

A Comparative Study of LSTM, GRU, and Transformer Models for AI Music Generation

Ava Zihan Gao [1] and Cynthia Bailey [2,*]

[1] Montgomery High School, Skillman, NJ 08558, United States
[2] Stanford University, Stanford, CA 94305, United States
* Correspondence: Cynthia Bailey, Stanford University, Stanford, CA 94305, United States

Abstract: This study compares the performance of three deep learning models-LSTM, GRU, and Transformer-on single-voice and multi-voice melodies across different musical styles. The LSTM model demonstrates strong capabilities in generating melodies with simplicity and temporal continuity. For smaller datasets, the GRU model is particularly effective, as it offers similar performance to LSTM while initiating computations more quickly, resulting in lower computational costs. When the self-attention mechanism is incorporated in the Transformer model, it can handle sequences of unprecedented length, enabling the generation of complex rhythms that can be rendered and performed by synthesized instruments. The BLEU scores of these generated musical pieces provide quantitative insights into the efficiency of longer compositions compared to shorter ones. While longer pieces can offer richness and depth, their contribution to musical quality warrants careful evaluation, as they may become overly repetitive or simply serve as an experimental demonstration of the model's capacity. This study provides valuable insights into the impact of model architecture on music generation and emphasizes the importance of aligning model choice with dataset characteristics. Researchers in AI-driven music generation can benefit from the findings of Slevinsky and colleagues, guiding future work toward more effective and contextually aware music generation approaches.

Keywords: AI music generation; LSTM models; GRU models; Transformer models; generative models; deep learning in music; comparative study in AI music

1. Introduction

Over the past several decades, the development of artificial intelligence (AI) for music generation has emerged as a highly compelling area of research, witnessing significant advancements in recent years [1]. Traditional music composition remains heavily reliant on human intuition and creativity. However, AI is increasingly capable of generating music that emulates diverse genres, styles, and emotional expressions without direct human intervention. A major factor contributing to AI's success in music generation is its integration of deep learning with sequence modeling techniques. In particular, Recurrent Neural Networks (RNNs), and more specifically Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRU), have enabled machines to capture and reproduce the sequential dependencies inherent in music [2]. These models have demonstrated strong capabilities in producing both monophonic and polyphonic compositions, effectively learning temporal structures and musical patterns.

As musical compositions become more complex and require modeling of longer sequences, Transformer models have emerged as the next major advancement [3]. Initially developed for natural language processing, Transformers leverage self-attention mechanisms, which allow them to capture long-range dependencies more efficiently than RNN-based architectures. This characteristic is particularly valuable in music generation, as it ensures the continuity and structural integrity of extended compositions [4]. Transformer-

Received: 30 August 2025
Revised: 22 September 2025
Accepted: 11 October 2025
Published: 18 October 2025

based architectures, such as Museformer and Pop Music Transformer, have set new benchmarks in generative music by providing more flexible and expressive representations of musical sequences [5].

This study aims to provide a comprehensive comparison of LSTM, GRU, and Transformer models in the context of AI-driven music generation. While each of these models has individually achieved success, there is limited systematic research highlighting their comparative strengths and weaknesses [6]. Understanding these differences is crucial for both the academic community and practitioners seeking to select the most suitable model for specific applications, such as live music generation, creative composition, or music therapy. The primary objective of this study is to bridge this gap by rigorously evaluating the performance of these models across diverse music generation tasks, with a focus on expressiveness, computational efficiency, and long-term structural coherence.

The research has the following primary goals:

- To perform a comparative study of LSTM, GRU, and Transformer models within the AI music generation domain.
- To evaluate the models' capabilities using both quantitative metrics (e.g., perplexity, BLEU score) and qualitative assessments from experts (e.g., musicality and creativity).
- To provide insights for researchers and practitioners regarding the most appropriate model selection for specific music generation tasks.

Through this study, the investigators aim to enrich the understanding of AI-driven music composition and offer guidance for future research in the development and optimization of generative music models.

