Exploring Music Style Transfer and Innovative Composition using Deep Learning Algorithms

Sujie He*
Modern Conservatory of Music University,
Shan Dong University of Art, Shandong, China

***Abstract*—Automatic music generation represents a challenging task within the field of artificial intelligence, aiming to harness machine learning techniques to compose music that is appreciable by humans. In this context, we introduce a text-based music data representation method that bridges the gap for the application of large text-generation models in music creation. Addressing the characteristics of music such as smaller note dimensionality and longer length, we employed a deep generative adversarial network model based on music measures (MT-CHSE-GAN). This model integrates paragraph text generation methods, improves the quality and efficiency of music melody generation through measure-wise processing and channel attention mechanisms. The MT-CHSE-GAN model provides a novel framework for music data processing and generation, offering an effective solution to the problem of long-sequence music generation. To comprehensively evaluate the quality of the generated music, we used accuracy, loss rate, and music theory knowledge as evaluation metrics and compared our model with other music generation models. Experimental results demonstrate our method's significant advantages in music generation quality. Despite progress in the field of automatic music generation, its application still faces challenges, particularly in terms of quantitative evaluation metrics and the breadth of model applications. Future research will continue to explore expanding the model's application scope, enriching evaluation methods, and further improving the quality and expressiveness of the generated music. This study not only advances the development of music generation technology but also provides valuable experience and insights for research in related fields.**

Keywords*—*Deep learning; style transfer; innovative composition; Generative Adversarial Networks

I. Introduction

Music, as one of the greatest inventions in human history, not only serves as a medium for cultural expression but also represents a cultural industry with tremendous potential for growth [1]. In recent years, with the rapid development of digital technology [2] [3] [4], the music industry is undergoing profound changes. As society's demand for music continues to expand, diversified music creation has become an inevitable trend. The demand for higher-quality music creation is evident in everything from the background music for short videos to the theme songs for movies and TV shows. However, traditional music composition methods, constrained by the need for specialized knowledge of music theory and instrumental skills, cannot meet the growing market demand. Against this backdrop, the use of computers to aid music composition and achieve automated music generation has emerged as a new research frontier.

Automatic music generation is a product of the intersection of information science and art studies, aimed at minimizing human intervention in computer-aided music composition [5]. It is not only a significant part of multimedia research but also a hot topic in artificial intelligence. Researchers are working on how to generate music that has both a clear style and conforms to audience aesthetics, delving into the deep connections behind music data. Therefore, an in-depth mining and analysis of music data features have significant theoretical and practical significance [6]. It can enrich the methods for generating music datasets and help build efficient music generation models, reducing the burden of manual composition and offering the possibility of music creation for non-professionals. By establishing objective music evaluation models, we can scientifically measure the quality and style consistency of automatically generated music.

Identifying the structure and style of music is core to the field of music generation. To track the development of elements such as melody, harmony, and rhythm, and to provide valuable information for music composition and automatic generation, advanced data processing techniques and algorithms are required. Although modern music generation technologies can create music in various styles, their application in automated composition and arrangement is still insufficient to capture the full complexity of music creation.

In the application of music generation, we need to optimize algorithms to generate music segments that are coherent and consistent in style. Considering the complexity of melody and harmony, the diversity of musical styles, and the uncertainty of melodic lines and harmonic progressions, we must enhance the algorithm's fitting ability to address these challenges. To this end, we attempt to introduce structures similar to squeeze-and-excitation models into music generation networks, forming a music generation model with feature extraction capabilities. Additionally, we incorporate attention mechanisms based on batch normalization, such as channel attention modules. Music generation models can draw inspiration from the Swin Transformer structure, introducing Swin Blocks to capture the long-range dependencies of music data and better extract deep music features.

II. Literature Review

This article will collect existing work in the field of automated music composition to highlight the shortcomings of current research.

A. Traditional Music Generation Methods

Over time, the field of music generation has experienced significant development, with early algorithms laying the groundwork for more complex systems. The earliest models of music generation operated on random principles within fixed parameters such as pitch, duration, and rhythm, often resulting in melodies lacking in musical coherence and artistic intent. The advent of sequence modeling algorithms marked a turning point in traditional automated music generation methods. These methods often rely on statistical probability methods such as Markov models, introducing a more structured composition technique that uses Markov chains and stochastic processes to predict future outcomes, greatly reducing the randomness problems in early music generation efforts [7].

David Cope's "Experiments in Musical Intelligence" (EMI) combined music language models by identifying repetitive structures in composers' works and reusing these patterns in new arrangements [8], thus generating music of a similar style. This method further demonstrated the potential of using Markov models and N-gram methods to create music in different styles [9] [10] [11]. Subsequently, Bretan and others [12] proposed a method based on the similarity ranking of musical fragments and the combination of new musical fragments to create new works from existing pieces. On the other hand, Pachet and others [13] introduced a method that uses chords to guide the selection of melodies. These techniques rely on the feature parameters of musical sequence data, using sequence models to achieve the desired musical output through signal reconstruction theory.

