Hindawi
Mobile Information Systems
Volume 2022, Article ID 2376888, 11 pages
https://doi.org/10.1155/2022/2376888

Research Article

Music Genre Classification Based on Deep Learning

Wenlong Zhang

School of Music and Dance, Yantai University, Yantai 264005, China

Correspondence should be addressed to Wenlong Zhang; 201001001730@ytu.edu.cn

Received 10 June 2022; Revised 3 July 2022; Accepted 12 July 2022; Published 21 August 2022

Academic Editor: Muhammad Zakarya

Human music life can be traced back to ancient times. The music art of human society is rich and colorful, which makes the music classification unable to classify efficiently and accurately. Moreover, the classification has become a daunting task. On this basis, this paper studies the method of deep learning for processing music classification. Not only is the design structure of music signal channel classified, but also all connected neural networks associated with the music are investigated to design an appropriate network model. According to different music sequence measurements, the feature sequence mechanism of music design feedback optimization is also investigated. The type probabilities of different calculated orbits are measured by *softmax* activation function, and the function value of cross loss is obtained. Finally, an Adam optimization algorithm is used as the optimization algorithm of the proposed network model. Subsequently, an independent adaptive learning planning rate is designed. By adjusting the network parameters, the first- and second-order estimates of the calculated gradient are classified. The experimental outcomes prove that the anticipated method can meritoriously increase the correctness of music classification and is helpful for music channel classification. Moreover, we also observed that the number of neurons in the network has also a significant impact over the training and testing errors.

1. Introduction

The creation and performance of early popular music were mostly commercial, and it was carried out in cities and towns, which was different from folk music with strong rural color. At the same time, it does not have the standardization and stability of art music [1]. These in early days, in many cases, were just oral. Therefore, some people say that popular music is different from art music and folk music. This, in fact, generally refers to a kind of music that is easy to understand, relaxed and lively, and easy to spread and has a large audience. Some people say that some particular music is "popular music" [2]. Music genre is an important label to describe music. Music tags play a virtuous part in pinpointing and separating digital music resources [3]. Therefore, from a huge amount of musical data, their identification and classification have become more daunting. Facing the enormous music catalogue, depending on manual explanation for classification will devour significant computational costs, resources, and time. Moreover, we believe that they will still not be able to meet the needs of the current times enriched by big data, Internet of things, and people's increasing interest in music. Therefore, music classification has gradually become a research hotspot.

At present, scholars in related fields have made theoretical research on the classification of music themes. For example, the authors in [4] proposed an engine system for classifying genres, which aims to replace these features by a new model. The model can also recommend music from vocal music that has been extracted from online music. Their experimental results show that this method not only has certain efficiency but also can effectively modulate speech pitch and construct separation masking based on neural recursion. It should be kept in mind that the voice signals mixed with music can be screened and deleted. The music pitch classification method based on the RNN model can improve the time trajectory of speech and music pitch values. Moreover, this can also determine that the unknown continuous pitch sequence belongs to speech or music. This method has significant classification performance without losing speech noise separation performance. Nevertheless, the previously mentioned approaches still have some

complications, such as low classification precision, poor effect, and lengthy computational time.

In order to solve the above complications, a classification method of music genres based on deep learning is proposed in this paper. Using deep learning, the data preprocessing is used to filter the music signals. Furthermore, using a fully connected neural network structure, the extraction of music genre features is completed. Finally, the attention mechanism is used to design a music genre classification network model. The music genre classification effect of the suggested method is better than those of other approaches, which can effectively improve the classification accuracy of the music genre. Moreover, our approach shortens the classification time significantly. The main contributions are as follows:

(i) We study the classification of the design structure of music signal channel, and the connected neural network associated with music is designed.

(ii) According to different music sequence measurements, the feature sequence mechanism of music design feedback optimization is studied.

(iii) The type probabilities of different calculated orbits are measured by softmax function, and the function value of cross loss is obtained.

(iv) Finally, an Adam optimization algorithm is used as the optimization algorithm of network model, and an independent adaptive learning planning rate is designed.

The remainder of the paper is organized as follows. In Section 2, we briefly discuss the basic theory of deep learning. Neural and back-propagation (BP) networks along with activation functions are discussed. In Section 3, fundamentals of music signal analysis are illustrated. In the fourth section, we discuss the classification of music genres and feature extraction and propose a neural network model. Experimental discussion and results are presented in Section 5. Finally, Section 6 summarizes the paper and presents directions for future research.

2. Basic Theory of Deep Learning

Deep learning is a branch of machine learning that deals with learning algorithms using deep neural networks. In fact, deep learning methods are developed from artificial neural networks (ANNs). It should be noted that ANNs are the most commonly used and representative model structure in the field of machine learning. Deep neural network (DNN) is a neural network, which is formed from the interconnection of various neurons and weights and may have many hidden layers and neurons [5]. Deep learning can learn higher-level feature expression from complex and large samples.

