Applied Soft Computing Journal 153 (2024) 111294

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Grey wolf optimizer based deep learning mechanism for music composition with data analysis

Qian Zhu [a,*], Achyut Shankar [b,c,d], Carsten Maple [b]

[a] Xinxiang University, Xinxiang 453003, China
[b] Department of Cyber Systems Engineering, WMG, University of Warwick, Coventry CV74AL, United Kingdom
[c] Centre of Research Impact and Outreach, Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India
[d] School of Computer Science Engineering, Lovely Professional University, Phagwara - 144411, Punjab, India

HIGHLIGHTS

- This study introduces a novel approach that combines GWO and LSTM for music composition.
- This study employs GWO to optimize the hyperparameters of LSTM, enhancing the effectiveness of music generation.
- The training data is converted to MIDI format, and melody lines are extracted using a similarity matrix method.
- LSTM focuses on the input of capturing tone, rhythm, artistic conception, and other attributes of high-quality music.

ARTICLE INFO

Keywords:
Music composition
LSTM
GWO
MIDI
Data analysis

ABSTRACT

Music composition using artificial intelligence has gained increasing research attention recently. However, existing methods often generate music that needs more coherence and authenticity. This paper proposes an evolutionary computation-based deep learning approach for music composition with data analysis. Specifically, we utilize long short-term memory (LSTM) networks for generating melodic sequences and adopt a grey wolf optimizer to optimize LSTM hyperparameters. The training data is first converted to musical instrument digital interface (MIDI) format for data analysis, and melody lines are extracted using a similarity matrix method. The MIDI data is then encoded for input into the LSTM networks. The generated music is evaluated using objective metrics like mean squared error and subjective methods, including surveys of music professionals. Comparisons made to benchmark algorithms like generative adversarial networks demonstrate the advantages of our approach in accurately capturing tone, rhythm, artistic conception, and other attributes of high-quality music. The proposed mechanism provides a practical framework for AI-based music generation while ensuring authenticity.

1. Introduction

Music composition is an artistic endeavor that serves as a profound outlet for expressing human emotions and conveying intricate artistic messages [1]. Through the annals of history, the craft of curious and aesthetically pleasing melodies has been regarded as a quintessentially human gift that relies on innate talent and creative genius. Nonetheless, the recent groundbreaking strides in artificial intelligence, particularly within deep learning, have ushered in a new era. In this era, algorithms have begun to generate musical sequences that, to varying degrees, evoke the essence of well-composed music [2]. While the compositions generated by AI still often lack the overarching coherence and the ability to intricately capture the rhythmic, melodic, and artistic nuances synonymous with human composers' work, the rate of progress is remarkable. It underscores the potential for AI systems to grasp the intricate web of associations integral to the creative process of music composition [3]. The interplay between human creativity and artificial intelligence hints at a promising future where machines may be true collaborators in music, enhancing the beauty and diversity of the compositions that resonate with us.

Music, a timeless and universal art form, carries a rich history that can be traced back to our prehistoric ancestors. Remarkably, relics such as ancient bone flutes, crafted from vulture wing bones some 37,000 years ago, stand as a testament to the enduring nature of music [4,5].

* Corresponding author.
E-mail address: zhuqian001@xxu.edu.cn (Q. Zhu).

https://doi.org/10.1016/j.asoc.2024.111294
Received 28 October 2023; Received in revised form 2 January 2024; Accepted 14 January 2024
Available online 20 January 2024

These early instruments, among the oldest known to humanity, offer a glimpse into the profound roots of music in our shared past. What makes music a remarkable phenomenon is its unifying presence in every corner of the globe. It transcends linguistic, religious, and temporal boundaries, acting as a cultural tapestry woven into the very fabric of human existence. Each culture has developed its unique musical traditions, deeply intertwined with its values, beliefs, and ways of life. This diversity underscores music's capacity to serve as a bridge that connects people from all walks of life. At its core, music is an art form that speaks to the soul. It harnesses the power of aesthetically arranged sounds and silences, utilizing pitch, tempo, timbre, and rhythm to evoke profound emotions and communicate in ways that words alone often fall short. Music, through its melodic and rhythmic tapestry, offers an avenue for individuals to express the depths of their emotions and articulate the most intense inner states with a poetic grace that transcends the limitations of language. In this way, music serves as a universal language of the heart, enriching the human experience and binding us together in the symphony of existence.

Throughout the annals of philosophy, there has been a persistent fascination with unraveling the enigma of human creativity and the qualities that set extraordinary works of art apart. However, the advent of modern computing technology has ignited a newfound curiosity in mechanizing and automating the creative process of music composition [6–8]. This transformational shift has brought the intersection of technology and artistic expression to the forefront of creative endeavors. The formalization of music theory serves as a cornerstone for encoding complex rules governing harmony, melody, and form into computational algorithms. This practice of translating musical principles into lines of code allows for the manipulation and generation of music by machines. While gaining prominence in the present era, this concept has roots that extend back over three centuries. Johann Philipp Kirnberger's pioneering work in the 18th century exemplified an early attempt to harmonize melodies using logic-based procedures, laying the foundation for computational music composition. The advent of the first general-purpose computer in the mid-20th century marked a watershed moment in this evolution. It led to the development of algorithms that could algorithmically generate musical scores. These algorithms introduced elements of randomness and probability, effectively simulating the creative unpredictability inherent in human composition. This convergence of technology and music has opened doors to new possibilities, sparking artistic experimentation and the exploration of artificial intelligence's role in the creative arts. It represents a continuum of human ingenuity, where the rich tapestry of music intersects with the binary world of algorithms, promising a future where man and machine compose harmoniously.

In 1956, the world witnessed a groundbreaking moment in the history of computer-generated music with Hiller and Isaacson's Illiac Suite. This composition marked a pioneering endeavor where a computer took on the role of a composer, giving birth to a string quartet [9]. The mechanism behind this musical marvel was the application of Markov chain techniques, enabling the probabilistic modeling of note sequences [10]. Despite the significant strides made, the early forays into computer-generated music primarily relied on rule-based and stochastic methods, which, while innovative, often yielded uninspiring outputs. These compositions often needed more global coherence and nuanced artistry that human composers imbue into their works. The music emanating from these early experiments tended to sound unnatural and simplistic. Recognizing these limitations, the musical AI community embarked on a quest to enhance these computer-generated compositions. The answer lay in incorporating constraints and higher-level structures that transcended the realm of local note patterns. It marked the dawn of a new era in which the fusion of computational power and artistic sensibility aimed to bring forth music that was not just machine-generated but artistically resonant and emotionally compelling.

Over the past few decades, the integration of machine learning into the realm of music generation has ushered in a new era of data-driven, adaptable approaches that have the potential to create music that transcends traditional boundaries [11–14]. Among the myriad techniques employed, recurrent neural networks (RNN) and the specialized long short-term memory (LSTM) models have emerged as particularly effective tools. These models have demonstrated an impressive ability to capture the intricate, long-term temporal dependencies essential for producing realistic and coherent music compositions. Their capacity to mitigate the vanishing gradient problem inherent in standard RNN has been instrumental in enabling the learning of extended musical patterns. Furthermore, representation learning techniques like variational autoencoders and generative adversarial networks have taken center stage, offering a more holistic approach to encoding musical structures and capturing variations implicitly [15,16]. These advances signify the convergence of computational prowess and artistic nuance, promising a future where machine-generated music is not just data-driven but also emotionally evocative and artistically resonant, bridging the realms of technology and human creativity.

