Neural Computing and Applications (2024) 36:6381–6401
https://doi.org/10.1007/s00521-024-09418-2

REVIEW

A review of intelligent music generation systems

Lei Wang[1] · Ziyi Zhao[1] · Hanwei Liu[1] · Junwei Pang[1] · Yi Qin[2] · Qidi Wu[1]

Received: 14 June 2023 / Accepted: 15 January 2024 / Published online: 19 February 2024

Abstract
With the introduction of ChatGPT, the public's perception of AI-generated content has begun to reshape. Artificial intelligence has significantly reduced the barrier to entry for non-professionals in creative endeavors, enhancing the efficiency of content creation. Recent advancements have seen significant improvements in the quality of symbolic music generation, which is enabled by the use of modern generative algorithms to extract patterns implicit in a piece of music based on rule constraints or a musical corpus. Nevertheless, existing literature reviews tend to present a conventional and conservative perspective on future development trajectories, with a notable absence of thorough benchmarking of generative models. This paper provides a survey and analysis of recent intelligent music generation techniques, outlining their respective characteristics and discussing existing methods for evaluation. Additionally, the paper compares the different characteristics of music generation techniques in the East and West as well as analysing the field's development prospects.

Keywords AI-generated content · Symbolic music generation · Automatic composition · Music representations

1 Introduction

As an area exploring automated creation, computer creativity has driven high-quality research in the direction of intelligent innovation, including storytelling [2] and art [70]. With the development of artificial intelligence, music has proven to have a very high capacity and potential for automatic creation. Intelligent music generation is dedicated to solving the problem of automatic music composition and is currently one of the most productive sub-fields in the field of computational creativity.

Symbolic music generation refers to representing music as a sequence of symbols, followed by feature learning and generative modeling [103]. Two essential properties of symbolic music composition are structural awareness and interpretive ability. Structure-aware models can produce naturally coherent music with long-term dependencies, including different levels of repetition and variation [43]; interpretive ability translates complex computational models into controllable interfaces for interactive performance [45]. Finally, this field can serve a variety of purposes, such as composers using the technology to develop existing material, to inspire and explore new ways of making and working with music, and listeners can customise or personalise music.

The artificial intelligence composition system is usually called the Music Generation System (MGS). There are basically 5 steps in music production: 1) composition; 2) arrangement; 3) sound design; 4) mixing and 5) mastering. To our knowledge, contemporary AI music generation encompasses a broad spectrum of music production stages, ranging from sparking creative inspiration and assisting with song structure to fully autonomous composition. These AI systems provide valuable tools and resources for music creators, enabling them to explore, innovate, and

✉ Ziyi Zhao
zhaozi1@tongji.edu.cn

Lei Wang
wanglei@tongji.edu.cn

Hanwei Liu
liuhw1@tongji.edu.cn

Junwei Pang
pangjw@tongji.edu.cn

Yi Qin
qinyi@shcmusic.edu.cn

[1] College of Electric and Information Engineering, Tongji University, Cao'an Street, Shanghai 201804, China

[2] Department of Music Engineering, Shanghai Conservatory of Music, Fenyang street, Shanghai 200031, China

interact with technology in various aspects of the music creation process. To conclude, our study mainly focuses on composition and arrangement, covering ideas to a sequence of symbols. The music clips created by generative algorithms encompass a variety of different instruments, contingent upon the complexity of the generation system and the specific objectives of music generation [119]. Whether monophonic or polyphonic, these segments include nearly all common instruments. Music generation involving instruments such as bass, drums, guitar, piano, violin, string instruments, wind instruments, electronic synthesizers, among others, has been extensively researched and applied [69, 120, 121]. Concurrently, the methods and tools for music generation are continuously evolving, enabling the computer-generated production of an increasing array of instruments and music genres. However, it is important to note that the generation of music involving various ethnic instruments still faces limitations and requires ongoing refinement and development.

Currently, a limited number of review papers on music generation systems are available. Herremans et al. [31] introduced the classification method of music generation systems and pointed out problems in music evaluation. Briot et al. [4] introduced algorithms based on deep learning based music generation system, and put forward corresponding countermeasures for some limitations in this field. Kaliakatsos et al. [46] analyzed and predicted the research fields that promote the development of intelligent music generation in the field of artificial intelligence; Loughran et al. [55] aimed the music generation system based on evolutionary calculation has analyzed the categories and potential problems; Plug et al. [70] introduced the latest research technology of music generation systems in the game field, and analyzed the prospects and challenges of the field. However, in these previous papers, some do not provide sufficient analysis of algorithms and models [31, 46, 55]. In Briot's work [4], there exists content chapters with weak thematic relevance, with the basic theory of neural networks serving as an example. These chapters have not substantially contributed to an in-depth exposition of the theme, leading to a dispersion of the work's content. Others have a more traditional and conservative outlook on the future [31, 46], and [70] is more focused on the application level, giving less information on the general method of music generation.

In light of the limitations of previous review work, this paper comprehensively analyses the most recent articles on the various sub-research directions of music generation, points out their respective strengths and weaknesses, and summarises them in a clear categorisation according to the generation method. In order to better understand the digital character of the music, we also present the two main representations of music as an information stream and list the commonly used datasets For the evaluation of generative effects, the most representative evaluation methods based on both subjective and objective aspects are selected for discussion. Our work concludes with a comparative analysis of the similarities and differences between Eastern and Western research in the field of AI composition based on different cultural characteristics and a look at the further development of the field of music generation as it matures.

2 Overview

The most basic elements of music include the pitch of the sound, the length of the sound, the strength of the sound and the timbre. These basic elements combine with each other to form the melody, harmony, rhythm and tone of the music. In the field of artificial intelligence composition, melody constitutes the primary purpose of an automatic music generation system. In most melody generation algorithms, the goal is to create a melody similar to the selected style, such as generating western folk music [62] or free jazz [48]. In addition to melody, harmony is another popular aspect of automatic music generation. When generating a harmony sequence, the quality of its output mainly depends on the similarity with the target style. For example, in choral harmony, this similarity is clearly defined through adherence to sound guiding principles [27], whereas in popular music, chord progressions primarily serve as accompaniments to the melody [90]. In addition, audio generation and style conversion are also popular directions in the field of intelligent composition. The overall framework of the specific music generation system is shown in Fig. 1. It is worth noting that many historical music generation systems do not include deep neural networks or evolutionary algorithms, but generate outputs based solely on musical rule constraints.

In the process of generating music, the musical content needs to be encoded in digital form as input to the algorithm, after which the intelligent algorithm is learnt and trained to eventually output different types of musical segments, such as melodies, chords, audio, style transitions, etc... Music has a clear hierarchical structure in time, from motives to phrases and then to segments. As shown in Fig. 2, where the overall structure of the music together with local repetitions and variants form the theme, rhythm and emotion of the piece; secondly, music usually consists of multiple voices (may be the same or different instruments), each individual voice part has a complete structure, and together they must be in harmony with each other. Combining these limitations, music generation is still a very challenging problem.

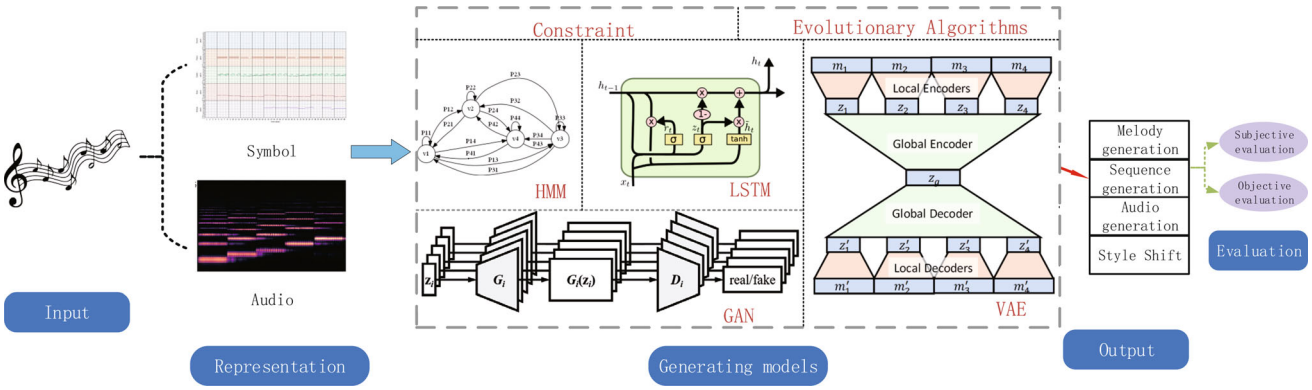


Fig. 1 Overview of music generation system

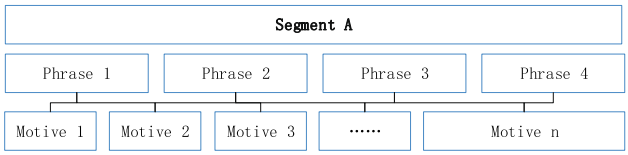


Fig. 2 The hierarchy of music

3 Representation

Music can be seen as a stream of information that conveys emotions at different levels of abstraction [102]. The input to a generative system is usually some representation of music, which has an important influence on the number of input and output nodes (variables) to the system, the correctness of training, and the quality of the generated content [4].

3.1 Specific formats

The representation of music is mainly divided into two categories: audio and symbolic. The main difference between the two is that audio is a continuous signal, while the symbol is a discrete signal.

3.1.1 Audio representation

The main ways in which music can be represented in audio are:

a. Waveform [67]
b. Spectrogram [57]

Among them, the spectrogram [57] can be obtained by the short-time Fourier transform of the original audio signal. The advantage of using audio signal representation is that the original audio waveform retains the original properties of music and can be used to produce expressive music [61]. However, this form of representation has certain limitations, because music fragments in the original audio domain are usually represented as continuous waveform $x \in [-1, 1]^T$, where the number of samples T is the audio duration and the sampling frequency (usually 16 kHz To 48 kHz), short-term music fragments will generate a lot of data. Therefore, modelling the raw audio can lead to a somewhat range-dependent system, making it challenging for intelligent algorithms to learn the high-level semantics of music, such as the Jukebox [14] automated record jukebox that takes approximately nine hours to render one minute of music.

3.1.2 Symbolic representation

The symbolic representation of music fragments is mainly as follows:

a. MIDI events

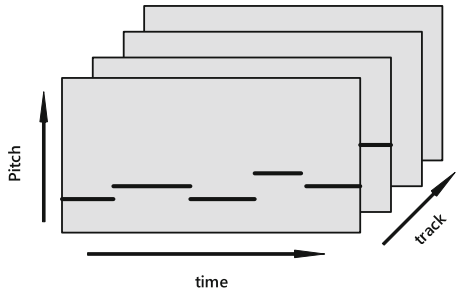
MIDI (Musical Instrument Digital Interface) is a digital interface of musical instruments to ensure the interoperability between various electronic musical instruments, software and equipment. Figure 3a shows the type of MIDI event. This method uses two symbols to indicate the appearance of notes: Note on and Note off, indicating respectively the beginning and end of the played notes. In addition, the note number is used to indicate the pitch of the note, which is specified by an integer between 0 and 127. This data structure contains an incremental time value to specify the relative interval time between notes. For example, Music transformer [37], MusicVAE [73], etc. convert melody, drums, piano, etc. into MIDI events and use them as input to the training network to generate music fragments. However, MIDI events cannot effectively retain the concept of multiple tracks playing multiple notes at once [34].

b.Piano-roll

The conversion of a music clip to Piano-roll format begins with a one-hot encoding of the MIDI file before converting it to image form. As shown in Fig. 3b, the

```
Track 0:
MetaMessage('set_tempo', tempo=500000, time=0)
MetaMessage('time_signature', numerator=4, denominator=4,
clocks_per_click=24, notated_32nd_notes_per_beat=8, time=0)
MetaMessage('end_of_track', time=548174)
Track 1: A.PIANO 1
MetaMessage('track_name', name='A.PIANO 1', time=0)
program_change channel=0 program=0 time=1536
control_change channel=0 control=7 value=127 time=0
note_on channel=0 note=72 velocity=57 time=0
note_on channel=0 note=48 velocity=45 time=2
note_off channel=0 note=48 velocity=0 time=86
note_off channel=0 note=72 velocity=0 time=32
```

(a) MIDI events(Extract from Twinkle Twinkle Little Star).