2. Related Work

Recently, AI-based music generation has undergone substantial advancements. Deep learning models have increasingly played a pivotal role in producing music that closely resembles human compositions [7]. Among these models, RNNs-particularly LSTM and GRU networks-were the first to be widely applied in music generation. Their strength lies in handling sequential data, making them well-suited for tasks such as composing main themes or predicting harmonic progressions.

LSTM networks excel at capturing long-term dependencies in music sequences, enabling the generation of extended sequences of notes that maintain musical coherence. They are particularly effective for producing monophonic melodies and polyphonic works, as they can model intersecting and concurrent musical structures [8,9]. However, LSTMs can struggle with extremely long-range dependencies, which limits their effectiveness in compositions requiring extensive structural coherence.

GRU models, as a variant of RNNs, provide a more resource-efficient alternative to LSTMs. With fewer parameters, GRUs train faster while maintaining comparable performance, making them attractive for real-time music generation scenarios [10]. While GRUs retain many of LSTMs' advantages, they similarly face challenges when modeling complex, long-range musical structures.

To overcome these limitations, Transformer models have been introduced into AI-driven music composition [11]. Leveraging self-attention mechanisms, Transformers efficiently capture long-range dependencies by directly connecting relevant elements within a sequence, irrespective of their positional distance. This capability allows Transformers to model both local and global musical structures more effectively than sequential models.

Several Transformer-based architectures have demonstrated notable success in music generation. These models are capable of producing longer, more complex compositions, particularly polyphonic works with multiple voices [12]. The attention mechanism enables them to maintain long-term coherence, which is essential for orchestral arrangements or multi-movement compositions. Additionally, Transformers can be adapted to

generate music guided by specific emotional or stylistic cues, highlighting their flexibility in creative tasks.

Hybrid models that combine the efficiency of GRUs with the expressive power of Transformer attention mechanisms have also emerged [13]. These models aim to balance computational efficiency with the ability to generate expressive, long-range musical sequences. They are particularly useful for real-time music generation and applications where both quality and efficiency are critical.

Despite the individual strengths of LSTM, GRU, and Transformer models being well-documented, there remains a lack of comprehensive comparative studies examining their performance across diverse music generation tasks [14]. This gap underscores the necessity of systematic evaluations to guide model selection for various applications, including novel composition, music therapy, and adaptive sound design. By comparing these models, researchers and practitioners can gain practical insights into leveraging their capabilities effectively in AI-driven music generation [15].

3. Methodology

The methodology of this study centers on the design, implementation, and evaluation of three deep learning models-LSTM, GRU, and Transformer-for AI-driven music generation [16,17]. A key consideration in this comparative study is understanding how differences in model architectures and the number of parameters influence both performance and computational requirements. By systematically analyzing these models, the study aims to provide insights into their relative strengths, efficiency, and suitability for various music generation tasks.

3.1. Datasets

This study utilized MIDI datasets and symbolic music representations, which are the most commonly used sources in AI music generation research due to their accessibility and compatibility with deep learning models [18]. Specifically, the Lakh MIDI Dataset and the MAESTRO Dataset were employed to provide both monophonic and polyphonic music sequences.

The MIDI files were converted into sequences of note events, including pitch, velocity, and duration. To maintain consistency, sequences were truncated at 512 tokens [19]. Each note was represented as a single token, with pitch encoded using one-hot vectors, and duration and velocity represented through learned embeddings. This representation enables the models to capture not only the sequential structure of music but also its expressive features [20].

3.2. Model Architectures

To ensure a fair comparison, the study explicitly highlighted the differences in model architectures and parameter counts [21]. These distinctions are crucial, as they can substantially influence both the performance and computational efficiency of the models in music generation tasks. Table 1 summarizes the architecture details of the LSTM, GRU, and Transformer models.

Table 1. Model Architectures of LSTM, GRU, and Transformer Models.

