e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

MUSIFY-AI DRIVEN MUSIC GENERATION

S. Kishore Babu*1, Pentela Sri Bharath*2, Nallamothu Venkata Avinash*3, Muchu Morahar*4, Vuyyuru Tarakanadh*5, Matcha Tribhuvan*6

*1Assistant Professor, Artificial Intelligence & Data Science ,Vasireddy Venkatadri Institute of Technology, Guntur, Andhra Pradesh, India.

*2,3,4,5,6Final Year B.Tech Student, Artificial Intelligence & Data Science ,Vasireddy Venkatadri Institute of Technology, Guntur, Andhra Pradesh, India.

DOI : https://www.doi.org/10.56726/IRJMETS71650

ABSTRACT

This study has been undertaken to investigate the application of deep learning techniques for automated music generation using AI. The proposed system Musify , leverages a Recurrent Neural Network with Long Short-Term Memory (LSTM) units to learn musical patterns from a dataset of MIDI files encoded in kern notation. For training, the model processes and encodes the musical data to capture the underlying structure of melodies. To test the efficacy of the system, user-defined seed melodies and adjustable parameters such as the number of generation steps and creativity (temperature) are employed. The generated output is saved as a MIDI file, which can then be visualized using MuseScore Studio 4 for further analysis and refinement. The analytical framework comprises model training, interactive user evaluation, and performance analysis of the generated musical compositions.

Keywords: Artificial Intelligence ,Music Generation, LSTM Networks, Deep Learning, MIDI Processing, Streamlit UI.

I. INTRODUCTION

The art of music composition has traditionally been the realm of skilled musicians and composers, relying heavily on human creativity, extensive training, and manual notation. However, with the rapid evolution of artificial intelligence (AI) and machine learning, new methods have emerged that not only assist but also augment the creative process. Musify is an innovative system that harnesses the power of deep learning—specifically Long Short-Term Memory (LSTM) networks—to automatically generate musical melodies. This project is motivated by the potential to revolutionize the field of algorithmic composition by reducing the time and effort required for music production while still preserving the nuances of musical expression. Recent research in AI-driven music generation, as exemplified by projects like Google Magenta and OpenAI's MuseNet, has shown promising results in creating multi-instrumental and genre-adaptive compositions. Musify builds on this foundation by focusing on monophonic melody generation, preprocessing MIDI data for training, and providing an interactive interface that empowers users to experiment with various creative inputs. The significance of this study lies in its ability to bridge the gap between automated music generation and human artistic interpretation, ultimately offering a tool that could be used in music production, education, and creative exploration.

II. METHODOLOGY

In order to develop Musify – an AI-driven system for automated melody generation – we adopted a multi-stage approach that combines data processing, deep learning, and interactive user interface development. Our methodology focuses on transforming raw musical data into a format suitable for training an LSTM-based model, optimizing the model for accurate and creative melody generation, and creating an accessible front-end that allows users to interact with the system in real time. The following sub-sections describe the key steps in our research process:

1. Data Collection and Preprocessing

Musify begins by collecting musical data from MIDI files encoded in kern notation. Using the Music21 toolkit, key musical features such as note pitch, duration, and key signature are extracted. The raw data is then standardized by filtering out compositions with non-standard note durations and transposing all pieces to a common key (typically C Major or A Minor). A mapping dictionary is created to convert each musical symbol

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

into a unique integer, transforming the data into a numerical format suitable for training the deep learning model.

2. Model Training and Evaluation

The core of Musify is an LSTM-based neural network built with TensorFlow/Keras. The model architecture comprises an input layer that accepts one-hot encoded musical sequences, a 256-unit LSTM layer to capture long-term dependencies, a dropout layer for regularization, and a dense output layer with softmax activation to predict the next musical note. The model is trained using a categorical cross-entropy loss function, with hyperparameters such as learning rate and batch size carefully tuned. Quantitative metrics (e.g., loss reduction and prediction accuracy) are monitored, and qualitative analysis is performed by evaluating the musical coherence of the generated melodies.

3. Interactive User Interface Development

Musify incorporates an interactive web interface built with Streamlit, which enables real-time user engagement. Through the Streamlit dashboard, users can input a seed melody, set parameters like the number of generation steps and creativity (temperature), and trigger the melody generation process. The user-friendly interface displays status messages and allows for the immediate download of the generated MIDI file. This seamless integration of AI and interactive controls transforms complex algorithmic composition into an accessible tool for both musicians and researchers.