(b) Piano-roll.

```
X: 1
T: Scotland The Brave
Z: GoneAway
R: march
M: 4/4
L: 1/8
K: Dmaj
A,D|D2 D3/2E/2|FD FA|d2 d3/2c/2|
```

(c) ABC notation(Extract from Scotland The Blave).

Fig. 3 Representation of music

x-axis represents the continuous time step, the y-axis represents the pitch, and the z-axis represents the MIDI track, where n tracks have n one-hot matrices.

The Piano-roll format quantizes the notes into a matrix form according to the beats. This representation form is popular in music generation systems because it is easy to observe the structure of the melody and does not require serialization of polyphonic music. The Piano-roll representation is one of the most prevalent methods for representing melodies. For instance, systems like Midi-VAE [5], MuseGAN [17], typically start by converting polyphonic music into the Piano-roll format before proceeding with further processing. However the limitation of using Piano-roll is that it cannot capture the vocal and other sound details such as timbre, velocity, and expressiveness [4], and it is difficult to distinguish long notes from repeated short notes (e.g. whole notes and multiple eighth notes) and has multiple zero integers resulting in low training efficiency.

c.ABC notation

ABC notation[1] encodes melodies into a textual representation, as shown in Fig. 3c, initially designed specifically for folk and traditional music originated in Western Europe, but later widely used for other types of music. A typical example of this encoding method, which has the advantage of being concise and easy to read, is Folk-rnn [76], which takes the ABC notation of a particular folk music transcription and uses it as input to an LSTM network to generate tunes in the folk music genre. Similarly, GGA-MG [22] uses ABC symbols for representation used to generate melodies. However, the application of this representation is limited by the fact that it encodes mainly monophonic melodies.[103] Lacking specific notation for polyphony or harmony, it does not excel at encoding complex polyphonic music.

3.2 Datasets

For deep neural network-based generative systems, beyond determining the appropriate musical representation, the selection of a suitable music dataset as the system's input is equally paramount. Popular public datasets contain music types such as multi-track, choral, piano and folk music, and representations such as audio, MIDI events and ABC symbols. Table 1 lists the types and encoding forms of several typical publicly available music datasets for statistical purposes.

Nsynth contains 305,979 notes, each with a unique pitch, timbre and envelope. For the 1,006 instruments from the commercial sample library, the four-second, single-single instrument is generated in the range of each pitch (21-108) and five different speeds (25, 50, 75, 100, 127) of a standard MIDI piano.

Lakh MIDI is a collection of 176,581 unique MIDI files, 45,129 of which have been matched and aligned with entries in the One Million Songs Dataset. It aims to facilitate large-scale retrieval of music information, both symbolic (using MIDI files only) and audio content-based (using information extracted from MIDI files as annotations for matching audio files).

JSB-Chorales offers three temporal resolutions: quarter, eighth and sixteenth. These 'quantizations' are created by retaining the tones heard on a specified time grid. Boulanger-Lewandowski (2012) uses a temporal resolution of quarter notes. This dataset does not currently encode fermatas and is unable to distinguish between held and repeated notes.

[1] http://abcnotation.com/wiki/abc:standard.

Table 1 Examples of music datasets

Dataset	Type	Format
Nsynth[2]	Multitrack	WAV
Lakh MIDI Dataset[3]	Multitrack	MIDI
JSB-Chorales[4]	Harmonized chorale	MIDI
Groove MIDI[5]	Drum	MIDI
Nottingham[6]	Folk tunes	ABC

[2]http://magenta.tensorflow.org/datasets/nsynth

[3]http://colinraffel.com/projects/lmd/

[4]http://www.jsbchorales.net/index.shtml

[5]http://magenta.tensorflow.org/datasets/groove

[6]http://abc.sourceforge.net/NMD/

Groove MIDI Dataset (GMD) consists of 13.6 h of unified MIDI and (synthetic) audio of human-performed, rhythmically expressive drum sounds. The dataset contains 1150 MIDI files and over 22,000 drum sounds.

Nottingham Music Dataset maintained by Eric Foxley contains over 1000 folk tunes stored in a special text format. Using NMD2ABC, a program written by Jay Glanville and some perl scripts, much of this database has been converted to ABC notation.

Based on a comprehensive survey of existing work, it is evident that music generation datasets, represented by the aforementioned datasets, predominantly focus on the collection and processing of Western music genres, with relatively limited attention given to Eastern music genres. While some researchers have gathered and processed music from Eastern genres, it is regrettable that only a few of these datasets are accessible. MG-VAE [58] digitized the folk songs from various regions of China as documented in "Chinese Folk Music Collection," converting them into 2000 distinct regional-style compositions presented in MIDI format. Moreover, the dataset curated by XiaoIce Band [90] focuses on Chinese pop music, suitable for generating multi-track pop music compositions. Jiang et al. [42] compiled a dataset comprising over a thousand Chinese karaoke songs, and the CH818 [33] dataset contains 818 Chinese Xpop (Chinese pop music) songs, from each of which 30-second segments evoking the strongest emotions were extracted and appended with emotional labels. Ajay et al. [116] introduce two large open data collections of Indian Art Music which currently form the largest annotated data collections available for computational analysis of Indian Art Music.

However, it is worth noting that despite the existence of these Eastern music datasets, their accessibility is restricted, possibly due to copyright-related concerns and other issues. To promote the advancement of the relevant research field, further concerted efforts are required to actively construct more comprehensive and diversified datasets of Eastern music, better catering to the research needs. And this necessitates extensive international collaboration, cultural compliance, data transparency, and technological innovation. By enhancing the diversity of datasets, promoting open sharing, utilizing data augmentation techniques, and harnessing community participation, it is possible to provide richer resources for research in Eastern music generation, thereby advancing understanding and facilitating the creative and cultural preservation of Eastern music.

4 Music generation systems

The quest for computers to create music has been pursued by researchers since the dawn of computing. In 1957, Lejaren Hiller produced the IlliacSuite [93], the melody of its fourth movement was the first musical composition in human history composed entirely by a computer using Markov chain. David Cope(1991) [94] introduced Experiments in Musical Intelligence (EMI), which used the most basic process of pattern recognition and recombinancy to create music that successfully imitated the styles of hundreds of composers.

In the contemporary era, the trajectory of music generation systems has traversed a developmental continuum, transitioning from early rule-based methodologies [77] to the presently ubiquitous deep learning-based paradigms [4]. This evolution is characterized by a gradual shift in the generated musical output, progressing from rudimentary and brief monophonic musical fragments to intricate polyphonic compositions that incorporate harmonious chords and protracted durations. This paper surveys the recent state-of-the-art in music generation and classifies them based on algorithm, genre, dataset, and representation, etc. The detailed classification results are presented in Table 2, which can be simply categorized into the following categories according to the purpose of generation:

- Melody generation
- Arrangement generation
- Audio generation
- Style transfer

The melody generator can produce melodies for instruments such as drums and bass, while the composition generator is capable of generating harmonies, lead melodies, and counterpoint, among others. Audio generation involves modeling music directly at the audio level, generating sound segments with different instruments and styles. This method presents challenges, including computational complexity and the difficulty of capturing

Table 2 List of music generation systems

	Name	Year	Algorithm	Application
Rule	T. Tanaka et al	2015	Constraint based System	melody generation
	MorpheuS	2017	Constraints(tonal and pattern)	Arrangement generation
	Chih-Fang Huang et al	2016	Markov chain	Melody generation
	AS Ramanto et al	2017	Markov chain	Melody generation
Markov Model	D Williams et al	2019	Hidden Markov Model(HMM)	melody generation
	Wavenet [13]	2016	Dilated Casual Convolutions	Audio generation
	Engel et al	2017	Wavenet/ AE	Timbre style transfer
	Manzelli et al	2018	WaveNet/biaxial LSTM	Audio generation
	TimbreTron	2019	WaveNet/CycleGAN	Composition style transfer
	Coconet	2019	Counterpoint by Convolution	Reconstruct partial scores
CNN	PerformanceNet	2019	ContourNet/TextureNet	Audio generation
	MelodyRNN	2016	lookback RNN/attention RNN	Melody generation
	Hadjeres et al	2017	Constraint-RNN/Token-RNN(LSTM)	Melody generation
	RL Tuner	2017	LSTM/RL	Melody generation
	StructureNet	2018	LSTM	Melody generation
	Makris et al	2019	LSTM/FF	Melody generation
	Keerti et al	2020	bi-LSTM/attention	Melody generation
	Folk-rnn	2016	LSTM	Arrangement generation
	Song From PI	2016	LSTM	Arrangement generation
	JamBot	2017	LSTM	Arrangement generation
	DeepBach	2017	bi-LSTM/pseudo-Gibbs sampling	Arrangement generation
	XiaoIce Band	2018	GRU/MLP/multi-task learning	Arrangement generation
	Amadeus	2019	LSTM/RL	Arrangement generation
	Performance RNN	2018	LSTM	Audio generation
RNN	Jin et al	2020	LSTM/AC RL	Composition style transfer
	C-RNN-GAN	2016	GAN/LSTM	Melody generation
	MidiNet	2017	GAN/CNN	Melody generation
	JazzGAN	2018	GAN/RA	Melody generation
	SSMGAN	2019	GAN/ SSM	Melody generation
	Yu et al	2019	GAN/conditional LSTM	Melody generation
	MuseGAN	2018	GAN	Arrangement generation
GAN	BinaryMuseGAN	2018	GAN/ BN	Arrangement generation
	LeadSheetGAN	2018	GAN/CNN	Arrangement generation
	CycleBEGAN	2018	CycleBEGAN-CNN/skip/recurrent	Timbre style transfer
	CycleGAN	2018	Cycle-GAN	Composition style transfer
	Play as You Like	2019	RaGAN/MUNIT	Composition style transfer
	WaveGAN	2019	DCGAN	Audio generation
GAN	GANSynth	2019	PGGAN	audio generation
	MusicVAE	2019	VAE/RNN	Melody generation
	GrooVAE	2019	VAE/LSTM	Melody generation
	Jiang et al	2019	attention-VAE	Melody generation
	MuseAE	2020	AAE/bi-LSTM	Melody generation
	MIDI-Sandwich2	2019	BVAE/MFG-VAE	Arrangement generation
	Rivero et al	2020	VAE	Arrangement generation
	Jukebox	2020	VQ-VAE	Arrangement generation
	MIDI-VAE	2018	parallel VAE/GRU	Composition style transfer
	MahlerNet	2019	conditional VAE/bi-RNN	Composition style transfer
VAE	MG-VAE	2019	VAE/bi-GRU	Composition style transfer

Table 2 (continued)