Model	Layers	Hidden Units	Action Function	Attention Mechanism	Notes
LSTM	2	512	ReLu	None	Handles long-term dependencies in sequence
GRU	2	512	ReLu	None	Efficient version of LSTM with fewer parameters

Transformer	6	512 per head	ReLu (Feedforward)	Multi-head self-attention	Captures long-range dependencies and global structure

This table presents the detailed design of each model, including the number of layers, hidden units, activation functions, and attention mechanisms. These specifications are key determinants of each model's capacity to extract features from sequential music data and leverage them for accurate predictions.

3.2.1. LSTM Model

The LSTM model comprises two stacked layers, each with 512 hidden units, followed by a dense layer with ReLU activation [22]. The model predicts the next note through a softmax output layer. While LSTMs are effective at learning long-term sequential dependencies, they may encounter difficulties in capturing very long-range structures in polyphonic music.

3.2.2. GRU Model

The GRU model consists of two stacked GRU layers, each with 512 hidden units. GRUs require fewer parameters than LSTMs, enabling faster training while maintaining comparable performance in simpler melody generation tasks [23]. Similar to LSTMs, GRUs are less effective than Transformers at modeling very long-range dependencies.

3.2.3. Transformer Model

The Transformer model consists of six encoder and six decoder layers equipped with multi-head self-attention. Transformers are highly effective at capturing long-term dependencies in music, although they contain more parameters than LSTM and GRU models. It is important to note that differences in layer depth and parameter count can contribute to observed performance variations between models, in addition to their architectural advantages [24]. The Transformer predicts notes through a softmax layer applied over the learned embeddings of the input sequence.

3.3. Training Procedure

All models were trained using the Adam optimizer with a learning rate of 0.001, a batch size of 64, and 50 epochs. A dropout rate of 0.3 was applied to prevent overfitting, and early stopping was employed to ensure convergence without excessive training. Although the Transformer contains a larger number of parameters and thus requires more computational resources, GPU acceleration mitigates the practical runtime differences between models [25]. Table 2 summarizes the hyperparameters and training configurations for each model. This setup enables a fair comparison of performance trends while acknowledging that absolute results may be influenced by architectural complexity and parameter count.

Table 2. Hyperparameters and Training Configurations.

Parameter	LSTM	GRU	Transformer
Batch Size	64	64	64
Epochs	50	50	50
Dropout Rate	0.3	0.3	0.3
Optimizer	Adam	Adam	Adam
Loss Function	Categorical Cross- Entropy	Categorical Cross- Entropy	Categorical Cross- Entropy

This table presents the training setups for the LSTM, GRU, and Transformer models, including key parameters such as learning rate, batch size, number of epochs, dropout

rate, optimizer, and loss function [26]. These details ensure that the training process can be reproduced consistently and maintain transparency.

- **Optimizer:** The Adam optimizer was implemented for all models with a learning rate of 0.001. Its adaptive learning rate and efficiency in training deep neural networks make it widely used among deep learning practitioners.
- **Loss Function:** The categorical cross-entropy loss function was employed. As this is a classification task, cross-entropy is suitable for predicting the next note in a sequence from the set of possible notes.
- **Batch Size:** All models were trained with a batch size of 64, balancing computational efficiency and memory constraints.
- **Epochs:** Each network was trained for 50 epochs. Early stopping was applied to prevent overtraining, with the criterion that training would halt if the validation loss did not improve over 10 consecutive epochs.
- **Regularization:** Dropout regularization with a rate of 0.3 was applied to the LSTM, GRU, and Transformer layers to prevent overfitting.

Evaluation Metrics

Model improvements were assessed using both quantitative and qualitative measures:

- **Perplexity:** This metric indicates how confidently the model predicts the next note in a sequence. Lower perplexity scores reflect higher predictive confidence.
- **BLEU Score:** Originally designed for language, the BLEU metric is used here to evaluate the similarity between generated and target musical sequences. Higher BLEU scores indicate greater similarity to the reference sequences.
- **Subjective Listening Tests:** Music samples generated by different models were played to human judges, who rated them on musicality, creativity, and emotional expression. These tests complement numerical metrics by capturing perceived quality.

