International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18 (2026) pp. 1-11

https://doi.org/10.70917/ijcisim-2026-0111
Article

Research on Automatic Creation Method of Documentary Background Music Based on Deep Generative Modeling

Hexi Wang and Mingjie Wang *

School of Art and Media, Beijing Normal University, Beijing 100091, China; wmjjmw12345@126.com

Abstract: Addressing issues such as lengthy production cycles and insufficient adaptability in the creation of background music for documentaries, this paper proposes an automatic composition method based on deep generative models. A source separation model based on RNN is designed to simultaneously process multi-track features in mixed audio. A bidirectional LSTM music generation model is constructed to leverage bidirectional temporal modeling capabilities to capture the global structural features of music, thereby optimizing the generation of melodies and chords. Experiments use classic documentary soundtrack segments as training data, and model performance is validated through spectral analysis, spectrogram comparison, and human subjective evaluation. Results show that after 2,000 iterations, the frequency distribution of the generated music converges with the sample music, and the bidirectional LSTM structure converges faster and produces better results than unidirectional models. In subjective evaluations, the model significantly outperformed the control model in terms of naturalness (4.35 points), creativity (4.52 points), and musicality (4.47 points), with a score difference of less than 0.5 points compared to real music.

Keywords: documentary background music; source separation model; music generation model; RNN; bidirectional LSTM

1. Introduction

Documentaries, with their authentic visual recordings, carry and convey information across various domains such as society, culture, and nature [1-2]. In the creation of documentaries, the importance of sound design cannot be overstated. It is not merely a simple overlay or splicing of sounds but an art form meticulously planned and designed [3]. Sound design not only enhances the expressive power of visuals, making the imagery more vivid and three-dimensional, but also deepens the theme, providing audiences with a more profound understanding and insight [4-5]. It not only elevates the creative standards of documentaries, making them more mature and refined in terms of artistic style and expression, but also enhances their dissemination effectiveness [6-8]. Among these elements, background music, as one of the most direct means of evoking emotions, plays an indispensable role in shaping the spiritual world and conveying inner emotions [9-10]. In many outstanding documentaries, the “documentary” principle is evident in the handling of background music, emphasizing not only technical aspects of pre-recording but also the flexible use of synchronized sound in post-production to strengthen the documentary consciousness of the overall creative process [11-14]. By employing music and sound effects that closely resemble the raw, natural elements of life, and under the backdrop of soothing, elongated background music, the integration of music and visuals is achieved, further deepening the theme [15-16].

Television documentaries, like other visual programs, unfold through the combined sensory experience of sight and sound. As a result, numerous scholars have studied methods for generating background music for video content. Literature [17] designed an automated music generation framework called VidMuse that aligns with input video. It uses video as a condition to generate high-quality, diverse, and coherent soundtracks consistent with video content through long-short term modeling. Literature [18]

indicates that the combination of music and video can exert a strong appeal on users. In response to the time-consuming process of music selection and the laborious task of music creation, it proposes the use of complex artificial intelligence architectures to automatically generate background music, thereby effectively complementing visual content. Literature [19] proposes a multi-modal music generation model based on the Diffusion Transformer (DiT), which supports the generation of music consistent with the semantic and emotional content of the input from both single-modal and multi-modal data. Literature [20] addresses the issue of background music generation for short videos, proposing the use of a cross-modal recommendation algorithm (CMVAE) to match video content with relevant music, while employing a weighted fusion method to enhance the model's generalization performance and effectiveness. Literature [21] applies a learning-based method to generate raw waveform samples given input video frames, thereby producing natural and realistic sound effects for video input data. Although the music generated by the methods in the above literature has achieved good results in terms of sound quality, and the music played by various instruments has a certain degree of coherence and structure, it cannot meet the requirements of background music design principles, techniques, and their impact on the overall effect of the work in documentaries, and further research is still needed.

This paper combines the dual requirements of background music source separation and music generation in documentaries, designing an RNN-based source separation model to address the efficient separation of multiple audio tracks in mixed audio. A bidirectional LSTM music generation model is constructed, with detailed discussions on model architecture, training methods, and parameter optimization. Experimental analysis evaluates the model's performance, comparing generation results across different iteration counts. Spectrogram and spectrogram analyses are used to examine the experimental effects of different network structures on music sequence generation. Control experiments are set up, and subjective evaluations are combined to validate the quality of the music generated by the model.

2. Design of a Deep Learning-Based Automatic Music Generation Model

In the artistic expression of documentaries, background music is one of the core elements for conveying emotions, reinforcing narrative logic, and shaping the audience's immersive experience. Traditional creative models heavily rely on the subjective compositions of professional composers, which have limitations such as long production cycles, high costs, and difficulty in quickly adapting to different visual content and emotional tones. With the rapid development of artificial intelligence technology, music generation methods based on deep learning have provided a new approach to addressing this issue. However, existing music generation models are primarily designed for general scenarios such as pop music and classical music, and lack sufficient adaptability to special scenarios like documentaries, which have clear narrative directions and strong emotional orientation. The melodies generated often lack emotional synergy with the visual content, failing to meet professional creation requirements.