2.1. Neural Networks. Deep learning is developed from artificial neural networks. Furthermore, neural networks are abstracted from the structure of biological neural networks. In the network, information is transmitted and activated through the interconnection between basic units, known as neurons, which in fact imitates the process of information

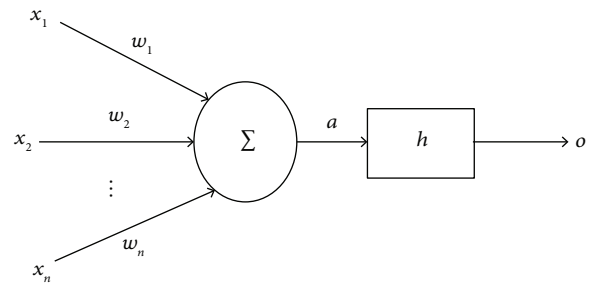

Figure 1: The neuron structure diagram.

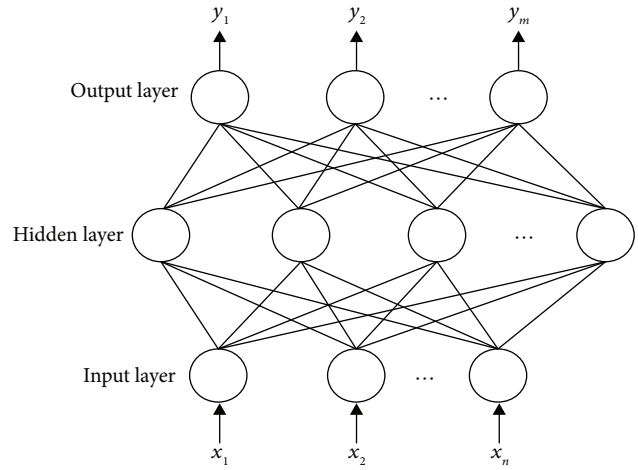

Figure 2: Structure diagram of a fully connected neural network.

transmission between the biological neurons [6]. The basic unit of the neural network is called neuron, and several neurons are connected with each other in such a way that communications occur among them [7]. The basic structure of the neuron is as shown in Figure 1.

In Figure 1, x_1 is the input signal, and the arrow starting from the input signal represents the connection. Each connection corresponds to a particular weight w_1. After the input signal passes through these connections, it is weighed and summed to obtain a (a usual output of the hidden neurons). Finally, the previous output goes through a nonlinear function in order to get output o. It should be noted that the nonlinear function h is called the activation function that is used to tune the performance of the network [8]. The process of neuron input to output can be described by mathematical expression as follows:

$$o = h\left(\sum_{i=1}^{n} x_i w_i + b\right). \tag{1}$$

In formula (1), b is the bias term of the neuron. Multiple neurons with the same inputs form a hidden layer. The input of one layer of neurons is used as the input of the next layer of neurons, and the basic neural network is formed according to this connection method. The input of a neuron can come from either the input signal or the output of other neurons [9]. The structure of the fully connected neural network is shown in Figure 2.

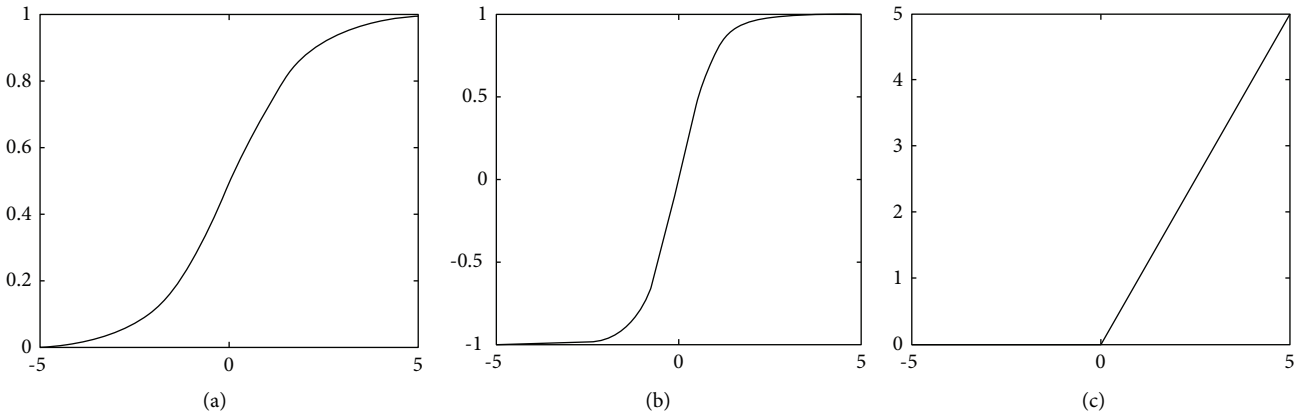

Figure 3: Three activation functions and images. (a) Sigmoid. (b) tanh. (c) ReLU.