Despite progress, traditional probabilistic models such as Markov chains have a significant limitation: they can only generate subsequences that already exist in the training dataset. In areas where innovation and creativity are crucial, these algorithms inherently lack the ability to generate truly novel and creative content. Developing music generation systems that can not only replicate but also innovate and further push the boundaries of musical computational creativity remains a challenge.

B. Deep Learning Generation Methods

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

In recent years, the rapid development of deep learning technology has made breakthrough progress in multiple fields, especially in the processing of sequential data such as computer vision, speech recognition, and natural language processing, sparking intense interest in the application of deep learning in the field of music generation.

In the research of automated music generation, Mangal and others [14] used Long Short-Term Memory networks (LSTM) and Recurrent Temporal Restricted Boltzmann Machines (RTRBM) models, achieving certain results. Johnson [15] explored new paths for polyphonic music generation through the CharRNN model. Nayebi [16] used Recurrent Neural Networks (RNN) to generate music based on MIDI files.

In the exploration of generating more complex music sequences, Franklin [17] proposed using RNNs to represent the possibility of multiple notes sounding simultaneously. Additionally, Huang's team [18] proposed a new music generation framework based on Deep Belief Networks (DBN). Hadjeres and others [19] used an RNN model combined with Gibbs sampling techniques to successfully generate multi-part gospel music.

On a different path from RNN models, Sabathe and others [20] introduced Variational Autoencoders (VAEs) to generate music by learning the distribution of music fragments. Concurrently, researchers like Yang, Mogren, and others [21] used Generative Adversarial Networks (GANs) to compose music, a method that takes random noise as input to produce new melodic sequences.

Overall, the application of deep learning in music generation is in rapid development, with different deep learning models continually pushing the limits of music generation technology. Despite many challenges, deep learning models have already shown great potential in imitation, innovation, and exploring the complex structure of music. With the deepening of research and the maturity of technology, future music generation systems are expected to become more intelligent, creating a richer and more diverse range of musical works.

C. Research Gaps

Although the field of music generation has made a series of advancements, there are still important gaps in existing research. There are several key musical elements that have not been fully considered in the generation process, such as the duration of notes, the handling of rests, the diversity of musical styles, and the musical formats of input models. Based on this, future research needs to address the following issues:

1) Limitations of Probabilistic Models in Music Generation: The traditional probabilistic models currently in use have some feasibility in music generation, but due to the diversity and evolution of music, these models may not be able to adapt to new musical trends in a timely manner. Moreover, building effective probabilistic models requires a deep foundation in musical theory. Traditional methods also rely on a lot of manual feature design and extraction, leading to low efficiency and a large workload.

2) Lack of Uniformity in Music Data Preprocessing: Currently, there is no unified standard for music data preprocessing methods, resulting in different studies adopting their own methods, such as music generation methods based on variational autoencoders and melody algorithms based on digital signal processing. These methods often neglect the rhythmic nature of music, such as the length of notes and pauses. Even in generation models that consider both melody and rhythm, there are problems with the compatibility of pitch and rhythm during training. This lack of standardized representation hinders the universality and compatibility between different music generation methods.

3) Limitations of Deep Learning Models in Music Generation: While deep learning models such as LSTM have shown potential in music generation, they usually cannot generate long-term melodic sequences. Existing large text generation models, such as BERT and GPT-2, perform excellently in text generation but face data representation issues when directly applied to music generation. Due to the fundamental differences between music signal representation and text, existing language generation models cannot be directly applied to music generation.

In summary, future research needs to develop new models and techniques to address these challenges in music generation, to truly enhance the novelty of musical composition and the acceptance of the audience.

III. Automatic Music Generation Methods based on Generative Adversarial Networks

Music generation has been achieved using the CHSE-GAN model based on the segmentation of music text into measures. The current state of research is first elucidated, followed by an introduction to GAN networks, and then music segments are generated using the segmentation of music text into measures. Finally, this method's potential in music generation is described by comparing it with other generation models in terms of loss rate, accuracy, and other indicators.

A. Model Introduction

In the contemporary field of music composition, deep learning technologies such as Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs) have begun to be explored for constructing musical works. With their advanced data processing capabilities, they exhibit notable creative potential.

Generative Adversarial Networks (GANs) are a unique unsupervised learning framework, designed around the concept of two neural networks contesting with each other to promote the learning process. The generator network (G) is responsible for transforming a random noise vector z into the data space, simulating samples from the real data distribution. Meanwhile, the discriminator network (D) has the task of outputting a scalar value, predicting whether a given sample is from the real data distribution or produced by the generator G.