	Name	Year	Algorithm	Application
	Music Transformer	2019	transformer	Arrangement generation(piano)
	LakhNES	2019	transformer XL	Arrangement generation
	MuseNet	2019	Transformer	Arrangement generation
	Guan et al	2019	self-attention/GAN	Arrangement generation
	Zhang et al	2020	transformer/GAN	Arrangement generation
	Transformer VAE	2020	transformer/VAE	Melody generation
Transformer	Choi et al	2020	Transformer/VAE	melody generation
	Muńoz et al	2016	MA/fitness:Harmonic rules	Melody generation
	Jeong et al	2017	EA/fitting function: Multi-objective (stability, tension)	Melody generation
	GGA-MG	2020	GA/fitting function: LSTM network	Melody generation
	S Hickinbotham et al	2016	GA/fitness function: Interactive	Live-coding
	Kaliakatsos et al	2016	EA/PSO	Arrangement generation(interactive)
	Olseng O et al	2018	EA/ fitness function:multi-objective	Arrangement generation
			EA/ fitness function:multi-objective	Arrangement generation
EA	EvoComposer	2019	Harmony rules, corpus statistical analysis	Arrangement generation

semantic structure from raw audio. Style transfer encompasses timbre conversion, arrangement style transformation, etc., and attempts to apply a specific style (such as a composer or musical genre) or a particular instrument (e.g., drums, bass) to the original musical composition using automatic arrangement techniques. Detailed classification information for each music generation system can be found in Table 2. Although many music generation algorithms feature combinations and nesting, such as the use of Variational Auto-Encoder (VAE) combined with Recursive Neural Networks (RNN) [73], music generation algorithms can be generally categorised as follows:

- Rule Based Systems
- Markov Model
- Deep Learning: LSTM, VAE, GAN et al.
- Evolutionary Computation, EC

4.1 Rule based music generation

The rule based music generation system is a non-adaptive model, which generates music based on the theoretical knowledge of music. Including rule based on grammar [77] and its variant L-system, rule-based constraints: such as similarity [50], rhythm rules [30], etc. These constraints help guide the process and direction of intelligent music composition. The performance of a rule based generation system largely depends on the creativity of the developer, the musical concept of the inventor, and how this abstraction is expressed in terms of the corresponding variables [46]. T.Tanaka et al. [77] proposed a formula for generating music rhythm based on grammatical rules, focusing on the structure and redundancy of music, and controlling some global structure of the note rhythm sequence, but the project is still a theoretical expression, has not yet been practiced. Miguel et.al. [16] introduced the user's emotional input based on the use of rules derived from music theory knowledge and proposed a bottom-up approximation approach to design a two-level multi-agent structure for music generation in a modular way. Although the problem of musical expressiveness is well solved, the system is still at a rather early stage of development, handling user input very rigidly and performing poorly when dealing with fuzzy concepts. Besides, as Fig. 4 shows, MorpheuS [30] uses Variable Neighborhood Search (VNS) to assign pitches by detecting repeated rhythms in music to generate polyphonic music with specific tension and repeating rhythms.

Rule based generative systems can effectively constrain the exploration space. However, music is organized on multiple abstract levels, even in highly structured and rule-bound compositions like Bach chorales [36], complex rules may not be sufficient to capture deeper-level structures. Rule based music generation systems to some extent limit the diversity of their outputs. As the system involves more analysis and rules, it tends to introduce higher degrees of ambiguity.

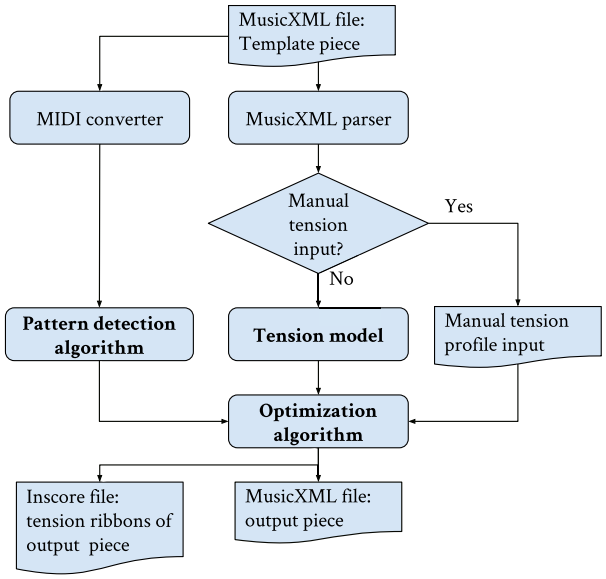


Fig. 4 Overview of MorhpeuS' architecture [30]

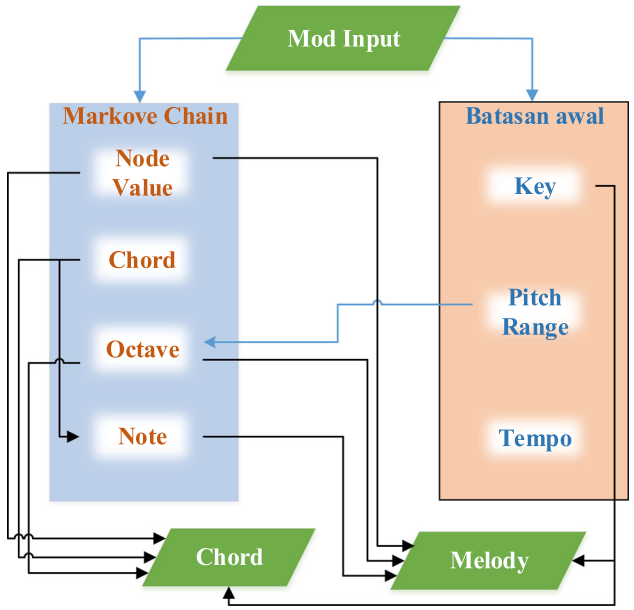


Fig. 5 Programmable Markov music generation system [71]

4.2 Markov model

Markov model [35, 71, 85] is a probability-based generation model for processing time series data, i.e. data where there is a time series relationship between samples. The probability of occurrence of specific elements in the data set and the probability of well-defined features can be captured, such as the occurrence of notes or chords, the transition of notes or chords, and the conditional probability of generating chords on a given note.

Due to the temporal nature of music segments, Markov models are highly suitable for generating music elements such as melodies (note sequences) [9].Chi-Fang Huang et al. [35] established a Markov chain switching table of the Chinese pentatonic scale by analyzing the modes of Chinese folk music. They generate the next note of the current input melody according to the Markov chain switching table and the rhythm complexity algorithm, and re-present a specific style of music. A S Ramanto et al. [71] used four Markov chains to model the components of a musical fragment including tempo, pitch, note value and chord type, and assigned parameter values for different moods to each part, thus ensuring that the music generation system could adapt to different moods of the input. The model of the procedural music generation algorithm used in the literature is shown in Fig. 5. D Williams et al. [85] also generated melodies based on emotions, with the difference that the approach used a Hidden Markov Model (HMM) for modelling, showing that emotionally relevant music can be composed using a Hidden Markov Model.

Compared with the deep learning frame, the advantage of the Markov model is that it is easier to control, allowing constraints to be attached to the internal structure of the algorithm [4]. However, from a creative perspective, Markov models may yield more novel combinations for shorter sequences, but for longer, structured sequences, they might non-innovatively reuse many elements from the corpus, leading to an excessive repetition of segments.

4.3 Deep learning methods for music generation

Content generation is an extended area of deep learning. With the achievements of Google's Magenta[2] and CTRL (Creator Technology Research Lab)[3] in music generation, deep learning as a method for music generation is garnering increasing attention. Unlike grammar-based or rule-based music generation systems, deep learning based music generation systems can learn the distribution and relevance of samples from an arbitrary corpus of music and generate music that represents the musical style of the corpus through prediction (e.g. predicting the pitch of the next note of a melody) or classification (e.g. identifying the chords corresponding to the melody).

4.3.1 Convolutional neural networks

Convolutional neural networks (CNNs) are primarily used for processing image data that does not involve time-series characteristics, making it relatively challenging to handle music data with time-series features. As a result, there are currently fewer music generation models based on CNNs.

[2] https://magenta.tensorflow.org/.

[3] https://www.francoispachet.fr/.

However, it is still feasible to generate music through improvements in model architecture or sophisticated data preprocessing techniques.

The audio generation model WaveNet [67], introduced by Google Deepmind, including its variants TimbreTron [60] (timbre conversion), Manzelli et al. [20] (audio generation) provide effective improvements to convolutional neural networks to enable them to generate music clips. As Fig. 6 shows, Wavenet generates new music clips with high fidelity by introducing extended causal convolution (Dilated Casual Convolutions), which predicts the probability distribution of the current occurrence of different audio data based on the already generated audio sequences. By introducing extended convolution, the deep model has a very large perceptual field, thus dealing with the long-span time-dependent problem of audio generation. However, there is no significant repetitive structure in the generated music fragments and the generated audio is less musical. Manzell et al. improved this method by using an LSTM (Long Short-Term Memory) network combined with the Wavenet architecture to generate cello-style audio clips based on a given melody from the MusicNet dataset [122], such as the Happy Birthday and a C major scale. Where the LSTM network is used to learn the melodic structure of different musical styles, the output of this model is used as input to the Wavenet audio generator to provide the corresponding melodic information for the generation of the audio clip.

The Wavenet model can also be used for other types of music generation tasks, with Engel et al. [20] constructing a Wavenet-like encoder to infer hidden temporal distribution information and feeding it into a Wavenet decoder, which effectively reconstructs the original audio and enables conversion between the timbres of different instruments. Similarly, TimbreTron [38] is a musical timbre conversion system that applies 'image' domain style transfer to the time-frequency representation of audio signals. The timbre transfer is performed in the log-CQT domain using CycleGAN (Cycle Generative Adversarial Networks) by first calculating the Constant Q Transform of the audio signal and using its logarithmic amplitude as an image, and finally using the Wavenet synthesiser to generate high-quality waveforms, enabling the transformation of instrument timbres such as cello, violin and electric guitar. In addition, Coconet [36] used a convolutional alignment network to automate the filling of incomplete scores by modelling the tenor, baritone, bass and soprano in a Bach chorus as an eight-channel feature map, using the feature map of randomly erased notes as input to the convolutional neural network.

4.3.2 Recurrent neural networks

Recurrent Neural Networks (RNN) are a class of neural networks for processing time series, which are very suitable for processing music clips. However, RNN cannot solve the long time dependency problem due to the gradient problem that occurs when processing long time sequence data, which is prone to gradient explosion (disappearance). LSTM, as a variant of RNN, effectively alleviates the long time dependency problem of RNN by introducing cell states and using three types of gates, namely input gates, forgetting gates and output gates, to hold and control information. LSTM networks are mainly used in music generation systems to generate music clips instead of RNN networks.

The combination of LSTM and other algorithms can effectively generate melodies, polyphonic music, etc. The MelodyRNN (Fig. 7) proposed by the Google Brain team [83] built a lookback RNN and an attention RNN in order to improve the ability of RNNs to learn long sequence structures, where the attention RNN introduced an attention mechanism (attention) to model higher-level structures in music to generate melodies. Keerti et al. [48] also introduced an attention mechanism and used dropout to reduce overfitting to generate jazz music. Rl Tuner [39] used an RNN network to provide partial reward values for a Reinforcement Learning (RL) model, with the reward values consisting of two components: (1) the similarity of the next note generated to the note predicted by the reward RNN; (2) the degree of satisfaction with the user-defined constraints. However, this method is based on overly simple melodic composition rules and can only generate simple monophonic melodies. Other music generation systems for melody generation include Hadjeres et al. [27], which proposes an Anticipation-RNN neural network structure for generating melodies for interactive Bach-style choruses; StructureNet [62] generates simple monophonic accompanying music based on LSTM networks; and Makris et al. [59] constructed a separate LSTM network for each drum type, with a feed-forward (FF) network serving as the conditional layer. Its input consists of a one-hot matrix containing 26 features, including rhythm and time signature, and it can generate sequences for drums without prior training on time signatures.