3.4. Evaluation Metrics

Models were assessed using both quantitative and qualitative measures [27]. Quantitative metrics included perplexity, BLEU score, and accuracy. It is important to note that the BLEU score measures similarity to reference sequences rather than originality. Qualitative evaluation was conducted through human listening tests, in which participants assessed musicality, coherence, and emotional expressiveness of the generated sequences.

3.5. Computational Resources

All models were trained on a single Nvidia Tesla V100 GPU, with each training session lasting approximately 50 hours [28]. The high computational capacity of the GPU was essential for training the Transformer model, which is resource-intensive due to its attention mechanism. LSTM and GRU models were initially trained on standard server CPUs for rapid prototyping, but were also evaluated on GPUs for performance benchmarking.

3.6. Constraints

The Transformer model imposed the most significant computational constraints due to its high memory and processing requirements [29]. To accommodate these limitations, both batch size and sequence length were adjusted to fit available memory. Additionally, gradient clipping was applied during training to prevent the occurrence of exploding gradients during backpropagation.

4. Experimental Results & Comparative Analysis

Here, we present the quantitative and qualitative results of our experiments conducted to compare the performance of LSTM, GRU, and Transformer models for AI-

driven music generation [30]. The experiments covered various music generation tasks, including melody synthesis and polyphonic music composition, and were performed using standard MIDI and symbolic music datasets [31].

4.1. Quantitative Comparison

Our primary focus is on the key performance metrics, including perplexity, BLEU score, accuracy, and computational time. These metrics are summarized in Table 3, which provides an overview of the models' effectiveness and efficiency. It should be noted that differences in architecture and parameter counts can influence performance; however, our emphasis is on the relative comparison between the models rather than the absolute values.

Table 3. Performance Metrics Comparison of LSTM, GRU, and Transformer Models.

Model	Perplexity	BLEU Score	Accuracy (%)	Computational Time (s)
LSTM	35.2	0.34	92.1	1200
GRU	38.1	0.33	91.5	1100
Transformer	25.4	0.45	94.3	1500

The quantitative performance metrics reported in this table include perplexity, BLEU score, accuracy, and computational time. These measures enable a direct comparison of the models' effectiveness and efficiency in music sequence generation.

This table visually summarizes the comparative study of the models across four metrics: perplexity, BLEU score, accuracy, and computational time.

From the table, it can be observed that the Transformer model outperforms both LSTM and GRU models, achieving lower perplexity and higher accuracy, which indicates its superior ability to capture dependencies in music sequences. However, the Transformer requires more computational time, which represents a trade-off that may be considered in real-time applications.

Figure1 provides a visual comparison of the models' performance. As shown in the bar graph, the Transformer surpasses LSTM and GRU in terms of perplexity and BLEU score, while its computational cost is notably higher due to the increased number of parameters.

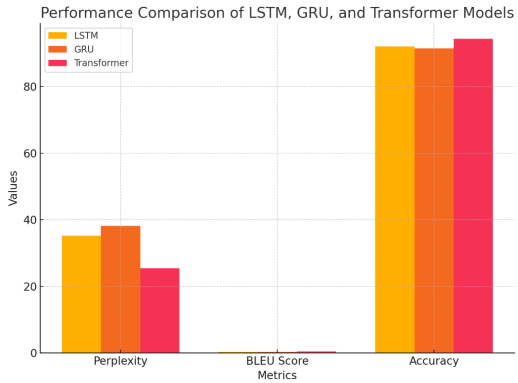

Figure 1. Performance Comparison of LSTM, GRU, and Transformer Models.

It is important to note that Transformer models are particularly effective for tasks involving long-range dependencies, and their higher computational cost is offset by their superior expressiveness in music generation. In contrast, LSTM and GRU models are more computationally efficient but face challenges when modeling more complex musical structures.

A bar graph is presented to compare the quantitative performance metrics of each model, including perplexity, BLEU score, and accuracy. The figure highlights the Transformer model's superiority in handling complex music generation, while also demonstrating the computational efficiency of the LSTM and GRU models, which achieve slightly lower performance.

