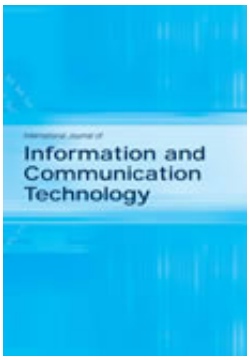
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Application and performance analysis of LSTM networks in polyphonic popular music generation

Juncheng Fang

DOI: 10.1504/IJICT.2025.10072366

Article History:

Received:	14 May 2025
Last revised:	23 May 2025
Accepted:	25 May 2025
Published online:	05 August 2025

Application and performance analysis of LSTM networks in polyphonic popular music generation

Juncheng Fang

Conservatory of Music,
Sichuan University of Science and Engineering,
Zigong, 563000, China
Email: F13880184178@126.com

Abstract: Deep learning-based music generating techniques have slowly shown notable advancement in the area of popular music composition with the fast evolution of artificial intelligence technology. This work intends to look at how long-short-term memory (LSTM) networks are used in polyphonic pop music generation and their performance. An LSTM-based generative model is therefore created to properly catch the temporal dependencies in popular music and produce melodies and harmonies following the rules of music. Experimental findings indicate that, particularly in the coordination between several voices, the LSTM network can better preserve the harmony and consistency of the song when producing polyphonic music. At last, this study offers a perspective for future research considering the constraints of the present work; with the ongoing enhancement of dataset diversity and model optimisation, smart music composition will become more and more relevant in the domain of music composition.

Keywords: LSTM networks; polyphonic music generation; popular music; temporal dependence.

Reference to this paper should be made as follows: Fang, J. (2025) 'Application and performance analysis of LSTM networks in polyphonic popular music generation', *Int. J. Information and Communication Technology*, Vol. 26, No. 29, pp.19–38.

Biographical notes: Juncheng Fang received his PhD from the Mahasarakham University, Thailand. Currently, he is working at the Conservatory of Music, Sichuan University of Science and Engineering, China. Her research interests include popular music, ethnic folk music and artificial intelligence.

1 Introduction

1.1 Background and significance of the study

Music generation has slowly moved from experimental investigation to practical application in recent years as artificial intelligence and music art have become more deeply integrated (Sturm et al., 2019). Generative music technology has become a key instrument for encouraging advances in music production techniques, particularly in the areas of digital entertainment, intelligent arrangement, and assisted creation. Deep learning techniques' inclusion in this process greatly enhances the quality and variety of

music production, hence transforming automated music creation systems from static rule-driven to dynamic model learning. LSTM has grown to be one of the most often used network models in the music generating sector as a very representational structure in sequence modelling because of its capacity to manage complicated dependencies in time series.

Because of its rhythm, melodic simplicity and structural consistency, popular music has drawn most interest in music generation studies. Most current automatic generation systems, meanwhile, still concentrate on monophonic melodies and lack thorough modelling of the synergistic interactions between polyphonic components like harmony and accompaniment. Polyphonic pop music, in contrast to monophonic music, not only demands obvious melodic logic but also harmony, rhythmic consistency, and stylistic coherence across several tracks, which raises more expectations on the timing comprehension and structural control capacity of the generation model (Dean and Evans, 2024).

LSTM network has achieved remarkable results in melody generation, rhythm control, music prediction and other tasks, and can effectively capture long-term dependency information in note sequences by virtue of its gating mechanism, so better simulating the structural characteristics of music such as repetition, change and progression. Further use of LSTM to polyphonic popular music generation suggests that the model must handle the synergetic patterns between several parallel tracks concurrently, which not only increases the investigation of LSTM in terms of modelling capacity but also presents a fresh difficulty for the data representation, network structure design, and generation strategy.

This study aims to provide a framework for automatic synthesis of polyphonic popular music based on LSTM networks and to methodically evaluate its performance and generative effects in modelling multidimensional music structures. This work, on the one hand, can increase the musicality and integrity of the intelligent composition system, therefore broadening its practical application value in music creation, education, games, and other contexts; on the other hand, by means of the examination of the applicability and limits of LSTM in the polyphonic modelling task, it also offers the theoretical support and practical foundation for the design of subsequent more complex generative models.

1.2 Current status of domestic and international research

An important area in the study of intelligent music creation, the polyphonic popular music generation job combines several aspects like melodic modelling, harmonic coordination, rhythmic control, etc. Deep learning technology is developing quickly, so academics have always sought to include several neural network architectures to improve the expressiveness of music generating tools. The present available studies can be roughly classified as follows.

Originally developed for image identification, convolutional neural network (CNN) has also been utilised for music modelling because of its benefits in local feature extraction. Through convolution operations, some research have turned music into 2D piano roll images, built grids for pitch and time axis, and pulled out patterns in note fragments to seize short-term structures like chords and rhythmic blocks (Siphocly et al., 2021). CNN-based models, for instance, may simultaneously analyse data from several voices, stressing the alignment characteristics of the rhythmic and accompaniment layers,

and have performed well in polyphonic style imitation projects. Still, CNNs have some constraints when it comes to more organised pop music and little capacity to predict great distances in time.

Especially in enhancing the naturalness and diversity of produced samples, generative adversarial network (GAN) has been extensively employed in music generation in recent years. Typically, researchers build a generator-discriminator adversarial system in which the generator produces music clips and the discriminator decides if these are actual music (Wang et al., 2021). Usually, MidiNet creates melodies using a mixed CNN-GAN design, hence improving the uniqueness of melodies while preserving stylistic consistency. Though GAN has had great success in the picture domain, in polyphonic music generation the discrete nature of the note sequences and the intricacy of the music structure continue to highlight issues with unstable model training process and imprecise assessment criteria.

Music creation is being opened by the emergence of transformer model and self-attention mechanism. Researchers have begun to include the transformer framework into polyphonic modelling assignments since it can be long-distance modelling and parallel computing (Wu et al., 2023). Music transformer, for instance, can produce intricate musical passages with many voices and great performance in preserving rhythmic, melodic, and accompaniment coordination by modelling the structural relationship between several notes using relative position encoding; OpenAI's MuseNet goes to multi-instrumental, multi-style composite music generation based on the transformer model. Though Transformer is more expressive, its architecture is complicated, it uses a lot of processing power, and it is very reliant on the number and quality of training data.

Graph neural network (GNN) has also been applied to capture graphical links between structures and voice components in music. Treating notes as graph nodes and creating edge connections between various voices, rhythms, and chords, some researchers have tried to recreate intricate note-to-note interaction patterns (de Lemos Almada and Carvalho, 2022). Such techniques are potentially useful in exposing the combinatorial logic of music and are particularly appropriate for conveying dynamic coupling interactions between several voices. The use of GNN in music production, on the other hand, is still exploratory and mostly concerned with assisted generation or structural analysis; it is still challenging to produce high-quality music creation independently.