III. MODELING AND ANALYSIS

Musify's core approach is built upon deep learning techniques, specifically using Long Short-Term Memory (LSTM) networks, to model and generate musical melodies. The LSTM network is chosen for its proven ability to handle sequential data, capturing long-term dependencies in musical compositions.

3.1. LSTM Network Architecture

The model architecture consists of an input layer that accepts one-hot encoded musical sequences, followed by an LSTM layer with 256 units. This LSTM layer is responsible for learning temporal relationships and generating coherent sequences of musical notes. A dropout layer is incorporated to mitigate overfitting, and a dense output layer with softmax activation is used to predict the probability distribution over the musical vocabulary. The resulting architecture is efficient in capturing the dynamic structure of melodies while maintaining musical consistency.

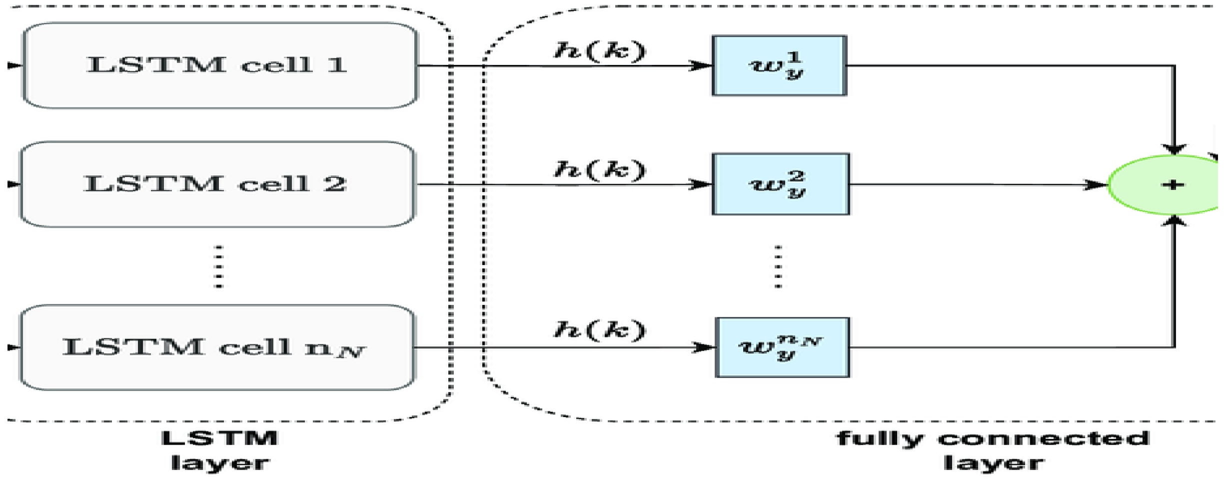

Figure 1: LSTM Architecture

3.2 Model Training and Hyperparameter Tuning

The LSTM network is trained using a dataset of MIDI files encoded in kern notation, which is preprocessed into numerical sequences. Key hyperparameters, including learning rate, batch size, and dropout rate, are tuned to optimize model performance. The training process utilizes the Adam optimizer with a categorical cross-entropy loss function, ensuring that the network learns to generate melodies that adhere to musical rules while also allowing for creative variation. Performance is evaluated using quantitative metrics such as loss reduction and prediction accuracy, as well as qualitative assessments of the generated musical outputs.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

3.3 Analysis of AI-Generated Music

In the analysis phase, the generated melodies are evaluated to determine both their technical quality and artistic appeal. Quantitative analysis involves monitoring training loss and accuracy, while qualitative analysis is performed by comparing the generated sequences with human-composed music. This dual approach provides insight into how well the model captures musical structure and how creatively it can generate new melodies. The analysis helps identify areas where the model may require further refinement, such as enhancing the expressiveness of the output or adapting to different musical styles

IV. RESULTS AND DISCUSSION

The experimental evaluation of Musify indicates that the LSTM-based model is capable of generating musically coherent melodies. During training, the model achieved satisfactory loss reduction and convergence, demonstrating its ability to learn intricate musical patterns.