These two networks compete with each other during training, adjusting their parameters to enhance their own performance: the generator G tries to produce more realistic data, while the discriminator D strives to more accurately distinguish between real and generated data. This adversarial training process can be viewed as a minimax game where both the generator and discriminator have their own objective functions, which are in opposition to each other. They evolve together until a dynamic equilibrium is reached.

$$\min_G \max_D V(D,G) = \mathrm{E}_{\mathrm{x}\sim p_{\delta\alpha\tau\alpha}(x)}[\log D(x)] + \mathrm{E}_{\mathrm{z}\sim p_z}[1 - \log(D(G(\mathrm{z}))] \quad (1)$$

In this, P_{data} represents the distribution of real data x, and P_z denotes the prior distribution of z. Nevertheless, the application of GANs in music composition is still at an exploratory stage, and this method still shows limitations in capturing the complex interactions of musical elements in time and space. Compared to the short sequences generated in text, music composition deals with much longer time series, which makes it difficult for the network to grasp the profound connections between sequences during learning.

In view of this, this study adopted the CHSE-GAN model, which is designed under the influence of GAN concepts, for music composition. The model combines a discriminator and a generator, and specifically, allows the discriminator to pass feature information to the generator, supporting the generator to learn and understand data of longer time series, which to some extent eases the challenge of dealing with complex musical structures. This paper will first provide an overview of the foundations of Generative Adversarial Networks, followed by an in-depth discussion of the structure and functions of the CHSE-GAN model.

B. CHSE-GAN Music Generation Method

In this section, we introduce the CHSE-GAN (Channel Attention and Squeeze-Excitation based Generative Adversarial Networks), which is specially designed for music generation. Its network structure has been adjusted to suit the characteristics of music data, as shown in Fig. 1. Based on CycleGAN, CHSE-GAN has made the following improvements to enhance its application in the field of music composition:

Introduced a channel attention mechanism based on batch normalization, NAM (ch). Traditional channel attention methods calculate weights through complex network structures, which may not be sufficient to capture the complex patterns in music. By extracting the scaling factors from batch normalization as channel attention weights, we can effectively distribute weights to features within the network without increasing network complexity and extra parameters, thereby strengthening the focus on important musical features.

Music features often have rich hierarchical levels and subtle dynamic changes, thus a single feature extraction structure may not capture them adequately. CHSE-GAN introduces a Squeeze-Excitation (SE) attention mechanism into the residual network to form a new Res-SE module. Combined with the channel attention mechanism based on batch normalization, it creates a new backbone network for feature extraction, enhancing the generator's perception of complex musical structures and details, and improving the capture of musical features.

As shown in Fig. 2, the generator network structure of CHSE-GAN consists of three main parts: downsampling, the backbone network, and upsampling. Specifics are as follows:

Downsampling part: Three downsampling operations using convolutional layers with a stride of 2 are performed to expand the receptive field and reduce dimensions. The first layer uses a 64-dimensional 7×7 convolution kernel to capture a broader range of musical structure information.

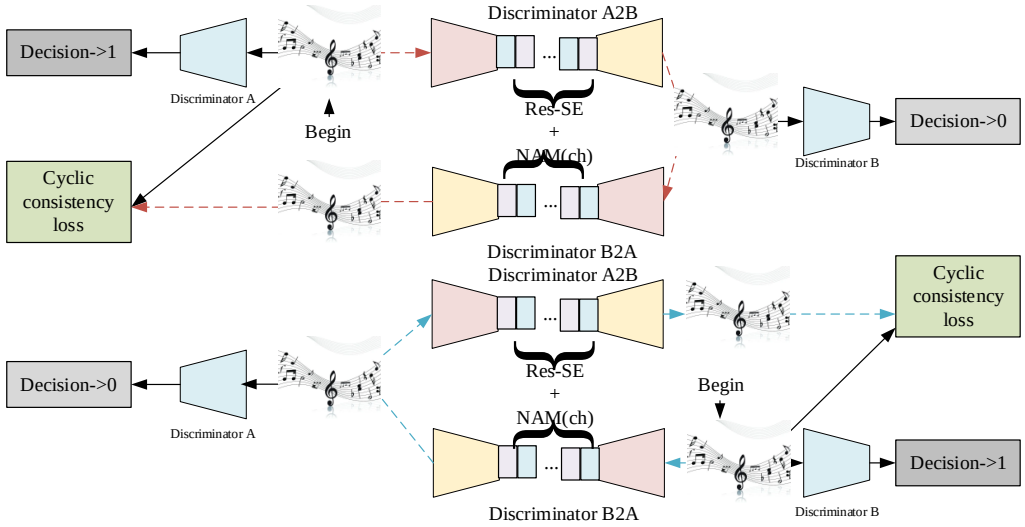


Fig. 1. Network structure diagram based on CHSE-GAN music generation.