From bottom to top, as shown in Figure 2, the input layer takes inputs, passing through several neuron layers, and the output layer creates the output. The network structure, in Figure 2, has only one hidden layer, and this type of neural network is also called a single hidden layer feedforward neural network. In deep learning, multiple hidden layers can also be set, and each hidden layer is set with a different number of neurons according to the actual situation to improve the learning capability. The connection weight matrix of each layer and the previous layer is multiplied by the output value of the neuron of the previous layer, and the bias term of this layer is added to obtain a linear output. Subsequently, the obtained linear output then passes through the activation function of this layer performing nonlinear transformation to get the output of this layer of neurons [10]. The process of neurons in each layer from receiving input to calculating output can be described by a calculation formula as follows:

$$\begin{aligned} z^l &= W^l a^{l-1} + b^l \\ a^l &= f^l\left(z^l\right). \end{aligned} \tag{2}$$

In formula (2), z^l is the linear output vector of neurons in layer l, which is calculated from the output vector a^{l-1} of neurons in layer $l-1$, the connection weight matrix W^l of layer l, and the bias term b^l of layer l. Furthermore, a^l is the nonlinear output vector of the l layer neuron obtained by the linear output z^l of layer l neuron through the activation function $f^l(\cdot)$ of layer l.

Let us again refer to the basic architecture of the neural network, as shown in Figure 2, starting from the input layer, along the direction from input to output. For example, according to the above process, a series of linear and activation operations are carried out for the input vector, connection weight matrix, and offset term of each layer [11]. All these parameters are calculated layer by layer until the target prediction result is obtained at the output layer. This process is a forward propagation process.

2.2. Back-Propagation (BP) Algorithm. The input layer, hidden layer, and output layer are the three components that make up the front end, middle, and end of the BP neural network. It is assumed that $x0 = -1$; the beginning of the imported input is the input vector, whose formula is $x = (x1, x2, \ldots, xi, \ldots, xn)T$; the middle of the neural network is the hidden layer, which will slow down training. The output vector is the result of the generated data, and its formula is $y = (y1, y2, \ldots, yi, \ldots, yn)T$. $y0 = -1$ can be provided as an additional assumption. The algorithm is a part of a unique programme, and, right now, one of the most cutting-edge fields is neural network. The result of combining the two is BP neural network. The topology of the BP neural network is shown in Figure 3. This research employs the modified BP neural network model to evaluate music classification, which can successfully eliminate the difficulties of instability and slow convergence of the classic model and can comprehensively improve the accuracy of the evaluation findings [12]. Topological structure of BP neural network model is shown in Figure 4.

In this first step, we calculate the error of the output layer according to the error loss function and then transfer it layer by layer to the middle layers in some form and update the parameters of each layer [13, 14]. Through continuous iteration, the error of loss function calculation is minimized and the parameters converge. The back-propagation algorithm adopts the gradient descent method, as illustrated in equation (3), to update the parameters:

$$\begin{aligned} w_{ij}^l &= w_{ij}^l - \eta \nabla w_{ij}^l \\ b_i^l &= b_i^l - \eta \nabla b_i^l. \end{aligned} \tag{3}$$

In formula (3), η is the learning rate, and ∇w_{ij}^l and ∇b_i^l are the gradients of the error loss function to the connection weight w_{ij}^l and the paranoid term b_i^l, respectively. It can be seen that the key of the back-propagation algorithm is to find the gradient of the error loss function to the parameters [15]. The calculation process is given in the following steps.

Step 1: Calculate the loss error according to the target prediction and expected output of the output layer using the following equation:

$$L = \gamma\left(a^N, y\right). \tag{4}$$

In formula (4), L is the loss error, a^N is the target prediction vector of the output layer, y is the target expectation vector, and the function $\gamma(\cdot)$ denotes the loss function.

