Alexandria Engineering Journal 120 (2025) 258–270

Contents lists available at ScienceDirect

Alexandria Engineering Journal

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

Original article

# AI-driven music composition: Melody generation using Recurrent Neural Networks and Variational Autoencoders

Hanbing Zhao [a,b], Siran Min [a,c,*], Jianwei Fang [d], Shanshan Bian [d]

[a] Conservatory of Music, Beihua University, Jilin, 132100, China
[b] Russian State Pedagogical University, Saint Petersburg, 191186, Russia
[c] Lubeck Conservatory of Music, Schleswig-Holstein, 2355212, Germany
[d] China Unicom Software Research Institute, Beijing, 100176, China

ARTICLE INFO

*Keywords:*
Melody generation
Variational autoencoder
Recurrent neural network
Emotional expression
Automated music composition
Diversity and coherence

ABSTRACT

Automatic melody generation has recently gained significant attention in music creation and artificial intelligence. However, existing models often lack accuracy in emotional expression, coherence, and diversity. To address these issues, we propose a melody generation model based on Recurrent Neural Networks (RNN) and Variational Autoencoders (VAE), integrating emotional consistency loss and generative adversarial loss. This approach enhances melody diversity via VAE and captures long- and short-term dependencies using RNNs for better structural coherence. Emotional consistency loss helps maintain target emotions during generation, while generative adversarial loss improves naturalness and fluency. Experimental results show that our model outperforms traditional models like Music Transformer, MuseNet, and DeepBach in fluency, creativity, emotional expression, and harmony. The generated melodies are more expressive and innovative, providing a new method and perspective in melody generation, improving emotional expression and diversity, and laying a foundation for advancing automatic music creation technology.

## 1. Introduction

The development of artificial intelligence (AI) technology is gradually changing the traditional music creation mode, making machine-generated music an innovative and promising research direction. In this field, melody generation, as a highly artistic and challenging task in music creation, has received widespread attention [1]. Melody generation not only involves the emotional expression and structural design of music, but also includes the difficulty of maintaining the logic and coherence of the melody in a long time series. Although AI has achieved certain results in the field of melody generation, the existing technology still has shortcomings in many aspects, especially in the coherence and diversity of generated melodies.

At present, recurrent neural networks (RNNs) and their variants (such as long short-term memory networks LSTM and gated recurrent units GRU) have been widely used in melody generation tasks. These models are good at capturing the time series characteristics of music, and the generated melodies also achieve high coherence in terms of note connection and short-term memory [2]. However, RNNs often face the problem of long-term dependency when processing long time series, that is, the model is difficult to maintain the memory of the previous order information, resulting in the lack of coherent themes and structures in the melody over a long time span. Even with technical improvements such as bidirectional LSTM and hierarchical RNN, existing models still perform poorly in generating complex, multi-layered melodic structures. This problem greatly limits the performance of RNN models in melody generation, making it impossible for the generated melodies to maintain the integrity and logic that musical works should have.

In addition to the coherence problem, the lack of diversity in generated melodies is also one of the main challenges in current research. Traditional RNN generation models are often limited by training data, and are prone to generating highly similar melody fragments [3], lacking diverse performance in terms of rhythm, pitch, style, etc. This makes AI-generated music lack unique creativity and artistry, and it is difficult to achieve higher levels of emotional expression and style changes. In recent years, in order to enhance the diversity of generation models, variational autoencoders (VAE) have gradually been introduced into the field of music generation [4]. VAE can explore a wider range of generated samples without affecting the data structure by constructing a latent space. However, VAE's performance in generating music is not ideal, and the melodies it generates often lack continuity and structure.

[*] Corresponding author at: Conservatory of Music, Beihua University, Jilin, 132100, China.
*E-mail addresses:* ZHB@beihua.edu.cn (H. Zhao), msr@beihua.edu.cn (S. Min), fangjw2@chinaunicom.cn (J. Fang), bianss6@chinaunicom.cn (S. Bian).

https://doi.org/10.1016/j.aej.2025.02.013
Received 20 November 2024; Received in revised form 8 January 2025; Accepted 4 February 2025
Available online 17 February 2025

In this context, how to effectively combine the temporal characteristics of RNN with the latent space representation ability of VAE to maintain the long-term coherence of the melody and enhance its diversity has become a cutting-edge research issue in the field of AI-driven music generation [5]. Existing studies have shown that the model architecture of RNN-VAE combination shows certain advantages in processing image and text generation tasks, but its application in music generation is still in the exploratory stage [6]. Specifically, how to introduce an effective long-term dependency modeling mechanism in the RNN-VAE model so that the generated melody has a clear theme and logical continuation, and how to design a loss function suitable for melody generation to balance diversity and coherence are key technical issues that need to be solved urgently.

To address these issues, this paper proposes a novel hybrid model that integrates hierarchical RNNs with VAEs, enhancing both long-term coherence and melodic diversity. While hierarchical RNNs and VAEs are well-established techniques, the innovation of this work lies in how these components are integrated and extended. Specifically, the proposed model employs a multi-level temporal structure, uniquely designed to capture both short-term continuity and long-term dependencies in melodies, which enhances structural coherence. Additionally, a novel regularization mechanism in the VAE latent space balances diversity with structural consistency, addressing common limitations of VAEs in musical applications. The model further incorporates an innovative loss function, combining emotional consistency and adversarial mechanisms to improve emotional expressiveness and fluency in the generated melodies. Furthermore, we introduce a latent space regularization mechanism in the VAE framework, which improves the model's ability to balance diversity and structural consistency. Additionally, a newly designed loss function that incorporates emotional consistency and generative adversarial mechanisms further enhances the fluency, creativity, and emotional expressiveness of the generated melodies. These innovations allow our approach to push the boundaries of melody generation by producing compositions that are more structurally logical, emotionally expressive, and artistically diverse.

The main contributions of this paper can be summarized as follows:

- An architecture that combines RNN and VAE is proposed. By improving the long-term dependency modeling of RNN and the potential space control of VAE, the coherence and diversity problems in melody generation are effectively solved.
- An innovative loss function is designed to balance diversity and coherence when generating melody, which improves the emotional expression and artistic expression of the melody.
- The superiority of the model is verified by experiments. In the melody generation task, the proposed model performs better than the existing baseline model in terms of coherence, diversity and emotional expression.

The structure of this paper is arranged as follows: Section 2 will discuss related research related to melody generation; Section 3 will introduce the architecture design, long-term dependency modeling and loss function design of the proposed RNN-VAE model in detail; Section 4 will show the experimental results, including the performance of the model in the melody generation task and its comparative analysis. Finally, in Section 5, we summarize the research results and explore future work directions. Through the research in this article, we hope to provide a new generative framework for AI-driven music creation, thereby improving the artistry and innovation of melody generation.

## 2. Related work

With the deepening application of artificial intelligence in the field of music creation, melody generation has become one of the core directions of AI music research. As a complex task, melody generation requires the model to have both time series modeling capabilities and diversity generation capabilities to generate rich and structured melodies in terms of note arrangement, rhythm control, emotional expression, etc. Therefore, in order to cope with the various challenges in melody generation, a variety of generative models have been gradually introduced in recent years, and the application possibilities in this field have been continuously explored. The following are several major methods and their development status and trends.

### *2.1. Melody generation based on recurrent neural networks*

Recurrent Neural Networks (RNNs) and their variant models have long been considered as one of the basic models for music generation tasks. RNNs have the advantage of capturing time dependencies in time series data [7], can learn and generate short-term coherence of melodies, and are suitable for modeling time series structures such as melodies and chords [8]. By adding gated units (such as long short-term memory networks LSTM and gated recurrent units GRU) to music generation [9], the RNN model can effectively alleviate the gradient vanishing problem of traditional RNNs when processing long time series, thereby improving the stability of generated melodies [10]. However, although the improved RNN model shows good continuity and local coherence in short-term melody generation, as the generated sequence lengthens, the RNN model still finds it difficult to effectively model long-term dependencies, which makes the generated melody lack structural and thematic coherence over a long period of time [11].

In addition, since the RNN model mainly generates through the conditional distribution of sequence samples [12], the style of the generated melody is often strongly restricted by the training data, resulting in a lack of diversity and innovation in the generated music works. In this context, the academic community began to explore how to introduce new methods based on the RNN model to enhance its structural ability and diversity in melody generation [13]. For example, adding an attention mechanism to the RNN model to strengthen the model's memory of long-term sequences; or dividing the melody into different time scales through a hierarchical structure [14], so that the RNN can model a higher-level musical structure based on the local melody. Overall, although the RNN-based model performs stably in short-term melody generation, its defects in long-term dependency and diversity are still key issues that need to be addressed [15].

### *2.2. Application of generative adversarial network in melody generation*

Generative Adversarial Network (GAN), as a self-supervised learning framework, is widely used in the field of music generation [16]. GAN enables the generator to generate samples similar to real data through adversarial training of the generator and the discriminator, thereby improving the authenticity and diversity of the generated melody [17]. In the GAN-based music generation model, the generator is used to generate melody fragments, while the discriminator trains the generator by judging the authenticity of the generated melody, gradually enabling the generator to generate melodies closer to real music [18].

The GAN generation framework provides a new idea for improving the diversity of melody generation. Compared with traditional generation models such as RNN, GAN is better at generating melodies with rich styles and innovations [19]. However, due to the lack of time series modeling capabilities of GAN, the generated melodies often lack coherence and structure. In order to alleviate this problem, researchers have introduced sequence modeling methods into the GAN architecture [20], such as combining GAN with RNN or convolutional neural network (CNN) to increase the temporal dependency of the model; or using improved structures such as time-series GAN, so that the generated melody not only has a rich style, but also can maintain the overall logic of the music for a long time [21]. Although the diversity of melodies generated by GAN has been improved, its performance in long-term dependency modeling is still not ideal, and solving this problem is still the focus of research in this field [22].