LSTM networks have a longer history of study in polyphonic music creation than the structures and a more developed approach system. Representing recurrent neural networks, LSTM uses a gating method to effectively handle the gradient vanishing problem of conventional RNN in long-term dependency modelling, hence allowing it to seize the temporal logical links between notes. Many research have produced melodic, harmonic, and rhythmic data using single-layer or multi-layer LSTM architectures, and they have obtained polyphonic outputs using parallel modelling or shared timelines. Currently one of the most often employed models in music generation research, LSTM is also flexible in tasks like structural control and stylistic migration. Furthermore, LSTM network training is consistent and needs low data size, making it especially appropriate for fast testing and deployment on small and medium-sized music datasets.

Though new structures like transformer and GAN show promise in terms of representation complexity and generation quality, the usefulness and performance of LSTM networks in polyphonic popular music production remain typical. A methodical

study of the modelling approach, generation effect, and performance comparison of LSTM under this task not only assesses its present benefits but also offers theoretical foundation and experimental reference for following multi-model fusion investigation.

2 Relevant technical basis

2.1 Characteristics of popular music and polyphonic music

Pop music is extensively spread in many settings including commercial entertainment, film and television soundtracks and network communication as a mainstream genre in modern music culture. Its melody is lovely, rhythm is obvious, structure is tight, and it has great auditory appeal and general acceptance. From the viewpoint of technical modelling, popular music has very formal and repetitive qualities, which offers a basic framework for its generative modelling in the domain of artificial intelligence. But the modelling complexity of the generating job has grown dramatically with the growth of music generation research from monophonic melodies to polyphonic structures, and more sophisticated vocal control and semantic alignment techniques are urgently required.

Structurally speaking, popular music usually has several repeating and symmetrical passage forms like verse, chorus, intro, interlude and outro (Okoro, 2021). With obvious section borders and melodic directions, most of these structures are based on the structuring of periodic time units such eight or sixteen bars. Pop music often uses conventional chord progressions to keep harmonic stability in terms of harmonic arrangement, augmented by simultaneously, rhythmic or rhythmic variations are added to boost auditory diversity. These qualities give pop music algorithmically sound in terms of style development, melodic extension and rhythmic imitation.

Conversely, polyphonic music adds many independent melodic lines over a single melody, creating a three-dimensional weaving of pitch, rhythm, and harmony. Common voices in polyphonic structures are the primary melody, harmony, bass voice, drum and percussion tracks, and special orchestration tracks (e.g., synthesisers, violins, woodwinds, etc.). The various voices are synchronised in the time dimension, and at the same time they have different functions in terms of musical semantics: the main melody is responsible for conveying the theme of the music, the harmony voices reinforce the mood and rhythm, the bass voices support the harmonic roots, and the percussion controls the overall rhythmic framework. Polyphonic music must handle both vertical coordination (harmonic consistency) and horizontal coherence (temporal logic of development) in the work of automatic generation given its obvious division of labour and synergistic character.

Technical modelling of polyphonic music presents three fundamental difficulties, the first being the complexity of the alignment and coordinating interaction among the voices. Though distinct voices must be rhythmically aligned, they frequently vary in pitch, rhythmic pattern, beginning and stopping times, etc. For generative systems, the key difficulty is to properly model these asymmetric but linked time series connections. The key difficulty for the generative system is the efficient representation of these asymmetric but connected time series relationships. The second is the great need for harmonic plausibility. In the generating outcomes, unreasonable chord progressions or intervallic superposition can cause dissonance or perhaps out-of-tune sounds, compromising the audibility and beauty of the music. Thus, the model must either

explicitly or implicitly grasp the principles of harmonic grammar, including those governing the avoidance of discordant intervals and the regulation of fourths and octaves. Equally difficult is rhythmic and orchestral control. The rhythmic beat pattern should reflect the melody, especially in the drum track and percussion section; different instruments should be spread across the frequency domain to prevent overlap; and orchestration conventions in the music style (e.g., electro-pop prefers low-frequency drums and synthetic underscoring, while jazz prefers brass and bass, etc.) must be followed.

Another significant difficulty is the multimodality of polyphonic music data. Usually made up of several MIDI files, polyphonic music features varying note counts as well as varying start and end times. When input to the model is a key component of data pre-processing in polyphonic modelling, how can multi-track MIDI data be consistently encoded to provide consistent temporal alignment and processability (Krause and Müller, 2023). Finally, the model is also more in demand from the cross-cutting requirements of style and mood regulation. Not only the auditory aesthetics and structural consistency of the melody should be ensured in popular music generation, but also the conversion between various styles (e.g., lyric, electronic, rock, etc.) has to be realised, and even the control of emotional development curves, such as the transition from padding to outbursts, etc., which puts forward the dual requirements of style recognition and generation control capability to the model.

Unlike conventional single-melody modelling, polyphonic music generation demands the model to not only grasp the logic of music unfolding in time but also create a hierarchical awareness of the internal structure and generation mechanism. Though early music generation techniques were largely focused on template splicing or Markov chains, managing the long-term dependency and structural coupling of polyphonic components is challenging. Deep learning is now driving more and more studies to build polyphonic output using structures with strong sequence modelling capabilities; among these, LSTM networks, with their long-term memory mechanism for sequence data, show strong adaptability in capturing rhythmic changes, melodic development and polyphonic synchronisation, and become one of the important technological routes for music AI research.

2.2 LSTM network structure and principle

A particular architecture suggested addressing the gradient vanishing and gradient explosion issues experienced by conventional RNNs while handling long time dependence concerns. By adding several gating mechanisms, LSTM can effectively capture and keep long temporal dependencies in sequence data, thereby excelling in tasks like speech recognition, natural language processing, and music synthesis (Yu et al., 2019). When learning long term dependencies, LSTM can better manage information flow and memory transfer than conventional RNNs, which makes it especially appropriate for handling lengthy time sequence data.

LSTM's gating system is its main feature. Every LSTM unit has several gates to regulate memory updating and information flow. Specifically, the LSTM network is made up of output gates, cell states, input gates, and forgetting gates. Every gate has a unique purpose and is meant to enable the network to save and forget data, hence preventing information overload or loss.

The first gate in the LSTM that governs the flow of information from the previous moment to the present one is called the forgetting gate. Based on the present input and the concealed state of the prior moment, the forgetting gate determines which information should be kept and which should be thrown away. Thus, the LSTM can choose forget unnecessary information as required, therefore preventing the gradient vanishing issue that might arise in conventional RNNs under long-term dependency concerns. The design of the forgetting gate lets the LSTM dynamically adjust the information delivery in time-series data without causing the network memory to become too reliant on prior inputs.