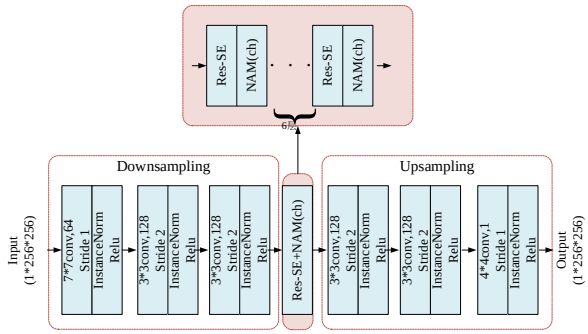


Fig. 2. Schematic diagram of the network structure of the CHSE-GAN algorithm generator.

Backbone network: Consists of NAM(ch) and the new Res-SE modules, which, through a combination of Squeeze-Excitation attention and channel attention based on batch normalization, enhance the extraction and expression capabilities for musical features.

Upsampling network: After extracting deep features, the SE module adjusts the output of the convolution from the backbone network, then NAM(ch) redistributes channel weights after each residual block. Upsampling is constructed using deconvolutional layers, with batch normalization and ReLU activation functions applied after each layer to restore the data to the original spatial dimensions of the musical signal.

With this carefully designed network structure, CHSE-GAN can generate music works that are rich in expressiveness and dynamism, providing a powerful tool for automated music composition and style transformation.

1) Batch Normalization-based Channel-wise Attention: In deep learning models for music generation, it is crucial to effectively utilize the time-frequency features in music signals. To enhance the ability to extract these deep features, we can adopt an attention mechanism based on batch normalization. This mechanism can reinforce the model's focus on important parts of music features, thereby improving the quality of music generation. Batch Normalization is commonly used in deep learning to speed up the training process and improve model performance. In music generation models, we can use the statistical parameters obtained during the batch normalization process to calculate channel attention. Specifically, the parameters of batch normalization are used not only for feature normalization but can also serve as weight information to adjust feature mappings on different channels. This method is known as the Normalization-based Attention Module (NAM).

For instance, we can design an NAM(ch) module to adaptively readjust the feature weights on each channel without adding extra network parameters. The NAM(ch) module can be placed after each part of the residual network structure to enhance the fine expression of musical spectral features. The computational flowchart of NAM(ch) is shown in Fig. 3, where ω_i represents the weights, and γ_i represents the scaling factors for each channel. The pseudo-code for the batch normalization-based channel attention algorithm is as follows.

Algorithm 1: Channel Attention Algorithm Based on Batch Normalization

Initialize

Step 1: $\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i$

Step 2: $\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2$

Step 3: $\hat{x}_\iota \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}}$

Step 4: $y_i \leftarrow \gamma \hat{x}_i + \beta = \mathrm{BN}_{\gamma,\beta}(x_i)$

Step 5: $\omega_{\mathrm{i}} = \frac{\gamma_i}{\sum_{j=0}\gamma_j}$

Step 6: $M_S = \mathrm{Sigmoid}(\omega_i \times y_i)$

$$\mathrm{B}_{\mathrm{o}\upsilon\tau} = \mathrm{BN}(\mathrm{B}_{\mathrm{i}\nu}) = \gamma \frac{\mathrm{B}_{\mathrm{i}\nu} - \mu_{\mathrm{B}}}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} + \beta \; \omega_{\mathrm{i}} = \frac{\gamma_i}{\sum_{j=0}\gamma_{\mathrm{j}}} \quad (2)$$

2) Res-SE Module Based on Residual Blocks and Squeeze-and-Excitation Attention: In music generation models, the Res-SE module, which combines residual blocks and squeeze-and-excitation attention mechanisms, has been proven to significantly enhance the representational capacity of music features. The design of this structure is inspired by successful

experiences in the field of image processing, but it has been optimized for the characteristics of music data.

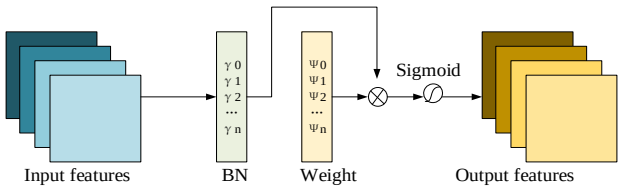


Fig. 3. Schematic diagram of batch normalization channel attention calculation structure in CHSE-GAN.

As shown in Fig. 4, taking music signal processing as an example, suppose we have an input with a time-frequency representation of size 256×64×64, which represents 256 channels, each with 64 time steps and 64 frequency components. The input first goes through two layers of convolution, batch normalization, and ReLU activation function, and then it is divided into two branches:

i. The first branch is passed through directly without *processing* to preserve the original features of the music signal.

ii. The second branch is enhanced through an SE (Squeeze-*and*-Excitation) module. This module first employs a global average pooling operation to "squeeze" each channel, reducing the 64×64 feature map of each channel to a single scalar to capture the global contextual information.