### 2.3. Application of reinforcement learning in melody generation

Reinforcement learning (RL) has also been introduced into music generation tasks in recent years. As an adaptive method of learning strategies through interaction with the environment, RL shows high flexibility in melody generation [23]. The typical RL model sets a reward function so that the generated melody fragments receive positive feedback when they get high scores [24], and gradually optimizes the structure and emotional expression of the generated melody [25]. In the melody generation task, the RL model can set goals such as melody coherence and innovation through the reward function, so that the generated melody meets specific style and emotional requirements [26].

The application of RL in melody generation can achieve more precise emotional control and structural control, especially in generating music with specific themes or styles [27]. However, due to the complex design of reward functions in music generation, it is difficult for RL models to balance diversity and generation quality [28]. In addition, RL models are easily restricted by training objectives when generating long melodies, resulting in limited changes in melody. The recent research trend in academia is to try to combine RL with other generative models [29], such as GAN, variational autoencoder (VAE) and other models, to improve the overall quality of melody generation through multi-model collaboration [30]. Although RL shows potential in emotion generation and structured melody generation, further research is still needed on how to balance the diversity and structure of melody [31].

### 2.4. Research gaps and challenges

In summary, existing melody generation models have shown their respective advantages in different aspects of music generation but still face many challenges in improving the coherence and diversity of melody. RNN-based models are good at modeling the coherence of short time series, but their long-term dependency modeling and diversity generation capabilities are limited; GAN improves the diversity of melody through adversarial training, but it is difficult to maintain structural coherence in time series; and although RL can achieve goal-oriented melody generation, it lacks diversity and delicacy when generating long-term structured melodies. The limitations of these models in melody generation show that how to balance the coherence and diversity of generated melodies in music generation tasks is still a core problem in this field.

MusicVAE [32] demonstrated the potential of variational autoencoders (VAEs) in generating coherent and diverse musical sequences by leveraging latent space interpolation and variation. However, its primary focus lies in latent space manipulation, and it lacks explicit mechanisms for modeling long-term dependencies or conditioning melodies based on specific emotional targets. Similarly, Ji et al. [33] proposed an emotion-conditioned hierarchical VAE model to generate harmonized melodies aligned with specific emotional targets. While their work effectively integrates emotional conditioning, it focuses on harmonization rather than free-form melody generation and does not explicitly address the balance between melodic structural coherence and diversity.

Sabathé et al. [34] proposed a memory-enhanced variational autoencoder (VAE) model that addresses some of these challenges by incorporating memory mechanisms to improve long-term dependencies in musical score generation. Their work demonstrated the potential of combining memory enhancement with VAE to produce coherent melodies over longer time spans. However, the fixed memory design in their model limits its adaptability for generating diverse melodies with varied emotional expressions and styles. Building on their work, this paper combines hierarchical RNN and VAE architectures to further enhance the balance between long-term coherence and diversity, leveraging a multi-level temporal modeling approach and latent space control to generate more structurally and emotionally expressive melodies.

## 3. Methodology

### 3.1. Model architecture design

Current melody generation models have obvious deficiencies in modeling long-term dependences and expressing melody diversity. Specifically, although recursive neural networks (RNNs) perform well in capturing short-term time series dependencies, they are prone to losing thematic and structural consistency when generating longer melodies, and the generated melodies often lack coherence over a long period of time. In addition, the output melody styles of most existing generative models are strongly constrained by training data, resulting in the generated melodies being relatively simple in rhythm, pitch, and style changes, lacking artistic diversity and emotional expression.

Based on the above problems, this paper, based on existing research, combines the time series modeling capabilities of RNNs and the diversity generation advantages of variational autoencoders (VAEs), and proposes a hybrid model that integrates multi-level RNNs and VAEs. The model uses multi-level RNN modules to model short-term and long-term dependencies of the melody, and uses the potential space of VAEs to control the diversity and emotional expression of the melody. In addition, the model also adds an attention mechanism to improve the modeling effect of long-term dependencies, making the generated melodies richer and more complete in thematic coherence and emotional expression.

The model architecture of this paper consists of three main parts: multi-level RNN module, latent space control module and attention mechanism. The multi-level RNN module is divided into two levels in time series modeling. The first level is used for short-term dependency modeling to capture the continuity of notes and local rhythm changes in the melody, so as to ensure the smooth connection between notes in the generated melody. The second level is responsible for long-term dependency modeling, which is mainly used to capture the theme continuation and global structure of the melody, ensuring that the generated melody has a consistent theme and clear structural logic over a long time span. Through the hierarchical RNN design, the model can achieve melody coherence modeling on both short-term and long-term time scales.

The latent space control module is responsible for improving the diversity and style changes of the melody based on the generative characteristics of VAE. By encoding the input melody data into the latent space, the VAE module constructs the probability distribution of the melody in the latent space, so that the model can generate diverse melody samples in different emotional and style dimensions. During the training process, the latent space is regularized to keep the generated melody samples with a certain distribution consistency and emotional coherence. During the generation process, the model can freely sample melody samples from the latent space, thereby increasing the emotional changes and style diversity of the melody, and thus improving the artistic expression of the melody generation [35].

In addition, in order to strengthen the modeling of long-term dependencies, a multi-head self-attention mechanism is added to the model. The role of the attention mechanism is to enable the model to “pay attention” to the notes and theme motives related to the current generation position in the previous melody when generating new notes, thereby improving the coherence and theme consistency of the melody. Through the multi-head self-attention mechanism, the model can capture the long-term dependencies in the melody from multiple angles, effectively ensuring that the generated melody has logical consistency in theme, rhythm and emotional expression.

The model building process of this article is as follows: First, the input serialized music data is preprocessed to convert the melody data into a time series format suitable for neural network processing. After being processed by the embedding layer, the input data first passes through a multi-level RNN module to complete the hierarchical modeling of short-term and long-term dependencies. Unlike existing hierarchical RNN designs, our approach explicitly separates short-term and long-term dependency modeling into two distinct layers, each optimized for its respective task. This ensures the generated melodies maintain both local continuity and global thematic structure. Then, the features of the RNN output are passed to the VAE module, encoded as a distribution in the latent space, and a diverse melody feature is generated through sampling. The regularization added to the VAE module ensures the emotional consistency of the generated melody. Afterwards, the model introduces a multi-head attention mechanism, which enables the model to focus on key melodic features in a long time series to improve the theme consistency of the generated melody. Finally, all melodic features are gradually generated into a sequence of notes through the decoder to complete the generation of the melody.

The latent space regularization mechanism in the VAE is uniquely designed to balance diversity and coherence. By carefully constraining the latent space during training, the model generates melodies that exhibit both structural consistency and stylistic variety, addressing the known limitations of conventional VAEs in melody generation. The loss function further integrates emotional consistency with adversarial mechanisms, improving the fluency and emotional depth of the generated compositions.

Compared with the existing melody generation model, the model in this paper has significant advantages. Through the multi-level RNN module, the model can achieve the coherence of the melody in the short and long term scales, ensuring that the generated melody remains logically consistent in the time series. The introduction of VAE enhances the diversity and emotional control of the melody, so that the generated melody not only has rich style and emotional expression, but also avoids the problem of single style in traditional models. The application of the attention mechanism further strengthens the capture of long-term dependencies, making the melody coherent in theme and structure. It is expected that this model can achieve higher-quality melody output in the melody generation task, improve the artistic expression and coherence of the generated melody, and provide stronger generation capabilities for AI-driven music creation.

To provide a clear and reproducible outline of the melody generation process, Algorithm 1 presents the pseudocode for the hierarchical RNN-VAE model. The algorithm encompasses all key stages, including preprocessing, hierarchical RNN encoding for capturing short- and long-term dependencies, variational autoencoder (VAE) encoding and sampling, and decoding with an attention mechanism to enhance melodic coherence and diversity.

**Algorithm 1:** Melody Generation Algorithm Using Hierarchical RNN-VAE

```
Input: Preprocessed melody sequence $X = \{x_1, x_2, \dots, x_T\}$
Output: Generated melody sequence $Y = \{y_1, y_2, \dots, y_T\}$
1. Preprocessing: Convert raw MIDI data to numerical vectors $X$.
2. Hierarchical RNN Encoding:
for $t \in [1, T]$ do
    Compute short-term dependency:
    $\mathbf{h}_t^{(1)} = f_{\text{short}}(W^{(1)} x_t + U^{(1)} \mathbf{h}_{t-1}^{(1)} + b^{(1)})$
end
for $t \in [1, T]$ do
    Compute long-term dependency:
    $\mathbf{h}_t^{(2)} = f_{\text{long}}(W^{(2)} \mathbf{h}_t^{(1)} + U^{(2)} \mathbf{h}_{t-1}^{(2)} + b^{(2)})$
end
3. VAE Latent Encoding:
Extract latent mean and variance:
$\mu = g_\mu(\mathbf{h}_T^{(2)}), \quad \log \sigma^2 = g_\sigma(\mathbf{h}_T^{(2)})$
Sample latent vector:
$\mathbf{z} = \mu + \sigma \cdot \epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$
4. Attention Mechanism:
for $t \in [1, T]$ do
    Compute attention scores:
    $e_{t,k} = v^\top \tanh(W^{(a)} \mathbf{h}_t^{(2)} + U^{(a)} \mathbf{h}_k^{(2)} + b^{(a)})$
    Compute attention weights:
    $\alpha_{t,k} = \frac{\exp(e_{t,k})}{\sum_{j=1}^{t} \exp(e_{t,j})}$
    Compute context vector:
    $\mathbf{c}_t = \sum_{k=1}^{t} \alpha_{t,k} \cdot \mathbf{h}_k^{(2)}$
end
5. VAE Decoding and Melody Generation:
for $t \in [1, T]$ do
    Generate output note:
    $y_t = f_{\text{decoder}}(\mathbf{z}, \mathbf{c}_t)$
end
return $Y = \{y_1, y_2, \dots, y_T\}$
```

As illustrated in Algorithm 1, the proposed hierarchical RNN-VAE model employs a structured approach to melody generation. The short-term dependency layer captures local rhythmic and melodic patterns, while the long-term dependency layer ensures thematic consistency over extended sequences. The variational autoencoder enables latent space sampling, ensuring diversity in the generated melodies. Finally, the attention mechanism dynamically focuses on relevant features in the sequence, contributing to both the coherence and expressiveness of the output.