To address these challenges, this paper focuses on the task of automatically creating background music for documentaries and proposes a method based on deep generative models.

2.1. RNN Sound Source Separation Model

The model uses the amplitude spectrum of the audio signal as feature input. In each training cycle t, the model receives the amplitude spectrum of the mixed signal and the amplitude spectra of its target sound sources B_t (bass), D_t (drums), M_t (music), and V_t (vocals). In the output of cycle t, the model generates the estimated signals for each sound source, namely $\hat{B}_t, \hat{D}_t, \hat{M}_t$ and $\hat{V}_t$. The RNN model structure used for sound source separation is shown in Figure 1.

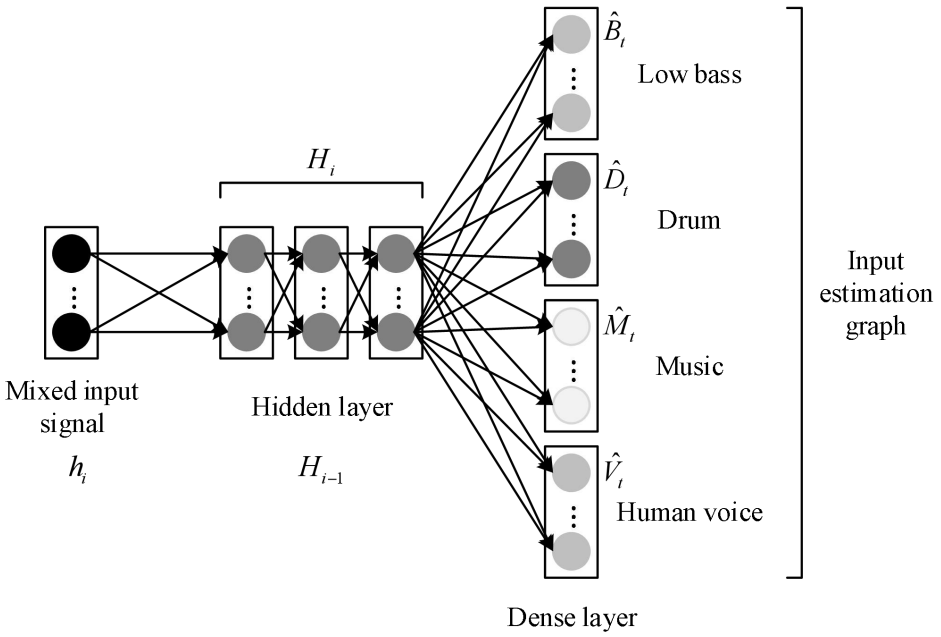

Figure 1. Source Separation RNN model.

In order to perform source separation more efficiently, the model is trained to process all constituent sound sources simultaneously, rather than creating separate networks for each sound source. During training, audio files selected from the dataset and their respective component audio files are read as time-series audio signals. These signals are segmented based on the specified sampling rate (S_t) and duration (T), generating data segments of length $S_t \times T$. For each segment of each sound source, the short-time Fourier transform (STFT) is used to calculate the frequency domain counterpart of the time domain signal. Based on observations, the window size and shift size of the STFT are set to 1024G and 512G, respectively, in this dataset. This processing enables the model to effectively process and separate different components in the audio, thereby improving separation quality and accuracy.

During training cycle t, each entry in a batch consists of the mixed input signal h_t and the individual sound sources B_t (bass), D_t (drums), M_t (music), and V_t (vocals). The mixed signal is fed into a multi-RNN unit comprising three gated recurrent unit (GRU) layers, each with 256 units. This multi-RNN unit is connected to a dense layer for each source. The dense layer uses the rectified linear unit (ReLU) as the activation function and outputs the estimated signals for each sound source, labeled as $B_t^{'}, D_t^{'}, M_t^{'}$ and $V_t^{'}$, respectively. The estimated signals are further processed to optimize the masking function, as shown in Equations (1) to (4):

$$\hat{B}_t = \frac{B_t^{'}}{C_t^{'}} \odot h_t \tag{1}$$

$$\hat{D}_t = \frac{D_t^{'}}{C_t^{'}} \odot h_t \tag{2}$$

$$\hat{M}_t = \frac{M_t^{'}}{C_t^{'}} \odot h_t \tag{3}$$

$$\hat{V}_t = \frac{V_t^{'}}{C_t^{'}} \odot h_t \tag{4}$$

In the formula, $C_t^{'} = B_t^{'} + D_t^{'} + M_t^{'} + V_t^{'}$ represents the amplitude spectrum of the combined estimate, and © is the element-wise multiplication operator, also known as the Hadamard product.