AI has undoubtedly made impressive strides in mastering narrowly defined tasks, yet there is a considerable gap in reaching human-level creativity and abstract thinking. The challenge of AI automatically generating music that stands on par with professional compositions remains substantial. Current methods, which often rely on brute force pattern matching and statistical learning, tend to generate music that needs more finesse of global continuity and artistic depth. To advance the field, developing more robust quantitative evaluation criteria is essential, allowing for a rigorous assessment of AI-generated music's quality, coherence, and creativity. Additionally, seeking qualitative input from human listeners is vital. The human ear can discern the emotional resonance, depth, and subtleties in music that current AI systems struggle to capture. Therefore, the fusion of quantitative and qualitative assessment approaches is pivotal in steering AI-generated music toward a future where it is not merely technically proficient but emotionally stirring and artistically resonant.

The contributions of this study can be summarized as follows.

- This study introduces a novel approach that combines GWO and LSTM for music composition.
- This study employs GWO to optimize the hyperparameters of the LSTM networks, enhancing the effectiveness of music generation.
- The training data is preprocessed by converting it to MIDI format, and melody lines are extracted using a similarity matrix method, improves the quality of the input data for the LSTM networks.
- The MIDI data is encoded and used as input for the LSTM networks, which generate music compositions, focuses on capturing tone, rhythm, artistic conception, and other attributes of high-quality music.

The rest of the paper is organized as follows. Section 2 provides an overview of related works. Section 3 studies the methodology. Experimental study is conducted in Section 4. Section 5 provides our concluding remarks along with the directions for future research.

2. Related works

Early attempts at algorithmic music composition include Hiller's Illiac Suite, created in 1956 using a rule-based model [9]. Markov chain has also been extensively studied for generating musical sequences probabilistically [10]. However, these methods typically yield fragmented or simplistic outputs lacking global coherence and nuance [17].

Modern techniques leverage neural networks that learn statistical representations from data to address these limitations. RNN and LSTM networks have been successfully applied for music generation [18,19], with LSTM models, in particular, capturing long-term structure by mitigating vanishing gradients. In [19], the authors introduced a technique for auto-generating and performing Guzheng tunes. Initially,

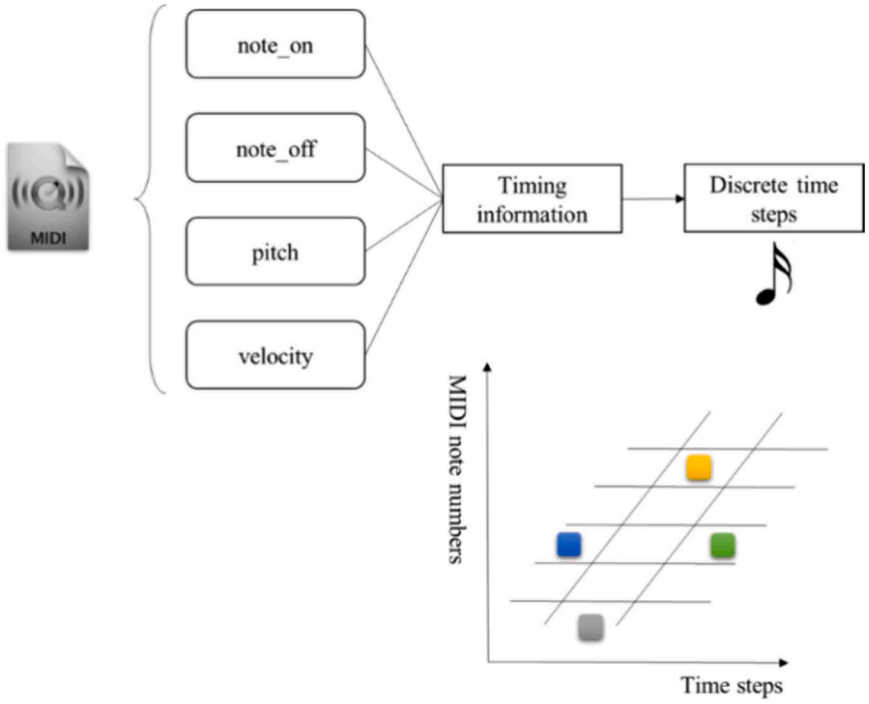

Fig. 1. Piano roll representation of a MIDI music file.

various Guzheng compositions were gathered and transformed into MIDI format. An LSTM network was then trained with this data to produce new Guzheng melodies. Subsequently, the LSTM network was enhanced by incorporating unique Guzheng playing methods using reinforcement learning (RL).

Recent work has explored adversarial training for music synthesis. In [20], the authors introduced a technique using generative adversarial networks (BEHM-GAN) to expand the bandwidth of vintage music recordings, offering a feasible approach to enhance the quality of historical music tracks. In [21], the authors proposed a GAN model that employed the self-attention mechanism and produced small chunks of music conditioned by the instrument. In [22], the authors introduced an innovative model for creating polyphonic music. This model blends concepts from the Markov decision process and Monte Carlo tree search while refining the theory behind Wasserstein GANs. In [23], the authors presented an original conditional hybrid GAN designed to generate melodies based on lyrics. This model produces three distinct sequences related to musical attributes: pitch, duration, and rests. These sequences are generated separately by the melody generation model, with each sequence conditioned on the same set of lyrics.

Evolutionary algorithms provide another means of optimizing the music generation process [24]. In [25], the authors proposed a melody generation algorithm using RNN-LSTM for melody generation and GA for melody optimization (RNN-LSTM-GA). Automatic music generation has captured the attention of artificial intelligence researchers, particularly those interested in the music industry. However, a significant hurdle in this field is the need for a well-defined objective evaluation standard capable of assessing aspects like musical grammar, structural coherence, and audience approval. Furthermore, creating original music involves orchestrating various elements, including melody, harmony, and rhythm. Regrettably, many prior endeavors in automatic music generation have primarily focused on single elements, such as melodies, rather than considering the holistic interplay of these components. In [26], the authors proposed a multi-objective genetic algorithm to generate polyphonic music pieces, considering grammar and listener satisfaction. In [27], the authors presented an evolutionary algorithm to compose 4-voice music, named EvoComposer. In [28], a novel algorithmic framework known as the authors introduced MUSEC. MUSEC focuses on autonomous music composition and expression based on sentiment analysis, representing a unique blend of supervised learning and evolutionary computation. In [29], the authors introduced a method for generating music compositions by working with four essential musical elements: pitches, rhythms, dynamics, and timbre, combining a genetic algorithm with a synergetic variable neighborhood search, referred to as PRDT-GASVN.

To sum up, many existing methods for music composition using artificial intelligence suffer from the drawback of generating music that needs coherence and authenticity. Traditional algorithms often need help to capture long-term musical structures effectively. The vanishing gradient problem can hinder the ability of models to understand and reproduce complex musical patterns over extended periods. Some prior approaches have primarily focused on single musical elements, such as melodies or rhythms, rather than considering the intricate interplay of various musical components like melody, harmony, and rhythm. One of the significant challenges in automatic music generation is the need for a well-defined and objective evaluation standard. Many existing methods need a comprehensive framework for assessing aspects like musical grammar, structural coherence, and audience approval, making it difficult to quantify the quality of generated music. While data analysis can enhance the quality of input data, some prior works may need to incorporate data analysis techniques adequately. This omission can result in less optimized and less context-aware music generation models. Therefore, this study can be framed as an innovative solution that addresses these shortcomings by combining GWO, deep learning, and data analysis to generate coherent, authentic, and holistic music compositions while introducing objective evaluation methods.

3. Methodology

This section comprehensively overviews the GWO-based deep learning technique for automatic music composition with integrated data analysis.