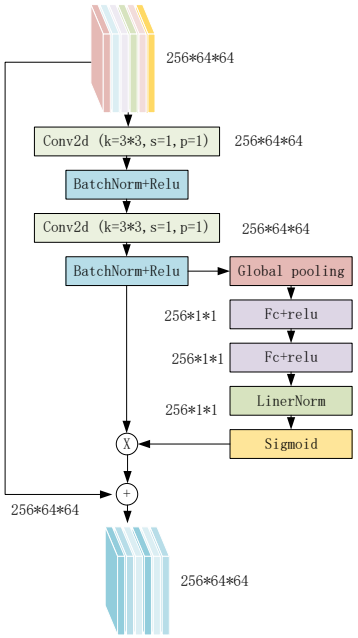


Fig. 4. Schematic diagram of Res-SE network structure in CHSE-GAN.

Next, this global information is processed through two layers of fully connected layers, which include ReLU activation functions, to capture the inter-channel dependencies and generate a weight for each channel. These weights are further transformed through a Sigmoid activation function to obtain the final weights for each channel, which determine which channels are important.

Finally, we multiply these weights by the original input to recalibrate the features. This process allows the model to emphasize those channel features that are important for music generation while suppressing the less important parts.

IV. Automatic Music Generation Methods Based on Generative Adversarial Networks

This section will validate the effectiveness of the proposed method based on a self-made experimental dataset.

A. Experimental Environment

The hardware environment for the experiments in this chapter is shown in Table I:

TABLE I. Experimental Software and Hardware Environment

CPU	Intel i7 8700k
GPU	GTX 3080
Memory	32G
Operating System	Ubuntu 18.04
CUDA	11.1
Main Frameworks	Pytorch, Keras
Main Programming Language	Python 3.6

To explore the automatic generation of pop music, this study has selected the widely popular POP909 music dataset as the training resource. The characteristic of the POP909 dataset is the clear division of its melody tracks, making the melodies easy to extract and process separately. The preprocessing of music data adopted the method introduced in Chapter 3, converting MIDI files into text format, with the help of the music21 toolkit in the Python platform.

The experimental part designed four different studies to comprehensively evaluate music generation performance. These four areas are: comparison of music generation effects at different tempos, performance comparison of various music generation models, comparison of the quality of generation with other algorithms, and evaluation using the music evaluation model introduced in Chapter 6. By analyzing the results of different tempos, the comprehensive performance of the MT-CHSE-GAN network can be assessed, and its applicability for generating different types of music can be determined. Compared with algorithms such as Rank-GAN and Seq-GAN, this study aims to verify the advantages of MT-CHSE-GAN in the field of music composition. Finally, based on the evaluation model in Chapter 6, the music generated by MT-CHSE-GAN is compared and analyzed with real music samples, melodies produced by LSTM networks, and MT-GPT-2 networks, to examine from a more objective perspective whether the MT-CHSE-GAN network can meet the standards of real music melody generation.

B. Experiment

After a series of in-depth training, the innovative CHSE-GAN model we used, which is based on bar segmentation, successfully created numerous musical works. By careful listening tests, we noticed that most of these works exhibit smooth and pleasant characteristics. To give everyone an intuitive feeling, we randomly selected some sample fragments to showcase this achievement. Before that, it should be noted that since this work adopted a way of expressing music as text,

all generated music works initially exist in text form. To convert these text data into audible music, we used the music21 toolkit under the Python environment to achieve the transformation from text to MIDI format. Subsequently, we used MusicScore 3 software to open it in the form of a score for a more detailed display, with the specific effects shown in Fig. 5.

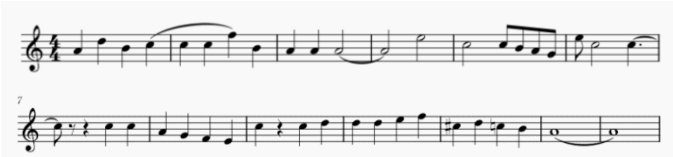


Fig. 5. Generated fragment display.

During the model training, the accuracy of the model was recorded, as shown in Fig. 6 below.

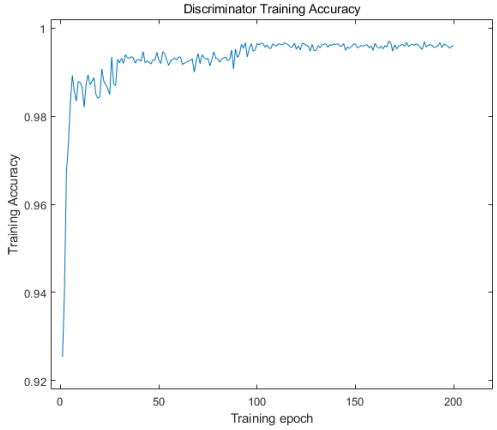


Fig. 6. Model accuracy curve.