2.2. LSTM Music Training Generation

2.2.1. Data Representation

For LSTM, let the input be represented as a vector V, which consists of two heat vectors M and C, representing melody and chords, respectively, then:

$$V = (M, C) \tag{5}$$

A piece of music is represented by a $T \times 180$ dimensional vector (155 dimensions for melody and 25 dimensions for chords), where T represents the number of time steps. In the melody vector M, the 155 dimensions include pitch, duration, and cosine sign. Using hold state and rest state, the first 153 dimensions of M represent the pitch values from 0 to 152. Dimension 154 is the rest state, meaning the note is empty. The last dimension is the hold state, representing the duration of the previous pitch.

Similar to the melody vector M, the first 24 dimensions of the chord vector C represent the most common primary and secondary chords, without considering inversions (i.e., different root notes within a chord). The last dimension is the no chord symbol NC. For chords not among the 24 most common chords, they are matched with one of the most common chords with the same pitch. For example, C-major7 (C7) matches C major, C-minor7 (Cm7) matches C minor, and C-augment (Caug) matches C major. The chord vector C does not have a state, as melody generation is more related to chord values than duration.

For the WaveNet model, the dimension of the melody vector M is $T \times 128$. Each channel represents a pitch, with the lowest pitch 0 indicating a rest. If the pitch value is the same in two or more consecutive time steps, it is considered a sustained note. For the conditional vector, all chords are hashed into 24 types, and the chord input is a hot vector on the hash table. Each input chord vector C has a shape of $T \times 25$ (with an additional channel representing NC), where the number of time steps T is the same as the corresponding melody vector.

2.2.2. LSTM Architecture

In the proposed LSTM model, we developed a bidirectional model from a unidirectional model and adjusted the data to fit both models.

LSTM consists of four gates: cell gate c, input gate i, output gate o, and forget gate f, whose relationships are shown in Equations (6) to (10):

$$f_t = \sigma_g(W_f x_i + U_f h_{i-1} + b_f) \tag{6}$$

$$i_t = \sigma_g(W_i x_i + U_i h_{i-1} + b_i) \tag{7}$$

$$o_t = \sigma_g(W_o x_i + U_o h_{i-1} + b_o) \tag{8}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \tag{9}$$

$$h_i = o_i \odot \sigma_h(c_i) \tag{10}$$

The detailed structure of the bidirectional LSTM model is shown in Figure 2. The left side is the input layer, followed by seven bidirectional LSTM layers and a fully connected layer. To speed up convergence, skip connections are used in all recurrent layers except the first layer.

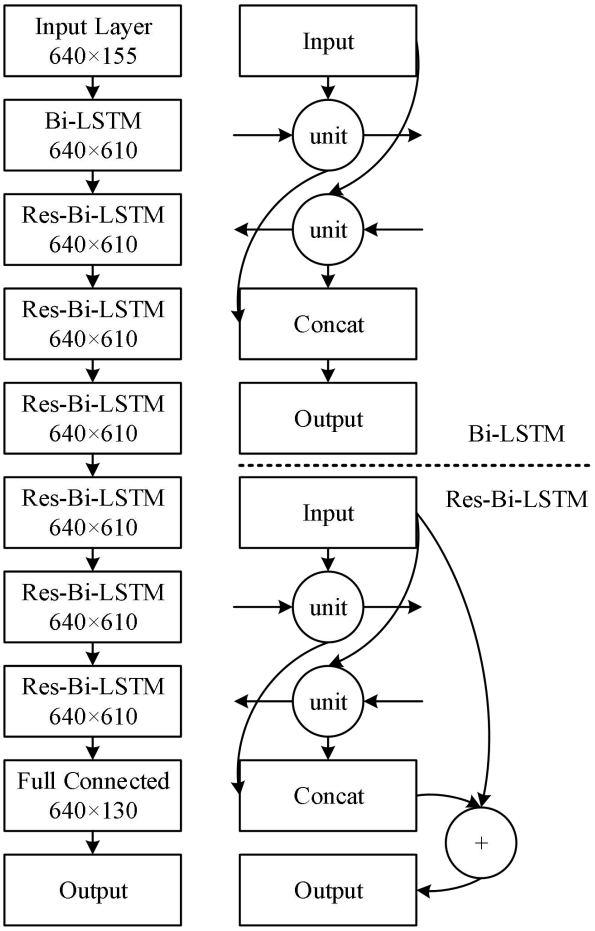

Figure 2. Bidirectional LSTM model.