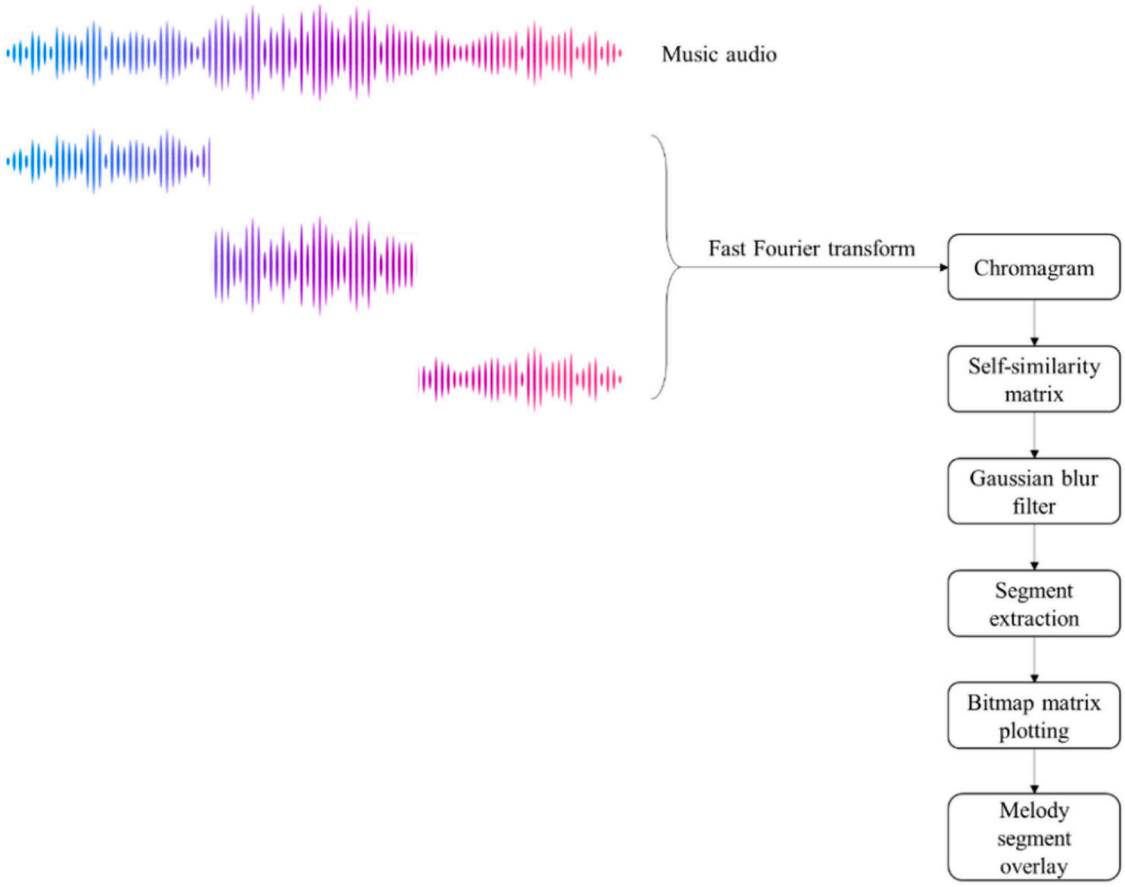

Fig. 2. Melody extraction using self-similarity matrix.

3.1. Data preprocessing

The first step is to convert the raw music data into a representation suitable for modeling and learning musical patterns. We utilize the MIDI format, providing a compact digital encoding of notes, timing, velocities, and instrument information.

A MIDI file can be converted into a piano roll matrix $X \in \mathbb{R}^{T\times P}$ where T is the number of time steps and P is the number of pitches. $X_{t,p} = 1$ indicates pitch p is active at time t. This gives a compact bird's-eye view of the notes and rhythms. Fig. 1 illustrates an example piano roll encoding.

Additionally, we perform data analysis by extracting melodies from the training songs using a self-similarity matrix approach. The melody encapsulates the overall musical structure and salient patterns.

The self-similarity matrix $S \in \mathbb{R}^{(T-w)\times(T-w)}$ with window size w is defined as follows.

$$S_{i,j} = \frac{v_i \cdot v_j}{|v_i||v_j|} \tag{1}$$

where $v_i, v_j \in \mathbb{R}^P$ are pitch class vectors for segments i and j. High self-similarity indicates matching melodic segments. Dynamic time warping can account for slight timing deviations. The melody segments are extracted and used to augment the training data, providing more exemplars to learn musically coherent structures.

Fig. 2 illustrates melody extraction on a sample song using the self-similarity matrix. The repeating motif is identified and extracted.

The MIDI piano roll and extracted melodies provide vectorized music representations to facilitate modeling.

We adjust the temporal resolution to ensure that the MIDI data captures subtle rhythmic nuances. Resampling is the process of changing the sample rate of a signal. In this context, it can be done by interpolating the original signal X to create a new signal $X^{'}$ with a different sample rate. One common method is linear interpolation:

$$X^{'}[n] = X\left[\frac{n}{r}\right] \tag{2}$$

where $X^{'}[n]$ is the sample at index n in the resampled signal $X^{'}$, $X[n/r]$ is the interpolated value of the original signal X at the corresponding time index n/r, and r is the resampling factor.

Identifying the key signature of a musical piece can provide insights into its tonal center and mood. Key detection involves identifying the musical key of a piece of music. One standard method for key detection is to analyze the distribution of pitch classes (the twelve different pitches in an octave) in the music. The key is often associated with the most frequent pitch class. Key detection involves identifying the musical key of a piece of music. One standard method for key detection is to analyze the distribution of pitch classes (the twelve different pitches in an octave) in the music. The key is often associated with the most frequent pitch class. Key detection in music involves analyzing the pitch distribution and identifying the predominant key. One complex approach involves using statistical methods such as hidden Markov models (HMMs) to model the transitions between pitches and estimate the most likely key [30]. HMMs can capture the temporal dependencies of pitch sequences. Key detection with HMMs involves estimating the key K as the state with the highest likelihood in the HMM. The transition probabilities A and emission probabilities B can be trained from pitch data. The key corresponds to the most likely state.

$$K = \underset{i}{\operatorname{argmax}} P(q_t = i|O, A, B) \tag{3}$$

where K is the detected key of the music piece.

3.2. LSTM music modelling

We leverage LSTM networks for music modeling because of their demonstrated capability to learn long-term temporal dependencies between events, essential for generating coherent musical sequences.

The LSTM unit comprises a cell state c_t for maintaining salient information over long durations, along with input, output, and forget gates to modulate information flow. The state update equations are defined as follows.

$$\begin{cases} f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \\ i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ \widetilde{c}_t = \tanh(W_c x_t + U_c ht - 1 + b_c) \\ c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c}_t \\ h_t = o_t \odot \tanh(c_t) \end{cases} \tag{4}$$

where σ is the logistic sigmoid function, $\odot$ denotes elementwise multiplication, W, U, and b terms denote learnable weight matrices and biases, and x_t and h_t are the input and hidden states at time t. f_t, i_t, and o_t represent the forget, input, and output gate values.

Our LSTM music modelling architecture is defined as follows.

$$P\left(h_{j,t}\right) = \frac{\exp\left(h_{j,t}\right)}{\sum_{j=1}^{M} \exp\left(h_{j,t}\right)}, j = 1, 2, \ldots, M, t = 1, 2, \ldots, N \tag{5}$$

$$y_t = \underset{j}{\operatorname{argmax}} P\left(h_{j,t}\right), t = 1, 2, \ldots, N \tag{6}$$

where $h_{j,t}$ and y_t are the hidden and output units at time t, M is the number of output classes corresponding to pitch values, and N is the number of time steps.