Regarding arrangement generation, Folk-rnn first used LSTM networks to model musical sequences transcribed into ABC notation for generating folk music. DeepBach [28] used a bi-directional LSTM (Bi-LSTM) to generate Bach-style choral music considering two directions in time: one aggregating past information and the other aggregating future information, demonstrating that the sequences generated by the bi-directional LSTM were more musical. XiaoIce Band [90] proposes an end-to-end

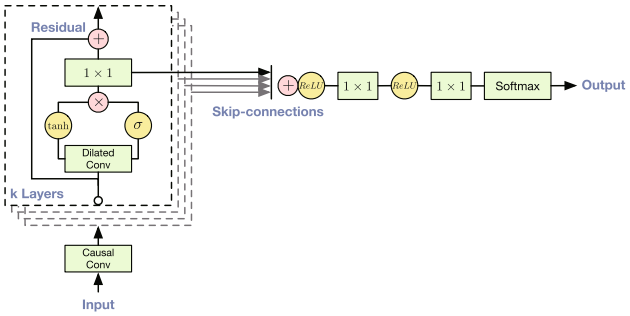


Fig. 6 WaveNet: overview of the residual block and the entire architecture [67]

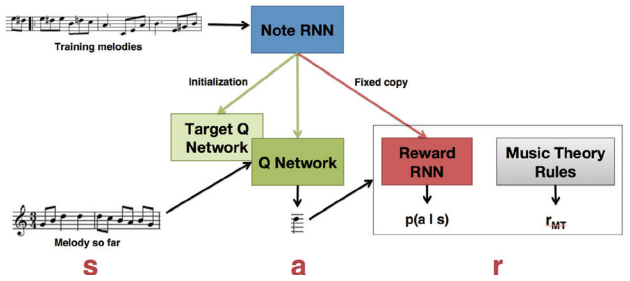


Fig. 7 A Note RNN is trained on MIDI files and supplies the initial weights for the Q-network and Target-Q-network, and final weights for the reward RNN [83]

Chinese multi-track pop music generation framework that uses a GRU network to handle the low-dimensional representation of chords and obtains hidden states through encoders and decoders, ultimately generating music for multi-instrument tracks. Amadeus [49] uses an explicit duration encoding approach: multiple audio streams represent note durations and uses RL's reward mechanism to improve the structure of the generated music. Jambot [6] employs a chord LSTM network to predict chord sequences, and a polyphonic LSTM network to generate polyphonic music based on the predicted chord sequences. Song From PI [12] uses a layered RNN network model, where the two bottom LSTM networks generate the melody and the two top LSTM networks generate the drums and chords.

In addition, audio generation models such as Performance RNN [68] and PerformanceNet [84] are considered to be the possible future development directions of music generation systems [19]. Music performance is a necessary condition for giving meaning and feeling to music. Direct synthesis of audio with a library of sound samples often leads to mechanical and dull results. Performance RNN converts a MIDI file of a live piano piece into a musical representation of multiple one-hot vectors with 413 dimensions, and the method specifies a time "step" to a fixed size (10ms) rather than a note time value, thus capturing more expressiveness in the note timing. PerformanceNet is the first attempt at score-to-audio conversion using a fully convolutional neural network with a symbolic representation of the music (Piano-roll) as input and an audio representation (sound spectrum map) as output. The model consists of two sub-networks: a convolutional encoder/decoder structure is used to learn the correspondence between Piano-roll and the acoustic spectrogram; a multi-segment residual network is further used to optimise the results by adding spectral textures for overtones and timbres, ultimately generating music clips for violin, cello and flute.

4.3.3 Generative adversarial nets

Generative Adversarial Nets (GAN) [25] is a state-of-the-art method for generating high-quality images. The idea behind the method is to train two networks simultaneously: a generator and a discriminator, with the two lattices playing against each other, with the discriminator eventually being unable to distinguish between the truth and falsity of the real data and the generated data as the optimal result. Researchers have been working on applying this to more sequential data, such as audio and music. For music generation, the problems are that:

(1) music is of a certain time series
(2) music is multi-track/multi-instrumental
(3) monophonic music generation cannot be used directly for polyphonic music

Numerous researchers have improved GAN networks to generate melodies, arrangements, style transformations, etc.

For melody generation, C-RNN-GAN [63] uses LSTM networks as a generator and discriminator model, but the method lacks a conditional generation mechanism to generate music based on a given antecedent. MidiNet [87] improves the method by inserting the melody and chord information generated in the previous stage as a conditional mechanism in the middle convolution layer of the generator to restrict the generation of note types, thus improving the innovation of generating single jazz pieces, and the diagram of this model is shown in Fig. 8. JazzGAN [79]

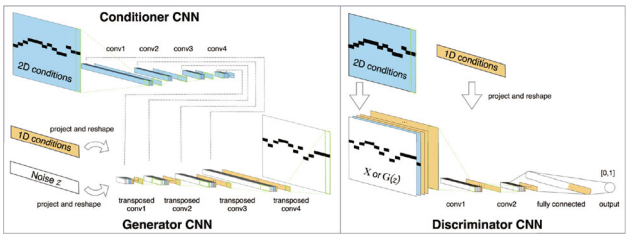


Fig. 8 System diagram of the MidiNet model for symbolic-domain music generation [87]

developed a GAN-based model for single jazz generation, using LSTM networks to improvise monophonic jazz music over chord progressions, and proposed several metrics for evaluating the musical features associated with jazz music. SSMGAN [41] used the GAN model to generate a self-similar matrix (SSM) to represent musical self-repetition, and subsequently fed the SSM matrix into an LSTM network to generate melodies. Yu et al. [88] first proposed a conditional LSTM-GAN for melody generation from lyrics, containing an LSTM generator and LSTM discriminator, both with lyrics as conditional inputs.

For arrangement generation, MuseGAN proposed a GAN model with three types of generators to construct correlations between multiple tracks, containing independent generation within tracks, global generation between tracks, and composite generation, but the method produced many over-fragmented notes. BinaryMuseGAN [18] improved upon the aforementioned method by introducing binary neurons (BN) as inputs to the generator. This representation makes it easier for the discriminator to learn decision boundaries. It also introduces chromatic features to detect chord characteristics, reducing the excessive fragmentation of notes. LeadSheetGAN [53] uses a functional score (Lead Sheet) as a conditional input to generate a Piano- roll, and this approach produces results that contain more information at the musical level due to the clearer melodic direction of the functional score compared to MIDI data.

In terms of style transfer and audio generation, CycleGAN [7] introduces a style loss function for style conversion and a content loss function to maintain content consistency based on Cycle-GAN [91] in order to achieve conversion of jazz, classical and pop music, while using two discriminators to determine whether the generated music belongs to the merged set of two music domains. CycleBEGAN [86] uses BEGAN network [3] to stabilize the training process, while introducing jump connections to improve the clarity of melody and lyrics, and recursive layers to improve the accuracy of pitch to achieve male and female voice conversion. In addition, Jin et al. [45] used the LSTM network as a generator, adding music rules as a reward function in reinforcement learning to a control network that takes advantage of the AC (Actor Critic) reinforcement learning algorithm to select for diversity of elements and avoid the drawback of GAN networks generating samples that are too similar, thus improving the creativity of stylistic music generation. The same goes for Play as You Like [57] which implements the conversion of classical, jazz and pop music. For audio generation, WaveGAN [15] achieved the first attempt to apply GAN to the synthesis of raw waveform audio signals, and GANSynth [21] improved the method by generating the entire sequence in parallel rather than sequentially, which appended a one-hot vector representing pitch and achieved independent control of pitch and timbre based on a PGGAN networ [29], with the final results 50,000 times faster than the standard Wavenet [67], and the generated music clips are better than Wavenet and WaveGAN [15].

Despite the outstanding performance in music generation, GAN still have shortcomings:

(1) They are difficult to train
(2) They are poorly interpretable, and there is still a lack of high-quality research results on how GANs can model text-like data or musical scores.

4.3.4 Variational auto-encoder

Variational Auto-Encoder (VEA) is essentially a compression algorithm for encoders and decoders, which has been able to analyse and generate information such as pitch dynamics and instrumentation in polyphonic music. The use of VAE aims to solve two problems [72]: music recombination and music prediction, but when the data is multimodal, VAE does not provide a clear mechanism for generating music, usually using a hybrid coding model based on VAE, such as a combination of VAE networks and RNN networks.

Some typical examples are MIDI-VAE [5], Fig. 9 and MusicVAE [73]. Among them, MIDI-VAE constructs three pairs of coders/decoders sharing a potential space to automatically reconstruct the pitch, intensity and instrumentation of a musical composition for musical style conversion. This work represents the first successful attempt to apply style conversion to a complete musical composition, but the small amount of data used for style feature training results in short lengths of the generated music. MusicVAE uses hierarchical decoder to improve the modelling of sequences with long-term structure, using a bidirectional RNN as an encoder, allowing the model to interpolate and reconstruct well. Then, GrooVAE [24] as a

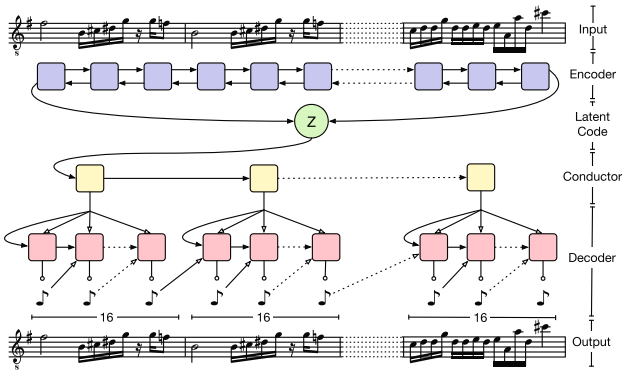


Fig. 9 Schematic of the hierarchical recurrent variational autoencoder model, MusicVAE [73]

variant of MusicVAE, generates drum melodies by training on a corpus of electronic drums recorded from live performances. Wei et al. [96] did not adopt an end-to-end approach to learning hierarchical representations but instead introduced a novel model based on EC^2 -VAE. This model utilizes a context-constrained learning approach for long-term symbolic music representations, employing contrastive learning methods and a hierarchical prediction model to train and optimize these long-term representations. It can decompose the pitch and rhythm components of melodies (32 bars) without increasing the dimensions of the long-term representations. Dubnov et al. [99] designed a vanilla polyphonic VAE using only linear layers to learn a latent representation of the musical surface. They presented a theoretical framework of musical creativity and surprisal formulated in terms of information theoretical relations between fullrate (high-fidelity) encoding of the musical data, and a lower complexity latent encoding that models reduced informational aspects of musical structure.

For the generation of Eastern popular and folk music, MG-VAE [58] was the first study to apply deep generative models and adversarial training to oriental music generation, which was able to convert music into folk music with a specific regional style. Jiang et al. [42] developed a music generation model with style control. This method is based on MusicVAE and utilizes global and local encoders to construct three music generation models. It attempts to find an appropriate approach to represent musical structure at the levels of bars, phrases, and songs. Ultimately, it can generate melodies with style control.