The difference between unidirectional and bidirectional LSTM models lies in the amount of chord information. In unidirectional models, only the preceding chord progression is used when generating the current note. However, bidirectional models allow the entire chord sequence (i.e., the global structure) to be added to the generation process. In this case, the conditional probability of the bidirectional LSTM model should be modified to:

$$p(M_{T-i+1,T} \mid C_{1,T}) = \prod_{i=1}^{T-i} p(m_i \mid m_1, \cdots, m_{i-1}, .c_1, \cdots, c_r) \tag{11}$$

It can be noted that the string conditions $c_1, \cdots, c_r$ differ from the unidirectional formula $c_1, \cdots, c_i$. The bidirectional model is based on the complete string series.

2.2.3. Model Training

The training process for LSTM will be conducted using the Python programming language. The input and output of the note sequences will be processed using one-hot encoding and decoding. The libraries used during training include TensorFlow V1.5, Keras V2.1.5, Music21 V5.2.0, Torch V1.8.0, and Mido V1.2.0, among others.

The training process generates new musical works. First, the MIDI files of documentary music are decomposed into multiple segments as defined by the program. Based on experience, each segment of the musical sequence is set to a length of 100. Additionally, considering that the first 8 notes of each musical work lack contextual information, these notes are not included in the network. For the LSTM network model, the algorithm is implemented using the deep learning framework "Keras" in Python. First, a sequential neural network is created. Then, the network layers are configured using the cross-entropy loss function and RMSProp optimization method. The final parameter values are as follows: the hidden layer 1 has 512 input neurons, the hidden layer 2 has numpitch, Dropout is 25%, the activation function is "softmax", epochs=2000, batch_size=512, loss='categorical_crossentrop', optimizer='RMSProp', learning_rate=0.0005. The output sequence length is set, and within the sequence length, the output state

of the LSTM network is converted into the probability of each note, and the output is calculated using the loss function. Since the generated music often lacks melody, the chorus melody is extracted from the training sample fragments and mixed with the original documentary music as training samples. The LSTM model is then run again to obtain the final results.

For the note generation part, there are chord and single-note generation results. In MIDI format, the generated chords are similar to [45.21.78], where each note in the chord is separated by "•." Each note is extracted and converted into an integer, then converted into note format at the corresponding midi_number and placed into the note stack. The notes in the stack are then converted into new chords, The converted chords are then sent to the music output. For single-note generation results, the data is directly converted into new notes. To prevent cross-over in each iteration, the offset for each iteration is set to 0.5.

RNN models are prone to the vanishing gradient problem, which prevents the loss from decreasing. At the same step size, the final loss is not significantly different from the initial loss, resulting in underfitting and preventing the attainment of an optimal trained model. Additionally, based on the generation results of the final RNN model, most music segments exhibit gaps, further validating the issues with the RNN model. For RNN networks, the model may require a simpler structure. In contrast, the LSTM model achieves convergence at the same stride length, demonstrating that LSTM can learn more content than RNN in time-series data training and achieve better results.

For automatic composition of other instruments or music, this paper recommends using a simpler RNN network with a single-layer or double-layer structure for small datasets. If overfitting is observed during training, the number of layers in the network structure can be appropriately increased. If the RNN still fails to achieve convergent training results, it is recommended to replace the RNN network with an LSTM or GRU network during training. The gated recurrent unit (GRU) is another variant of the RNN network. Its gating system achieves a faster convergence rate compared to LSTM, making it an excellent alternative. The LSTM network structure proposed in this paper can also serve as the initial network structure for future training on datasets of the same size.

2.2.4. Parameter Design

During parameter discussions, all parameters except those under discussion will be fixed. The number of input segments randomly selected is referred to as the batch size. The larger the batch size, the better the LSTM can learn, but the overall computational load of the network increases, requiring more time.

In addition to the impact of batch_size on network training, LSTM has many other parameters, such as the learning rate. As a hyperparameter controlling the learning speed of various networks, the learning rate is also an essential parameter setting in LSTM. If the learning rate is set too low, it will slow down the model's convergence speed. Conversely, if it is set too high, oscillations may occur during training, preventing the LSTM network from converging. To achieve a good learning rate, a larger learning rate is typically used at the beginning of LSTM training to accelerate the training speed of the network. As the network approaches the minimum loss point, the learning rate is gradually reduced to bring the network closer to the convergence point. In the training process of this model, the learning rate was initially set to 0.5, with a momentum of 0.65 to exponentially decay the learning rate every 100 steps, ultimately setting the learning rate to $\alpha = 0.48 \times 10^{-3}$.

Based on the parameter discussion, for automatic composition of other instruments or music, in the case of small data samples, this paper suggests that the initial training parameters can be set as follows: Dropout = 15%, activation function = "softmax", epochs = 1200, batch_size = 1024, loss = 'categorical_crossentropy', optimizer = 'RMSProp', learning rate = 0.0005. Adjust these parameters appropriately based on training results. If overfitting occurs, you can not only reduce the number of network layers but also increase the Dropout rate, for example, from 5% to 25%. If you need to accelerate training, it is recommended to increase the batch size and learning rate appropriately, such as adjusting the batch size from 256 to 1024 and increasing the learning rate by a factor of ten as a reference.