Updating the present memory, the input gate then determines which new information should be recorded to the cell state of the LSTM. The input gate determines which information is significant and should be kept in the long-term memory of the network by means of an update value calculated from the current input data and the concealed state of the previous moment. The LSTM can constantly update its memory depending on fresh data using the input gate, hence enabling it to capture important information at various time steps in the sequence and produce more accurate forecasts.

A key component of the LSTM, the cell state denotes the long-term memory of the network. Present throughout the sequence in the network, the cell state transfers information across the time dimension and remains stable across time steps (Li et al., 2019). Unlike conventional RNNs, whose memory may be lost owing to gradient loss, the cell state's architecture lets the LSTM to effectively keep significant information over lengthy time spans. The LSTM may constantly change the cell state to keep long-term memory while handling complicated time-series data by means of interaction between forgetting gates and input gates.

The output gate computes the LSTM network's output value at the present instant depending on the cell state and the current input. The output gate not only influences the present network output but also controls the hidden state update sent to the LSTM cell at the following instant. The output gate's function is to offer information for the prediction of the future instant and to make sure the LSTM can produce sensible outputs depending on the present inputs and internal memories.

By means of the design of these gating mechanisms, the LSTM is able to flexibly and selectively update or forget the information when processing sequence data, which eliminates the limitations of conventional RNNs in handling long term dependencies. Particularly in time series, this LSTM architecture allows it to effectively capture long-term dependencies, which is especially clear in music generating activities.

LSTM's benefits are especially clear in polyphonic music production. Polyphonic music is the coordination and harmony of several independent voices; LSTM can efficiently manage the interdependence between these voices, producing works that follow the rules of music and are inventive. LSTM not only catches the melodic evolution but also creates harmonic intervals across several voices, hence producing intricate harmonic structures. By means of LSTM training, the model can learn to coordinate rhythm and harmony across several voices, hence producing musical fragments that fit the musical style and emotional expression.

2.3 Common encoding methods for music generation

The music generation process depends on how to convert the music data into a format appropriate for computer processing. With each note comprising information like its start

time, pitch, and intensity, it depicts a musical composition as a chronological succession of notes. The time-pitch-velocity triangle, which more succinctly represents the fundamental rhythmic and melodic structure of the music, is the most usual form. The encoding of the music affects the generation outcomes by means of the model's interpretation and manipulation of the music data during generation. Usually, music encoding techniques consist mostly of audio-based spectrum encoding, piano roll encoding, and note sequence encoding.

A simple and natural method of music coding is noting sequence coding. Pitch and intensity are shown straight in MIDI encoding; note duration in this encoding is typically based on the time data of following notes (Wu et al., 2019). When matched with the conventional representation of musical scores, this makes the encoding of note sequences very interpretable.

Note sequence encoding is defined by the formula:

$$N = (T_1, P_1, V_1), (T_2, P_2, V_2), \ldots, (T_n, P_n, V_n) \tag{1}$$

where T_i indicates the note's start time, P_i indicates the note's pitch which is often stated in MIDI code, and V_i indicates the note's strength or loudness. Musical works can thus be precisely shown as time sequences appropriate for learning and generation utilising sequence modelling techniques.

Its simplicity and clarity of understanding are benefits of this encoding method, especially when handling jobs like melody production. Thus, note data in musical works may be precisely represented and instantly applied for sequence modelling. On the other hand, this encoding method's disadvantage is rather clear; especially with complicated rhythmic frameworks or polyphonic music, where the note sequences have limited expressive power, it does not reflect the continuity between notes very effectively. Though note sequence coding is useful for certain straightforward music generating chores, piano roll chart coding could be more relevant when confronted with more complicated musical works.

A two-dimensional matrix called piano roll chart encoding shows the temporal characteristics of music; the horizontal axis denotes time and the vertical axis pitch (Benetos et al., 2018). Every column in the matrix represents a time step; every row relates to a pitch. The matrix's associated element is marked 1 if a note is played at a particular moment; if the note is not played, the corresponding matrix position is 0, meaning the note is off. Especially in polyphonic music creation, piano roll-up diagrams can more easily capture the temporal linkages of notes than note sequence encoding and can properly show the interrelationships between several voices. The piano roll diagram formula is stated as:

$$R = \{r_{i,j}\}, \quad r_{i,j} \in \{0, 1\} \tag{2}$$

where $r_{i,j}$ indicates if the i^{th} pitch in the piano roll chart is played at the j^{th} time point, 1 indicates the note is played and 0 indicates the note is not played. By changing the time step and pitch range, this matrix can be suited to various musical works.

The benefits of the piano roll chart are its clear representation of note timing and pitch information as well as its ease of representation of complicated rhythmic and melodic structures via the matrix form, which makes it especially appropriate for musical works with many voices or long-time spans. This method has significant drawbacks,

particularly for works with a wide pitch range or extended time spans since the dimensions of the matrix may get quite big, creating storage and processing problems.

Another popular coding method in certain audio generating projects is audio-based spectral coding. Unlike symbolic note representations, spectral coding transforms the audio signal into a frequency-domain representation by means of spectrum analysis, hence capturing timbre, harmony, and other subtle audio characteristics in the audio in more depth. Short-time Fourier transform (STFT) and Mel frequency cepstrum coefficient (MFCC) are two often employed spectral analysis techniques (Abdul and Al-Talabani, 2022).

A two-dimensional matrix with the horizontal axis denoting time, the vertical axis denoting frequency, and every matrix element denoting the magnitude or energy at the relevant time and frequency point is a popular depiction of a spectrogram. The following formula can represent this matrix:

$$S(t, f) = \left|X(t, f)\right|^2 \tag{3}$$

where $X(t, f)$ is the complex spectrum produced by Fourier transform; $S(t, f)$ is the magnitude spectrum at time t and frequency f; the absolute value squared is the magnitude spectrum. Especially for audio signal creation tasks, spectral coding is appropriate since it catches the frequency domain qualities of audio signals, particularly regarding timbre and harmonic aspects.

Though spectral coding offers more precise audio data, it struggles with high computing and storage requirements. Particularly in the creation of sound, the high-dimensional representation of the spectrogram complicates the generation process and demands more computing power.

Apart from the previously stated coding techniques, rhythmic and timbre coding are also key factors in music creation. Rhythmic coding often uses beats, beat numbers, and note spacing to indicate the rhythmic pattern in music. On the other hand, timbre coding allows the generative model to more accurately mimic the timbral qualities of several instruments when producing music by means of the spectrum analysis of the audio signal.