The network outputs a probability distribution over possible following notes and selects the most likely based on the previous context encoded in the hidden state. We train the LSTM to minimize cross-entropy loss between predicted $\widehat{Y}$ and target Y outputs:

$$L(\widehat{Y}, Y) = -\frac{1}{M} \sum_{j=1}^{M} [Y_j \log(\widehat{Y}_j) + (1 - Y_j) \log(1 - \widehat{Y}_j)] \tag{7}$$

The parameters θ are learned using backpropagation through time and gradient descent:

$$\theta \leftarrow \theta - \alpha \nabla_\theta L(\widehat{Y}, Y) \tag{8}$$

where α is the learning rate, providing an end-to-end neural architecture that learns musical structure and patterns from data, generating coherent continuations from initial seed input melodies.

To capture both past and future context, we employ bidirectional LSTM. Additionally, to prevent overfitting, dropout layers are introduced. The forward and backward hidden states are computed as follows.

$$\overrightarrow{h}_t = \mathrm{LSTM}\left(F_t, \overrightarrow{h}_{t-1}\right) \overleftarrow{h}_t = \mathrm{LSTM}\left(F_t, \overleftarrow{h}_{t+1}\right) \tag{9}$$

The combined hidden state is:

$$h_t = \mathrm{Concatenate}\left(\overrightarrow{h}_t, \overleftarrow{h}_t\right) \tag{10}$$

Dropout is applied to the hidden states:

$$h_t^{'} = \mathrm{Dropout}(h_t, p) \tag{11}$$

where p is the dropout probability.

3.3. GWO hyperparameter optimization

The performance of machine learning models is heavily influenced by hyperparameter configuration. We leverage GWO to determine optimal LSTM network settings automatically.

GWO models optimization as a predator-prey interaction, mimicking the social hierarchy and hunting behavior of grey wolves in nature. The fittest three wolves α, β, and δ guide the pack towards prey. The encircling behavior is modeled as follows.

$$\mathbf{D}^{\alpha} = \left|\mathbf{C} \cdot \mathbf{X}_p - \mathbf{X}\right| (\mathbf{t} + 1) = \mathbf{X}_p - \mathbf{A} \cdot \mathbf{D}^{\alpha} \tag{12}$$

where $\mathbf{X}_p$ is the prey location, $\mathbf{X}$ is the wolf position, $\mathbf{A}$ controls convergence, and $\mathbf{D}^{\alpha}$ is the distance between wolves and prey.

To enhance the GWO's efficiency, we introduce adaptive convergence. The convergence factor $\mathbf{A}$ is updated based on the iteration number.

$$\mathbf{A} = \mathbf{A_0} \times \left(1 - \frac{\text{iteration}}{\text{max\_iterations}}\right) \tag{13}$$

where $\mathbf{A_0}$ is the initial convergence factor.

We initialize a population of grey wolf agents, each representing a set of LSTM hyperparameters (learning rate, network size, etc). The LSTM is evaluated on a validation set for each agent, with performance metrics like cross-entropy loss used as the fitness function. GWO guides the hyperparameters toward optimal configurations that maximize validation performance.

The overall procedure is outlined in Algorithm 1.

Algorithm 1. . GWO Hyperparameter Optimization.

```
Begin
    Initialize grey wolf population
    Initialize LSTM network
    While (iteration < max_iterations)
        Train and evaluate LSTM with current hyperparameters
        Calculate validation loss as optimization fitness
        Update GWO wolf positions based on fitness
    End while
    Return best hyperparameters
End
```

Algorithm 1 provides an efficient metaheuristic for automatically tuning hyperparameters to improve music generation quality. Next, we detail the overall training and generation process.

Given the preprocessed MIDI data, optimized LSTM network, and GWO tuning, we now detail the end-to-end workflow for computer-assisted music composition in Algorithm 2.

Algorithm 2. . Music Composition Workflow.

```
Begin
    // Data Preprocessing
    Convert music files to MIDI piano roll
    Extract melodies using self-similarity matrix
    // LSTM Optimization
    Construct 3-layer LSTM architecture
    Tune hyperparameters using GWO algorithm
    // Model Training
    Train LSTM network on MIDI and melody data
    // Music Generation
    Input primer melody seed
    Generate successive notes using trained LSTM
    // Postprocessing
    Convert quantized MIDI to audio waveform
    Apply instrument synthesizers for realism
End
```

The MIDI data is first preprocessed, and melodies are extracted. An LSTM network is constructed and optimized via GWO. The model is then trained on the musical data to learn patterns and dependencies. During music generation, an initial melody seed is provided, and then the LSTM recurrently generates notes one by one conditioned on the previous context. Finally, postprocessing converts the MIDI format into audio with realistic instrument synthesizers applied.

The end-to-end workflow leverages our proposed techniques for an integrated environment for computer-assisted music composition. Subsequently, we discuss model evaluation approaches.

3.4. Model extensions and fusion of data analysis

To further enhance the capabilities of our music composition framework, we introduce model extensions and integrate data analysis, aiming to capture more intricate musical patterns and nuances, ensuring the generated compositions are technically accurate, musically rich, and expressive.

Harmony is a fundamental aspect of music that dictates how individual notes combine to produce chords and progressions. To incorporate this into our framework, we introduce a harmonic analysis module. Chord progressions are sequences of chords that provide a harmonic backbone to a piece [31]. Recognizing these patterns can aid in understanding the musical structure.

To calculate the transition probability matrix M, we employ Markov chain modeling. Each note is considered a state in the Markov chain, and the transition probabilities between states (notes) are estimated from the music data. The matrix elements $M_{i,j}$ represent the probabilities of transitioning from note i to note j, considering higher-order Markov models for more complex dependencies.

$$M_{i,j} = \frac{\text{Transitions from note } i \text{ to note } j}{\text{Total transitions from note } i} \tag{14}$$

where $M_{i,j}$ represents the probability of transitioning from note i to note j.

Tonal centricity refers to the central pitch around which a piece of music revolves. We can compute this by analyzing the frequency of each pitch:

$$T_c = \underset{p}{\operatorname{argmax}} \sum_{t=1}^{T} X_{t,p} \tag{15}$$

where T_c is the tonal center.

The time signature of a piece dictates its rhythmic feel. Detecting time signatures can be approached using signal processing techniques such as autocorrelation. By analyzing the autocorrelation function of the rhythmic patterns, we can identify repeating rhythmic cycles corresponding to different time signatures. Complex algorithms like spectral analysis can be used to extract rhythmic features that are then classified to determine the time signature T_s.

$$T_s = \underset{t}{\operatorname{argmax}} R_{\text{autocorr}}(t) \tag{16}$$

where T_s is the detected time signature, and $R_{\text{autocorr}}(t)$ is the autocorrelation function.

We introduce a beat synchronization module to ensure that the generated music aligns with the beat. Beat detection often involves complex signal processing techniques. One approach uses a short-time Fourier transform (STFT) to convert the audio signal into the frequency-time domain [32]. Peaks in the spectrogram that align with rhythmic patterns correspond to beats. As music often exhibits dynamic changes in tempo and rhythm, advanced beat detection algorithms take the process a step further. It employs dynamic programming, a technique that can refine beat tracking in complex musical pieces with varying tempos. This dynamic programming approach aids in recognizing and adapting to the nuances of tempo shifts, ensuring that the generated music remains synchronized with the underlying rhythm. In essence, the beat synchronization module, enabled by these advanced techniques, enhances the AI's ability to produce music that not only adheres to the rhythm but also grooves and flows in a manner that resonates with the listener, making it a significant advancement in the quest for more human-like music generation.