MahlerNet [56] constructed a conditional VAE to model the distribution of latent states. Two bidirectional RNN networks (one considering music reconstruction and one considering context) constitute an encoder, and the decoder outputs duration, pitch, and instrument, realized the use of any number of instruments to simulate polyphonic music of any length. MIDI-Sandwich2 [51] introduces a hierarchical multimodal fusion generative VAE (MFG-VAE) network based on RNN. Initially, it utilizes multiple independent Binary VAE (BVAE) models with non-shared weights to extract feature information for different instruments in various tracks. MGF-VEA, serving as the top-level model, blends features from different modes, and finally, multi-track music is generated through the decoding of BVAE. This method improves the refinement method of BinaryMuseGAN [18], transforms the binary input problem into a multi-label classification problem, and solves the problem of the difficulty of distinguishing the refining steps of the original scheme and the difficulty of descending the gradient. MuseAE [80] applied Adversarial Autoencoders to music generation for the first time. The advantage of this approach over VAE lies in its use of adversarial regularization instead of KL divergence, which, in principle, can enforce it to adhere to any probability distribution, thereby offering greater flexibility in selecting the prior distribution for latent variables. The Jukebox [14] built a system that can generate a variety of high-fidelity music in the original audio domain, but this method still requires 9 h to render one minute of music.

4.3.5 Transformer

A piece of music often has multiple themes, phrases or repetitions on different time scales. Therefore, generating long-term music with complex structures has always been a significant challenge. The music generation system is usually based on a sequential RNN (or LSTM, GRU, etc.) network, but this mechanism brings about two problems: Calculating from left to right or calculating from right to left limits the parallel ability of the model; Part of the information will be lost during the sequence calculation. This led Google to propose the classical network architecture Transformer [81], which uses an attention mechanism instead of the traditional Encoder-Decoder framework that combine the inherent patterns of CNNs or RNNs. The core idea of this architecture is to use the Attention mechanism to indicate correlations between inputs, with the following advantages:

(1) Data parallel computing
(2) Visualization of self-reference
(3) Solve the problem of long-term dependence better than RNN, and has achieved great success in the task of maintaining long continuity in timing.

However, the feasibility of this model in music generation tasks is limited because the spatial complexity of the intermediate vectors representing relative positions in the algorithm is extremely high (quadratic in the sequence length). Music Transformer [37] significantly reduces the spatial complexity of the intermediate vectors representing relative positions to the order of sequence length, making it applicable for piano music composition. However, for longer music generation, the resulting compositions suffer from poor musical quality and structural disorder. LakhNES [120] used transfer learning to generate chip music by pre-training in a large-scale dataset using Transformer's latest extension Transformer-XL, and then fine-tuning in a small-scale chip music dataset. MuseNet [69] used the same network as GPT-2 (a large Transformer-based language model [8]), which is trained using the Transformer's recomputed and optimised kernel to generate a four-minute musical composition consisting of ten different instruments, but the model does not necessarily arrange the music according to the input instruments, generating each note by calculating the probability of all

possible notes and instruments. Figure 10 shows the flow chart of this Transformer self-encoder.

In addition, some researchers have used Transformer in combination with other models to address problems such as the poor musicality of long sequences. Guan et al. [26] proposed a combination of Transformer and GAN, first using a self-attentive mechanism based on GAN to extract spatial and temporal features of music to generate consistent and natural monophonic music, and then using a generative adversarial structure with multiple branches to generate multi-instrumental music with harmonic structure between all tracks. Zhang et al. [89] also used the Transformer in combination with GAN to make the generation of long sequences guided by the adversarial goal, which provides powerful regularisation conditions that can force the Transformer to focus on learning the global and local structure of the music. Transformer VAE [43] combines Transformer with VAE, effectively addressing the limitations of VAE in handling time series structures and the non-explanatory nature of Transformer's latent states. Choi et al. [11] similarly used the Transformer (decoder) to obtain a global representation of the music, thus better control over aspects such as performance style and melody.

Several recent works have introduced innovative music generation frameworks, including user-friendly interfaces, Transformer-based methodologies, and novel representations, which have demonstrated improvements in music style fidelity, enriched melodic content, and faster sequence generation. Guo et al. [97] introduce a new music generation framework with a user-friendly interface for infilling, utilizing a transformer-based approach that supports customizable control tokens, including tonal tension and track polyphony. Their research explores the impact of these tokens on music generation, demonstrating improved stylistic fidelity compared to previous methods. The model is made available in a Google Colab notebook for interactive music creation. Zou et al. [101] proposed a

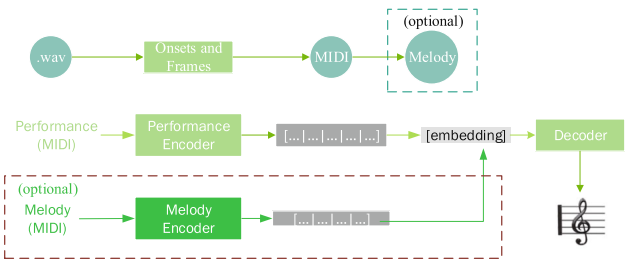


Fig. 10 The flow chart of Transformer self-encode: first transcribes the.wav data file into a MIDI file using the Onsets and Frames framework, which is then encoded into a performance representation to be used as input. The output of the performance encoder is then aggregated across time and (selectively) combined with melodic embedding to produce a representation of the entire performance, which is then used by the Transformer decoder when reasoning [69]

transformer based framework MELONS that can generate enjoyable long-term melodies with obvious structure and rich contents by using a structure generation net and a melody generation net. One of the key ideas is to represent structure information using graph which consists of eight hierarchical relations among pairwise bars. Dong et al. [98] propose a new multitrack music representation that encoded each musical event as a tuple of six variables (i.e., type, beat, position, pitch, duration, and instrument) to significantly shorten the sequence length of multitrack music and also allow a diverse set of instruments. Furthermore, a decoder-only transformer model with multi-dimensional input and output spaces called Multitrack Music Transformer (MMT) was proposed to generate longer sequences at a faster inference speed. Besides, Yu et al. [100] proposed Museformer with a novel fine- and coarse-grained attention. The fine-grained attention is applied to the structure-related bars for learning the structure-related correlations, and the coarse-grained attention is applied to the summarization of the other bars for getting a sketch of them.

4.3.6 Evolutionary algorithms

The field of evolutionary computing encompasses various techniques and methods inspired by natural evolution. At its core lies the Darwinian search algorithm based on highly abstracted biological models. The advantage of applying them to music generation systems is that they can be optimized in the direction of high relative quality or adaptability, thus compressing or extending the data and processes required for the search. It provides an innovative and natural means of generating musical ideas from a set of specifiable original components and processes [95]. EA requires three basic criteria to be considered in advance: 1. Problem domain: defining the type of music to be created; 2. Individual Representation: the representation of the music, melody, harmony, pitch, duration, etc.; 3.Fitness Measure.

Fitness measure is a very important part of genetic algorithms to evaluate the goodness of a solution, where the fitness statistics of different music generation systems are shown in Table 2. In music generation systems, fitness measure includes interactive adaptation, similarity to a music corpus, rule-based metrics, etc. Early evolutionary algorithms commonly used interactive adaptation measure, which requires significant human intervention and can suffer from adaptation bottlenecks; poor robustness and wider applicability in music corpus or rule-based fitness measures, as well as limitations on the creative potential of the system.

For melody generation, Munoz et al. [65] employ unfigured bass harmonization techniques to create bass

melodies. They use harmonic rules as an adaptability metric, taking the bass line as input, and in a parallel manner, they find subspaces for suitable melodies. By utilizing a local composer intelligent agent with self-adaptation, they produce high-quality four-part musical compositions. Jeong et al. [40] proposed a multi-objective evolutionary algorithm for automatic melodic synthesis that produces multiple melodies at once, which measures the psychoacoustic effects of the resulting music based on the stability and tension of the sound as an adaptive metric. Lopes et al. [54] proposed an automatic melodic synthesis method based on two adaptive functions, one defined using Zipf's law [92] and the other using Fux's law [23]. GGA-MG [22] uses LSTM as an adaptive function, first providing a sufficient dataset of random and artificial samples using an initial Generative Genetic Algorithm (GGA), and then using a Bi-LSTM neural network for training as an adaptive function for the main GGA to generate melodies similar to those of human compositions.

In arranging music, MetaCompose [74], Fig. 11, was used to generate real-time music, using a multi-objective adaptive function to construct a chord generator, melody generator, and accompaniment generator, but the method lacked the generation of harmonies. EvoComposer [13] EvoComposer [81] enhances this approach by generating harmonies for four voice types (tenor, bass, soprano, alto) based on a given choice. It defines an adaptability function that adheres to classical music rules while incorporating statistical information extracted from an existing music corpus to absorb specific composer styles. Olseng [66] used four adaptation measures: local, global melodic features and local, global harmonic features with the aim of making the melodies generated by the algorithm sound harmonious and pleasant. Kaliakatsos [47] uses Particle swarm optimization (PSO) to converge on user-preferred rhythmic and pitch features, and then uses a genetic algorithm to compose music based on these features in a 'top-level' evolution.

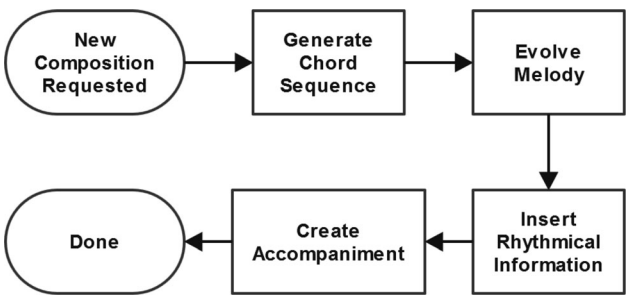


Fig. 11 Steps for generating a composition using evolutionary algorithms [74]

5 Evaluation

In the field of music generation, evaluating the effectiveness of generation is a major challenge. Existing evaluation methods mainly combine subjective and objective evaluation, where subjective evaluation includes Turing tests, questionnaires, etc., and objective evaluation includes quantitative indicators based on musical rules, similarity to corpus or style, etc. Agres et al. [61] provide an overview of current evaluation methods for music generation systems, providing provides motivation and tools. These metrics are interrelated and together influence the construction of the music generation system and the quality of the resulting pieces.

5.1 Subjective evaluation

Subjective evaluation of music generation systems is usually done by performing human listening tests to evaluate the performance of the system in terms of melodic output. This method can be used in a similar way to the Turing test or questionnaires. For example, in MG-VAE [58], listeners are guided to rate the musicality (i.e. whether the song has a clear musical style or structure) and stylistic significance (i.e. whether the style of the song matches a given regional label) of the generated music, combined with several quantitative metrics based on musical rules to evaluate the merits of the music generation system. Chen et al. [10] define a manual evaluation metric that includes Interactivity i.e. whether the interaction between chords and melody is good, complexity i.e. whether the piece is complex enough to express the theme, and structure i.e. whether there is some repetition, forward movement, variation in the sample. Performance RNN collects people's ratings of the generated music; Jaques et al. [39, 53, 84] combine human ratings to judge the performance of the music generation system. In [39], a user study is conducted via Amazon Mechanical Turk in which participants are asked to rate which of two randomly selected melodies they preferred on a Likert scale. In [84], an online user study is conducted to evaluate the result of arrangement generation. After listening to the music, respondents are asked to vote for the best model among the three, in terms of harmonicity, rhythmicity and overall feeling, respectively. Moreover, they are asked to rate each sample according to the same three aspects. In [53], they evaluate the generated music clips by a user study and compare the mean opinion score (MOS). Specifically, 156 adults of which 41 are music professionals or students major in music are recruited and asked to rate the music pieces following metrics in a five-point Likert scale.