3. Analysis and Comparison of the Performance of Music Generation Model Experiments

This paper implements music sequence modeling based on the popular deep learning frameworks Torch7 and Lua. MP3 files of classic documentary soundtracks are selected as training samples, and the dataset after data preprocessing is approximately 750 MB.

3.1. Comparison of Experimental Results after Different Numbers of Iterations

First, we verify the convergence of the model for the music data processed in this paper. For music files generated with different iteration counts, we perform spectral analysis on the generated music and sample music. The spectral sequences of the sample music are labeled as S, while those generated after 500, 1000, and 2000 iterations are labeled as G1 to G3. The comparison results are shown in Figure 3. At 500 iterations, the generated music sequence contains many frequencies not present in the sample music, resulting in a disorganized music sequence. By 1000 iterations, the frequency distribution generally aligns with that of the sample music, though some frequencies from the sample music sequence are missing in the generated frequencies. By the 2000th iteration, the frequency distribution of the generated music sequence has largely aligned with that of the sample music. As the number of iterations increases, the frequency of learning and adjusting the weight parameters also increases, which to some extent improves the model's accuracy.

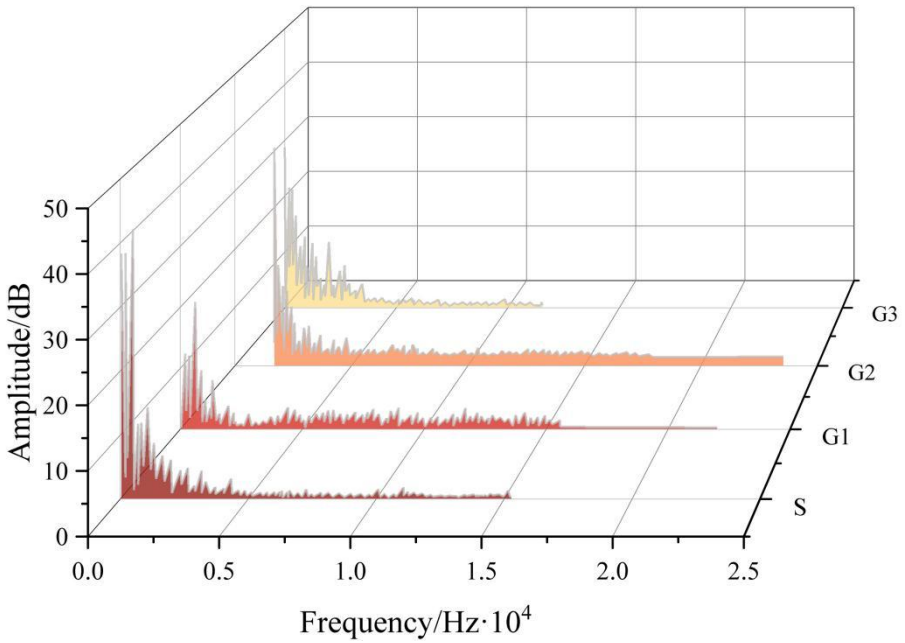

Figure 3. Generates the music spectrum.

3.2. Comparison of Experimental Results under Different Network Structures

3.2.1. Spectrum Analysis

To ensure the fairness of the experiment, the process was iterated 2,000 times. A portion of the sample sequence was extracted, and after Fourier transformation, the spectral diagrams of the sample music and generated music were obtained. The comparison results of the spectral diagrams are shown in Figure 4. It can be seen that the overall frequency distribution ranges of the generated music sequence and the sample music sequence are basically consistent, indicating that the LSTM network has learned the contextual relationships in the sample music time series and can generate music of a similar style. In the experiment, it was found that the bidirectional LSTM network did not perform as well as other network structures in terms of music generation after 2,000 iterations. Therefore, additional experiments were conducted with 1,000 and 3,000 iterations for the bidirectional LSTM network. The conclusion was that the music generated after 1,000 iterations had the best quality. During the experiments, it was observed that the bidirectional LSTM required longer training time per iteration, but it could converge in fewer iterations and also produced better music generation results. This is because the bidirectional LSTM network includes a forward LSTM and a backward LSTM. The forward LSTM captures feature information from the preceding text, while the backward LSTM captures feature information from the subsequent text. Compared to a unidirectional LSTM, this allows for the capture of more feature information, enabling the expression of richer musical information. Therefore, under the same training set conditions, fewer iterations are required to capture sufficient feature information and generate high-quality music sequences.