3 Application and implementation of LSTM networks in polyphonic pop music generation

3.1 Data preparation and coding

The preparation and encoding of input data is vital in polyphonic popular music creation. This work encodes music data using piano roll. Every voice is seen as a distinct note sequence if one is to be able to create polyphonic music. To help LSTM networks learn polyphonic music, the piano roll map data of every voice part is combined in this paper into a multi-dimensional matrix, with each dimension signifying a distinct voice part. Such a technique has the benefit of guaranteeing independence between each vocal part while maintaining their interrelationships in the whole musical structure, which is vital for producing coherence and harmony in polyphonic music.

Moreover, the input data is often sliced using a sliding window to more accurately capture the temporal characteristics of the notes. Given a sliding step of 1 and a window length T, every input sample can be shown as a succession of notes of length T in a

continuous music sequence (Silva et al., 2018). At every time point t, the input data will be a vector of notes with T time steps; each vector will include information on the notes of the current time step and its prior T time steps. Thus, the LSTM network can effectively learn the temporal relationships between sounds and produce constant and inventive musical sequences.

The following formula can show the encoding of the piano roll map:

$$P(t,i)=\begin{cases}1, & \text{if note } i \text{ is played at time step } t\\ 0, & \text{otherwise}\end{cases} \tag{4}$$

where $P(t, i)$ indicates the time step t and whether the note i is active or not. The input sequence may also be shown as:

$$X=\{x_1, x_2, \ldots, x_T\} \tag{5}$$

where x_t indicates the note vector of the t^{th} time step comprising the note information of the present time step and its prior T time steps.

This encoding allows the input data to give the LSTM network sufficient information for the model to seize the temporal dependencies of the music and produce polyphonic pop music.

3.2 LSTM network structure

LSTM networks create melodies and harmonies by learning the relationships between note sequences, hence generating polyphonic pop music. LSTM is especially able to handle synergistic interactions between several voice parts by using its own memory system to capture note sequence properties across long time spans. Specifically, the creation of each note depends on previously produced notes and coordinated interactions with other voices; the gating mechanism of LSTM allows the network to dynamically control the flow of information and memory, therefore guaranteeing that the produced music is consistent and harmonic.

First, LSTM's forgetting gate decides whether the state information of the prior moment should be lost (Wang et al., 2019). The forgetting gate's function in polyphonic pop music creation is to keep those qualities that are useful for the present note production and to discard the obsolete information unrelated to the present generation. The equation determines it:

$$f_t=\sigma\left(W_f\cdot\left[h_{t-1}, x_t\right]+b_f\right) \tag{6}$$

where f_t indicates the output of the forgetting gate at the present time, W_f is the weight matrix of the forgetting gate, h_{t-1} is the hidden state at the prior moment, x_t is the input at the present time, which indicates the characteristics of the note, b_f is the bias term, and σ is the sigmoid activation function, which limits the output of the forgetting gate to a range between 0 and 1 and controls the degree of information discarded.

The input gate next decides which new note information should be added to the memory of the network to affect the next produced notes. Especially when polyphonic music is being created, the input gate determines how the melodies and harmonies of the several voices are coordinated with one another to guarantee that the produced notes fit the harmony. The input gate is determined by the formula:

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \tag{7}$$

where i_t is the output of the input gate, W_i is the weight matrix of the input gate, and b_i is a bias term regulating the model's acceptance of the input note information at the present time.

Simultaneously, LSTM has to provide a candidate memory unit to modify the memory state at the present time. The candidate memory unit's function in polyphonic music creation is to determine which note data should be added to the memory while preserving a valid time series dependency (Ycart and Benetos, 2020). The candidate memory unit is computed using the formula:

$$\tilde{C}_t = \tanh\left(W_C \cdot \left[h_{t-1}, x_t\right] + b_C\right) \tag{8}$$

where $\tilde{C}_t$ is the candidate memory cell at the present time, W_C is the weight matrix, b_C is the bias term regulating the inclusion of incoming information, and tanh is the hyperbolic tangent function maintaining the output of the candidate memory cell within the appropriate range.

Updating the memory cells lets the LSTM finally decide the hidden state ht of the current moment, which will be sent to the network as the input for the next moment to produce the next note. The LSTM in the polyphonic generation task not only creates the current note depending on the prior note but also creates the notes of other voices depending on the relative positional relationship between each voice, therefore preserving the coordination of melody and harmony.

This structural design allows LSTM to capture intricate time series dependencies while guaranteeing harmony and consistency between melody and harmony in the polyphonic popular music generation task, hence strongly supporting automatic music production.

3.3 Training process

The performance of LSTM networks in polyphonic popular music generation depends on the training procedure. The LSTM network, which performs well in managing complicated music creation activities with its adjustable learning rate and quick convergence, is trained using the Adam optimisation method in this work.

The LSTM network, which can record the intricate temporal correlations between notes and between many voices via its gating mechanism, is fed pre-processed data. The LSTM network's aim is specifically to produce melodies and harmonies by learning the temporal dependencies of notes and the interactions between various voice components, therefore conforming to the laws of music.

Typically, the loss function of the LSTM network uses mean square error (MSE) to gauge the discrepancy between the expected notes and the actual notes during training (Edalatifar et al., 2022). Specifically, the loss function is stated as:

$$L = \sum_{t=1}^{T} \left(y_t - \hat{y}_t\right)^2 \tag{9}$$

where y_t is the actual note, $\hat{y}_t$ is the note forecasted by the LSTM network, and T is the total note sequence length. The aim of the loss function is to reduce the difference

between the expected and actual notes, therefore enhancing the quality of the model's creation.

The LSTM network updates the parameters using the Adam optimisation technique during back propagation (Reyad et al., 2023). The Adam optimisation algorithm's gradient update formula is:

$$\theta_t = \theta_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \dot{o}} \tag{10}$$

where θ_t is the parameter of the present time step, η is the learning rate, $\hat{m}_t$ and $\hat{v}_t$ are the deviation corrections of the mean and squared mean of the gradient, and ϵ is a small constant to avoid the divide-by-zero error. The Adam optimisation technique makes the network parameter update more stable and efficient by able to adaptively change the learning rate depending on the gradient information of each parameter.

The training method in this work not only emphasises the creation of individual voices but also explicitly addresses the synergistic generation between several voices. Sharing the LSTM network's hidden layer states allows the voice components to preserve temporal and pitch coordination throughout the creation process. The generative efficacy of the model is assessed after each round of training until convergence.

The training of the LSTM network consists overall not just of note creation but also of harmonic and temporal dependency learning across several voices. Adam's optimisation technique helps the LSTM network to effectively manage the polyphonic popular music creation challenge and finally produce musical works with high-quality melodies and harmonies.