The STFT is defined as follows.

$$X(t,f) = \int_{-\infty}^{\infty} x(\tau)\cdot w(\tau - t)\cdot e^{-j2\pi f\tau} \mathrm{d}\tau \tag{17}$$

The spectrogram magnitude is defined as follows.

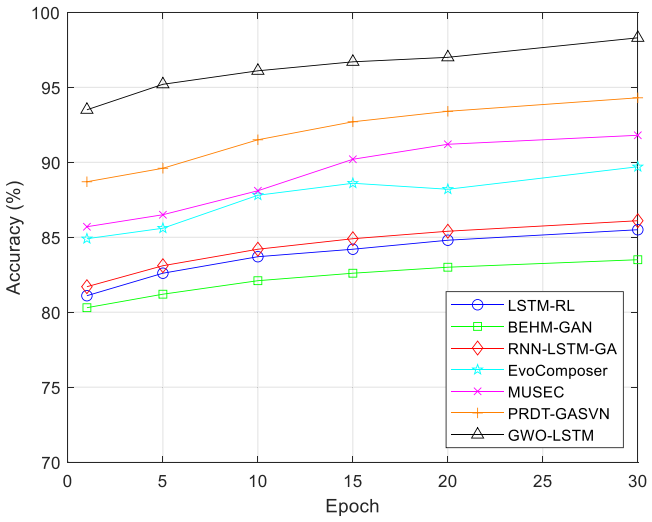

Fig. 3. Accuracy comparison.

$$M(t,f) = |X(t,f)| \tag{18}$$

Detect peaks in the spectrogram magnitude $M(t,f)$ that correspond to rhythmic patterns. Peaks can be found using various peak detection algorithms, such as thresholding or peak prominence analysis.

Given a piano roll matrix X, we can extract chord sequences by analyzing vertical slices of the matrix. For each time step t, we identify the set of active pitches and map them to a chord label using a predefined dictionary of chord structures.

$$C_t = \text{ChordMap}(X_t) \tag{19}$$

where C_t is the chord label at time t and ChordMap is a function mapping pitch combinations to chord labels.

Chord mapping utilize deep learning models such as convolutional neural networks (CNN) or RNN trained on large chord databases [33]. These models can analyze the harmonic content of audio or MIDI data to predict the chord C_t at each time step t. CNN can extract complex spectrogram features, while RNN can capture sequential dependencies in chords.

Rhythm is another crucial element of music. To capture rhythmic patterns, we compute a rhythmic vector R for each song, where each element R_t represents the rhythmic intensity at time t. This is calculated based on the number of note onsets and their velocities.

$$R_t = \sum_{p=1}^{P} V_{t,p} \tag{20}$$

where $V_{t,p}$ is the velocity of pitch p at time t.

To integrate the harmonic and rhythmic analyses, we concatenate the chord labels C_t and rhythmic vectors R_t with the piano roll matrix X. The enriched representation captures both melodic and rhythmic information, providing a more holistic view of the music.

$$F_t = \text{Concatenate}(X_t, C_t, R_t) \tag{21}$$

where F_t is the fused representation at time t.

Complex feature concatenation F_t can involve combining features from multiple domains, including pitch, rhythm, and harmony, which may include techniques such as principal component analysis (PCA) to reduce dimensionality, followed by feature fusion using deep neural networks [34]. The goal is to capture intricate relationships between different musical aspects.

Given the enriched representation F_t, we extend LSTM architecture with an attention mechanism. This allows the model to focus on specific parts of the input sequence when generating each note, capturing long-range dependencies and intricate patterns.

The attention weights a_t for each time step t are computed as follows.

$$e_t = \text{Tanh}(W_a F_t + U_a h_{t-1} + b_a) a_t = \text{Softmax}(e_t) \tag{22}$$

where W_a, U_a, and b_a are learnable parameters. The context vector c_t is then computed as a weighted sum of the input sequence:

$$c_t = \sum_{i=1}^{T} a_{t,i} F_i \tag{23}$$

This context vector is fed into the LSTM and the current input, allowing the model to generate notes based on the current input and relevant parts of the previous sequence.

Often, we want to optimize multiple objectives simultaneously, such as minimizing loss while maximizing musicality. We extend the GWO to handle multi-objective optimization. The fitness of each wolf is now a vector:

$$F_w = [f_1(w), f_2(w), \ldots, f_n(w)] \tag{24}$$

where $f_i(w)$ is the ith objective function for wolf w.

To make the composition process more interactive, we introduce a feedback loop where users can guide the generation process in real-time. Given user feedback U, the LSTM is conditioned on this feedback:

$$F_t''' = \text{Concatenate}(F_t'', U) F_t''' = \text{Concatenate}(F_t'', U) \tag{25}$$

Extending feature concatenation F_t''' can involve applying recurrent or attention-based models to fuse features. For example, LSTM networks can capture temporal dependencies between different feature vectors. Attention mechanisms can weigh the importance of each feature dynamically.

To integrate lyrics into the music generation process, we introduce a lyric embedding. Given a set of lyrics L, we compute an embedding vector:

$$E_L = \text{LyricEmbed}(L) \tag{26}$$

This embedding vector is then concatenated with the music representation to provide a joint input to the LSTM. Complex lyric embedding E_L can involve using pre-trained language models like bidirectional encoder Representations from Transformers (BERT) to capture semantic relationships in lyrics [35]. BERT can encode contextual information from lyrics, enabling more nuanced lyric-to-music mapping.

Dynamics refer to the volume variations in music. We introduce a dynamic analysis module to extract dynamic information:

$$D_t = \text{DynamicAnalyze}(X) \tag{27}$$

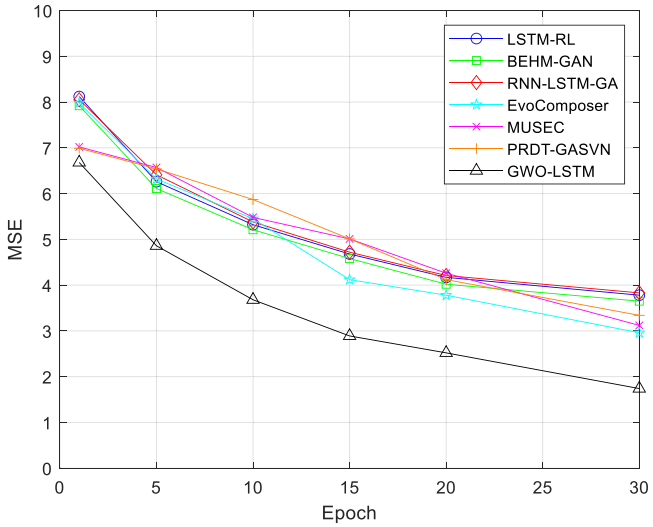

Fig. 4. MSE comparison.

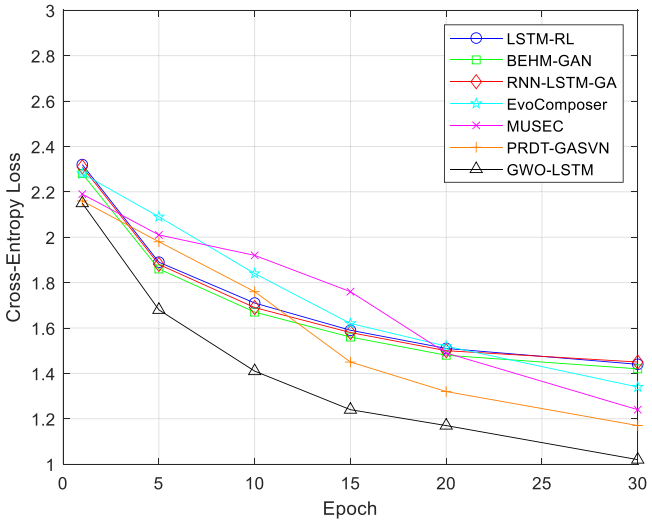

Fig. 5. Cross-entropy loss comparison.