While human feedback may be the most reasonable method for evaluating generated music, it has certain limitations. Human users are unable to rate a large volume of music, and user fatigue may occur within the first few minutes of rating or selecting music. Additionally, due to individual differences in audience taste, the uncertainty in music ratings or selections increases, resulting in inconsistent ratings. While Sturm et al. [75] extensively discuss the advantages of evaluating music in the form of concerts or by extending concerts to music competitions, such as setting a concert as one of the most natural ways to experience music, which can to some extent reduce the fatigue of subjects compared to laboratory settings, this method inevitably introduces differences due to variations in performers, venues, and environments.

5.2 Objective evaluation

Objective evaluation of music generation systems includes similarity to a corpus or style, quantitative indicators based on musical rules, etc. Similarity is central to the success metrics in music generation systems, therefore, an important challenge becomes finding the right balance between similarity and novelty or creativity. Evaluating the creativity (sometimes equated with novelty) of generated music is a complex topic and is discussed at length by Agres et al. [1]

Different objective evaluation criteria can be chosen depending on the type of generated music. Zhang et al. [89] devised seven objective metrics to compare generated musical compositions with real compositions, these include: number of different note repetitions, count of different pitches, number of triplets, etc. GANSynth [21] compared generated music with real compositions by using manual evaluation, counting the number of difference Bins (a measure of generated music fragment diversity), initial scores (to evaluate the GAN network), pitch accuracy, pitch entropy, etc. JazzGAN [79] employs three metrics, including the pattern collapse metric for observing intervals and durations of repeated and incoherent notes, the creativity metric to measure the extent to which generated segments replicate sequences from the corpus and the number of variant sequences, and the harmony and chordal metric to evaluate the relationship between the generated musical sequence and a given chord sequence in order to assess the generative system. MuseGAN [17] has designed two aspects for evaluating the differences between generated music and real music in a single track and between different tracks, which include within a track: the proportion of empty bars, the number of pitch classes per bar, the proportion of eligible notes, and the ratio of notes per 8-beat or 16-beat interval, and between tracks: the pitch distance between tracks. In [39], objective metrics are selected to assess the results, including 1) the number of notes belonging to some excessively repeated segment; 2) the ability to play in key, resolve melodic leaps, and play motifs; 3) the number of melodies that start with the tonic note; 4) melody auto-correlation, and 5) repeated motifs. In [84], four objective metrics are selected: 1) Empty bars; 2) used pitch classes; 3)qualified note and 4) tonal distance.

A major challenge in the objective evaluation of music production systems is that there is no single objective evaluation criterion for judging the 'goodness' of a system. These criteria are limited in that they are only applicable to the current type of music generation task and cannot be applied to all music generation systems, and their reasonableness has yet to be considered.

6 Analysis and perspectives

6.1 Current state of intelligent music generation

Due to the fact that music generation systems (1) can provide a far greater volume of content than composers can offer, their cost per minute of music is much lower than that of composers; (2) they can customize and personalize music based on user settings; (3) they can inspire composers and provide assistance; (4) they can be used for music education, among other applications. Therefore, this field has very extensive prospects for development and commercial value. The company cases that have already landed include Melodrive's focus on instant music generation in game scenarios, Amper Music to generate personalized customized original music in a few seconds, and Ambient Generative Music to automatically generate personalized music in different environments.

With the development of artificial intelligence, there are also certain difficulties in the rapid development of artificial intelligence composition. The current music generation systems can generate polyphonic music with a higher quality and a shorter period of time containing different styles and instruments in a shorter period of time [45]. However, the construction of a long time series of music fragments with a complete structure [43] and the evaluation of generated music [1] have become a common problem for artificial intelligence composition researchers. Although the latest music generation system can generate longer-term music fragments, such as MuseNet, which can use 10 different instruments to generate 4-minute music works. However, the latest music generation algorithms still struggle to effectively handle time series issues. As the generated music segments become longer, the musicality of the produced segments deteriorates, resulting in increasingly chaotic structures or higher levels of repetition.

Currently, researchers are focusing on optimizing generation algorithms or imposing constraints on generated music segments based on music theory knowledge to alleviate the temporal dependency issues in music generation systems.

6.2 Comparative analysis of music generation research in eastern and western contexts

From a technological perspective, there is a noticeable disparity in the development of music generation between the Western and Eastern context. While western music generation research has reached a relatively mature stage, with companies like OpenAI [69], Google [104], and AIVA [105] offering publicly accessible online music generation tools, research in the Eastern world is still in its nascent stages in this field. Although some Eastern enterprises and research institutions in China, Japan and India are actively exploring this field [90, 106–110], research on music generation with a focus on Eastern indigenous music genres remains relatively limited. This discrepancy can be attributed to several factors, represented by differences in musical structure and thought development due to cultural backgrounds, the scarcity of Eastern music databases, the absence of effective evaluation methods, and the lack of interdisciplinary professionals bridging the gap [44].

Cultural differences play a profound role in shaping the technological characteristics of symbolic music generation. Eastern music differs notably from Western classical music (especially symphonic compositions) in terms of structure, musical thinking, and artistic expression [32, 112, 114]. Regarding musical structural power and thinking, Eastern music genres exhibits a distinct understanding that typically construct single-line monophonic music and emphasize the aesthetics of the music. Eastern classical music. For example, Chinese classical music primarily revolves around pentatonic scales and there is a preference for ensemble playing with a group of vocal parts or instruments in traditional Japanese music [111]. Conversely, Western music tends to be polyphonic with an emphasis on major and minor scales, focusing on harmonic logic and using techniques like repetition, contrast, and extension to create atmospheres. Moreover, in the context of artistic expression, Eastern and Western music genres vary significantly. Western music pay more attention to profundity and seriousness, highlighting the contrast between subjectivity and objectivity. In contrast, Eastern traditional music places more significance on the unity of subjectivity and objectivity [82]. And there exists considerable content for improvised performance [113], resulting in more flexible and diverse rhythm. From an information theory perspective, this increases the difficulty of labeling and analyzing traditional Eastern music data. Considering the intrinsic cultural differences like structural, content, and performance differences between Eastern and Western music, music generation systems need extensive adjustments and optimization to be applicable to either, thereby reducing the universality between Eastern and Western music generation systems.

The lack of diverse and high-quality Eastern music corpora is another significant factor restricting the development of music generation systems primarily focused on Eastern music genres. While datasets like MG-VAE [58] have gathered folk music from various regions in China and collections like XiaoIce Band [90] and CH818 [33] cover Chinese pop music, these data resources are still relatively limited. Compared to a variety of predominantly Western music datasets, the insufficiency in both the quantity and quality of Eastern corpora hinders music generation algorithms and models from fully grasping the unique features of Eastern music, thereby impacting the quality of generated music segments. In other Eastern countries, such as India, despite the presence of a rich traditional music culture, there are still challenges in systematically organizing and digitizing it [116]. Similarly, music in the Middle Eastern region, such as Arabic and Persian music, boasts a profound history and unique characteristics, but is limited by challenges related to digitization and data integration [115]. Data pertaining to these musical forms remains relatively scarce. Given that the music of these regions features distinct scales, rhythms, and instrument characteristics, the lack of comprehensive music databases hampers the development of generative technologies tailored to these musical styles. Furthermore, due to variations in development levels and stages, the intricate and complex task of collecting various types of Eastern ethnic music across different cultures and regions is encountered. And issues related to song copyrights also pose constraints on the advancement of music generation research.

Another challenge is the lack of effective evaluation methods and standardized objective assessment criteria for music generation systems. Firstly, this implies that, in automatically generated music segments, there is often a reliance on human ratings to identify those with superior structure and sound quality. Meanwhile, eastern music encompasses a diverse and unique musical tradition, including various genres and styles from China, India, Japan, and other [117]. Secondly, in comparison to the extensive research on Western music genres and structures, the analysis of various Eastern music genres remains limited, making it difficult to obtain objective data suitable for evaluating the effectiveness of Eastern music generation systems. These two issues collectively constrain the further advancement of Eastern music generation research, as the absence of effective evaluation methods and limited

structural analysis hinder the improvement of the quality of Eastern music generation systems.

It is well-established that music theory and strict rules potentially play a crucial role in enhancing generative systems [118]. However, in practice, most researchers in the East studies tend to specialize in a singular domain, lacking interdisciplinary knowledge to effectively integrate the foundational principles of music with cutting-edge generative models. Such imbalance has resulted in unequal progress in the subfields of Eastern music generation and most researchers channel their primary efforts towards optimizing generative algorithms, inadvertently neglecting the potential impact of music theory in refining the generative systems. Moreover, due to the intricacies of music theory, it proves exceptionally challenging for researchers to independently undertake the comprehensive learning and adept modeling of musical theoretical constructs without professional guidance.

In summary, research on music generation with a focus on Eastern indigenous music genres faces inherent challenges such as cultural disparities, insufficient musical databases, inadequate evaluation criteria, and a lack of interdisciplinary knowledge. In order to address these challenges, it is imperative to augment research funding and educational programs, raise public awareness. Concurrently, there is a necessity to bolster interdisciplinary collaboration, curate data comprehensively, formulate standardized evaluation criteria, and promote international cooperation. These measures are essential for advancing the research and development of high-quality music generation systems with cultural relevance. Furthermore, it is worth noting that in today's globalized society, musical cultures have been intermingling and fusing with each other. As music evolves, there is a noticeable shift in perspective from the binary distinction between the East and the West towards differentiating and contrasting music from the standpoint of local, traditional, and modern characteristics. This trend serves to emphasize the diversity and complexity of music, and it also facilitates the further development of music generation models. Therefore, it is anticipated that future research will engage in in-depth discussions and analyses from this perspective.

6.3 Trends in artificial intelligence music

Whether it's addressing the limitations of Eastern music or exploring the commonalities between Eastern and Western music generation, both can serve as directions for the future development and enhancement of music generation systems. The ultimate goal is to generate beautiful songs that meet individual preferences and possess complete structures. By examining the development and current state of AI composition, the trends in the evolution of music generation systems are as follows.

6.3.1 Music generation technology moves towards maturity

Currently, music generation systems predominantly focus on generating either lyrics or melodies. For cross-modal music generation, such as simultaneous generation of music and lyrics or music and video, this will be a crucial direction in the future development of artificial intelligence in music. This direction necessitates the extraction and fusion of features across different modalities, presenting certain technical challenges. In the future development of music generation systems, improvements can be standardized through the use of more robust music data architectures, including Piano-roll, MIDI events, sheet music, and audio, along with more standardized testing datasets and evaluation metrics.

Furthermore, given that deep learning architectures are the mainstream generation methods for music generation systems, enhancing the interpretability of deep networks and their controllability becomes one of the future directions for music generation technology. Additionally, addressing the issue of temporal dependencies in generating music, focusing on segment-level audio generation rather than note-level audio generation, represents a research trend in future music generation systems.

6.3.2 Emotional expression of music generation and its coordinated development with robot performance

Music is one of the most important forms of human emotion expression. Musical emotions are conceptually regarded as an expression of human emotion that is difficult to quantify, and it undergoes rich changes with the progress of music. The current degree of music generation intelligence is at a low level, based more on the perspective of signal analysis, without introducing a human recognition system for musical emotions, without anthropomorphic musical composition thinking, and with human-computer interaction systems limited to shallow information exchange. Therefore, the integration of multi-channel information such as computer vision and computer hearing to recognise the human spectrum of musical emotions and audio expression systems, and then the optimisation of the emotional dimension of music generation using techniques such as deep learning is the main goal of the construction of future interactive intelligent composition systems.