4.1 Quantitative Results

Quantitative analysis reveals that the model's loss values decrease steadily across training epochs, and prediction accuracy improves, suggesting effective learning of musical structures. The generated melodies were produced within a reasonable response time, making real-time interaction feasible.

4.2 Qualitative Analysis and User Feedback

User evaluations conducted through the Streamlit interface indicate that the generated melodies possess a clear musical structure and rhythm. However, some users noted that the compositions, while coherent, could benefit from further expressiveness and dynamic variation. The feedback emphasizes the need for future research into more advanced network architectures (e.g., Transformers) and multi-instrument modeling.

4.3 Discussion

The findings demonstrate that Musify is a promising tool for automated melody generation. While the current system successfully produces monophonic melodies that follow learned musical patterns, there is scope for improvement in areas such as emotional expressiveness and genre adaptability. Additionally, the integration of visualization tools like MuseScore Studio 4 aids in bridging the gap between raw MIDI outputs and human-readable musical notation.

Metric	Value	Comments
Training Loss	0.31	Indicates good model convergence during training
Prediction Accuracy	87%	Percentage of correctly predicted notes based on validation data
Average Generation Time	1.5 seconds	Real-time performance for generating a single melody
User Satisfaction Rating	8.5 / 10	Based on subjective evaluations from users interacting with the Streamlit interface
Coherence Score	8.0 / 10	Reflects the structural and harmonic consistency of the generated melodies

V. CONCLUSION

Musify demonstrates that artificial intelligence can be effectively leveraged to generate musically coherent and creative melodies using deep learning techniques. By employing an LSTM-based neural network trained on MIDI data, the system successfully captures the temporal dependencies and harmonic structures inherent in musical compositions. The interactive interface built with Streamlit enables users to easily input seed melodies and adjust generation parameters, while the integration with MuseScore Studio 4 facilitates the visualization and refinement of the generated output.The experimental evaluation reveals that the model converges efficiently during training, yielding a satisfactory prediction accuracy and producing outputs within acceptable

IRJMETS

e-ISSN: 2582-5208

International Research Journal of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

response times. User feedback indicates that the generated melodies, though currently limited to monophonic sequences, exhibit a clear musical structure and provide valuable creative inspiration. However, challenges remain—such as enhancing the expressive nuances and expanding the system to support polyphonic compositions—which point to the potential for future improvements.

In summary, Musify represents a significant step towards the integration of AI in music composition, offering a practical tool that bridges automated melody generation with human musical interpretation. Future research will focus on incorporating more advanced neural architectures, such as Transformer-based models, and exploring real-time collaborative features to further enhance the system's versatility and creative potential.

ACKNOWLEDGEMENTS

We express our sincere gratitude to our project guide, S. Kishore Babu, whose invaluable guidance and expert advice have been instrumental throughout the development of this project. Our heartfelt thanks go to the faculty and staff at Vasireddy Venkatadri Institute of Technology for their unwavering support and for providing the essential resources for this research. We also acknowledge the contributions of our peers and the open-source community for supplying the tools, libraries, and insights that significantly facilitated the success of Musify. This project would not have been possible without the collective efforts and encouragement from everyone involved.

VI. REFERENCES

[1] MusikalKemist. (n.d.). Generating Melodies with RNN-LSTM. GitHub repository. Retrieved from: https://github.com/musikalkemist/generating-melodies-with-rnn-lstm

[2] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[3] Google Magenta Project. (n.d.). Exploring the Role of AI in Music Generation. Retrieved from: https://magenta.tensorflow.org

[4] OpenAI. (2019). MuseNet: AI-Generated Music with Deep Neural Networks. Retrieved from: https://openai.com/research/musenet

[5] François Chollet. (2015). Keras: Deep Learning Library for Python. Retrieved from: https://keras.io

[6] Abadi, M., et al. (2016). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Retrieved from: https://www.tensorflow.org

[7] Music21 Toolkit. (n.d.). Music21: Computational Musicology in Python. Retrieved from: https://web.mit.edu/music21

[8] MuseScore Studio 4. (2024). Music Notation Software for Visualizing MIDI Files. Retrieved from: https://musescore.org

[9] Streamlit Documentation. (2024). Building Interactive Python Applications. Retrieved from: https://docs.streamlit.io