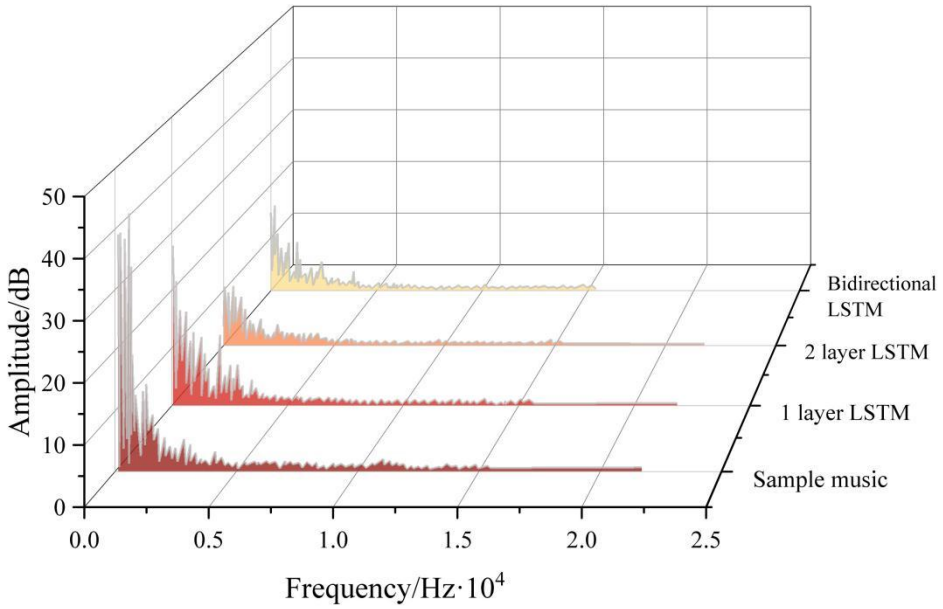

Figure 4. Comparison result of the spectrogram.

3.2.2 Spectrogram Analysis

The spectrograms of the generated music and sample music were extracted and analyzed under 2,000 iterations of a single-layer LSTM. The comparison results between the spectrograms of the generated music and sample music are shown in Figure 5. The spectrograms of the generated music and the sample music are very similar, with a difference of approximately 50–100 Hz. This indicates that the differences between the two are minimal. Therefore, it can be verified that the music automatically generated by the model in this paper possesses melody features similar to those of the sample music.

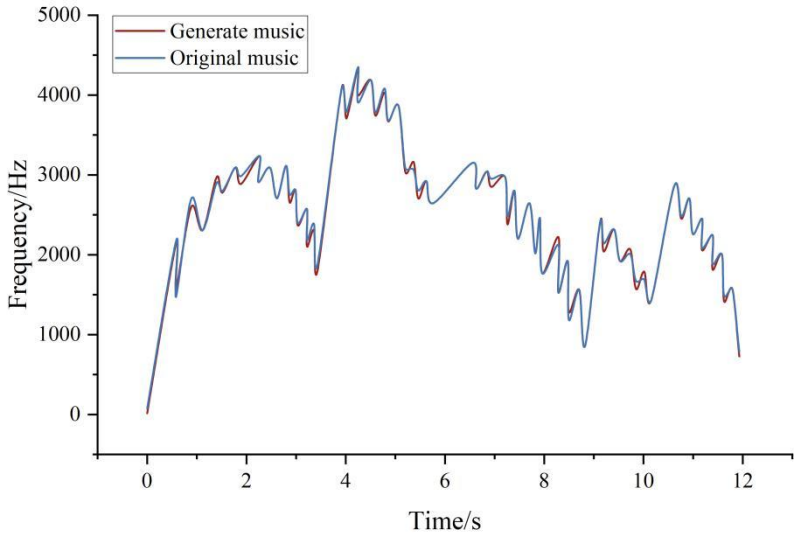

Figure 5. Comparison results of the spectrogram.

3.3. Analysis of Music Generation Results

3.3.1 Analysis of Results

Select existing music segments, extract the melody segments of the existing music through an encoder, and then input them into the model to generate music after making detailed adjustments. The manually drawn melody progression and the melody progression of the generated music are shown in Figure 6. It can be seen that the music generated by the model has tonality, with the generated music

being in D minor, and the melody is richer, indicating that the model has the ability to generate AI-generated music works with a certain degree of musicality.