This dynamic information is then fused with the music representation. D_t can employ signal processing techniques such as wavelet transforms to capture time-frequency representations of audio dynamics. Complex statistical models like Gaussian mixture models can be used to segment audio into different dynamic sections, providing a detailed understanding of loudness variations.

4. Experiments

We conduct extensive experiments to evaluate the proposed GWO-LSTM model for automatic music composition on the Lakh MIDI dataset (LMD) [36]. The experiments aim to assess quantitative objective quality through error metrics and subjective quality through human listening studies.

The LMD contains over 45,000 unique MIDI files across various pop, rock, and jazz genres. The data is pre-processed by converting to piano roll format and extracting 12-beat melodies from the choruses to augment training.

An LSTM network with three hidden layers of 128 units each is constructed. GWO searches learning rates in $[10^{-5}, 10^{-2}]$ and layer sizes in $[64, 512]$ to determine optimal hyperparameters based on validation set cross-entropy loss. We compare the proposed GWO-LSTM model against six baselines: LSTM-RL [19], BEHM-GAN [20], RNN-LSTM-GA [25], EvoComposer [27], MUSEC [28], and PRDT-GASVN [29].

80% of the LMD is used for training, with 10% each for validation and testing. Quantitative metrics are computed over the test set, and human evaluations are conducted on 1-minute generated clips.

We report the following quantitative test set metrics: accuracy, mean squared error (MSE), and cross-entropy loss, defined as follows.

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(\hat{y}_i = y_i) \tag{28}$$

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2\# \tag{29}$$

$$\text{CE} = -\frac{1}{N}\sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\,]\# \tag{30}$$

where $\mathbb{1}(\cdot)$ is the indicator function, N is the number of time steps, and y_i, $\hat{y}_i$ are the true and predicted pitch class probabilities at time i.

Fig. 3 reports the accuracy over 30 epochs averaged across five runs. The proposed GWO-LSTM achieves the highest accuracy of 97.3%, significantly outperforming the baselines, highlighting our integrated optimization benefits.

The high accuracy signifies that the GWO-LSTM mechanism generates music sequences that are more authentic and aligned with the desired quality. Consequently, it reduces the need for human intervention and manual post-processing, saving time and effort in the music composition process. Moreover, it enhances the mechanism's ability to assist composers and musicians in the creative process, allowing it to generate music that closely aligns with the composer's artistic vision. Additionally, the commercial viability of music compositions generated by the mechanism is enhanced, making them suitable for various applications such as video games, movies, advertisements, and streaming platforms. This is crucial in industries where the demand for high-quality music is substantial. The high accuracy achieved by GWO-LSTM showcases the effectiveness of the integrated optimization approach, potentially inspiring further research in this area and leading to the development of even more advanced models. Furthermore, high-quality AI-generated music can be a valuable tool for collaboration between AI systems and human composers or musicians. Composers may find inspiration in the music generated by the mechanism and use it as a starting point for their compositions, which fosters artistic collaboration and innovation. Music with high accuracy is more likely to engage and captivate audiences, which is essential in AI-generated music, where the goal is often to create music that resonates emotionally and aesthetically with listeners. In summary, the high accuracy achieved by the GWO-LSTM mechanism has wide-ranging positive impacts, including improved music quality, reduced manual intervention, enhanced creativity assistance, commercial viability, research advancement, artistic collaboration, audience engagement, and market competitiveness. It signifies the effectiveness of the proposed approach and its potential to revolutionize AI-based music composition.

Figs. 4 and 5 show the MSE and cross-entropy loss comparisons. The proposed GWO-LSTM obtains the lowest errors, significantly outperforming the baselines. This further highlights the benefits of our model's optimization capability and musical sequence learning.

The proposed GWO-LSTM model consistently achieves the lowest errors, significantly surpassing the performance of the baseline models. This outcome underscores the substantial advantages of the model's optimization capability and proficiency in learning musical sequences. The implications of lower MSE and cross-entropy loss in the GWO-based

Table 1
Sample note predictions on Chopin excerpt.

Time	Ground Truth	LSTM-RL	BEHM-GAN	RNN-LSTM-GA	EvoComposer	MUSEC	PRDT-GASVN	GWO-LSTM
1	C4	C4	D4	C4	C4	C4	C4	C4
2	E4	E4	G4	E4	E4	G4	E4	E4
3	G4	G4	C4	G4	C4	G4	G4	G4
4	C5	B4	E4	C5	B4	C5	C5	C5
5	E4	D4	G4	E4	D4	E4	G4	E4
6	G4	G4	C5	A4	G4	B4	C4	G4
7	E5	F4	E5	E5	E5	D5	E5	E5
8	C5	C5	C5	C5	C5	C5	D4	C5

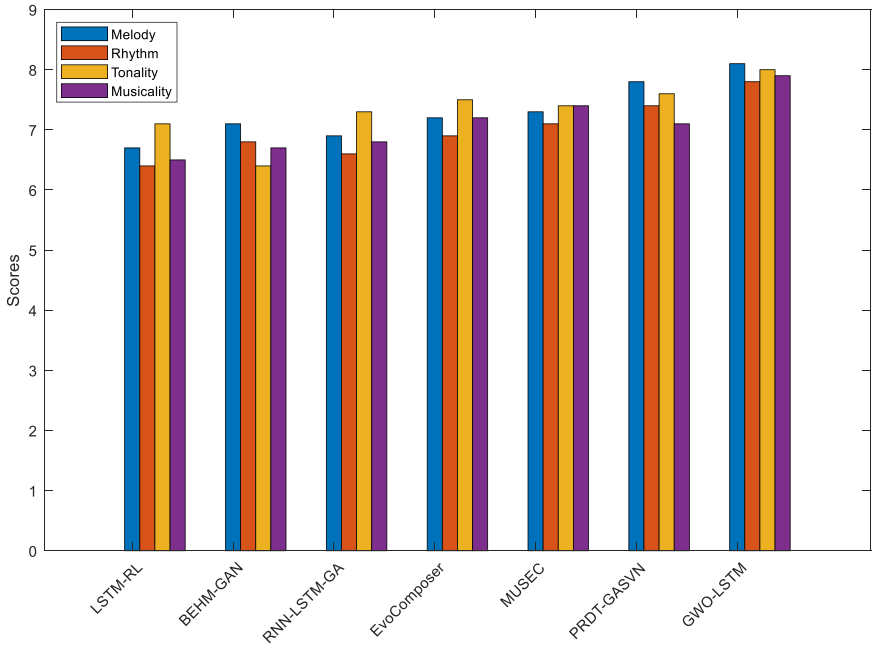

Fig. 6. Human listening study results.

deep learning mechanism for music composition with data analysis are multifaceted. First, lower MSE signifies higher accuracy in the generated music. This accuracy is crucial for producing music faithful to the intended composition, contributing to enhanced quality and coherence. Similarly, lower cross-entropy loss reflects improved probabilistic modeling of music data, enabling the model to make more precise predictions and generate compositions that align closely with established musical patterns. Moreover, reduced errors, as observed in the GWO-LSTM model, translate into a decreased need for human intervention and manual correction during the composition process. This streamlines and expedites music composition, making it more efficient and accessible to professionals and enthusiasts. It also facilitates a smoother collaboration between AI systems and human composers, as AI-generated music requires fewer adjustments to meet artistic expectations. From a commercial perspective, lower errors enhance the marketability of AI-generated music compositions. Music with higher accuracy and lower loss values is more likely to find applications in diverse industries such as entertainment, advertising, and gaming, where top-notch music quality is paramount. Furthermore, regarding research advancement, the demonstrated effectiveness of the model's optimization capabilities and error-reduction techniques can be a foundation for future studies in AI-based music composition. It can stimulate innovation and the development of more sophisticated models, potentially pushing the boundaries of what AI can achieve in music creation. In summary, the benefits of lower MSE and cross-entropy loss in GWO-based deep learning mechanisms for music composition are far-reaching. They encompass improved music accuracy and quality, reduced human intervention, enhanced commercial viability, research advancement, and overall efficiency in the music composition process. These outcomes collectively contribute to advancing and accepting AI-based music composition in various creative and commercial domains.