Furthermore, research in the field of music visualization based on robotic systems [52] and emotionally interactive music therapy robots [78] integrates robotics technology with musical expression, with robots as the central element.

In the future, achieving intelligent composition and collaborative performance of music robots under the context of affective computing is a significant highlight in the integrated development of intelligent music generation and performance.

7 Conclusion

The involvement of artificial intelligence in artistic practices has become a common phenomenon, and using AI algorithms to generate music is a highly active research area.

This paper systematically outlines the latest developments in the field of music generation from the perspective of algorithm types and briefly introduces various forms of musical representation and evaluation criteria. Furthermore, the paper analyzes the current state of AI composition on a global scale, particularly in the contexts of Eastern and Western traditional music, and compares the distinct developmental features of the two. In particular, this paper examines the opportunities and challenges for future music generation systems. It is hoped that this review contributes to a better understanding of the current status and trends in AI-based music generation systems.

Acknowledgements Figure 4-11 are cited from relevant paper on music generation, and we extend our sincere appreciation to the authors for their contribution.

Author's contributions Methodology: LW; Formal analysis and investigation: ZZ; Writing—original draft preparation: ZZ, HL; Data curation: JP; Writing—review and editing: YQ, SL, ZZ; Supervision: QW, LW; Funding acquisition: LW.

Data availability Not applicable.

Code availability Not applicable.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Agres K, Forth J, Wiggins GA (2016) Evaluation of musical creativity and musical metacreation systems. Comput Entertain CIE 14(3):1–33 (**Publisher: ACM New York, NY, USA**)
2. Avdeeff M (2019) Artificial intelligence and popular music: SKYGGE, flow machines, and the audio uncanny valley. In: Arts, volume 8, page 130. Multidisciplinary Digital Publishing Institute. Issue: 4
3. Berthelot D, Schumm T, Metz L (2017) Began: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717
4. Briot J-P, Hadjeres G, Pachet F-D (2020) Deep learning techniques for music generation. Springer International Publishing, Cham, Computational Synthesis and Creative Systems
5. Brunner G, Konrad A, Wang Y, Wattenhofer R (2018) MIDI-VAE: Modeling dynamics and instrumentation of music with applications to style transfer. arXiv preprint arXiv:1809.07600
6. Brunner G, Wang Y, Wattenhofer R, Wiesendanger J (2017) JamBot: music theory aware chord based generation of polyphonic music with LSTMs. In: 2017 IEEE 29th international conference on tools with artificial intelligence (ICTAI), pp 519–526, Boston, MA. IEEE
7. Brunner G, Wang Y, Wattenhofer R, Zhao S (2018) Symbolic music genre transfer with CycleGAN. arXiv:1809.07575 [cs, eess, stat]
8. Budzianowski P, Vuli I (2019) Hello, It's GPT-2—How can i help you? Towards the use of pretrained language models for task-oriented dialogue systems
9. Carnovalini F, Rodà A (2020) Computational creativity and music generation systems: an introduction to the state of the art. Front Artif Intell 3:14
10. Chen K, Zhang W, Dubnov S, Xia G, Li W (2019) The effect of explicit structure encoding of deep neural networks for symbolic music generation. In: 2019 International workshop on multilayer music representation and processing (MMRP), pp 77–84. IEEE
11. Choi K, Hawthorne C, Simon I, Dinculescu M, Engel J (2020) Encoding musical style with transformer autoencoders. In: International conference on machine learning, pp 1899–1908. PMLR
12. Chu H, Urtasun R, Fidler S (2016) Song From PI: a musically plausible network for pop music generation. arXiv:1611.03477 [cs]
13. De Prisco R, Zaccagnino G, Zaccagnino R (2020) EvoComposer: an evolutionary algorithm for 4-voice music compositions. Evolution Comput 28(3):489–530 (**Publisher: MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info**)
14. Dhariwal P, Jun H, Payne C, Kim JW, Radford A, Sutskever I. Jukebox: a generative model for music. arXiv preprint arXiv:2005.00341
15. Donahue C, McAuley J, Puckette M (2019b) Adversarial audio synthesis. arXiv:1802.04208 [cs]
16. Delgado M, Fajardo W, Molina-Solana M (2009) Inmamusys: Intelligent multiagent music system. Exp Syst Appl 36(3):4574–4580
17. Dong H-W, Hsiao W-Y, Yang L-C, Yang Y-H (2018) Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment. In: Thirty-second AAAI conference on artificial intelligence
18. Dong H-W, Yang Y-H (2018) Convolutional generative adversarial networks with binary neurons for polyphonic music generation. arXiv:1804.09399 [cs, eess, stat]
19. Dong H-W, Yang Y-H (2019) Generating Music with GANs https://salu133445.github.io/ismir2019tutor ial/pdf/ismir2019-tutorial-slides.pdf. Accessed 11 Jan 2022
20. Engel J, Resnick C, Roberts A, Dieleman S, Norouzi M, Eck D, Simonyan K (20170) Neural audio synthesis of musical notes with waveNet autoencoders. In: International Conference on Machine Learning (pp. 1068-1077). PMLR
21. Engel J, Agrawal KK, Chen S, Gulrajani I, Donahue C, Roberts A (2019) Gansynth: Adversarial neural audio synthesis. arXiv preprint arXiv:1902.08710
22. Farzaneh M, Toroghi RM. GGA-MG: Generative genetic algorithm for music generation. arXiv preprint arXiv:2004.04687

23. Fux JJ, Edmunds J (1965) The study of counterpoint from Johann Joseph Fux's Gradus ad parnassum. Number 277. WW. Norton & Company
24. Gillick J, Roberts A, Engel J, Eck D, Bamman D (2019) Learning to groove with inverse sequence transformations. In: International conference on machine learning (pp. 2269–2279). PMLR
25. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Adv Neural Inf Process Syst, 27
26. Guan F, Yu C, Yang S (2019) A GAN model with self-attention mechanism to generate multi-instruments symbolic music. In: 2019 International joint conference on neural networks (IJCNN)
27. Hadjeres G, Nielsen F (2017) Interactive music generation with positional constraints using anticipation-RNNs. arXiv preprint arXiv:1709.06404
28. Hadjeres G, Pachet F, Nielsen F (2017) DeepBach: a steerable model for bach chorales generation. In: International conference on machine learning, pp 119–127.PMLR
29. Han C, Murao K, Noguchi T, Kawata Y, Uchiyama F, Rundo L, Nakayama H, Satoh S (2019) Learning more with less: Conditional PGGAN-based data augmentation for brain metastases detection using highly-rough annotation on MR images. In: Proceedings of the 28th ACM International conference on information and knowledge management, pp 119–127
30. Herremans D, Chew E (2019) MorpheuS: generating structured music with constrained patterns and tension. IEEE Trans Affect Comput 10(4):510–523
31. Herremans D, Chuan C-H, Chew E (2017) A functional taxonomy of music generation systems. ACM Comput Surv 50(5):1–30
32. Hu X, Lee JH (2012, October) A cross-cultural study of music mood perception between american and chinese listeners. In: ISMIR (pp. 535–540)
33. Hu X, Yang Y-H (2017) The mood of Chinese Pop music: pepresentation and recognition. J Assoc Inf Sci Technol
34. Huang A, Wu R (2016) Deep learning for music. arXiv preprint arXiv:1606.04930
35. Huang C-F, Lian Y-S, Nien W-P, Chieng W-H (2016) Analyzing the perception of Chinese melodic imagery and its application to automated composition. Multimedia Tools Appl 75(13):7631–7654
36. Huang C-ZA, Cooijmans T, Roberts A, Courville A, Eck D (2019) Counterpoint by convolution. arXiv preprint arXiv:1903.07227
37. Huang C-Z A, Vaswani A, Uszkoreit J, Shazeer N, Simon I, Hawthorne C, Dai AM, Hoffman MD, Dinculescu M, Eck D (2018) Music transformer. arXiv preprint arXiv:1809.04281
38. Huang S, Li Q, Anil C, Oore S, Grosse RB (2019) Timbretron: A wavenet (cyclegan (cqt (audio))) pipeline for musical timbre transfer. arXiv preprint arXiv:1811.09620
39. Jaques N, Gu S, Turner RE, Eck D (2017) Tuning recurrent neural networks with reinforcement learning
40. Jeong J, Kim Y, Ahn CW (2017) A multi-objective evolutionary approach to automatic melody generation. Exp Syst Appl 90:50–61 (**Publisher: Elsevier**)
41. Jhamtani H, Berg-Kirkpatrick T (2019) Modeling self-repetition in music generation using generative adversarial networks. In: Machine learning for music discovery workshop, ICML
42. Jiang J (2019) Stylistic melody generation with conditional variational auto-encoder
43. Jiang J, Xia GG, Carlton DB, Anderson CN, Miyakawa RH (2020) Transformer VAE: a hierarchical model for structure-aware and interpretable music representation learning. In: ICASSP 2020—2020 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 516–520. ISSN: 2379-190X
44. Jie CHEN (2015) Comparative study between Chinese and western music aesthetics and culture
45. Jin C, Tie Y, Bai Y, Lv X, Liu S (2020) A style-specific music composition neural network. Neural Process Lett 52(3):1893–1912
46. Kaliakatsos-Papakostas M, Floros A, Vrahatis MN (2020) Artificial intelligence methods for music generation: a review and future perspectives. Nat Inspired Comput Swarm Intell, pp 217–245. Publisher: Elsevier
47. Kaliakatsos-Papakostas MA, Floros A, Vrahatis MN (2016) Interactive music composition driven by feature evolution. SpringerPlus 5(1):1–38 (**Publisher: Springer**)
48. Keerti G, Vaishnavi A, Mukherjee P, Vidya AS, Sreenithya GS, Nayab D (2020) Attentional networks for music generation. arXiv preprint arXiv:2002.03854
49. Kumar H, Ravindran B (2019) Polyphonic Music composition with LSTM neural networks and reinforcement learning. arXiv preprint arXiv:1902.01973
50. Leach J, Fitch J (1995) Nature, music, and algorithmic composition. Comput Music J 19(2):23–33 (**Publisher: JSTOR**)
51. Liang X, Wu J, Cao J (2019) MIDI-Sandwich2: RNN-based Hierarchical Multi-modal Fusion Generation VAE networks for multi-track symbolic music generation. arXiv:1909.03522 [cs, eess]. arXiv: 1909.03522
52. Lin P-C, Mettrick D, Hung PC, Iqbal F (2018) Towards a music visualization on robot (MVR) prototype. In: 2018 IEEE international conference on artificial intelligence and virtual reality (AIVR), pp 256–257. IEEE
53. Liu H-M, Yang Y-H (2018) Lead sheet generation and arrangement by conditional generative adversarial network. arXiv:1807.11161 [cs, eess]
54. Lopes HB, Martins FVC, Cardoso RT, dos Santos VF (2017) Combining rules and proportions: A multiobjective approach to algorithmic composition. In: 2017 IEEE congress on evolutionary computation (CEC), pp 2282–2289. IEEE
55. Loughran R, O'Neill M (2020) Evolutionary music: applying evolutionary computation to the art of creating music. Genet Program Evol Mach 21(1):55–85 (**Publisher: Springer**)
56. Lousseief E, Sturm BLT, Sturm BL (2019) MahlerNet: Unbounded Orchestral Music with Neural Networks. In: the Nordic sound and music computing conference 2019 and the interactive sonification workshop (pp. 57–63)
57. Lu C-Y, Xue M-X, Chang C-C, Lee C-R, Su L (2019) Play as you like: timbre-enhanced multi-modal music style transfer. Proc AAAI Conf Artif Intell 33:1061–1068
58. Luo J, Yang X, Ji S, Li J (2019) MG-VAE: Deep Chinese folk songs generation with specific regional style. arXiv:1909.13287 [cs, eess]
59. Makris D, Kaliakatsos-Papakostas M, Karydis I, Kermanidis KL (2019) Conditional neural sequence learners for generating drums' rhythms. Neural Comput Appl 31(6):1793–1804
60. Manzelli R, Thakkar V, Siahkamari A, Kulis B (2018) Conditioning deep generative raw audio models for structured automatic music. arXiv preprint arXiv:1806.09905
61. Manzelli R, Thakkar V, Siahkamari A, Kulis B (2018) An end to end model for automatic music generation: Combining deep raw and symbolic audio networks. In: Proceedings of the musical metacreation workshop at 9th international conference on computational creativity, Salamanca, Spain
62. Medeot G, Cherla S, Kosta K, McVicar M, Abdallah S, Selvi M, Newton-Rex E, Webster K (2018) StructureNet: inducing structure in generated melodies. In: ISMIR, pp 725–731
63. Mogren O (2016) C-RNN-GAN: Continuous recurrent neural networks with adversarial training. arXiv:1611.09904 [cs]