The changes in loss rate are shown in Fig. 7 below.

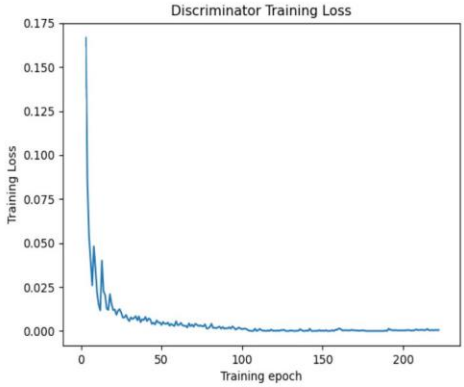


Fig. 7. Model loss rate curve.

As can be seen from the figures above, the MT-CHSE-GAN model, which used adversarial pre-training, reached convergence quickly with both the accuracy rate and loss values stabilizing around 100 epochs.

1) Comparison of Music Generation at Different Tempos

By training the model with pop music at different tempos, the following generated music was obtained, as shown in Fig. 8.

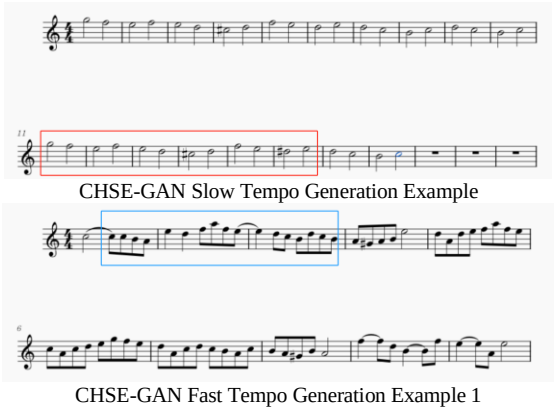


Fig. 8. MT- CHSE-GAN generates fast and slow music comparisons.

In the examples shown in Fig. 8, we can see that the two slow tempo music segments on the left were generated by the MT-CHSE-GAN after training, while the two fast tempo segments on the right also used the MT-CHSE-GAN model. Although the slow tempo music has a pleasant melody, there is a minor issue: some notes are repeated too often, as indicated by the red box in the figure (a note's duration exceeds one measure). In contrast, the fast tempo segments avoid this issue; their melodies progress in a stepwise fashion, with a clear rhythm, as shown by the notes marked with a blue box. From this comparison, we can observe that in the generation of slow tempo music, the model still has room for improvement in handling long-duration notes, while in the generation of fast tempo music, the model performs relatively well.

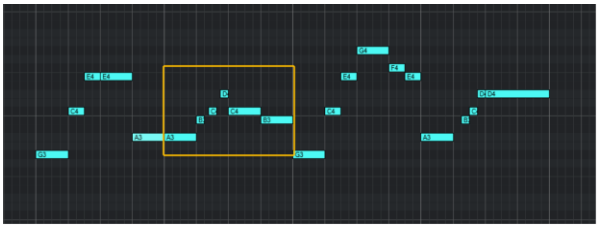
2) Comparison of Music Generation for Different Models

Next, we compared the music generated by different models (see Fig. 9). From top to bottom, the figure sequentially shows music segments generated by the MT-CHSE-GAN, Rank-GAN, and Seq-GAN networks. As indicated by the pink arrows in the figure, the melody segments generated by the MT-CHSE-GAN and Seq-GAN networks show clear high and low fluctuations, with most of the melodic changes revolving around the theme pitch and returning to the theme pitch at the end, which is consistent with the melodic development pattern of pop music.

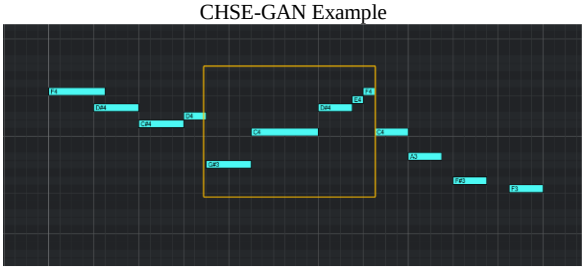
On the other hand, the music segment generated by the Rank-GAN network shows an overall descending trend, with the melody starting high and gradually descending. This type of melodic structure is often inconsistent with the typical composition of pop music and may give a sense of oppression and unease. This indicates that, in handling long sequence melodies, the MT-CHSE-GAN and Seq-GAN networks have better generative effects compared to the Rank-GAN network.