3.4 Generation process

The LSTM network in polyphonic popular music creation runs recursively on temporal dependencies and vocal synergy information acquired during training to produce note sequences. Starting with an initial note or a starting sequence, the generation process produces the first note depending on these inputs; subsequently, using the produced notes as inputs for the next step, the network continues to produce the following notes. A whole musical fragment is produced by this method.

Every time step, the LSTM network calculates a new note depending on the present input and the prior output. The generation process not only depends on the data of individual voices but also includes their interrelationships. For instance, the LSTM network must change the melody's pitch and rhythm to match the harmony part's notes, so guaranteeing the coordination between the melody and the harmony. This generation method shows LSTM's benefit in handling polyphonic pop music; it can seize the time link between harmonic notes and coordinate the production of several vocal parts.

Specifically, at every time step, the LSTM network produces a probability distribution showing the generation probability of the next note among the potential notes. The LSTM picks the next note and feeds it as input for the next step by sampling this probability distribution. The following equation can reflect this procedure:

$$p\left(y_t \left| y_{t-1}, \ldots, y_1 \right.\right) = \text{softmax}\left(W_h h_t + b_h\right) \tag{11}$$

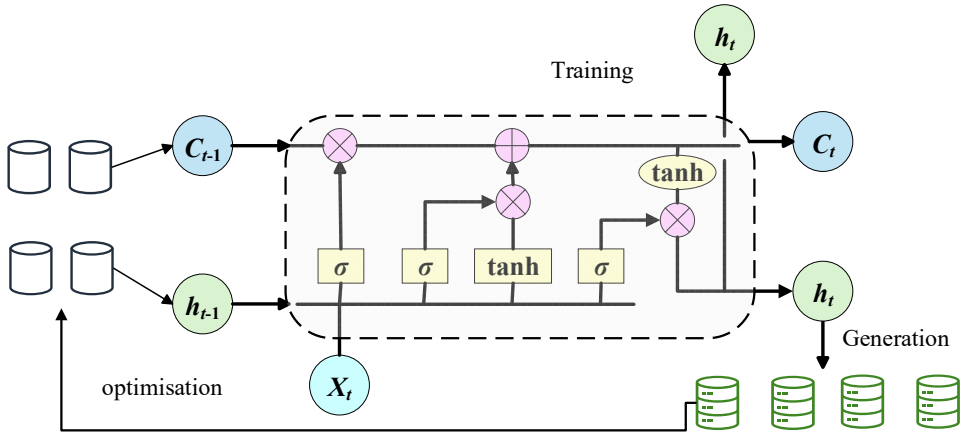
where $p(y_t|y_{t-1}, \ldots, y_1)$ indicates the likelihood of producing the note y_t at time t given the prior moment's note sequence $y_{t-1}, \ldots, y_1$; h_t is the hidden state of the LSTM network at time step t, W_h and b_h are the corresponding weight matrices and bias terms, and the softmax function converts the output into a probability distribution and selects the most likely note as the next generated note.

LSTM can efficiently combine the previously produced notes with the notes now by means of this recursive generation process, therefore guaranteeing that the produced polyphonic music not only follows the temporal logic of the music but also preserves the harmony and coordination between the voices.

3.5 Post-processing process

The produced note sequences have to be fixed and adjusted during post-processing to guarantee their harmonic and rhythmic lawfulness. Adjusting the spacing between notes helps the harmonic progression to fit the standards of music theory. The timing of the notes may be optimised to guarantee rhythmic fluidity and a natural feel to the music by examining the location of each note on the timeline. Rhythmic optimisation for polyphonic music is not only the modification of separate voices but also the consideration of the rhythmic coordination between the voices to prevent awkward rhythmic leaps between notes (Barrett, 2022).

Figure 1 Flow of generating music (see online version for colours)



The post-processing algorithm can change the produced outcomes based on the note relationship to help this. The timing distribution of the notes can be optimised by changing the variances in the timing values of the notes, hence enhancing the smoothness of the rhythm. The adjustment procedure can be stated as follows:

$$\Delta t_{t,t-1} = smooth\left(y_t, y_{t-1}, \Delta t\right) \tag{12}$$

where each y_t is a note produced at moment t, the smooth function indicates the smoothing of note length variations, and Δt is the time value difference between notes. This approach helps to minimise artificial rhythmic jumps, hence optimising the produced polyphonic music more in rhythmic harmony.

All in all, the first five stages span the whole data preparation to final music generating process; see Figure 1.

4 Experiments and performance analysis

4.1 Dataset introduction and evaluation metrics

The Lakh MIDI Dataset (LMD), released by a research team at the University of California, San Diego (UCSD), was built from audio metadata from the Million Song Dataset matched, cleaned, and format normalised with music files from several MIDI sharing sites. Especially demonstrating good qualities in terms of polyphonic structure and stylistic variety, LMDs are more appropriate for modelling challenges in contemporary popular music than conventional classical music datasets. The 'LMD-clean' subset is chosen as the training base and the fragments with unusual lengths, extreme note distributions or incomplete structures are further screened out to create a high-quality sample set for modelling, thus guaranteeing the quality and representativeness of the training data for generating the model.

Some pop music clips with polyphonic material like the main theme, harmony, and accompaniment are chosen, converted to MIDI format, and beat, pitch, duration, and other characteristics standardised in this paper to serve as the foundation for the generation of input for the following LSTM model. Table 1 displays the dataset information.

Table 1 Dataset summary (Lakh MIDI Dataset – LMD-clean subset)

Item	*Description*
Dataset name	Lakh MIDI Dataset (LMD-clean subset)
Total files	21,425 MIDI files (approximately 8,000 used after filtering)
Musical styles	Pop, rock, electronic, jazz, and other modern genres
Format support	Standard MIDI (multi-track, polyphonic structure)
Source and availability	Public and open access (UCSD repository/GitHub)
Pre-processing steps	Pitch normalisation, rhythm quantisation, part separation and alignment

The model can learn and train polyphonic music creation depending on the input of melodic, harmonic and rhythmic characteristics, so guaranteeing that the produced material has a high degree of consistency and musicality in terms of style, structure and rhythm. This work presents two assessment metrics which are measured from the angles of note accuracy and melodic diversity respectively, to more objectively assess the quality of the produced music and the model performance.

First, Note Accuracy is used to assess the degree of overlap in pitch and temporal value between produced notes and actual notes.

$$Note\ accuracy = \frac{N_{matched}}{N_{generated}} \tag{13}$$

where $N_{matched}$ refers to the number of produced notes that precisely correspond with the actual notes; $N_{generated}$ is the model's entire note count. A higher index indicates that the model generation outcomes are closer to the original music structure and that the timing learning capacity is better (Briot and Pachet, 2020).