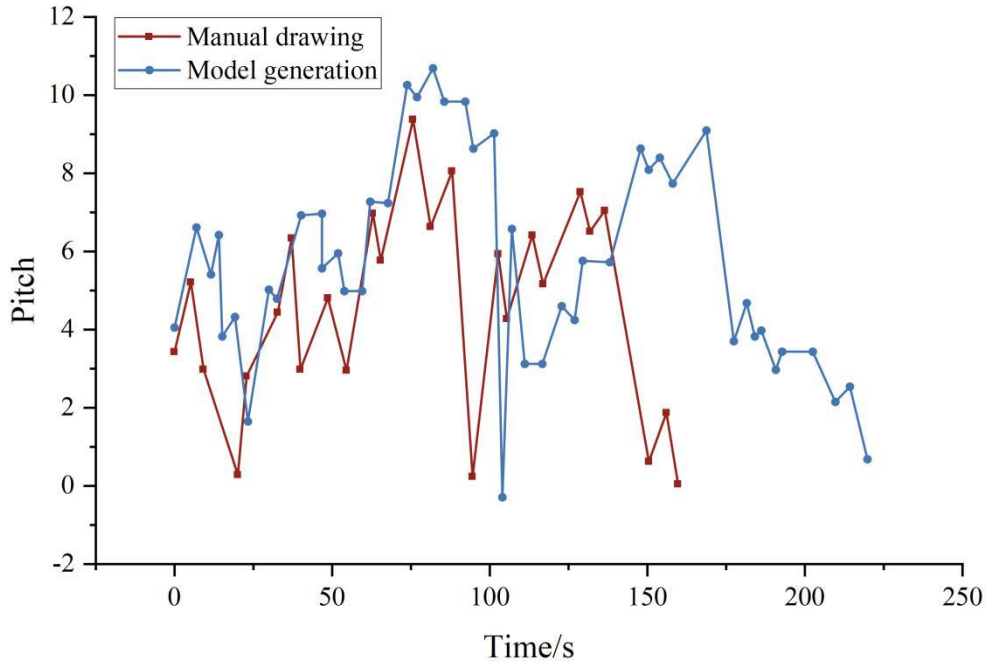

Figure 6. Comparison of melody trends.

3.3.2. Control Trial

Since musical works are related to artistic aesthetics, human sensory evaluation is also a very important criterion for assessment. This paper adopts a human evaluation method, using naturalness, creativity, and musicality to evaluate the artistic value of musical works. Naturalness refers to whether the work sounds harmonious and natural. It is used to measure the degree of alignment between music and human auditory experience. For example, whether it can stimulate the listener's senses, whether it can evoke natural emotions in the listener, and whether it has tonality, etc. Creativity refers to whether the work is interesting and has unexpected novelty. It is used to measure whether the music generated by the neural network is innovative, avoiding the production of monotonous and boring works. Musicality refers to whether the work is subjectively pleasant to the ear and conforms to natural music theory. It can measure whether the music conforms to the natural major scale and is also a key factor in determining whether the model has learned the complex structural relationships of the musical data attributes.

This study invited 20 volunteers to score each piece of music based on naturalness, creativity, and musicality. Each volunteer listened to three groups of five melodies each and provided subjective evaluations on a scale of 1 to 5. Bi-LSTM, EC2-VAE, and VD-ED were used as control models. The results of the subjective metric comparisons between the proposed model, control models, and real documentary music are shown in Figure 7. Analysis revealed that the music pieces generated by the proposed model outperformed other models in terms of naturalness, creativity, and musicality. The average scores for the proposed model were 4.35, 4.52, and 4.47, respectively, surpassing the second-best VD-ED model by 26.45%, 24.86%, and 26.27%, respectively. The difference from the real music scores was within 0.5 points, indicating that the proposed model can generate music works that better align with human auditory experience.

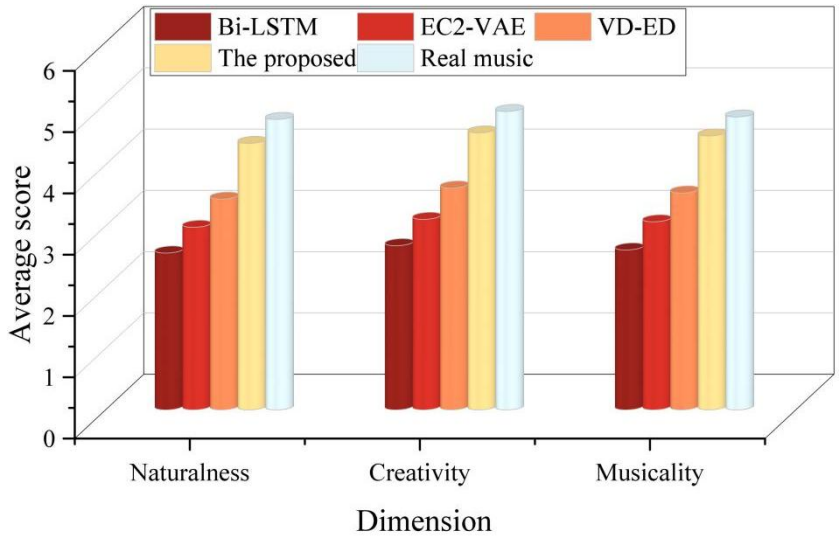

Figure 7. Comparison results of subjective indicators.

4. Conclusion

This paper addresses the unique requirements of creating background music for documentaries by proposing an automatic composition method based on deep generative models. The main experimental conclusions are as follows.