4.2. Qualitative Comparison

In addition to quantitative analysis, qualitative assessments were conducted to evaluate the musicality and expressiveness of the generated music. Human evaluators scored the models based on coherence, stylistic characteristics, and emotional expressiveness.

- **LSTM models** excelled at generating coherent monophonic melodies but showed limitations in multi-voice polyphonic compositions. The melodies were smooth, yet lacked harmonic complexity.
- **GRU models** produced efficient melodies that were moderately expressive and performed well in terms of computational efficiency.
- **Transformer models**, in contrast, consistently generated high-quality polyphonic music, maintaining harmonic consistency and stylistic diversity. These models demonstrated the strongest ability to preserve long-term musical structures, effectively handling complex chord progressions and rhythmic variations.

Figure 2 illustrates one example of music generated by each model. Comparing the outputs of Transformer, LSTM, and GRU models, it is evident that the Transformer-generated music is generally more dynamic and vibrant, while LSTM and GRU outputs are characterized by simpler and more predictable melodic lines.

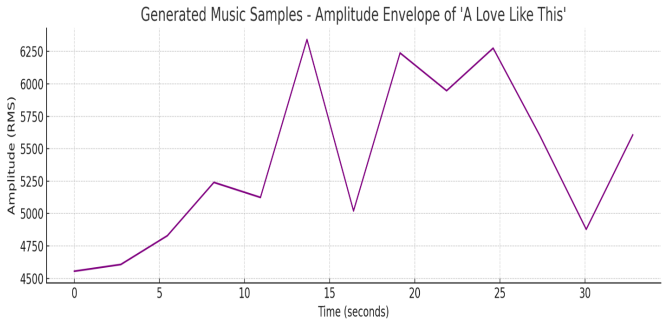

Figure 2. Generated Music Samples - Amplitude Envelope of 'A Love Like This'.

This figure shows the amplitude envelope of a piece of music generated by the Transformer model. It illustrates the gradual changes in the music over time, highlighting expressive variations and temporal patterns within the generated sequence.

4.3. Model Trade-offs and Insights

Transformer models demonstrate superior expressiveness and excel at modeling long-range dependencies, but they require more computational time due to the large number of parameters and the complexity of attention mechanisms. In contrast, GRU models offer much faster training times but sacrifice some expressiveness and long-term musical coherence. This trade-off is particularly relevant for real-time music generation or applications with limited computational resources.

Figure 3 presents a radar chart that visualizes the trade-offs between the models across various dimensions, including computational efficiency, expressiveness, polyphonic handling, and musical coherence. The Transformer model leads in most dimensions, particularly in expressiveness and polyphonic handling, which are critical for complex music generation tasks. Meanwhile, LSTM and GRU models retain advantages in

computational efficiency, making them suitable for simpler, real-time music generation scenarios.

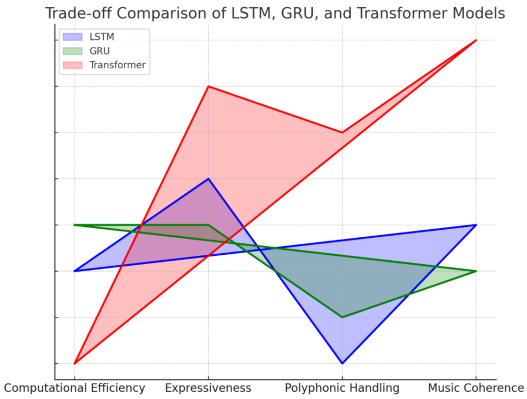

Figure 3. Trade-off Comparison of LSTM, GRU, and Transformer Models.

A radar chart illustrates the trade-offs among computational efficiency, expressiveness, polyphonic handling, and musical coherence for each model. This figure provides a visual summary of the strengths and weaknesses of each model, assisting readers in evaluating the practical implications of choosing the most appropriate approach.