Pitch class histogram entropy is used next to assess the complexity and diversity of the produced music in terms of pitch distribution using the formula below:

$$H = -\sum_{i=1}^{12} p_i \log_2 (p_i) \tag{14}$$

where p_i is the likelihood of the i^{th} pitch class showing in the entire produced song. A significant base for assessing whether the produced music has too much repetition or pitch bias, this index can show the richness of the melody and the expressiveness of the general musicality. In the music, a higher entropy number indicates a richer pitch dispersion and a greater musicality of the produced outcome (Danieli and Frank, 2022).

The two measures taken together form a quality evaluation system for the produced outcomes that will guide future experiments comparing the benefits and drawbacks of the produced outcomes under various model parameters and training settings.

4.2 Effect of the number of LSTM layers on the quality of generation

This paper investigates in experiment 1 how the number of LSTM model layers influences the generation quality of polyphonic pop music. LSTM models with varying number of layers were configured for the experiments to help study the relationship between model depth and generation outcomes: one-layer LSTM, two-layer LSTM, three-layer LSTM, four-layer LSTM, and five-layer LSTM. The goal is to know how the model depth influences the temporal consistency and musicality of the produced outcomes by changing the number of layers of LSTM.

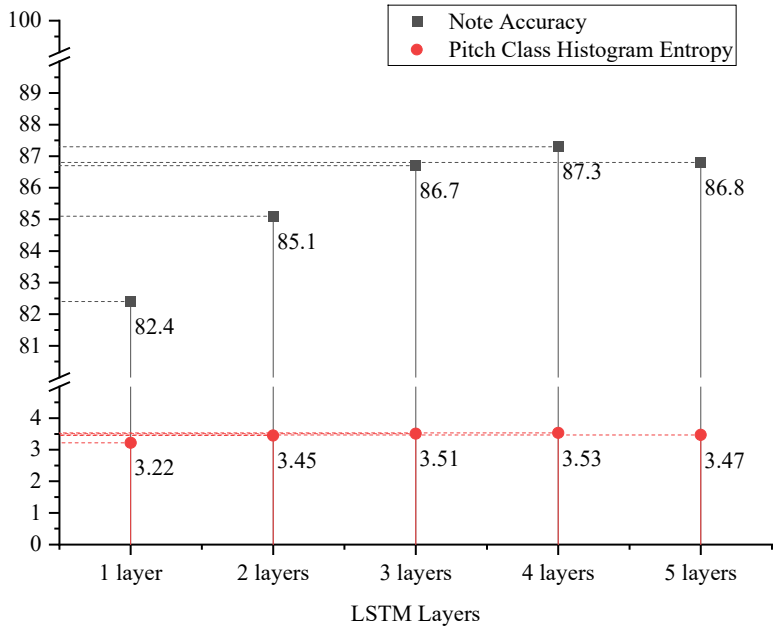
By comparing the note accuracy and pitch category histogram entropy of polyphonic music produced under various layer configurations, the experiment's main goal was to evaluate the variations in the performance of several model depths in terms of note prediction accuracy and melodic diversity.

Presented via Figure 2, the experimental findings show the note accuracy and pitch category histogram entropy for several LSTM layer configurations.

More LSTM layers raise note accuracy as well as pitch category histogram entropy. Specifically, when compared to the one-layer LSTM, the two-layer LSTM raises the note accuracy by 2.7 percentage points and the pitch category histogram entropy rises, suggesting that the model performs better in note production and melodic diversity. The performance of the three-layer and four-layer LSTMs is closer, with a slight rise in note accuracy and pitch category histogram entropy, and the four-layer LSTM marginally outperforms the three-layer LSTM, suggesting that the model tends to stabilise after a particular level of increase in the number of layers. This suggests that the model generation effect tends to stabilise after raising the number of layers to a particular level.

The five-layer LSTM's findings were, nevertheless, somewhat lower than those of the four-layer LSTM, especially in terms of note accuracy, which may suggest that using more LSTM layers in this experimental setup could cause overfitting or unstable model training.

Figure 2 Performance of different LSTM layer configurations (see online version for colours)



This study indicates, therefore, that while too many LSTM layers might lower model performance, the LSTM model can perform better in polyphonic pop music generation with a network architecture of 2 to 4 layers. Appropriate decisions regarding model depth must be made. The experimental findings offer a reference point for later model tuning and optimisation.

4.3 Effect of number of input voices on generation effectiveness

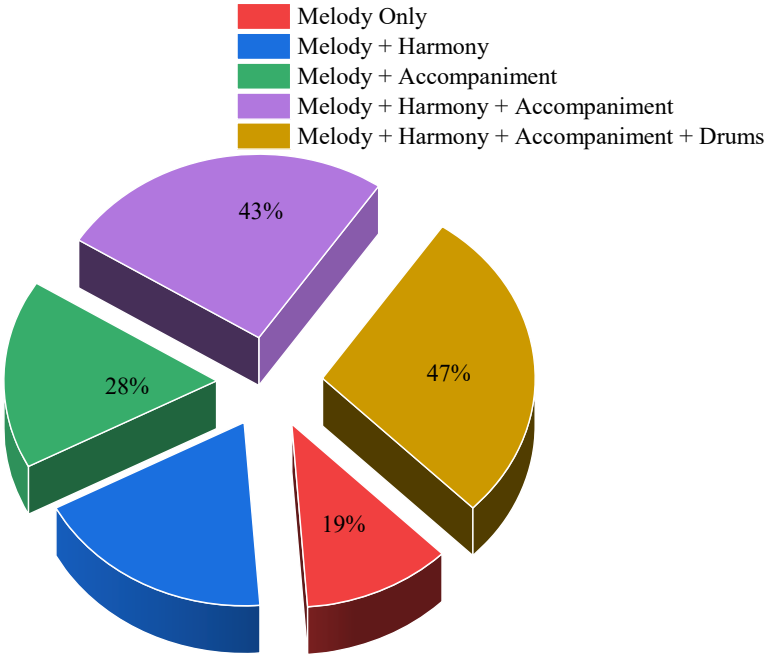
This paper of experiment 2 looks at how the number of input voices influences the quality of polyphonic popular music creation. Given the polyphonic music production task's need for co-ordination between melody, harmony, and rhythm, changes in the number of input voices greatly influenced the temporal and structural dependencies acquired by the model.

The experimental input setups were defined as: full voice input (main melody, harmony, accompaniment, percussion), main melody only, main melody + harmony, main melody + accompaniment, and main melody + harmony + accompaniment. Under each set of settings, 100 music samples were produced; five judges with musical professional backgrounds were asked to rate the samples in three areas: harmony, structural integrity, and stylistic consistency; those with an average score of more than 4 were deemed high-quality samples. Figure 3 displays the experimental findings.