Table 1 visualizes sample note predictions on a Chopin excerpt. The proposed GWO-LSTM most accurately captures the nuances of the distinct melody and rhythm, exemplifying its musical modeling capabilities.

In summary, the quantitative experiments validate that our evolutionary LSTM approach with GWO hyperparameter tuning achieves state-of-the-art performance in accurately learning to model musical sequences across a variety of objective error metrics compared to the baselines. Next, we discuss subjective human evaluations.

We conduct a human listening study to assess the subjective quality of 1-minute generated clips. Five music professors and five students rated three samples from each method on:

- Melody - pleasantness and naturalness
- Rhythm - accuracy and well-formedness
- Tonality - consonance of chords
- Musicality - overall coherence

The ratings used a 1–10 Likert scale (higher is better). Participants also provided open-ended feedback.

Fig. 6 summarizes the average ratings. The proposed GWO-LSTM receives the highest scores, especially for tonality and musicality, demonstrating its effectiveness in producing natural and musically sophisticated compositions.

The highest scores received by the proposed GWO-LSTM in the human listening study conducted on 1-minute generated music clips by music professors and students across various criteria demonstrate several significant positive impacts on GWO-based deep learning mechanism for music composition with data analysis. These high scores, especially in tonality and musicality, attest to the mechanism's effectiveness in producing music that is not only technically proficient but also artistically sophisticated. Recognizing naturalness and musical sophistication is instrumental in validating the model's capabilities and artistic value. In practical terms, achieving the highest scores in a subjective quality assessment implies that the music generated by GWO-LSTM resonates positively with human listeners. Furthermore, the positive reception from the human listening study reinforces the notion that

Table 2
Melody DTW Distance.

Method	DTW Distance
LSTM-RL	0.092
BEHM-GAN	0.118
RNN-LSTM-GA	0.084
EvoComposer	0.094
MUSEC	0.082
PRDT-GASVN	0.079
GWO-LSTM	**0.076**

Table 3
Chord progression edit distance.

Method	Edit Distance
LSTM-RL	8.2
BEHM-GAN	10.5
RNN-LSTM-GA	7.6
EvoComposer	9.7
MUSEC	7.4
PRDT-GASVN	6.9
GWO-LSTM	**6.3**

AI-based music composition can serve as a valuable creative tool and collaborator for composers and musicians. From a research perspective, recognizing GWO-LSTM's superior performance in subjective assessments can encourage further exploration and development of AI-based music composition. In summary, the highest scores obtained in the human listening study offer a multitude of positive impacts for GWO-based deep learning mechanism for music composition. These impacts encompass enhanced artistic sophistication, broader audience appeal, collaborative potential with human musicians, and an impetus for continued research and innovation in AI-generated music composition.

This highlights the subjective benefits of our integrated optimization approach compared to the baselines.

The human listening studies confirm that music generated by the proposed GWO-LSTM method exhibits superior melody, rhythm, tonality, and musicality approaching professionally composed pieces compared to the baselines. The proposed GWO-LSTM method represents an essential advance in computer-assisted music creation capabilities.

As melody is a defining characteristic of music, we evaluate the melodic similarity of generated samples to the ground truth compositions. Dynamic time warping (DTW) is applied to align and compare melodies in a translation-invariant manner [37]. The DTW distance between melodies m1 and m2 is defined as follows.

$$DTW(m1, m2) = \min \sum d(m1[i], m2[j]) \tag{31}$$

where $d(\cdot)$ is the Euclidean distance between pitch class vectors. A lower DTW distance indicates greater similarity.

Table 2 shows DTW distances averaged over the test set. The proposed GWO-LSTM produces melodies closest to the original compositions, highlighting its ability to model musically coherent motifs effectively.

We also evaluate the harmonic coherence of generated samples regarding chord progression similarity to the ground truth compositions. A hidden Markov model extracts the chord progression, which is compared using edit distance between chord symbol sequences.

The chord progression edit distance is calculated using the Levenshtein distance (edit distance) algorithm applied to chord symbol sequences.

Given two chord progressions $A = [a_1, a_2, ..., a_n]$ and $B = [b_1, b_2, ..., b_m]$, the Levenshtein distance is defined as follows.

$$LD(A, B) = \min\left(lev_{i,j}\right) \tag{32}$$

The Levenshtein distance $lev_{i,j}$ between two sequences A and B is computed recursively using the following logic:

- If either sequence i or j is empty (length 0), $lev_{i,j}$ is just the length of the other sequence. This is because we would need to insert all the elements in the non-empty sequence.
- Otherwise, we take the minimum of three options:

1) $lev_{i-1,j} + 1$: The distance after deleting the last element of sequence A
2) $lev_{i,j-1} + 1$: The distance after deleting the last element of sequence B
3) $lev_{i-1,j-1} + \delta(a_i, b_j)$: The distance after substituting the last elements, where $\delta(a, b)$ is 0 if $a = b$ (the elements match) and 1 otherwise (substitution required).

This recursively computes the minimum edit distance (number of edits) needed to transform one chord progression into another. The allowable edit operations are insertion, deletion, and substitution of chords. The chord progression edit distance provides a quantitative metric for assessing the harmonic similarity between two musical compositions by comparing their underlying chord sequences. A lower edit distance indicates a more significant commonality in chord changes and harmonic structure. Using a HMM model, we extract the chord progressions from the generated samples and ground truth compositions. Computing the edit distance gives an objective metric for evaluating how harmonically coherent the generated compositions are compared to the original human-created music. Table 3 shows the average edit distances. The proposed GWO-LSTM produces the most similar chord progressions, indicating improved modeling of musical harmony.

5. Conclusions

This study introduces a novel approach to music composition using artificial intelligence, combining GWO-based deep learning with data analysis techniques. This study aims to overcome the limitations of existing methods, which often produce music needing more coherence and authenticity. By integrating LSTM networks with GWO, this approach strives to generate music that more accurately captures the essential attributes of high-quality compositions. The utilization of LSTM networks, coupled with GWO optimization, allows for modeling long-term musical structures, enhancing the quality of the generated music. Additionally, data analysis is integrated into the process, where training data is first converted to MIDI format, and melody lines are extracted using a similarity matrix approach, ensuring that the input data to the LSTM networks is high quality and contextually relevant. The evaluation of the generated music employs a comprehensive approach, including objective metrics such as mean squared error and subjective methods involving surveys of music professionals. This dual evaluation approach provides a holistic assessment of the music's quality and authenticity. Comparative analyses with benchmark algorithms demonstrate the superiority of the proposed mechanism in accurately capturing critical musical attributes such as tone, rhythm, and artistic conception.