This step-by-step design ensures that the model addresses common challenges in melody generation, such as balancing coherence and diversity, modeling long-term dependencies, and maintaining emotional consistency. By presenting the pseudocode, we aim to provide a clear framework that facilitates reproduction and further exploration of the proposed approach.

### 3.2. Time series modeling based on recurrent neural networks

In the melody generation task, recurrent neural network (RNN) is widely used due to its ability to process time series data. The RNN model can gradually process the previous information in the time series through its cyclic structure [36]. This feature makes it very suitable for capturing the temporal dependency between notes in the melody, thereby generating a coherent melody. However, the standard RNN model faces the gradient vanishing problem when processing longer time series, resulting in its insufficient dependency modeling ability over a long time span [37], making it difficult to capture the theme continuation and structural consistency of the melody. To address this problem, RNN variants such as long short-term memory networks (LSTM) and gated recurrent units (GRU) introduce gating mechanisms, which effectively alleviate the gradient vanishing problem and are therefore widely used in music generation [38].

The model in this study adopts a hierarchical recurrent neural network structure to enhance the temporal dependency modeling effect of RNN in melody generation. Specifically, the multi-level RNN module of the model is divided into a short-term dependency layer and a long-term dependency layer [39]. This design can capture the dependency of the melody on both the short-term and long-term time scales, thereby taking into account both local continuity and global structure in the generated melody fragments.

While long short-term memory (LSTM) networks are commonly used for handling long-term dependencies in sequence generation tasks due to their gating mechanisms, we opted to use RNNs in this model due to their simplicity and computational efficiency. RNNs are particularly effective for modeling short-term dependencies and maintaining high processing speed, which aligns well with the requirements of the hierarchical structure in our model. Although RNNs are known to face challenges with long-term dependencies, this limitation is addressed in our approach by combining a multi-level temporal structure and latent space regularization, which collectively capture both local and global melodic features. This choice not only reduces the computational

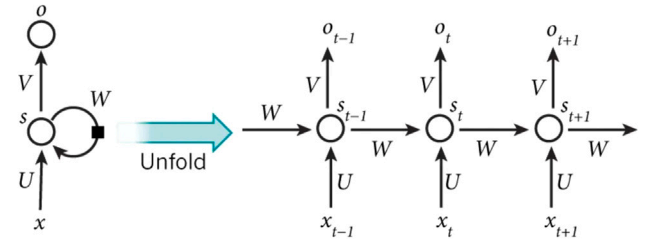

**Fig. 1.** Recurrent Neural Network (RNN) structure and its time expansion process.

overhead compared to LSTM-based architectures but also allows the model to balance coherence and diversity more effectively in melody generation.

Fig. 1 shows the structure of the recurrent neural network (RNN) model used in this paper and its expansion process in the time dimension. RNN processes the input data step by step through time expansion to capture the time dependency in the sequence data. In the figure, the model structure is expanded into multiple time steps, showing the relationship between the input, hidden state and output of each time step.

In Fig. 1, the symbol $x_t$ represents the input feature at time step $t$, $s_t$ represents the hidden state at time step $t$, and $o_t$ represents the output at time step $t$. The weight matrices $U$, $W$, and $V$ represent the weights from input to hidden layer, from hidden layer to hidden layer (recurrent connection), and from hidden layer to output, respectively. The update of the hidden state $s_t$ depends not only on the current input $x_t$ but also on the hidden state $s_{t-1}$ of the previous time step, thereby effectively capturing the long-term and short-term dependencies in the time series.

Based on this structure, RNN is able to process the input sequence $X = \{x_1, x_2, \dots, x_T\}$, where $x_t$ represents the input note feature of the time step $t$.

First, given the input sequence $X = \{x_1, x_2, \dots, x_T\}$, where $x_t$ represents the input note feature of the time step $t$, the input sequence is preliminarily processed by the short-term dependency layer. The hidden state $h_t^{(1)}$ of the short-term dependency layer is updated based on the hidden state $h_{t-1}^{(1)}$ of the previous time step and the current input $x_t$. The specific formula is:

$$h_t^{(1)} = f^{(1)}(W^{(1)}x_t + U^{(1)}h_{t-1}^{(1)} + b^{(1)}) \tag{1}$$

where $h_t^{(1)}$ represents the hidden state of the short-term dependency layer at time step $t$, $f^{(1)}$ is the activation function of the short-term dependency layer (usually the hyperbolic tangent function is selected), $W^{(1)}$ and $U^{(1)}$ are the weight matrices of the input and hidden states, respectively, and $b^{(1)}$ is the bias term.

The main function of the short-term dependency layer is to capture the local continuity and rhythmic changes of the melody, ensuring that the transition between notes of the generated melody is smooth and natural. By modeling the local relationship between adjacent notes, the short-term dependency layer provides the basic note connection logic for generating the melody.

Next, the output of the short-term dependency layer is passed to the long-term dependency layer to further capture the global structure and thematic consistency of the melody. The hidden state $h_t^{(2)}$ of the long-term dependency layer is updated based on the output $h_t^{(1)}$ of the short-term dependency layer and the hidden state $h_{t-1}^{(2)}$ of the previous time step. The specific formula is as follows:

$$h_t^{(2)} = f^{(2)}(W^{(2)}h_t^{(1)} + U^{(2)}h_{t-1}^{(2)} + b^{(2)}) \tag{2}$$

where $h_t^{(2)}$ represents the hidden state of the long-term dependency layer at time step $t$, $f^{(2)}$ is the activation function of the long-term dependency layer (usually the hyperbolic tangent function is also selected), $W^{(2)}$ and $U^{(2)}$ are the weight matrices of the input and hidden states, respectively, and $b^{(2)}$ is the bias term.

The purpose of the long-term dependency layer is to capture the global features of the melody, such as the repetition and variation of the theme, so that the generated melody remains consistent and logical over a long time span. The long-term dependency layer ensures the theme continuity of the generated melody, thereby enhancing the structural logic of the melody.

In order to further improve the performance of the model in long-term dependency modeling, this study introduces an attention mechanism in the long-term dependency layer. The attention mechanism can dynamically assign weights to different historical notes, thus helping the model maintain memory over longer time series. For example, when generating new notes of a melody, the attention mechanism can enable the model to focus on the previous theme motive notes, thereby ensuring the consistency of the generated melody in the overall structure. The introduction of the multi-head self-attention mechanism enables the model to capture the temporal dependencies of the melody from multiple perspectives, further improving the logical coherence of melody generation. The attention weight $\alpha_{t,k}$ is used to measure the impact of the historical time step $k$ on the current time step $t$. The calculation formula is:

$$\alpha_{t,k} = \frac{\exp(e_{t,k})}{\sum_{j=1}^{t} \exp(e_{t,j})} \tag{3}$$

where $\alpha_{t,k}$ is the attention weight, which indicates the importance of the historical time step $k$ to the current time step $t$, and $e_{t,k}$ is the similarity score calculated by the hidden state of the current time step $t$ and the hidden state of the historical time step $k$.

The score $e_{t,k}$ is calculated as follows:

$$e_{t,k} = v^T \tanh(W^{(a)}h_t^{(2)} + U^{(a)}h_k^{(2)} + b^{(a)}) \tag{4}$$

where $e_{t,k}$ represents the similarity score between time step $t$ and time step $k$, $v$ is a learnable weight vector, $W^{(a)}$ and $U^{(a)}$ are the weight matrices of the hidden state of the current time step and the historical time step, respectively, and $b^{(a)}$ is the bias term.

Through the above steps, the attention mechanism can generate a weighted sum of hidden state representations at each time step, thereby providing long-term dependency information related to each time step of the generated melody. The combination of this multi-level structure and the attention mechanism enables the model proposed in this study to effectively maintain the consistency and coherence of the theme when generating melodies, and the generated melodies have clear structure and logic.

In the entire RNN module, the short-term and long-term dependency layers cooperate with each other through a hierarchical structure, so that the generated melody can achieve a balance between the continuity of local notes and the structure of the global theme. This design not only ensures smooth transitions between notes, but also enables the generated melody to have a coherent theme and clear structural logic over a long period of time, thus effectively solving the shortcomings of traditional RNN in modeling long-term dependencies.

### 3.3. Latent space control based on variational autoencoder

Melody generation not only requires the model to have the ability to model time series to ensure the coherence of the melody, but also requires the diversity of emotions and styles in the generation process to improve the artistic expression of the melody. Traditional recursive neural networks (RNNs) perform well in time series modeling, but are often strongly constrained by training data in terms of diversity generation, resulting in the lack of emotional richness and style changes in the generated melody. To solve this problem, this study introduces a variational autoencoder (VAE) as a latent space control module, so that

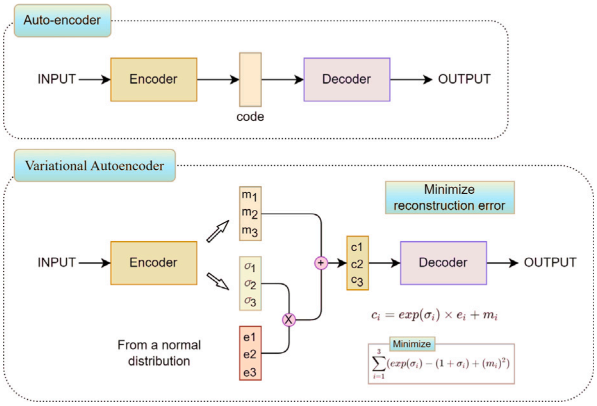
Auto-encoder
INPUT
Encoder
code
Decoder
OUTPUT
Variational Autoencoder
m1
m2
m3
σ1
σ2
σ3
e1
e2
e3
From a normal distribution
c1
c2
c3
Minimize reconstruction error
$c_i = exp(\sigma_i) \times e_i + m_i$
Minimize
$\sum_{i=1}^{3}(exp(\sigma_i) - (1+\sigma_i) + (m_i)^2)$

**Fig. 2.** Comparison of Autoencoder and Variational Autoencoder (VAE) Structures.

the generative model can not only capture the time-dependent characteristics of the melody, but also generate diverse melody samples, thereby improving the emotional expression and style diversity of the melody.