After an in-depth analysis of three different music melody generation models, we specifically observed the characteristics of note changes, especially the melody parts marked with yellow boxes in the figures. The melody fragments produced by the MT-CHSE-GAN network exhibit gentle note changes, with melodies that are smooth and orderly, rhythmically fluctuating around the tonic. In contrast, the melodies produced by the Rank-GAN and Seq-GAN networks (also marked with yellow boxes) show more dramatic note changes, sometimes with jumps between notes approaching an octave. Such abrupt

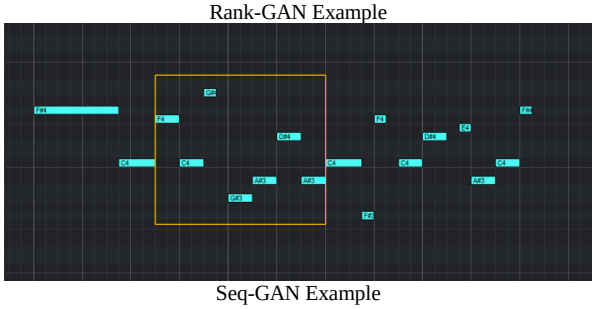
melodic changes may seem jarring and not quite in line with the conventions of pop music composition.



CHSE-GAN Example



Rank-GAN Example



Seq-GAN Example

Fig. 9. Generate samples for each model.

In evaluating the generated music, we scored several key aspects based on music theory, including the harmony of the melody, the logical coherence of the melody, the contour of the melody, and the tonality of the melody. We had the MT-CHSE-GAN, Rank-GAN, and Seq-GAN networks each generate 10 pieces of music, scored them, and calculated the average score for each item (out of a possible 10 points). The scoring results are summarized in the Table II shown below.

TABLE II. MELODY THEORY SCORE

	Melody Harmony	Melody correlation	Melody contour	Melodic tonality	Average score
MT-CHSE-GAN	8.4	7.4	7.6	8.2	7.9
Rank-GAN	6.9	5.8	6.4	7.2	6.575
Seq-GAN	7.3	6.1	7.2	7.1	6.925

Based on the melody evaluation metrics introduced in the previous chapters and through comparative analysis, we found that the music melodies generated by the MT-CHSE-GAN model are of significantly higher quality than those generated by the Rank-GAN and Seq-GAN models when trained with the same music text data. The melodies generated by the MT-CHSE-GAN network are closer to the style and texture of real music.

3) Comparative Analysis of Computational Complexity

In the experiments, an AMD Ryzen 7 4800H processor and RTX2060 graphics card were used for training and music generation of the various models. A comparison of the running time was made among the CHSE-GAN, Rank-GAN, and Seq-GAN models. In each model, music samples of 1024 characters in length were generated. The specific running time comparison results are presented in the table below in Table III. This comparison helps to assess the efficiency and resource consumption of different models when generating melodies.

TABLE III. COMPARISON OF RUNNING COMPLEXITY

Model	Time
CHSE-GAN	4.9s
Rank-GAN	6.5s
Seq-GAN	4.8s

As the data in the table shows, the time taken to generate 1024 characters varies slightly among the three models: the CHSE-GAN model requires approximately 4.9 seconds, the Rank-GAN model takes 6.5 seconds, and the Seq-GAN model needs 4.8 seconds. Although the generation times are similar, the CHSE-GAN has an advantage in terms of the quality of generation compared to the other two models.

4) Comparison of the Fit of Music Generated by Different Models

Based on a unified data representation, we trained the Rank-GAN and Seq-GAN networks and subsequently evaluated the models' performance. Maximum likelihood estimation (MLE) aims to minimize the cross-entropy between the true data distribution pp and the data distribution qq generated by the model. By quantifying MLE, we are able to assess the fit between the data and the model. This reflects not only the specific details of the data but also considers the details of the model.

Negative log-likelihood (NLL) was originally proposed in Seq-GAN research as an improved metric based on MLE, specifically to measure the degree of match between generated data and real data. Fig. 10 shows the training loss changes for NLL-test.

The NLL-test training loss curves from Fig. 10 indicate that the CHSE-GAN model converges more quickly and demonstrates better performance on this metric. Throughout the testing phase, CHSE-GAN consistently showed the best NLL performance, while Rank-GAN performed the worst. The NLL loss curves for Seq-GAN and Rank-GAN almost coincide before the solid line, but after the solid line, the performance of Rank-GAN declines compared to other stages. These results suggest that the music generated by the MT-CHSE-GAN network performs better in fitting real music.

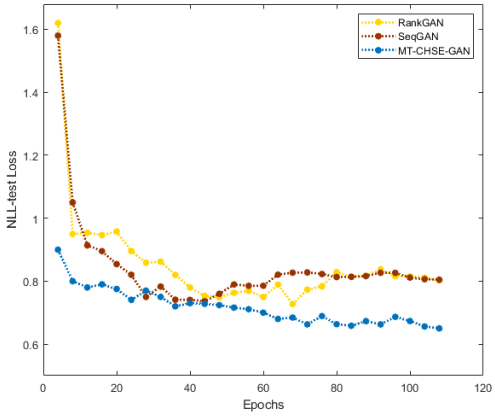


Fig. 10. NLL-test loss.