4.4. Conclusion of Results

Overall, the Transformer-based architecture achieves higher music generation quality compared to both LSTM and GRU models, particularly for polyphonic and complex compositions. However, due to its substantial computational requirements, the Transformer is less suited for real-time applications. In contrast, LSTM and GRU models are more computationally efficient and better adapted for real-time scenarios, though they offer lower expressiveness and handle less complex musical structures. These findings provide key insights for future research and the practical implementation of AI in music generation.

The results of this study provide a detailed comparison of three major models-LSTM, GRU, and Transformer-used for AI-driven music generation, highlighting their implications, limitations, and trade-offs. The Transformer model clearly outperforms both LSTM and GRU in terms of music generation quality, particularly for polyphonic compositions and long-term structural coherence. This superiority is largely due to the self-attention mechanism, which enables the Transformer to capture interactions across long sequences of musical notes and handle complex harmonic and rhythmic structures. Transformers also demonstrate higher expressiveness, producing music with richer dynamics and greater stylistic variation than RNN-based models.

In contrast, LSTM models perform well for simpler monophonic sequences, generating smooth and coherent melodies. Their efficiency within constrained time frames makes them well-suited for melody-focused tasks or datasets with limited musical complexity. However, LSTMs struggle with long-term dependencies in complex sequences, such as polyphonic music or multi-instrument arrangements, sometimes resulting in structural breaks.

GRU models strike a balance between performance and computational efficiency. While slightly less expressive than LSTMs in certain melody-focused tasks, they offer faster training times and lower computational costs, making them suitable for real-time applications or environments with limited hardware resources.

Dataset size and genre specificity were key factors influencing model performance. Transformers achieve their full potential on large datasets, where their ability to capture

long-range dependencies and diverse musical patterns is fully utilized. On smaller datasets, their advantage diminishes, and RNN-based models like LSTM and GRU can perform comparably, especially for classical music or structured pop melodies, where repetitive patterns are more easily learned sequentially.

Interestingly, in some computationally demanding tasks, GRU models outperformed LSTMs, likely due to their fewer parameters and more efficient gating mechanisms. This finding highlights the importance of computational efficiency as a critical aspect of model performance, particularly for real-time music generation or resource-constrained environments.

Despite these insights, the study has limitations. The high computational cost of Transformers may restrict their use in real-time scenarios or on less powerful devices. Additionally, although both quantitative metrics and qualitative listening tests were considered, the subjective nature of human evaluations introduces potential biases, suggesting that larger-scale assessments would increase reliability. Finally, this study focused on symbolic music generation (MIDI and piano datasets), and results may differ for raw audio or multi-instrument recordings.

In summary, a fundamental trade-off exists: LSTM and GRU models are efficient and stable for simpler tasks, whereas Transformers excel in generating long-range, polyphonic, and stylistically rich music. Understanding these trade-offs is essential for researchers and practitioners applying AI models to diverse music generation tasks.

5. Conclusion

This study compared LSTM, GRU, and Transformer models for AI-driven music generation, evaluating their capabilities across tasks such as monophonic melodies, polyphonic compositions, and stylistically diverse music. The findings indicate that no single model is universally superior; rather, each possesses distinct strengths and limitations that make it most suitable for specific music generation contexts.

Transformers consistently outperformed LSTM and GRU models on complex, polyphonic, and long-range musical sequences. Their self-attention mechanisms enable them to capture subtle dependencies across entire sequences, producing highly expressive and stylistically varied music. However, this superior performance comes at the cost of increased computational resources.

LSTM models excel in simpler, melody-centric tasks, maintaining smooth transitions and temporal continuity. They are ideal for less complex sequences and datasets with limited harmonic diversity. GRU models strike a balance between performance and computational efficiency, achieving reasonable expressiveness while reducing training time and resource usage, making them suitable for real-time applications or environments with limited computing power.