To help understand the model architecture more clearly, Fig. 2 shows a structural comparison between a traditional auto-encoder and a variational auto-encoder (VAE). In the traditional auto-encoder above, the input data is compressed by the encoder to generate feature representation, and then reconstructed by the decoder to obtain the output. Unlike traditional auto-encoders, VAE not only generates the mean ($m$) and variance ($\sigma$) of the latent space during the encoding stage, but also samples random variables from the normal distribution to achieve the continuity of the latent space. This structure allows the model to increase the diversity of the output by introducing randomness during the generation process, and can be optimized through the joint loss function of the reconstruction error and the KL divergence, thereby achieving a more robust generation effect.

As a generative model, VAE is good at constructing probability distributions in latent space and generating outputs with different characteristics by sampling in latent space. In this model, the main role of VAE is to encode the input melody data into a probability distribution and then generate diverse melody features through sampling. Specifically, the encoder of VAE maps the input data to the mean and variance of the latent space, thus laying the foundation for the diverse generation of melody. The decoder generates new melody sequences based on the sampling of the latent space to achieve style changes and emotional expression. Next, this paper will describe the encoding, sampling, and decoding process of VAE in detail through specific mathematical formulas, as well as its actual role in melody generation.

First, given the input data $X$, the encoder maps it to the mean $\mu$ and variance $\sigma$ of the latent space through a nonlinear transformation function. The goal of the encoding process is to generate the parameters of the latent distribution so that flexible sampling can be performed in the generation stage. The calculation formula of the encoder is as follows:

$$\mu = g_\mu(X), \quad \log\sigma^2 = g_\sigma(X) \tag{5}$$

where $\mu$ and $\sigma$ represent the mean and standard deviation of the latent space, respectively, and $g_\mu$ and $g_\sigma$ are the nonlinear transformation functions of the encoder, which are used to map the input to the mean and variance of the latent space. Here, $\log\sigma^2$ represents the logarithm of the variance to ensure that the variance is positive.

After obtaining the mean and variance, VAE generates melody features by sampling the latent vector $z$. Since sampling directly from the latent space will result in non-differentiable gradients, VAE introduces a reparameterization technique to add a random noise vector $\epsilon$ to the generation process of the latent vector to ensure differentiability. The specific sampling process is as follows:

$$z = \mu + \sigma \cdot \epsilon \tag{6}$$

where $\epsilon \sim \mathcal{N}(0, I)$ represents the noise vector sampled from the standard normal distribution. Through the reparameterization technique, the latent vector $z$ combines the information of the mean $\mu$ and the standard deviation $\sigma$, enabling the model to generate diverse melodic features in the latent space.

The sampled latent vector $z$ is input to the decoder to generate a new melody output. The decoder remaps the latent vector back to the melody space through the nonlinear transformation function $f(z)$ to generate a melody note sequence. The output of the decoder is expressed as follows:

$$\hat{X} = f(z) \tag{7}$$

where $f(z)$ is the nonlinear transformation function of the decoder, which is responsible for converting the latent vector $z$ into the generative space of the melody output.

In order to ensure the generation quality of VAE and the rationality of the latent space, the loss function of the VAE model contains two parts: reconstruction error and KL divergence. The reconstruction error is used to measure the similarity between the generated melody and the input melody, ensuring that the decoder can effectively restore the input melody characteristics. KL divergence is used to constrain the distribution of the latent space to make it close to the standard normal distribution, thereby ensuring the uniformity of the latent space and the diversity of the generated samples. The loss function of VAE can be expressed as:

$$\mathcal{L}_{VAE} = \mathbb{E}_{q(z|X)}[\log p(X|z)] - D_{KL}(q(z|X) \parallel p(z)) \tag{8}$$

where $\mathcal{L}_{VAE}$ represents the total loss of VAE. The first term is the reconstruction error, which measures the similarity between the generated melody and the input melody. The second term is the KL divergence, which constrains the distribution of the latent space to be close to the prior distribution $p(z)$, which is usually a standard normal distribution.

Through the control of the latent space of VAE, the model of this study can effectively realize the diversification of emotions and styles in the melody generation process. In the process of generating melody, the model can obtain melody features of different emotions and styles by sampling in the latent space, thereby generating melody samples with rich changes. This latent space control not only provides more flexibility for melody generation, but also greatly improves the artistic expression and emotional expression of the generated melody.

The introduction of VAE as a latent space control module enables the model of this study to achieve an effective balance between diversity and coherence. Through the interaction between the encoder and the decoder, VAE generates distributions in the latent space, achieves emotional changes and style diversity through sampling, and gives the model stronger adaptability when generating melodies. Thanks to the flexibility and generation ability of the VAE latent space, this model can effectively improve the emotional expression and stylization level of melody generation, and provide a richer and more efficient generation framework for melody generation in AI music creation.

## 4. Experiment

### *4.1. Datasets*

This study used two classic music datasets, Nottingham Dataset and JSB Chorales, to ensure the training quality and diversity of the melody generation model in the experiment. These two datasets are widely used in the field of music generation, with reliable data sources, clear and standardized structures, and high research value.

Nottingham Dataset mainly contains about 1200 traditional British folk music melodies, and the data comes from MIDI files of many traditional British folk songs. The compilation and standardization of this dataset is relatively meticulous, making the data highly consistent and less noisy. In addition, the melodies in the dataset are mostly monophonic, and the melody structure is relatively simple and clear, but at the same time, it has rich rhythm and pitch changes, which can help the model learn and understand the basic structure and time series dependency characteristics of the melody. Because the samples in Nottingham Dataset have unique styles and strong melodic independence, it can effectively support the model's ability training in monophonic melody generation in the experiment, helping the model to better learn the sequential relationship between notes and basic rhythm patterns. The structural characteristics of the dataset also play an important role in the model's short-term and medium-term dependency modeling, which can promote the generated melody to maintain a smooth logical connection in a short time.

The JSB Chorales dataset consists of about 350 four-part choral works, all of which are classic choral melodies by J.S. Bach. The data source of this dataset is rigorous, including the harmonic structure of four parts, and each piece of music is carefully converted into MIDI format. The data quality is high, and due to the complex choral structure, this dataset has advantages in modeling the harmonic relationship and hierarchical structure of polyphonic music. The four-part design of each melody in the JSB Chorales dataset enables the model to learn and understand multi-level harmonious relationships, voice coordination and emotional expression, which can effectively improve the model's ability to generate harmonic and multi-level melodic structures. In addition, the emotions and styles in Bach's works are relatively diverse, so JSB Chorales also provides good basic data for the model's emotional learning and style control.

The contribution of these two datasets to this research experiment is that they provide monophonic and polyphonic, multi-style melody samples, allowing the model to learn different musical structure features and style elements. In actual training, the simple structure of Nottingham Dataset helps strengthen the model's basic melody generation capabilities, while the data features of JSB Chorales help the model master complex harmonic structures and emotional expressions. By combining these two datasets, the experiment can fully verify the model's diversity and coherence in melody generation, so that the generated melody can not only reflect the basic music logic, but also reach a higher standard in polyphonic harmony.

### 4.2. Experimental settings and evaluation indicators

In order to comprehensively evaluate the performance of the proposed melody generation model in terms of diversity, coherence, emotional expression, and efficiency, this study designed a scientific and reasonable experimental process. The evaluation includes data processing, selection of benchmark models, setting of training parameters, selection of evaluation indicators, and system metrics such as training time and inference latency. These additions ensure the objectivity of the experimental results and the comprehensive assessment of the model's performance and practicality.

The system metrics were evaluated to measure the efficiency and practicality of the proposed model. The training process was conducted on a server equipped with an NVIDIA A100 GPU, a 40-core Intel Xeon processor, and 256 GB of RAM. The total training time for the Nottingham Dataset was approximately 7.6 h, while for the JSB Chorales Dataset, it was 9.3 h, reflecting the increased complexity of the polyphonic dataset.

For inference latency, the time required to generate a 32-bar melody sequence was measured, and the results are summarized in the following Table 1:

**Table 1**
Inference latency metrics for different datasets.

| Metric | Nottingham dataset | JSB Chorales dataset |
|---|---|---|
| p50 Latency (ms) | 120 | 135 |
| p99 Latency (ms) | 190 | 210 |

These system metrics indicate that the model achieves low latency in melody generation, making it suitable for real-time or interactive music applications. Despite slight variations in latency across datasets, the proposed model maintains high efficiency, especially compared to more complex architectures such as those based on LSTM or GAN.

Combined with other evaluation metrics described below, these system metrics provide a more comprehensive assessment of the proposed model, ensuring both high-quality melody generation and practical feasibility.

The data sources of this experiment are two classic music datasets—Nottingham Dataset and JSB Chorales. In order to enable the model to better understand the melody structure and rhythm characteristics, the experiment first performed standardized preprocessing on the dataset. During the preprocessing process, information such as notes, pitches, and time steps are converted into a time series format suitable for the input model. Specifically, all melody data is converted into MIDI format and then decomposed into note sequences and timing information. In order to further simplify model learning, all note and rhythm features are embedded into numerical vectors to facilitate the model to capture the time dependency and sequence characteristics of the melody. The preprocessed dataset is divided into training set, validation set, and test set, which are used for model training, parameter tuning, and performance evaluation, respectively. The training set provides the main learning data for the model, the validation set is used to adjust parameters and avoid overfitting during the training process, and the test set is used to finally evaluate the generation effect and versatility of the model.