V. Conclusion

This research is dedicated to exploring the important branch of artificial intelligence that is automatic music generation, with a particular focus on the application of deep learning technologies in this field and their practical value. In order to improve the stability of music generation models, a CHSE-GAN based model was developed, effectively addressing the issue of length in music melody generation. The model integrates music theory and mathematical statistics, and through the textualization and bar-wise processing of music data, as well as the introduction of the SE module and the channel attention module based on batch normalization, it enhances feature extraction capabilities without the need to add extra network parameters. Experiments show that CHSE-GAN can generate music of higher quality compared to traditional algorithms. Although research in music generation has made certain advances, its range of application is still relatively limited, and it lacks quantitative evaluation metrics. In particular, evaluation models that combine mathematical statistics with music theory knowledge still have great potential for development. Future work will continue to focus on expanding model applications, enriching evaluation methods, and improving the quality of generated music.

Acknowledgment

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g.” Avoid the stilted expression, “One of us (R. B. G.) thanks . . .” Instead, try “R. B. G. thanks.”

References

[1] Lam, M. W. Y., Tian, Q., Li, T., et al. (2024). Efficient neural music generation. Advances in Neural Information Processing Systems, 36.

[2] Liu, S., Zheng, P., & Bao, J. (2023). Digital Twin-based manufacturing system: a survey based on a novel reference model. Journal of Intelligent Manufacturing, 1-30.

[3] Liu, S., Zheng, P., Xia, L., et al. (2023). A dynamic updating method of digital twin knowledge model based on fused memorizing-forgetting model. Advanced Engineering Informatics, 57, 102115.

[4] Zheng, H., Liu, S., Zhang, H., et al. (2024). Visual-triggered contextual guidance for lithium battery disassembly: a multi-modal event knowledge graph approach. Journal of Engineering Design, 1-26.

[5] Copet, J., Kreuk, F., Gat, I., et al. (2024). Simple and controllable music generation. Advances in Neural Information Processing Systems, 36.

[6] Li, S., Dong, W., Zhang, Y., et al. (2024). Dance-to-music generation with encoder-based textual inversion of diffusion models. arXiv preprint arXiv:2401.17800.

[7] Fine, S., Singer, Y., & Tishby, N. (1998). The hierarchical hidden Markov model: Analysis and applications. Machine learning, 32, 41-62.

[8] Cope, D. (1989). Experiments in musical intelligence (EMI): Non-linear linguistic-based composition. Journal of New Music Research, 18(1-2), 117-139.

[9] Chordia, P., Sastry, A., & Senturk, S. (2011). Predictive table modelling using variable-length markov and hidden markov models. Journal of New Music Research, 40(2), 105-118.

[10] Van der Merwe, A., et al. (2011). Music Generation with Markov Models. IEEE multimedia, 18(3), 78-85.

[11] Pachet, F., & Roy, P. (2011). Markov constraints: steerable generation of Markov sequences. Constraints, 16(2), 148-172.

[12] Bretan, M., Weinberg, G., & Heck, L. (2016). A Unit Selection Methodology for Music Generation Using Deep Neural Networks. CoRR, abs/1612.03789.

[13] Pachet, F., Paris, C., Papadopoulos, A., et al. (2017). Sampling variations of sequences for structured music generation. In Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR'2017), Suzhou, China (pp. 167-173).

[14] Mangal, S., Modak, R., & Joshi, P. (2019). Lstm based music generation system. arXiv preprint arXiv:1908.01080.

[15] Johnson, D. D. (2017). Generating polyphonic music using tied parallel networks. In International conference on evolutionary and biologically inspired music and art (pp. 128-143). Cham: Springer International Publishing.

[16] Nayebi, A., & Vitelli, M. (2015). Gruv: Algorithmic music generation using recurrent neural networks. Course CS224D: Deep Learning for Natural Language Processing (Stanford), 52.

[17] Franklin, J. A. (2006). Recurrent Neural Networks for Music Computation. Informs Journal on Computing, 18(3), 321-338.

[18] Huang, Q., Huang, Z., Yuan, Y., et al. (2015). A New Method Based on Deep Belief Networks for Learning Features from Symbolic Music. In 2015 11th International Conference on Semantics, Knowledge and Grids (SKG) (pp. 231-234). IEEE.

[19] Hadjeres, G., & Pachet, F. (2017). DeepBach: a Steerable Model for Bach Chorales Generation. JMLR.org, 34(70), 1362-1371.

[20] Sabathe, R., Coutinho, E., & Schuller, B. (2017). 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) (pp. 3467-3474). IEEE.

[21] Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al. (2014). Generative adversarial nets. Advances in neural information processing systems, 2672-2680.References