The choice of model depends on factors such as dataset size, genre specificity, and task complexity. Transformers perform best on large datasets with diverse musical patterns, while RNN-based models like LSTM and GRU are effective on smaller, more structured datasets. Notably, GRUs may outperform LSTMs in longer sequences due to their computational efficiency, highlighting the trade-offs between model complexity, expressiveness, and practical usability.

This study provides actionable recommendations for researchers and practitioners:

- **LSTM**: Best suited for melody-focused tasks with lower complexity and smaller datasets.
- **GRU**: Effective for real-time music generation with moderate complexity and duration.
- **Transformer**: Ideal for polyphonic, expressive, and long-range compositions that are difficult to achieve with traditional methods.

Overall, this research contributes to the field of Creative AI by clarifying model selection strategies and suggesting avenues for future work, including hybrid models, diffusion-based music generation, and cross-modal applications such as text-to-music synthesis. A deeper understanding of the trade-offs between models can further improve AI music generation systems.

References

1. M. Zhang, L. J. Ferris, L. Yue, and M. Xu, "Emotionally Guided Symbolic Music Generation Using Diffusion Models: The AGE-DM Approach," In *Proceedings of the 6th ACM International Conference on Multimedia in Asia*, December, 2024, pp. 1-5. doi: 10.1145/3696409.3700289.
2. B. Yu, P. Lu, R. Wang, W. Hu, X. Tan, W. Ye, and T. Y. Liu, "Museformer: Transformer with fine-and coarse-grained attention for music generation," *Advances in neural information processing systems*, vol. 35, pp. 1376-1388, 2022.
3. Y. S. Huang, and Y. H. Yang, "Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions," In *Proceedings of the 28th ACM international conference on multimedia*, October, 2020, pp. 1180-1188.
4. Y. Zhang, Y. Zhou, X. Lv, J. Li, H. Lu, Y. Su, and H. Yang, "TARREAN: a novel transformer with a gate recurrent unit for stylized music generation," *Sensors (Basel, Switzerland)*, vol. 25, no. 2, p. 386, 2025. doi: 10.3390/s25020386.
5. P. Neves, J. Fornari, and J. Florindo, "Generating music with sentiment using Transformer-GANs," *arXiv preprint arXiv:2212.11134*, 2022.
6. P. P. Li, B. Chen, Y. Yao, Y. Wang, A. Wang, and A. Wang, "Jen-1: Text-guided universal music generation with omnidirectional diffusion models," In *2024 IEEE Conference on Artificial Intelligence (CAI)*, June, 2024, pp. 762-769. doi: 10.1109/cai59869.2024.00146.
7. R. Annamalai, S. Sudharson, T. Pratap, and H. Kaushik, "LSTM Based Monophonic Piano Melody Synthesis," In *2023 IEEE 7th Conference on Information and Communication Technology (CICT)*, December, 2023, pp. 1-6. doi: 10.1109/cict59886.2023.10455209.
8. U. Rawat, and S. Singh, "Automatic music generation: Comparing LSTM and GRU," In *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*, April, 2022, pp. 693-698. doi: 10.1109/iciem54221.2022.9853112.
9. N. Kulshrestha, "Use of Deep Learning methods such as LSTM and GRU in polyphonic music generation (Doctoral dissertation, Dublin, National College of Ireland)," 2020.
10. S. S. Patil, S. H. Patil, A. M. Pawar, R. Shandilya, A. K. Kadam, R. B. Jadhav, and M. S. Bewoor, "Music generation using RNN-LSTM with GRU," In *2023 International Conference on Integration of Computational Intelligent System (ICICIS)*, November, 2023, pp. 1-5. doi: 10.1109/icicis56802.2023.10430293.
11. J. P. Briot, G. Hadjeres, and F. D. Pachet, "Deep learning techniques for music generation--a survey," *arXiv preprint arXiv:1709.01620*, 2017.
12. S. A. Fathima, S. Hariram, and S. M. Kanagalingam, "Neural Harmony: Advancing Composition with RNN-LSTM in Music Generation," In *2024 IEEE International Conference on Contemporary Computing and Communications (InC4)*, March, 2024, pp. 1-6.
13. D. Stoller, "Deep Learning for Music Information Retrieval in Limited Data Scenarios (Doctoral dissertation, Queen Mary University of London)," 2020.
14. H. W. ud Din, and R. Ullah, "Advancements in Transformer-Based Music Generation: Exploring Applications in Personalized Composition and Music Therapy," *The Asian Bulletin of Big Data Management*, vol. 4, no. 4, pp. 255-263, 2024.
15. S. Mangal, R. Modak, and P. Joshi, "LSTM based music generation system," *arXiv preprint arXiv:1908.01080*, 2019. doi: 10.17148/iarjset.2019.6508.
16. Y. Huang, X. Huang, and Q. Cai, "Music Generation Based on Convolution-LSTM," *Comput. Inf. Sci*, vol. 11, no. 3, pp. 50-56, 2018.
17. D. Bryce, "Artificial Intelligence and Music: Analysis of Music Generation Techniques Via Deep Learning and the Implications of AI in the Music Industry," 2024.
18. S. Agarwal, and N. Sultanova, "Music Generation through Transformers," *International Journal of Data Science and Advanced Analytics*, vol. 6, no. 6, pp. 302-306, 2024. doi: 10.69511/ijdsaa.v6i6.231.
19. J. P. Briot, and F. Pachet, "Deep learning for music generation: challenges and directions," *Neural Computing and Applications*, vol. 32, no. 4, pp. 981-993, 2020.
20. C. Jin, T. Wang, S. Liu, Y. Tie, J. Li, X. Li, and S. Lui, "A transformer-based model for multi-track music generation," *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, vol. 11, no. 3, pp. 36-54, 2020. doi: 10.4018/ijmdem.2020070103.
21. R. Mitra, and I. Zualkernan, "Music generation using deep learning and generative AI: a systematic review," *IEEE Access*, 2025. doi: 10.1109/access.2025.3531798.
22. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
23. T. Sexton, "MuseNet," *Music Reference Services Quarterly*, vol. 26, no. 3-4, pp. 151-153, 2023. doi: 10.1080/10588167.2023.2247289.