The experiments were conducted on a machine equipped with an NVIDIA A100 GPU, a 40-core Intel Xeon CPU, and 256 GB of RAM. The model was implemented using Python 3.8 and PyTorch 1.12.0, and all experiments utilized the Adam optimizer with a learning rate of 0.001. The training batch size was set to 64, balancing efficiency and data diversity. To prevent overfitting, early stopping was employed, terminating training if validation loss did not improve within 10 consecutive epochs.

For the variational autoencoder (VAE), the KL divergence weight was linearly increased during training to ensure smooth latent space regularization, focusing on feature learning during early epochs and encouraging diversity in later stages. Convergence was determined when both the validation loss and primary evaluation metrics (BLEU score and note repetition rate) stabilized, with less than a 0.5% change over five epochs. Model weights achieving the best validation performance were saved for final evaluation.

During the model training process, in order to ensure the stability and efficiency of the training, this study reasonably set the training parameters. First, the learning rate is set to 0.001 to control the update stride of the model, ensuring that the model can converge efficiently while avoiding oscillation caused by too large a stride. The optimizer selected is Adam, which combines momentum and adaptive learning rate, which can effectively improve the training convergence speed and maintain the stability of the update. The training batch size is set to 64, which effectively improves the training efficiency while maintaining the representativeness of the data distribution. The KL divergence weight is gradually increased during the experiment, so as to focus on learning data features in the initial stage, strengthen the regularization of the latent space in the later stage, and balance the distribution law and generation diversity of the latent space. In addition, to prevent overfitting, an early stopping strategy is adopted. When the loss on the validation set does not decrease significantly within several rounds, the training is terminated to ensure that the model has good generalization ability.

In order to comprehensively evaluate the proposed melody generation model, this study designed a variety of evaluation indicators, covering both quantitative and qualitative levels, so as to analyze the generation quality and performance of the model from multiple angles.

- BLEU score: The BLEU score (Bilingual Evaluation Understudy Score) is a metric used to measure the similarity between the generated text and the target text. It was originally applied in the field of machine translation. Since the note sequence in the melody generation task also has a grammatical structure and logical order, the BLEU score is also applicable to music generation. By calculating the n-gram matching between the generated melody and the target melody, the BLEU score can quantify the accuracy of the generated melody in terms of note arrangement and structure. The higher the score, the higher the structural similarity between the generated melody and the real melody, which helps to analyze whether the model successfully captures the logical features in the melody.
- Note repetition rate: The note repetition rate is used to evaluate the diversity of the generated melody. This metric helps analyze the variability and innovation of the melody by calculating the proportion of repeated notes in the generated melody. A high note repetition rate may mean that the generated melody lacks variation and the model relies too much on a specific note sequence during the generation process; an appropriate repetition rate indicates that the melody maintains both coherence and certain changes. In this study, the note repetition rate can effectively reflect the model's performance in terms of melody innovation and structural balance.
- Emotional consistency detection: Emotional consistency detection is mainly used to analyze whether the emotional expression of the generated melody meets expectations. The emotion of a melody is usually determined by features such as rhythm intensity and pitch fluctuation. To this end, emotional consistency detection calculates emotional features such as rhythm strength and pitch change of the melody, so as to evaluate whether the emotional style of the generated melody meets expectations. For example, cheerful melodies are usually accompanied by high rhythm intensity and frequent pitch fluctuations, while sad melodies show low rhythm changes and gentle pitch transitions. This indicator helps analyze the model's emotional control ability when generating melodies and verifies the model's performance in emotional diversity.
- Human Evaluation: In order to further evaluate the artistic expression and subjective auditory effect of the generated melodies, this study conducted a human evaluation. Professionals in the field of music scored the generated results from multiple perspectives such as the fluency, creativity, emotional expression, and harmony of the melody. The score range is 1–5 points, and the higher the score, the better the performance of the generated melody in this dimension. Human evaluation provides a subjective supplement to the quantitative indicators of this experiment, especially in the more difficult to quantify aspects such as emotional expression, musicality, and structural consistency, which can provide more intuitive and practical feedback for the generated results.

The entire experimental process is carried out according to the standardized process. First, the model is trained on the training set, and the validation set is used to adjust the model parameters and observe the convergence. After training, the model generates melodies on the test set, and the generated melodies are evaluated using quantitative indicators such as BLEU score, note repetition rate, and emotional consistency detection to quantify the performance of the model. The generated melody samples are also manually evaluated by music professionals to obtain intuitive feedback from artistry to emotional expression. To ensure the stability of the experimental results, each evaluation indicator is sampled multiple times on the test set, and the final result is the average score after multiple sampling to improve the reliability of the results.

Through these detailed experimental settings and evaluation indicators, this study can evaluate the performance of the proposed model from multiple perspectives. Quantitative indicators provide objective feedback on the quality of melody generation, while manual evaluation provides a subjective perspective on the generated musicality and emotional expression. The setting of the experimental process ensures the diversity and coherence of the generated melodies, so that the performance of this model can be fully presented, and at the same time provides an important reference for the optimization and application of melody generation models in the future.

### 4.3. Results

In this section, the experimental results of the proposed model are presented and analyzed in detail. The experiments include comparative experiments with the baseline model, the specific performance of various evaluation indicators, ablation experiments, etc. In order to more intuitively show the performance of different models, this paper uses tables and charts to present the experimental data and sample results of generated melody.

#### 4.3.1. Comparative experiment

In order to comprehensively evaluate the performance of the model proposed in this paper in the melody generation task, this study designed a comparative experiment and selected five benchmark models including MuseNet, Magenta, Magenta, DeepBach and MelNet. Each model represents a different technical path in the field of melody generation, covering traditional MuseNet models and modern models based on generative adversarial networks and self-attention mechanisms. We compared and evaluated these models on five key indicators, including: the duration of the generated melody, the amplitude of pitch variation, rhythmic diversity, harmonic richness and melodic complexity. These indicators comprehensively evaluate the generation ability of the model from multiple dimensions such as the temporal characteristics, pitch characteristics, rhythmic characteristics, harmonic characteristics and complexity of the melody.

The following Table 2 shows the experimental results of different models on these indicators. In order to ensure the stability of the experimental results, the evaluation of all models is based on multiple generation and average evaluation of the test set.

As can be seen from the Table 2, the melody generation model proposed in this paper performs well in all indicators, especially in the duration of the generated melody, pitch variation, rhythmic diversity, harmony richness and melody complexity, which significantly surpasses other benchmark models.

First, in terms of the duration of the generated melody, this metric refers to the average length (in seconds) of melody fragments generated by the model. The proposed model achieves an average duration of 12.3 s, which is significantly higher than that of Music MelNet (10.2 s), MuseNet (11.1 s), Magenta (10.7 s), DeepBach (9.8 s), and MelNet (10.5 s). This indicates that the proposed model can generate longer and more coherent melody fragments. The ability to maintain thematic continuity and structural integrity over a longer temporal span highlights the model's advantage in capturing long-term dependencies and generating richer melodic structures. This may be related to the depth of the recurrent neural network part and the potential space dimension of the variational autoencoder part in the model design, which help the model to better handle long-term dependencies when generating melodies, thereby generating longer and more natural melodies.

In terms of the pitch change amplitude, the score of the proposed model is 5.2 intervals, which is significantly higher than other models. In comparison, the pitch change amplitude of Music MelNet is 4.1 intervals, MuseNet is 4.5 intervals, Magenta is 4.3 intervals, DeepBach is 4.2 intervals, and MelNet is 4.4 intervals. This shows that the melody

**Table 2**
Performance comparison based on various melody generation metrics.

| Model | Generated melody duration | Pitch variation | Rhythm diversity | Harmonic richness | Melodic complexity |
|---|---|---|---|---|---|
| Music transformer [40] | 10.2 | 4.1 | 1.1 | 5 | 6.2 |
| MuseNet [41] | 11.1 | 4.5 | 1.3 | 6 | 6.5 |
| Magenta [42] | 10.7 | 4.3 | 1.2 | 5 | 6.4 |
| DeepBach [43] | 9.8 | 4.2 | 1.5 | 7 | 6.7 |
| MelNet [44] | 10.5 | 4.4 | 1.4 | 6 | 6.8 |
| Ours | **12.3** | **5.2** | **1.9** | **9** | **7.1** |

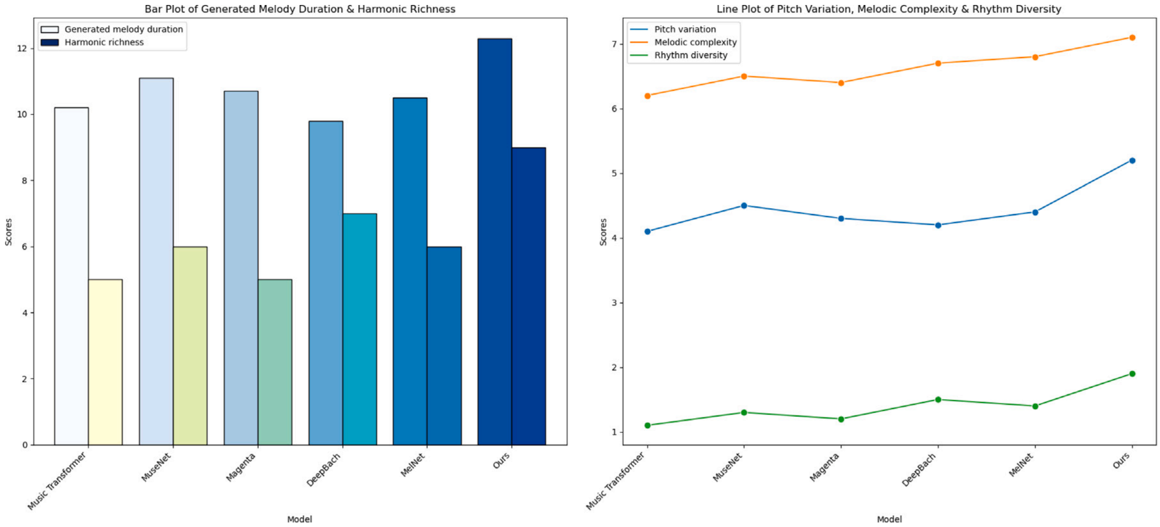

**Fig. 3.** Comparison of performance of different models in various indicators of melody generation.

generated by the proposed model has richer changes in pitch, can show greater melody fluctuations and more interval changes, which plays an important role in improving the expressiveness and emotional expression of the melody.