64. Mura D, Barbarossa M, Dinuzzi G, Grioli G, Caiti A, Catalano MG (2018) A soft modular end effector for underwater manipulation.: a gentle, adaptable grasp for the ocean depths. IEEE Robot Autom Mag , 4:1–1
65. Muñoz E, Cadenas JM, Ong YS, Acampora G (2014) Memetic music composition. IEEE Trans Evol Comput 20(1):1–15 (**Publisher: IEEE**)
66. Olseng O, Gambäck B (2018) Co-evolving melodies and harmonization in evolutionary music composition. In: International conference on computational intelligence in music, sound, art and design, pp 239–255. Springer
67. Oord Avd, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K (2016) WaveNet: a generative model for raw audio. arXiv:1609.03499 [cs]
68. Oore S, Simon I, Dieleman S, Eck D, Simonyan K (2020) This time with feeling: learning expressive musical performance. Neural Comput Appl 32(4):955–967
69. Payne C (2019) MuseNet.OpenAI Blog. https://openai.com/blog/musenet/. Accessed 11 Jan 2022
70. Plut C, Pasquier P (2020) Generative music in video games: state of the art, challenges, and prospects. Entertain Comput 33:100337 (**Publisher: Elsevier**)
71. Ramanto AS, No JG, Maulidevi DNU. Markov chain based procedural music generator with user chosen mood compatibility. In: Int J Asia Digital Art Des Assoc, 21(1):19–24
72. Rivero D, Ramírez-Morales I, Fernandez-Blanco E, Ezquerra N, Pazos A (2020) Classical music prediction and composition by means of variational autoencoders. Appl Sci 10(9):3053
73. Roberts A, Engel J, Raffel C, Hawthorne C, Eck D (2018) A hierarchical latent vector model for learning long-term structure in music. In: International conference on machine learning (pp 4364–4373). PMLR
74. Scirea M, Togelius J, Eklund P, Risi S (2016) Metacompose: A compositional evolutionary music composer. In: International conference on computational intelligence in music, sound, art and design, pp 202–217. Springer
75. Sturm BL, Ben-Tal O, Monaghan Ú, Collins N, Herremans D, Chew E, Hadjeres G, Deruty E, Pachet F (2019) Machine learning research that matters for music creation: a case study. J New Music Res 48(1):36–55 (**Publisher: Taylor & Francis**)
76. Sturm BL, Santos JF, Ben-Tal O, Korshunova I (2016) Music transcription modelling and composition using deep learning. arXiv preprint arXiv:1604.08723
77. Supper M (2001) A few remarks on algorithmic composition. Comput Music J 25(1):48–53
78. Tapus A (2009) The role of the physical embodiment of a music therapist robot for individuals with cognitive impairments: longitudinal study. In: 2009 Virtual rehabilitation international conference, pp 203–203. IEEE
79. Trieu N, Keller RM (2018) JazzGAN: Improvising with generative adversarial networks. In: MUME 2018: 6th international workshop on musical metacreation
80. Valenti A, Carta A, Bacciu D (2020) Learning style-aware symbolic music representations by adversarial autoencoders. arXiv:2001.05494 [cs, stat]
81. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser, Polosukhin I (2017) Attention is all you need. In: Adv Neural Inf Process Syst, pp 5998–6008
82. Veblen K, Olsson B (2002) Community music: toward an international overview. The new handbook of research on music teaching and learning, pp 730–753
83. Waite E, others (2016) Generating long-term structure in songs and stories. Web blog post. Magenta, 15(4)
84. Wang B, Yang Y-H (2019) PerformanceNet: score-to-audio music generation with multi-band convolutional residual network. Proc AAAI Conf Artif Intell 33:1174–1181
85. Williams D, Hodge VJ, Gega L, Murphy D, Cowling PI, Drachen A (2019) AI and automatic music generation for mindfulness, p 11
86. Wu C-W, Liu J-Y, Yang Y-H, Jang J-SR (2018) Singing style transfer using cycle-consistent boundary equilibrium generative adversarial networks. arXiv:1807.02254 [cs, eess]
87. Yang L-C, Chou S-Y, Yan Y-H (2017) Midinet: A convolutional generative adversarial network for symbolic-domain music generation. arXiv preprint arXiv:1703.10847
88. Yu Y, Srivastava A, Canales S (2021) Conditional LSTM-GAN for melody generation from lyrics. ACM Trans Multimedia Comput Commun Appl 17(1):1–20 arXiv:1908.05551
89. Zhang N (2020) Learning adversarial transformer for symbolic music generation. IEEE, Publisher, IEEE Transactions on Neural Networks and Learning Systems
90. Zhu H, Liu Q, Yuan NJ, Qin C, Li J, Zhang K, Zhou G, Wei F, Xu Y, Chen E (2018) XiaoIce Band: a melody and arrangement generation framework for pop music. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery and data mining, pp 2837–2846, London United Kingdom. ACM
91. Zhu J-Y, Park T, Isola P, Efros AA (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp 2223–2232
92. Zipf GK (2016) Human behavior and the principle of least effort: an introduction to human ecology. Ravenio Books
93. Hiller Jr LA, Isaacson LM (1957) Musical composition with a high speed digital computer. In: Audio engineering society convention 9. Audio Engineering Society
94. Cope D (1991) Recombinant music: using the computer to explore musical style. Computer 24(7):22–28
95. Miranda ER, Al Biles J (2007) Evolutionary computer music. Springer, Berlin
96. Wei S, Xia G (2022) Learning long-term music representations via hierarchical contextual constraints. arXiv:2202.06180 [cs, eess]
97. Guo R, Simpson I, Kiefer C, Magnusson T, Herremans D (2022) MusIAC: An extensible generative framework for music infilling applications with multi-level control. arXiv:2202.05528 [cs]
98. Dong H-W, Chen K, Dubnov S, McAuley J, Berg-Kirkpatrick T (2023) Multitrack music transformer. arXiv:2207.06983 [cs, eess]
99. Dubnov S, Chen K, Huang K. Deep musical information dynamics: novel framework for reduced neural-network music
100. Yu B, Lu P, Wang R, Hu W, Tan X, Ye W, Zhang S, Qin T, Liu T-Y (2022) Museformer: transformer with fine- and coarse-grained attention for music generation. arXiv:2210.10349 [cs, eess]
101. Zou Y, Zou P, Zhao Y, Zhang K, Zhang R, Wang X (2021) MELONS: generating melody with long-term structure using transformers and structure graph. arXiv:2110.05020 [cs, eess]
102. Schäfer T, Sedlmeier P, Städtler C, Huron D (2013) The psychological functions of music listening. Front Psychol 4:511
103. Ji S, Yang X, Luo J (2023) A survey on deep learning for symbolic music generation: representations, algorithms, evaluations, and challenges. ACM Comput Surv
104. Chrome Music Lab, Chrome's Song Maker. Accessed 22 Oct 2023, from https://musiclab.chromeexperiments.com/Song-Makerx
105. Aiva Technologies SARL. (Copyright 2016-2023). AIVA. Accessed 22 Oct 2023, from https://www.aiva.ai/
106. Choi K, Park J, Heo W, Jeon S, Park J (2021) Chord conditioned melody generation with transformer based decoders. IEEE Access 9:42071–42080. Conference Name: IEEE Access

107. Lee S-g, Hwang U, Min S, Yoon S (2018) Polyphonic music generation with sequence generative adversarial networks. arXiv:1710.11418 [cs, eess]
108. Mangal S, Modak R, Joshi P (2019) LSTM based music generation system. IARJSET 6(5):47–54 arXiv:1908.01080 [cs, eess, stat]
109. Shin A, Crestel L, Kato H, Saito K, Ohnishi K, Yamaguchi M, Nakawaki M, Ushiku Y, Harada T (2017) Melody generation for pop music via word representation of musical properties. arXiv:1710.11549 [cs, eess]
110. Wada Y, Nishikimi R, Nakamura E, Itoyama K, Yoshii K (2018) Sequential generation of singing F0 contours from musical note sequences based on WaveNet. In: 2018 Asia-Pacific signal and information processing association annual summit and conference (APSIPA ASC), pp 983–989. ISSN: 2640-0103
111. Matsue J (2015) Focus: music in contemporary Japan. Routledge
112. Mok AO (2014) East meets west: Learning-practices and attitudes towards music-making of popular musicians. Br J Music Educ 31(2):179–194
113. Nooshin L, Widdess R (2006) Improvisation in Iranian and Indian music. J Indian Musicol Soc 36:104–119
114. Son JH (2015) Pagh-paan's no-ul: Korean identity formation as synthesis of eastern and western music
115. Repetto RC, Pretto N, Chaachoo A, Bozkurt B, Serra X (2018) An open corpus for the computational research of arab-andalusian music. In: Proceedings of the 5th international conference on digital libraries for musicology, pp 78–86
116. Srinivasamurthy A, Gulati S, Repetto RC, Serra X (2021) Saraga: open datasets for research on indian art music. Emp Musicol Rev 16(1):85–98
117. Howard K (2016) Music as intangible cultural heritage: policy, ideology, and practice in the preservation of East Asian traditions. Routledge. Google-Books-ID: LYUWDAAAQBAJ
118. Carnovalini F, Rodà A (2020) Computational creativity and music generation systems: an introduction to the state of the art. Front Artif Intell, 3
119. Ji S, Luo J, Yang X (2020) A comprehensive survey on deep music generation: multi-level representations, algorithms, evaluations, and future directions. arXiv preprint arXiv:2011.06801
120. Donahue C, Mao HH, Li YE, Cottrell GW, McAuley J (2019) LakhNES: Improving multi-instrumental music generation with cross-domain pre-training. arXiv:1907.04868 [cs, eess, stat]
121. Simon I, Roberts A, Raffel C, Engel J, Hawthorne C, Eck D (2018) Learning a latent space of multitrack measures. arXiv:1806.00195 [cs, eess, stat]
122. Thickstun J, Harchaoui Z, Kakade S (2016) Learning features of music from scratch. arXiv preprint arXiv:1611.09827

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com