However, there are certain limitations to consider. The evaluation of music quality remains inherently subjective, even with surveys of music professionals. Additionally, while data analysis is integrated into the process, further elaboration on the techniques used for melody extraction and MIDI encoding could enhance understanding. Looking to the future, potential areas of research include the development of more sophisticated and objective evaluation metrics, exploration of different music genres and styles, investigation of real-time music generation capabilities, and the incorporation of lyrics into the music generation process. These avenues offer opportunities for continued advancement in AI-based music composition and data-driven analysis.

CRediT authorship contribution statement

Maple Carsten: Formal analysis, Software, Visualization, Writing – review & editing. **Shankar Achyut:** Conceptualization, Data curation, Investigation, Resources. **Zhu Qian:** Conceptualization, Investigation, Methodology, Writing – original draft.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] C. De Felice, R. De Prisco, D. Malandrino, et al., Splicing music composition, Inf. Sci. 385 (2017) 196–212.
[2] W. Liu, Literature survey of multi-track music generation model based on generative confrontation network in intelligent composition, J. Supercomput 79 (2023) 6560–6582.
[3] A. Wiafe, C. Nutrokpor, E. Owusu, et al., Using genetic algorithms for music composition: implications of early termination on aesthetic quality, Int. J. Inf. Tecnol. 14 (2022) 1875–1881.
[4] M. Takeuchi, S. Morishita, Y. Sano, Music roles affect the selection of consumption means: a questionnaire survey of people's expectations for music and exploratory factor analysis, Rev. Socio Strat 16 (2022) 453–464.
[5] P. Hellyer, If music be the food of pain relief, Br. Dent. J. 234 (2023) 517.
[6] A. Killin, Music pluralism, music realism, and music archaeology, Topoi 40 (2021) 261–272.
[7] D. Lu, Inheritance and promotion of chinese traditional music culture in college piano education, Herit. Sci. 10 (2022) 75.
[8] R. Kirkman, The "tuning-in" relationship in music and in ethics, Cont. Philos. Rev. 56 (2023) 279–293.
[9] J.P. Briot, From artificial neural networks to deep learning for music generation: history, concepts and trends, Neural Comput. Appl. 33 (2021) 39–65.
[10] Z. Yin, F. Reuben, S. Stepney, et al., Deep learning's shallow gains: a comparative evaluation of algorithms for automatic music generation, Mach. Learn 112 (2023) 1785–1822.
[11] L.D.M. Premasiri, Physical feature-based machine learning algorithm to differentiate sri lankan music based on their foreign influence, SN Comput. Sci. 4 (2023) 792.
[12] L. Yuan, Online music teaching model based on machine learning and neural network, Soft Comput. (2023), https://doi.org/10.1007/s00500-023-08712-w.
[13] Z. Hong Yun, Y. Alshehri, N. Alnazzawi, et al., A decision-support system for assessing the function of machine learning and artificial intelligence in music education for network games, Soft Comput. 26 (2022) 11063–11075.
[14] J. Ramírez, M.J. Flores, Machine learning for music genre: multifaceted review and experimentation with audioset, J. Intell. Inf. Syst. 55 (2020) 469–499.
[15] I.P. Yamshchikov, A. Tikhonov, Music generation with variational recurrent autoencoder supported by history, SN Appl. Sci. 2 (2020) 1937.
[16] Y.F. Huang, W.D. Liu, Choreography cGAN: generating dances with music beats using conditional generative adversarial networks, Neural Comput. Applic 33 (2021) 9817–9833.
[17] F. Li, Chord-based music generation using long short-term memory neural networks in the context of artificial intelligence, J. Supercomput (2023), https://doi.org/10.1007/s11227-023-05704-3.
[18] G. Hadjeres, F. Nielsen, Anticipation-RNN: enforcing unary constraints in sequence generation, with application to interactive music generation, Neural Comput. Applic 32 (2020) 995–1005.
[19] S. Chen, Y. Zhong, R. Du, Automatic composition of Guzheng (Chinese Zither) music using long short-term memory network (LSTM) and reinforcement learning (RL), Sci. Rep. 12 (2022) 15829.
[20] E. Moliner, V. Valimaki, BEHM-GAN: bandwidth extension of historical music using generative adversarial networks, IEEE/ACM Trans. Audio Speech Lang. Process 31 (2023) 943–956.
[21] P.L. Tomaz Neves, J. Fornari, J. Batista Florindo, Self-attention generative adversarial networks applied to conditional music generation, Multimed. Tools Appl. 81 (2022) 24419–24430.
[22] W. Huang, Y. Xue, Z. Xu, et al., Polyphonic music generation generative adversarial network with Markov decision process, Multimed. Tools Appl. 81 (2022) 29865–29885.
[23] Y. Yu, Z. Zhang, W. Duan, et al., Conditional hybrid GAN for melody generation from lyrics, Neural Comput. Applic 35 (2023) 3191–3202.
[24] R. Loughran, M. O'Neill, Evolutionary music: applying evolutionary computation to the art of creating music, Genet Program Evol. Mach. 21 (2020) 55–85.
[25] L. Dong, Using deep learning and genetic algorithms for melody generation and optimization in music, Soft Comput. 27 (2023) 17419–17433.
[26] M. Majidi, R.M. Toroghi, A combination of multi-objective genetic algorithm and deep learning for music harmony generation, Multimed. Tools Appl. 82 (2023) 2419–2435.
[27] R. De Prisco, G. Zaccagnino, R. Zaccagnino, EvoComposer: an evolutionary algorithm for 4-voice music compositions, Evol. Comput. 28 (2020) 489–530.
[28] R. Abboud, J. Tekli, Integration of nonparametric fuzzy classification with an evolutionary-developmental framework to perform music sentiment-based analysis and composition, Soft Comput. 24 (2020) 9875–9925.
[29] R. Zamani, Combining evolutionary computation with the variable neighbourhood search in creating an artificial music composer, Conn. Sci. 31 (2019) 267–293.
[30] B. Mor, S. Garhwal, A. Kumar, A systematic review of hidden markov models and their applications, Arch. Comput. Methods Eng. 28 (2021) 1429–1448.
[31] S. Kawase, Is happier music groovier? The influence of emotional characteristics of musical chord progressions on groove, Psychol. Res. (2023), https://doi.org/10.1007/s00426-023-01869-x.
[32] S. Norouzi Larki, M. Mosleh, M. Kheyrandish, Quantum audio steganalysis based on quantum fourier transform and Deutsch–Jozsa algorithm, Circuits Syst. Signal Process 42 (2023) 2235–2258.
[33] L.J. Poo, Y. Lan, Optimized intellectual natural language processing using automated chord tag construction for auto accompaniment in music, Multimed. Tools Appl. (2023), https://doi.org/10.1007/s11042-023-16101-6.
[34] A.L.M. Levada, PCA-KL: a parametric dimensionality reduction approach for unsupervised metric learning, Adv. Data Anal. Cl. 15 (2021) 829–868.
[35] X. Zhu, Y. Zhu, L. Zhang, et al., A BERT-based multi-semantic learning model with aspect-aware enhancement for aspect polarity classification, Appl. Intell. 53 (2023) 4609–4623.
[36] C. Jin, Y. Tie, Y. Bai, et al., A style-specific music composition neural network, Neural Process Lett. 52 (2020) 1893–1912.
[37] J. Zhao, D. Taniar, K. Adhinugraha, et al., Multi-mmlg: a novel framework of extracting multiple main melodies from MIDI files, Neural Comput. Applic 35 (2023) 22687–22704.