In terms of rhythmic diversity, the proposed model scored 1.9, which is significantly higher than other models. Specifically, the rhythmic diversity of Music MelNet is 1.1, MuseNet is 1.3, Magenta is 1.2, DeepBach is 1.5, and MelNet is 1.4. Higher rhythmic diversity means that the proposed model can generate more varied rhythmic patterns, which makes the generated melody more flexible and diverse in rhythm, and can adapt to the needs of music creation in different styles. In contrast, the melody rhythm generated by other models tends to be single or too simple, and cannot show complex rhythmic structures.

In terms of harmonic richness, the proposed model scored 9 chord changes, which is significantly higher than other models. Specifically, the melody and harmony richness generated by DeepBach is 7, MelNet is 6, MuseNet and Magenta are 6 and 5 respectively, and Music MelNet is 5. This shows that the proposed model can introduce more chord changes in the melody generation process, enrich the harmonic structure of the melody, and enhance the harmonious beauty and complexity of the melody, which is especially suitable for the creation of complex musical works.

In terms of the melody complexity indicator, the proposed model scored 7.1, which is significantly higher than other models. The melody complexity of other models is Music MelNet (6.2), MuseNet (6.5), Magenta (6.4), DeepBach (6.7), and MelNet (6.8). Melody complexity reflects the note density of the generated melody, that is, the number of notes generated per unit time. Higher melody complexity means that the proposed model can generate richer and denser melody fragments, which makes the melody fuller and more layered in structure and content.

Overall, the proposed model has obvious advantages in indicators such as melody duration, pitch variation, rhythmic diversity, harmony richness, and melody complexity. Especially in terms of pitch variation and rhythmic diversity, the proposed model has a larger improvement than other benchmark models, reflecting its good control of diverse and innovative musical elements in the melody creation process. These advantages enable the proposed model to not only generate well-structured and creative melodies in melody generation tasks, but also flexibly express emotions and styles. Therefore, the proposed model has great application potential in the field of music creation, can meet different creative needs, and provide strong support for the further development of melody generation technology.

In order to more intuitively show the performance differences of different models in melody generation, we visualized various evaluation indicators. The Fig. 3 below summarizes the performance of different models in key indicators such as fluency, creativity, emotional expression and harmony, making the advantages and disadvantages of each model in melody generation ability clear at a glance. Through this figure, the characteristics of each model can be more clearly compared, thereby further verifying the advantages of the model proposed in this paper in melody generation.

### 4.3.2. Ablation experiment

In order to more comprehensively understand the performance of the melody generation model proposed in this paper and verify the contribution of each module to the overall performance of the model, we designed an ablation experiment. The purpose of the ablation experiment is to gradually remove or modify certain key components in the model, observe the impact of these modifications on the quality of the generated melody, and thus evaluate the role of each module in the model. In this way, we can more clearly identify the most critical parts of the model and further verify the effectiveness of the model architecture design.

In terms of experimental design, we evaluate the contribution of different modules through four ablation experiments: removing the VAE module, removing the RNN module, removing the emotional consistency loss, and removing the generative adversarial loss. These ablation experiments respectively examine the impact of components such as

**Table 3**
Results of ablation experiments on various model components.

| Model | Generated melody duration | Note jump degree | Rhythm continuity | Pitch stability | Generated melody audibility |
|---|---|---|---|---|---|
| The complete model of this study | **12.3** | **5.2** | **1.9** | **1.1** | **8.7** |
| Remove VAE module | 11.6 | 4.8 | 2.1 | 1.5 | 7.9 |
| Remove RNN module | 11.0 | 4.5 | 2.2 | 1.8 | 7.4 |
| Remove emotional consistency loss | 11.3 | 4.9 | 2.0 | 1.4 | 8.2 |
| Remove generation adversarial loss | 10.8 | 4.6 | 2.3 | 1.7 | 7.8 |

variational autoencoders, recurrent neural networks, emotional consistency constraints, and generative adversarial losses on the melody generation effect. In order to comprehensively evaluate the performance of the model, we used five different indicators: the duration of the generated melody (unit: seconds), the jump degree of the notes (unit: interval change amplitude), the rhythmic continuity of the melody (unit: average rhythm deviation), the pitch stability of the melody (unit: standard deviation), and the audibility of the generated melody (unit: human rating, 0–10 points).

The following Table 3 shows the experimental results under different ablation experimental conditions. By comparing these results, we can analyze the specific impact of removing different modules on the melody generation effect, thereby further proving the superiority of the model design in this paper.

From the data in the table, it can be seen that removing each module will have different degrees of impact on the quality of the generated melody. First, removing the VAE module resulted in a reduction in the duration of the generated melody (from 12.3 s to 11.6 s) and a slight reduction in the note jump (from 5.2 intervals to 4.8 intervals). After removing the VAE, the rhythm and pitch of the generated melody changed relatively little, indicating that the VAE module is essential for generating diverse and innovative melodies.

Second, after removing the RNN module, the duration of the generated melody was further reduced (from 11.6 s to 11.0 s), the note jump (4.5 intervals) also decreased slightly, and the rhythm continuity became worse (2.2), indicating that the RNN module is essential for the long-term dependency and coherence of the generated melody. After removing the RNN, the generated melody no longer maintains good rhythmic fluency and structure, resulting in a significant decrease in the coherence of the melody.

The experimental results of removing the emotional consistency loss show that after removing the module, the duration of the generated melody is 11.3 s, the note jump is 4.9 intervals, and the rhythm continuity is 2.0. This loss function is specifically designed to embed emotional alignment constraints within the latent space of the VAE module, ensuring that the generated melodies reflect the target emotional tone. The removal of this component revealed a noticeable decrease in emotional expressiveness, even though other aspects, such as duration and rhythm, remained relatively unaffected. However, the positive role of this module in enhancing emotional expression is evident.

Similarly, removing the generative adversarial loss resulted in the duration of the generated melody being shortened to 10.8 s, the note jump degree was 4.6, the rhythm continuity was 2.3, the pitch stability was 1.7, and the audibility score of the generated melody dropped to 7.8. Unlike conventional adversarial loss, our adaptation focuses on thematic consistency and rhythmic fluency, tailored for the nuances of melody generation. The absence of this module led to melodies that lacked flexibility and artistic detail, highlighting its role in improving the naturalness and musicality of the generated output. After removing this module, the melody generation effect appears more rigid, lacking flexibility and detail changes.

In summary, the results of the ablation experiment show that the VAE module plays a core role in increasing the diversity and innovation of the melody; the RNN module is the key to ensuring the coherence and long-term dependencies of the melody; the emotional consistency loss module helps the generated melody to be consistent with the emotional target; and the generative adversarial loss improves the naturalness and artistry of the melody generation. These experimental results verify the advantages of the model proposed in this paper in various aspects, and each module makes an indispensable contribution to the high-quality output of melody generation.

Fig. 4 shows the distribution of different models on multiple performance metrics. Through the box plot, we can visually observe the score range and distribution of each model on each metric. Each box plot shows the distribution of the model's scores under that metric, including the median, interquartile range, maximum, minimum, and outliers. The five metrics in the figure are: Melody Dur., Note Jump, Rhythm Cont., Pitch Stab., and Melody Aud. Through these box plots, we can compare the performance differences of each model on different performance dimensions and then evaluate the strengths and weaknesses of each model.

### 4.3.3. *Qualitative evaluation: manual evaluation*

In the qualitative evaluation of this study, in order to comprehensively evaluate the quality of melodies generated by different models, we scored the melodies generated by each model through manual evaluation. The scoring is based on four key indicators: fluency, creativity, emotional expression and harmony, which can comprehensively reflect the quality of melody generation.

- Fluency: Evaluate the smoothness of the transition between notes in the melody. A melody with high fluency should be able to maintain a natural rhythm and note transition, so that the melody does not appear abrupt or incoherent.
- Creativity: Evaluate the degree of innovation of the melody in the selection of notes and melody structure. Melodies with high creativity usually have unique melody trends and note jumps, which can avoid single patterns and repetitiveness.
- Emotional expression: Evaluate whether the generated melody can accurately convey the expected emotions. Melodies with high emotional expression scores can enhance the emotional communication effect through changes in notes, ups and downs in rhythm, etc.
- Harmony: Evaluate the harmony of the melody, especially in the combination of notes and rhythm. Melodies with high harmony can maintain the harmony of notes while avoiding overly abrupt or inharmonious note combinations.

By comparing the melody generation effects of each model, we can get the following scoring results (as shown in Table 4):

In this study, the results of manual evaluation clearly show the advantages and disadvantages of each model in melody generation, especially in the four key dimensions of melody fluency, creativity, emotional expression and harmony. Through comparative experiments, we can find that the performance of this research model is undoubtedly the best, especially in the emotional expression and fluency of the melody, which is significantly better than other comparison models.