The experimental findings show that the general quality of the produced music is notably influenced by the rising number of input voices. Using just the main melody as input, the percentage of high-quality samples is just 19%, suggesting that the model struggles to produce well-structured and layered musical material without harmonic and rhythmic information. The percentage of high-quality samples rises to 31% and 28%

when harmony or accompaniment is added, respectively, suggesting that these two kinds of vocal parts help to improve melodic support and complement musical hierarchy, especially the harmony part's contribution to the general harmony of the music.

Figure 3 Proportion of high-quality samples by input voice configuration (see online version for colours)



Moreover, the percentage of high-quality samples almost doubles to 43% when the input includes the primary melody, harmony, and accompaniment components. Moreover, the percentage of high-quality samples rises to 47% with the addition of the percussion component. This implies that improving musical expression and strengthening artistic coherence depend much on rhythmic information. Generally speaking, the better the model is to catch the intricate time-dependent and synergistic characteristics of the vocal parts the richer the input information, thereby improving the coherence and artistic expression of the produced music.

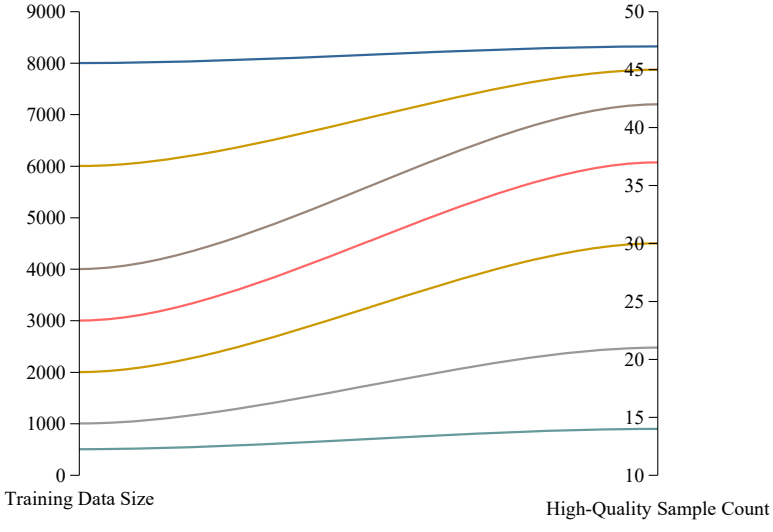
4.4 Impact of training data size on generation performance

The performance of polyphonic pop music production is methodically investigated in the third set of experiments in this work as influenced by training data size. Given the reliance of deep neural networks on the quantity of data, subsets of data of varying sizes were created for model training to see the impact of variations in the number of training samples on the generation results, while maintaining the LSTM network structure, optimiser and learning rate parameters consistently.

The studies split the training data across seven subsets of varying sizes, each with 500, 1,000, 2,000, 3,000, 4,000, 6,000 and 8,000 polyphonic pop music samples. Every model is rated by five music expert assessors on three aspects which produce 100

samples on a standard validation set. Judges found samples with an average score of 4.0 or above to be of good quality; the proportion of high-quality samples produced by each group of models was tallied correspondingly. Figure 4 displays the experimental findings.

Figure 4 High-quality sample proportion under different training data sizes (see online version for colours)



The general proportion of high-quality samples produced tends to rise as the number of training examples rises. The overall performance of the music produced by the model is bad when the number of training samples is 500; just 14% of the samples are assessed as good quality, suggesting the data volume is insufficient to enable the model to acquire complicated polyphonic patterns. The percentage of high-quality samples had climbed to 30% when the sample size was raised to 2,000, indicating the model's first understanding of musical patterns.

The percentage of high-quality samples rises to 42% once the data quantity is increased to 4,000 songs, suggesting that the model can learn the temporal dependence between notes and the coordination law between voices in a more in-depth way as the variety of data grows. Between 6,000 and 8,000 tracks, though, the surge tends to level off and the percentage of high-quality samples only increases from 45% to 47%, implying that the marginal benefit at this point has started to decline.

Especially at the initial stage, the amount of training data is a key influence on the model performance; when considered together, a reasonable extension of the training set can greatly enhance the music generation effect. On the other hand, it is demonstrated that the constant increase of data does not linearly enhance the model performance; a sensible selection of data size together with the optimisation of the model structure will be more practically relevant.

5 Conclusions

This research offers a methodical investigation on the use of LSTM networks in polyphonic popular music production. First, this article presents the features of polyphonic popular music and the benefits of LSTM in handling time series data; subsequently, it suggests an LSTM model relevant to polyphonic music generating. This work creates and implements a comprehensive generation framework using the LMD-clean subset dataset, comprising the preparation and encoding of input data, the design of LSTM network structure, the training phase, the generation process and the post-processing stage.

This work investigates the impact of various number of layers, number of input voices, and training data size on the generation effect by means of several sets of experiments; the findings indicate that while an increase in the number of input voices helps to improve the structural integrity of the music, an increase in the number of network layers and data size can significantly improve the quality of the generated music. Experimental validation thoroughly demonstrates the LSTM network's capacity to successfully capture the temporal dependencies in polyphonic music and produce harmonic melodies and harmonies, hence showing the possibility of LSTM in intelligent music production.

The LSTM network demonstrates great adaptability and generative capacity in producing polyphonic popular music, hence offering technological support with practical value for smart music creation.

Though the LSTM network suggested in this paper performs better in the polyphonic pop music generating challenge, it still has certain drawbacks. Though the LMD-clean subset dataset has a lot of pop music samples, its variety is still constrained, particularly in terms of style and complexity, and may not be enough to fully represent all kinds of pop music. The collection is mostly drawn from subgroups of popular music and lacks samples across eras, cultures, or other popular music styles. Future studies should also consider increasing the dataset to incorporate popular music data from other eras and areas to improve the generalisation capacity and adaptability of the model. Building datasets with different genres and styles also helps LSTM networks to better capture the varied characteristics of popular music and increase the expressiveness and diversity of the produced music.

Second, while LSTM offers benefits in capturing time-series dependency, there are still certain restrictions in processing dynamic changes and sophisticated musical emotional expression. Though the model's generative impact is sometimes not sufficiently subtle when producing complicated, emotionally rich polyphonic music, LSTM can efficiently reduce the issue of long-time reliance via its gating mechanism. LSTM might not function finely enough, particularly in the creation of rhythmic alterations, emotional transitions, and intricate harmonic structures. Future studies could thus aim to include more sophisticated model structures like Transformer or generative models based on reinforcement learning (Chen et al., 2021). While enhancing the depth of emotional expression and musical subtleties, these models can more efficiently capture and generate complicated structures with lengthy time spans via self-attentive mechanisms or reward signal-driven production.