By the 2000th iteration, the frequency distribution of the generated music sequences generally aligns with that of the sample frequency distribution. While the bidirectional LSTM requires longer training time per iteration, it achieves convergence in fewer iterations and produces better music quality, reaching optimal results at 1000 iterations. Under 2,000 iterations, the spectrograms of music generated by the single-layer LSTM are very similar to those of the sample music, with a difference of approximately 50–100 Hz, indicating minimal disparity between the two. Additionally, the music generated by the model exhibits tonality, being in the key of D minor, with richer melodies.

The music generated by the model in this paper outperforms other models in terms of naturalness, creativity, and musicality. The average scores for the model in this paper are 4.35, 4.52, and 4.47, respectively, surpassing the second-best VD-ED model by 26.45%, 24.86%, and 26.27%, respectively. The difference in scores compared to real music is within 0.5 points, indicating that the model in this paper can generate music that aligns more closely with human auditory experience.

References

1. Poulakis, N., & Stamatatou, C. (2022). Music, Documentaries and Globalization: From World Music to World Cinema. Popular Music Research Today: Revista Online de Divulgación Musicológica, 4(2), 7-21.
2. Carrillo Quiroga, P. (2024). Planning scientific documentary production for social impact, a practice-based approach. Media Practice and Education, 1-20.
3. Heise, T. S. (2018). Sounds from Brazil: brasilidade and the rise of the music documentary. In Screening songs in Hispanic and Lusophone cinema (pp. 249-263). Manchester University Press.
4. Leimbacher, I. (2017). Hearing voice (s): experiments with documentary listening. Discourse, 39(3), 292-318.
5. Gómez, L. L. (2023). Audiovisual Warfare: Music and International Persuasion in Documentary Films during the Spanish Civil War. Twentieth-Century Music, 20(2), 244-260.
6. Rangan, P. (2017). Audibilities: Voice and Listening in the Penumbra of Documentary: An Introduction. Discourse, 39(3), 279-291.
7. Birtwistle, A. (2016). Electroacoustic composition and the British documentary tradition. The Palgrave Handbook of Sound Design and Music in Screen Media: Integrated Soundtracks, 387-402.
8. Wöllner, C., Hammerschmidt, D., & Albrecht, H. (2018). Slow motion in films and video clips: Music influences perceived duration and emotion, autonomic physiological activation and pupillary responses. PLoS One, 13(6), e0199161.
9. Park, D. C. (2017). Effect of background music of TV documentary on audience's recall memory, flow, arousal of interest, evaluation. Journal of Digital Convergence, 15(10), 411-417.
10. Nosal, A. P., Keenan, E. A., Hastings, P. A., & Gneezy, A. (2016). The effect of background music in shark documentaries on viewers' perceptions of sharks. PLoS One, 11(8), e0159279.
11. Biewen, J., & Dilworth, A. (Eds.). (2017). Reality radio: telling true stories in sound. UNC Press Books.

12. Dowling, D. O., & Miller, K. J. (2019). Immersive audio storytelling: Podcasting and serial documentary in the digital publishing industry. Journal of radio & audio media, 26(1), 167-184.
13. Chovanec, J. (2017). Interactional humour and spontaneity in TV documentaries. Lingua, 197, 34-49.
14. Aston, J. (2017). Interactive documentary and live performance: From embodied to emplaced interaction. In I-Docs: The Evolving Practices of Interactive Documentary (pp. 222-236). Columbia University Press.
15. Yang, P. (2022). Sound Art and Pandemic: A Documentary Soundscape. Symbolon, 23(Special), 37-45.
16. Cox, G., Corner, J., Berkenhoff, A., Brereton, J., Bulley, J., & Connor, S. (2018). Auralising Action Space: channelling a sense ofplay in documentary sound design. In Soundings: Documentary film and the listening experience (pp. 86-103). University of Huddersfield Press.
17. Tian, Z., Liu, Z., Yuan, R., Pan, J., Liu, Q., Tan, X., ... & Guo, Y. (2025). Vidmuse: A simple video-to-music generation framework with long-short-term modeling. In Proceedings of the Computer Vision and Pattern Recognition Conference (pp. 18782-18793).
18. Bondapalli, A., Chokda, H., & Sudha Rani, K. M. (2024). Video Background Music Generator. Grenze International Journal of Engineering & Technology (GIJET), 10.
19. Zheng, J., Cao, M., & Zhang, C. (2025). VT2Music: A Multimodal Framework for Text-Visual Guided Music Generation and Comprehensive Performance Analysis. IEEE Access.
20. Yi, J., Zhu, Y., Xie, J., & Chen, Z. (2021). Cross-modal variational auto-encoder for content-based micro-video background music recommendation. IEEE Transactions on Multimedia, 25, 515-528.
21. Zhou, Y., Wang, Z., Fang, C., Bui, T., & Berg, T. L. (2018). Visual to sound: Generating natural sound for videos in the wild. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3550-3558).