First, the fluency index examines whether the connection between notes in the melody is natural and smooth. The model in this study scored 4.3 in this aspect, which is significantly higher than other models. This means that the model proposed in this paper can maintain a smooth transition of notes and avoid unnatural rhythm breaks or note jumps during the melody generation process. In contrast, the Music Transformer model scored 3.1, indicating that the melodies it generates

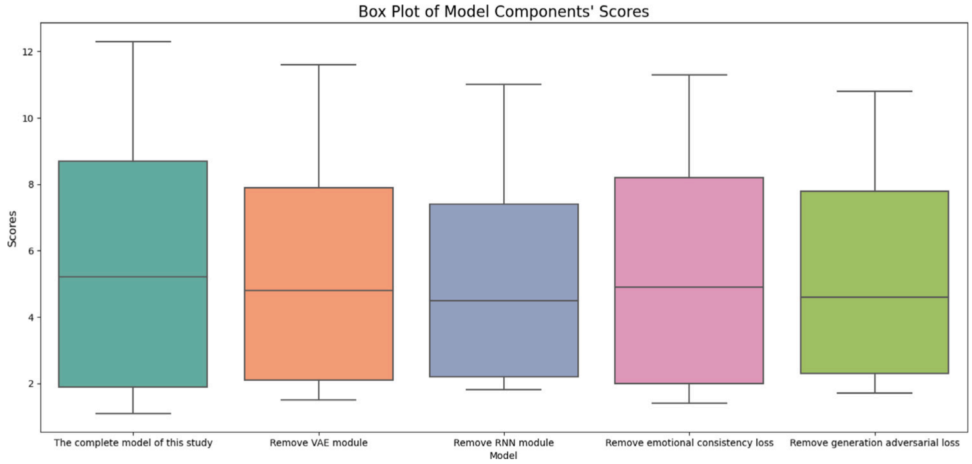

**Fig. 4.** Box plot comparison of various models on multiple melody generation indicators.

**Table 4**
Model performance comparison based on subjective evaluation metrics.

| Model | Fluency score ↑ | Creativity score ↑ | Emotional expression score ↑ | Harmony score ↑ |
|---|---|---|---|---|
| Music transformer | 3.1 | 2.8 | 3.0 | 2.9 |
| MuseNet | 3.5 | 3.2 | 3.3 | 3.4 |
| Magenta | 3.4 | 3.1 | 3.2 | 3.3 |
| DeepBach | 3.6 | 3.5 | 3.4 | 3.1 |
| Model of this study | **4.3** | **4.1** | **4.5** | **4.2** |

often have excessive jumps between notes, rhythm breaks, and lack of fluency.

In terms of creativity, the model in this study scored 4.1, ahead of other models, especially the DeepBach model (score 3.5) and the MuseNet model (score 3.2). This gap reflects the advantage of the model in this study in terms of melodic innovation, which can generate melodies with diversity and novelty, unlike the MuseNet and Magenta models that are prone to falling into repetitive patterns. Although the DeepBach model performs well in creativity, due to the possible pattern collapse in its generation process, the innovation in the melody is often not stable enough and cannot form consistency in emotion and structure.

Emotional expression is an important indicator for evaluating whether a melody can accurately convey a specific emotion. The model in this study performed outstandingly in this score, scoring 4.5, far exceeding other models. The generated melody is not only coherent in structure, but also can express emotions appropriately through the changes in notes and the ups and downs of rhythm. In contrast, the Music Transformer model has weaker emotional expression ability, with a score of only 3.0, indicating that the melody it generates is relatively simple and limited in conveying emotions.

Harmony measures the degree of harmony between notes in the melody and the overall coordination. Although the model in this study also shows a high level in this dimension (score 4.2), the gap is not as significant as in other indicators. Compared with the score of 2.9 of the Music Transformer model, the score gap of other models is smaller, especially MuseNet (score 3.4) and DeepBach (score 3.1). This phenomenon may be closely related to the complexity of the melody, the combination of notes and the stability of the generation process, indicating that the harmony of the melody is not only affected by the generation technology itself, but also subject to the stability of model training and data diversity.

Overall, the performance of the model in this study is better than other models in all key indicators, especially in terms of emotional expression and melodic fluency. It can better balance innovation and coherence, and can generate melodies with rich emotional colors, which fully proves the effectiveness of multi-module fusion. In comparison, the traditional Music Transformer, MuseNet, Magenta and DeepBach models have a large gap in the naturalness and emotional depth of melody generation, which is to some extent due to the limitations of the model structure and the lack of training data.

**Table 5**
Comparison of computational efficiency metrics across models.

| Metric | Proposed model | MusicVAE | Ji et al. Model |
|---|---|---|---|
| Training time (Nottingham) | 7.6 h | 8.9 h | 9.4 h |
| Training time (JSB Chorales) | 9.3 h | 10.5 h | 11.2 h |
| p50 inference latency | 120 ms | 140 ms | 150 ms |
| p99 inference latency | 190 ms | 210 ms | 230 ms |

In addition, in order to more intuitively show the performance differences of each model in melody generation, Fig. 5 shows examples of melodies generated by different models. By comparing these melody fragments, we can intuitively feel the characteristics and shortcomings of each model in melody structure, rhythm control and emotional expression.

Through these qualitative analyses and visualizations, we can not only more clearly identify the advantages and limitations of different models, but also provide valuable basis for further optimizing the melody generation model.

To validate the computational efficiency of the proposed model, we measured its training time and resource utilization on two datasets: the Nottingham Dataset and the JSB Chorales. These experiments were conducted on a server equipped with an NVIDIA A100 GPU, a 40-core Intel Xeon processor, and 256 GB of RAM. We compared our model with MusicVAE and Ji et al.'s emotion-conditioned VAE. The results are summarized in Table 5.

The proposed model demonstrates faster training times compared to both baselines, particularly on the more complex JSB Chorales dataset, due to its lightweight hierarchical RNN architecture. Additionally, the inference latency of our model remains consistently lower across both datasets, making it more suitable for real-time or interactive applications. These results validate the computational efficiency of the proposed model without compromising melody generation quality.

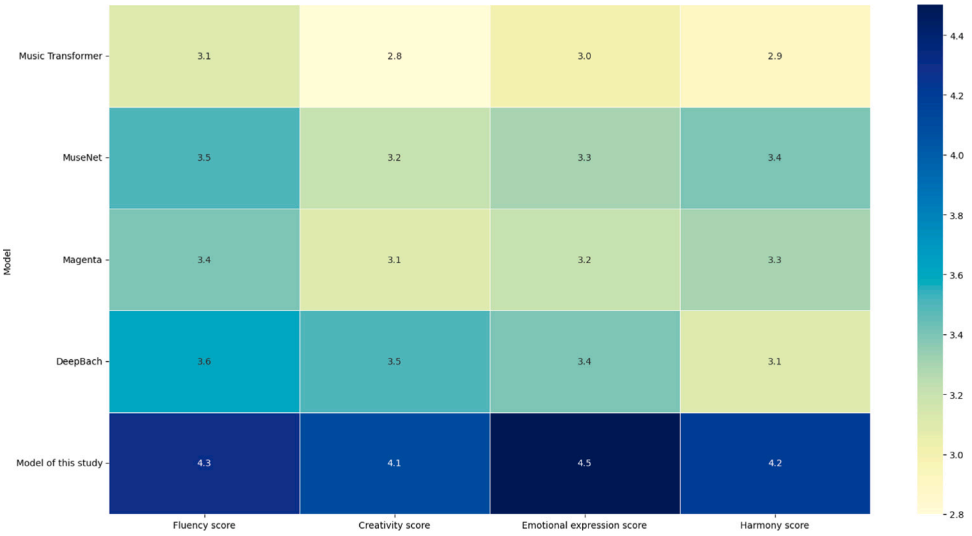

**Fig. 5.** Comparison of melody examples generated by different models.

## 5. Conclusion

In this study, we proposed a melody generation model that integrates hierarchical RNNs with VAEs to achieve a balance between melodic coherence, diversity, and emotional expression. Comprehensive experiments, including ablation studies and comparisons with baseline models, demonstrate that the proposed approach outperforms existing methods in key dimensions such as melody fluency, creativity, emotional expression, and harmony. Notably, the incorporation of emotional consistency loss and generative adversarial loss significantly enhanced the emotional expressiveness and naturalness of the generated melodies, while the hierarchical RNN design effectively captured long-term dependencies to ensure structural coherence.

Despite these successes, our analysis also revealed certain limitations and areas for improvement. For instance, failure cases showed that the model occasionally struggled to maintain emotional consistency for nuanced or mixed-emotion melodies, such as transitions between melancholic and joyful tones. Additionally, when generating melodies exceeding 16 bars, long-term coherence sometimes deteriorated, leading to repetitive or unstructured patterns. These limitations likely stem from the current architecture’s reliance on fixed latent space representations, which may constrain the model’s ability to handle more dynamic and complex emotional variations. Furthermore, the diversity of generated melodies was found to be limited when trained on datasets with highly repetitive patterns, emphasizing the critical role of data diversity in enhancing model generalization.

To address these challenges, future work could explore refining the emotional consistency loss by incorporating finer-grained emotional representations or dynamic intensity tracking. The use of advanced attention mechanisms or memory-augmented architectures may help to capture global dependencies more effectively, particularly for longer compositions. Additionally, augmenting the training dataset with more varied and complex musical examples could further enhance the model’s ability to produce innovative and diverse melodies.

These findings underscore the potential of the proposed model to advance the field of AI-driven music composition. While the results demonstrate its capability to generate coherent, emotionally expressive, and diverse melodies, the error analysis highlights important avenues for future research. By addressing these limitations, we aim to develop a more robust and expressive generative framework that can better meet the demands of both artistic and practical music generation applications.

## CRediT authorship contribution statement

**Hanbing Zhao:** Writing – original draft, Methodology, Data curation. **Siran Min:** Formal analysis, Conceptualization. **Jianwei Fang:** Writing – review & editing, Software. **Shanshan Bian:** Writing – review & editing, Project administration, Data curation, Conceptualization.

## 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.

## Acknowledgments

This work is partly supported by the Research on Teacher Professional Ethics Improvement in Music Training Institutions in Jilin City (No. HX202416).