Furthermore, while the manual evaluation method utilised in this study guaranteed the assessment accuracy of the produced music via a multi-dimensional scoring system, it still has some subjectivity and consistency issues. Variations in the aesthetic criteria of

the music by various critics could cause scoring discrepancies, therefore influencing the assessment of the produced outcomes. Future studies could investigate the addition of automated assessment criteria to measure and evaluate the quality of the produced music using an automated scoring system, hence enhancing the objectivity of the evaluation (Rupp, 2018). Combining deep learning and audio processing methods, applying GAN for music evaluation, or creating machine-learning-based music quality assessment standards would all help to increase assessment efficiency and lower the impact of human factors on the outcomes.

At last, even if this research has produced some outcomes regarding generative quality, there are still restrictions in musical production diversity and creativity. Currently, the LSTM model generates following notes using a great deal of historical data, hence limiting its creative material to some extent by the patterns found in the data.

Although polyphonic popular music generating already presents difficulties, future studies will open more creative avenues for music production as technology develops. Intelligent music production will achieve more major advancements in artistic expression, stylistic diversity and creative generation by constantly optimising the generation model, enlarging the dataset and adding new assessment systems. Artificial intelligence will surely play a bigger part in future music production, therefore providing a completely new experience for both musicians and fans.

Declarations

All authors declare that they have no conflicts of interest.

References

Abdul, Z.K. and Al-Talabani, A.K. (2022) ‘Mel frequency cepstral coefficient and its applications: a review’, *IEEE Access*, Vol. 10, pp.122136–122158.

Barrett, G.D. (2022) ‘‘How we were never posthuman’: technologies of the embodied voice in Pamela Z’s Voci’, *Twentieth-Century Music*, Vol. 19, No. 1, pp.3–27.

Benetos, E., Dixon, S., Duan, Z. and Ewert, S. (2018) ‘Automatic music transcription: an overview’, *IEEE Signal Processing Magazine*, Vol. 36, No. 1, pp.20–30.

Briot, J-P. and Pachet, F. (2020) ‘Deep learning for music generation: challenges and directions’, *Neural Computing and Applications*, Vol. 32, No. 4, pp.981–993.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A. and Mordatch, I. (2021) ‘Decision transformer: reinforcement learning via sequence modeling’, *Advances in Neural Information Processing Systems*, Vol. 34, pp.15084–15097.

Danieli, L. and Frank, M. (2022) ‘Entropy, pitch, and noise: organisation and disorganisation in the perception of closure for different types of spectra’, *Journal of New Music Research*, Vol. 51, Nos. 4–5, pp.378–387.

de Lemos Almada, C. and Carvalho, H. (2022) ‘Entropy, probabilistic harmonic space, and the harmony of Antonio Carlos Jobim’, *Musica Theorica*, Vol. 7, No. 1, pp.68–111.

Dean, R.T. and Evans, S.J. (2024) ‘Generalising personalised exploration and organisation of sonic spaces: metacultural approaches: metacultural approaches’, *Journal of Creative Music Systems*, Vol. 8, No. 1, pp.1–27.

Edalatifar, M., Ghalambaz, M., Tavakoli, M.B. and Setoudeh, F. (2022) ‘New loss functions to improve deep learning estimation of heat transfer’, *Neural Computing and Applications*, Vol. 34, No. 18, pp.15889–15906.

Krause, M. and Müller, M. (2023) 'Hierarchical classification for instrument activity detection in orchestral music recordings', *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 31, pp.2567–2578.

Li, C., Wang, Z., Rao, M., Belkin, D., Song, W., Jiang, H., Yan, P., Li, Y., Lin, P. and Hu, M. (2019) 'Long short-term memory networks in memristor crossbar arrays', *Nature Machine Intelligence*, Vol. 1, No. 1, pp.49–57.

Okoro, J.C.P. (2021) 'Evidences and applications of proverbs, parallelism, fables in Abigbo musical structure', *International Journal*, Vol. 9, No. 1, pp.23–35.

Reyad, M., Sarhan, A.M. and Arafa, M. (2023) 'A modified Adam algorithm for deep neural network optimization', *Neural Computing and Applications*, Vol. 35, No. 23, pp.17095–17112.

Rupp, A.A. (2018) 'Designing, evaluating, and deploying automated scoring systems with validity in mind: Methodological design decisions', *Applied Measurement in Education*, Vol. 31, No. 3, pp.191–214.

Silva, D.F., Yeh, C-C.M., Zhu, Y., Batista, G.E. and Keogh, E. (2018) 'Fast similarity matrix profile for music analysis and exploration', *IEEE Transactions on Multimedia*, Vol. 21, No. 1, pp.29–38.

Siphocly, N.N.J., El-Horbaty, E-S.M. and Salem, A-B.M. (2021) 'Top 10 artificial intelligence algorithms in computer music composition', *International Journal of Computing and Digital Systems*, Vol. 10, No. 1, pp.373–394.

Sturm, B.L., Ben-Tal, O., Monaghan, Ú., Collins, N., Herremans, D., Chew, E., Hadjeres, G., Deruty, E. and Pachet, F. (2019) 'Machine learning research that matters for music creation: a case study', *Journal of New Music Research*, Vol. 48, No. 1, pp.36–55.

Wang, B., Kong, W., Guan, H. and Xiong, N.N. (2019) 'Air quality forecasting based on gated recurrent long short term memory model in internet of things', *IEEE Access*, Vol. 7, pp.69524–69534.

Wang, Y., Song, X., Xu, T., Feng, Z. and Wu, X-J. (2021) 'From RGB to depth: domain transfer network for face anti-spoofing', *IEEE Transactions on Information Forensics and Security*, Vol. 16, pp.4280–4290.

Wu, D-C., Hsiang, C-Y. and Chen, M-Y. (2019) 'Steganography via MIDI files by adjusting velocities of musical note sequences with monotonically non-increasing or non-decreasing pitches', *IEEE Access*, Vol. 7, pp.154056–154075.

Wu, G., Liu, S. and Fan, X. (2023) 'The power of fragmentation: a hierarchical transformer model for structural segmentation in symbolic music generation', *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 31, pp.1409–1420.

Ycart, A. and Benetos, E. (2020) '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.

Yu, Y., Si, X., Hu, C. and Zhang, J. (2019) 'A review of recurrent neural networks: LSTM cells and network architectures', *Neural Computation*, Vol. 31, No. 7, pp.1235–1270.