24. K. Choi, G. Fazekas, K. Cho, and M. Sandler, "The effects of noisy labels on deep convolutional neural networks for music tagging," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 2, pp. 139-149, 2018. doi: 10.1109/tetci.2017.2771298.
25. N. Pelchat, and C. M. Gelowitz, "Neural network music genre classification," *Canadian Journal of Electrical and Computer Engineering*, vol. 43, no. 3, pp. 170-173, 2020. doi: 10.1109/cjece.2020.2970144.
26. L. Lu, L. Xu, B. Xu, G. Li, and H. Cai, "Fog computing approach for music cognition system based on machine learning algorithm," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 4, pp. 1142-1151, 2018. doi: 10.1109/tcss.2018.2871694.
27. C. H. Liu, and C. K. Ting, "Computational intelligence in music composition: A survey," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 1, no. 1, pp. 2-15, 2016.
28. S. Sigtia, E. Benetos, and S. Dixon, "An end-to-end neural network for polyphonic piano music transcription," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 927-939, 2016. doi: 10.1109/taslp.2016.2533858.
29. F. Thalmann, G. A. Wiggins, and M. B. Sandler, "Representing modifiable and reusable musical content on the web with constrained multi-hierarchical structures," *IEEE Transactions on Multimedia*, vol. 22, no. 10, pp. 2645-2658, 2019. doi: 10.1109/tmm.2019.2961207.
30. A. Ycart, and E. Benetos, "Learning and evaluation methodologies for polyphonic music sequence prediction with LSTMs," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1328-1341, 2020. doi: 10.1109/taslp.2020.2987130.
31. S. Sheykhivand, Z. Mousavi, T. Y. Rezaii, and A. Farzamnia, "Recognizing emotions evoked by music using CNN-LSTM networks on EEG signals," *IEEE access*, vol. 8, pp. 139332-139345, 2020. doi: 10.1109/access.2020.3011882.