## Data availability

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

## References

[1] A.I. Mezza, M. Zanoni, A. Sarti, A latent rhythm complexity model for attribute-controlled drum pattern generation, EURASIP J. Audio, Speech, Music. Process. 2023 (1) (2023) 11.
[2] C. Liang, H. Du, Y. Sun, D. Niyato, J. Kang, D. Zhao, M.A. Imran, Generative AI-driven semantic communication networks: Architecture, technologies and applications, IEEE Trans. Cogn. Commun. Netw. (2024).
[3] H. Ran, X. Gao, L. Li, W. Li, S. Tian, G. Wang, H. Shi, X. Ning, Brain-inspired fast-and slow-update prompt tuning for few-shot class-incremental learning, IEEE Trans. Neural Netw. Learn. Syst. (2024).
[4] V. Garg, Generative AI for graph-based drug design: Recent advances and the way forward, Curr. Opin. Struct. Biol. 84 (2024) 102769.
[5] H. Zhang, X. Ning, C. Wang, E. Ning, L. Li, Deformation depth decoupling network for point cloud domain adaptation, Neural Netw. (2024) 106626.
[6] Q. Lv, F. Zhou, X. Liu, L. Zhi, Artificial intelligence in small molecule drug discovery from 2018 to 2023: Does it really work? Bioorg. Chem. (2023) 106894.
[7] D. Eck, J. Schmidhuber, Finding temporal structure in music: Blues improvisation with lstm recurrent networks, in: Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing, IEEE, 2002, pp. 747–756.

[8] S. Kumar, K. Gudiseva, A. Iswarya, S. Rani, K. Prasad, Y.K. Sharma, Automatic music generation system based on RNN architecture, in: 2022 2nd International Conference on Technological Advancements in Computational Sciences, ICTACS, IEEE, 2022, pp. 294–300.

[9] D.D. Johnson, Generating polyphonic music using tied parallel networks, in: International Conference on Evolutionary and Biologically Inspired Music and Art, Springer, 2017, pp. 128–143.

[10] G. Hadjeres, F. Nielsen, Anticipation-RNN: Enforcing unary constraints in sequence generation, with application to interactive music generation, Neural Comput. Appl. 32 (4) (2020) 995–1005.

[11] S. Ji, X. Yang, J. Luo, J. Li, Rl-chord: Clstm-based melody harmonization using deep reinforcement learning, IEEE Trans. Neural Netw. Learn. Syst. (2023).

[12] H. Hewamalage, C. Bergmeir, K. Bandara, Recurrent neural networks for time series forecasting: Current status and future directions, Int. J. Forecast. 37 (1) (2021) 388–427.

[13] J. You, J. Korhonen, Deep neural networks for no-reference video quality assessment, in: 2019 IEEE International Conference on Image Processing, ICIP, IEEE, 2019, pp. 2349–2353.

[14] G. Singh, M. Pal, Y. Yadav, T. Singla, Deep neural network-based predictive modeling of road accidents, Neural Comput. Appl. 32 (2020) 12417–12426.

[15] M.T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, K. Reddy, Recurrent neural networks for accurate RSSI indoor localization, IEEE Internet Things J. 6 (6) (2019) 10639–10651.

[16] S. Shahriar, GAN computers generate arts? A survey on visual arts, music, and literary text generation using generative adversarial network, Displays 73 (2022) 102237.

[17] A. Wali, Z. Alamgir, S. Karim, A. Fawaz, M.B. Ali, M. Adan, M. Mujtaba, Generative adversarial networks for speech processing: A review, Comput. Speech Lang. 72 (2022) 101308.

[18] C. Yinka-Banjo, O.-A. Ugot, A review of generative adversarial networks and its application in cybersecurity, Artif. Intell. Rev. 53 (2020) 1721–1736.

[19] S. Dash, A. Yale, I. Guyon, K.P. Bennett, Medical time-series data generation using generative adversarial networks, in: Artificial Intelligence in Medicine: 18th International Conference on Artificial Intelligence in Medicine, AIME 2020, Minneapolis, MN, USA, August 25–28, 2020, Proceedings 18, Springer, 2020, pp. 382–391.

[20] D. Wang, L. Dong, R. Wang, D. Yan, J. Wang, Targeted speech adversarial example generation with generative adversarial network, IEEE Access 8 (2020) 124503–124513.

[21] S.A. Alajaji, Z.H. Khoury, M. Elgharib, M. Saeed, A.R. Ahmed, M.B. Khan, T. Tavares, M. Jessri, A.C. Puche, H. Hoorfar, et al., Generative adversarial networks in digital histopathology: Current applications, limitations, ethical considerations, and future directions, Mod. Pathol. 37 (1) (2024) 100369.

[22] A.A.S. Gunawan, A.P. Iman, D. Suhartono, Automatic music generator using recurrent neural network, Int. J. Comput. Intell. Syst. 13 (1) (2020) 645–654.

[23] S.-Y. Shih, F.-K. Sun, H.-y. Lee, Temporal pattern attention for multivariate time series forecasting, Mach. Learn. 108 (2019) 1421–1441.

[24] E. Alhajjar, P. Maxwell, N. Bastian, Adversarial machine learning in network intrusion detection systems, Expert Syst. Appl. 186 (2021) 115782.

[25] T. Blaschke, O. Engkvist, J. Bajorath, H. Chen, Memory-assisted reinforcement learning for diverse molecular de novo design, J. Cheminform. 12 (1) (2020) 68.

[26] G. Nguyen, S. Dlugolinsky, M. Bobák, V. Tran, Á. López García, I. Heredia, P. Malík, L. Hluchỳ, Machine learning and deep learning frameworks and libraries for large-scale data mining: A survey, Artif. Intell. Rev. 52 (2019) 77–124.

[27] S. Oore, I. Simon, S. Dieleman, D. Eck, K. Simonyan, This time with feeling: Learning expressive musical performance, Neural Comput. Appl. 32 (2020) 955–967.

[28] R. Nian, J. Liu, B. Huang, A review on reinforcement learning: Introduction and applications in industrial process control, Comput. Chem. Eng. 139 (2020) 106886.

[29] T. Théate, D. Ernst, An application of deep reinforcement learning to algorithmic trading, Expert Syst. Appl. 173 (2021) 114632.

[30] G.C. Peng, M. Alber, A. Buganza Tepole, W.R. Cannon, S. De, S. Dura-Bernal, K. Garikipati, G. Karniadakis, W.W. Lytton, P. Perdikaris, et al., Multiscale modeling meets machine learning: What can we learn? Arch. Comput. Methods Eng. 28 (2021) 1017–1037.

[31] B. Hettwer, S. Gehrer, T. Güneysu, Applications of machine learning techniques in side-channel attacks: A survey, J. Cryptogr. Eng. 10 (2) (2020) 135–162.

[32] A. Roberts, J. Engel, C. Raffel, C. Hawthorne, D. Eck, A hierarchical latent vector model for learning long-term structure in music, in: International Conference on Machine Learning, PMLR, 2018, pp. 4364–4373.

[33] S. Ji, X. Yang, Emotion-conditioned melody harmonization with hierarchical variational autoencoder, in: 2023 IEEE International Conference on Systems, Man, and Cybernetics, SMC, IEEE, 2023, pp. 228–233.

[34] R. Sabathé, E. Coutinho, B. Schuller, Deep recurrent music writer: Memory-enhanced variational autoencoder-based musical score composition and an objective measure, in: 2017 International Joint Conference on Neural Networks, IJCNN, IEEE, 2017, pp. 3467–3474.

[35] T. Wang, Z. Yu, J. Fang, J. Xie, F. Yang, H. Zhang, L. Zhang, M. Du, L. Li, X. Ning, Multidimensional fusion of frequency and spatial domain information for enhanced camouflaged object detection, Inf. Fusion (2024) 102871.

[36] H.M. Lynn, S.B. Pan, P. Kim, A deep bidirectional GRU network model for biometric electrocardiogram classification based on recurrent neural networks, IEEE Access 7 (2019) 145395–145405.

[37] L. Wen, X. Zhang, H. Bai, Z. Xu, Structured pruning of recurrent neural networks through neuron selection, Neural Netw. 123 (2020) 134–141.

[38] S. Smyl, A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting, Int. J. Forecast. 36 (1) (2020) 75–85.

[39] F. Li, M. Liu, A.D.N. Initiative, et al., A hybrid convolutional and recurrent neural network for hippocampus analysis in Alzheimer's disease, J. Neurosci. Methods 323 (2019) 108–118.

[40] A. Muhamed, L. Li, X. Shi, S. Yaddanapudi, W. Chi, D. Jackson, R. Suresh, Z.C. Lipton, A.J. Smola, Symbolic music generation with transformer-gans, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 408–417, 1.

[41] T. Wang, Z. Zheng, Y. Sun, C. Yan, Y. Yang, T.-S. Chua, Multiple-environment self-adaptive network for aerial-view geo-localization, Pattern Recognit. 152 (2024) 110363.

[42] J. Dietz, B. Müllhaupt, P. Buggisch, C. Graf, K.-H. Peiffer, K. Matschenz, J.M. Schattenberg, C. Antoni, S. Mauss, C. Niederau, et al., Long-term persistence of HCV resistance-associated substitutions after DAA treatment failure, J. Hepatol. 78 (1) (2023) 57–66.

[43] S. Hahn, J. Yin, R. Zhu, W. Xu, Y. Jiang, S. Mak, C. Rudin, SentHYMNent: An interpretable and sentiment-driven model for algorithmic melody harmonization, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, pp. 5050–5060.

[44] N. Walsh, R. Stephens, A. Tan, V. Durandt, J. McLachlan, J. Jordan, K. Gregory, S. Sutton, C. Barrow, A.N. Wong, Real-world outcomes of immunotherapy for melanoma brain metastases in New Zealand, JCO Oncol. Pr. (2024) OP–24.