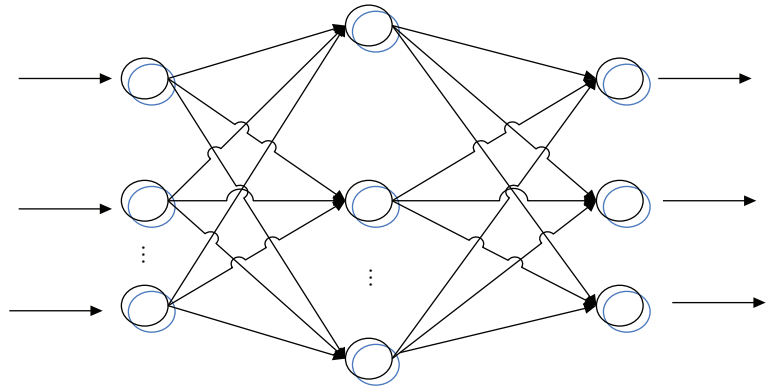

Figure 4: Topological structure of BP neural network model.

Step 2: Calculate the error term δ^l of layer l in the network according to the error loss L using the following equation:

$$\delta^l = \frac{\partial L}{\partial z^l} = \frac{\partial L}{\partial a^l}\frac{\partial a^l}{\partial z^l}. \tag{5}$$

Step 3: Calculate the error term of neuron i in layer l according to the chain rule, as illustrated in the following equation:

$$\delta_i^l = \frac{\partial L}{\partial z_i^l} = \sum_j \frac{\partial L}{\partial z_j^{l+1}}\frac{\partial z_j^{l+1}}{\partial z_i^l} = \left(\sum_j \delta_j^{l+1} w_{ij}^{l+1}\right) g_l'\left(z_i^l\right). \tag{6}$$

It can be seen from formula (6) that the error term of layer l is affected by the error term of layer $l+1$. In other words, the error of the network will propagate in the opposite direction layer by layer through the back-propagation algorithm.

Step 4: Calculate the connection weight of each layer and the gradient of the bias term according to the error term using the following equation:

$$\begin{aligned} \nabla w_{ij}^l &= \frac{\partial L}{\partial z_j^l}\frac{\partial z_j^l}{\partial w_{ij}^l} = \delta_j^l a_i^{l-1}, \\ \nabla b_i^l &= \frac{\partial L}{\partial z_i^l}\frac{\partial z_i^l}{\partial b_i^l} = \delta_i^l. \end{aligned} \tag{7}$$

As can be seen from formula (7), the gradient of the current layer connection weight w_{ij}^l strongly depends on the error term of the current layer neuron and the output of the previous layer neuron. Moreover, it can also be observed that the gradient of the current layer bias term b_i^l depends on the error term of the current layer neuron. Through substituting the above calculation results into formula (3), the parameter update of each round of the training process can be completed.

2.3. Activation Functions. The activation function achieves delinearization, turning the neural network into a nonlinear model and bringing the network model the ability to solve linear inseparable problems [16]. There are various activation functions that are related to neural network and each function can be replaced with another one in order to boost the accuracy of the model. Few of the well-known and largely used activation functions comprise the tanh function, ReLU (Rectified Liner Units) function, sigmoid function, and the softmax function. Among these, the *softmax* function is often used in the classification tasks [12, 17]. It should be noted that an appropriate activation function is selected according to the needs of the task and the characteristics of the network layer. The three activation function images are illustrated in Figure 3.

In the next discussion, we offer a brief description and mathematical model of each activation function. In the later sections, we will demonstrate that these functions have impacts on the network accuracy and prediction outcomes.

(1) tanh: the tanh function is a hyperbolic tangent function, which maps variables to the values among the range $[-1, 1]$. However, the tanh function has the problem of gradient saturation; that is, the derivative of the function at both ends is almost zero. This easily causes the problem of gradient disappearance in the training process of the neural network back-propagation, which makes the training speed of the network model very slow or difficult to converge. The function's mathematical expression is given in the following equation:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{8}$$

(2) Sigmoid: the sigmoid function image is similar to the tanh function, and the problem of gradient disappearance is also prone to occur. The function's mathematical expression is given in the following equation:

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{9}$$

(3) ReLU: the ReLU function is a linear rectification function and a nonsaturated activation function, which can solve the problem of the disappearance of the gradient caused by the derivative tending to zero. The ReLU function sets the negative value to 0 and performs truncation processing. The ReLU function is easier in the process of derivation calculation and can speed up the convergence speed of the network model [18]. The mathematical expression of the ReLU function is given by the following equation:

$$f(x) = \max(0, x). \tag{10}$$

(4) Softmax: the softmax function is generally used in the output layer of the neural network to complete the classification task. In the multiclassification process, the main task and function of the softmax function is to use the original output, calculate a new output, and map the value range to [0, 1]. In this way, the output of the neural network becomes the probability distribution of the target label. The function's mathematical expression is illustrated in the following equation:

$$f(x_i) = \frac{\exp(x_i)}{\sum_k \exp(x_k)}. \tag{11}$$

3. Fundamentals of Music Signal Analysis

3.1. Overview of Music Genres. Since the emergence of human beings, music has developed with the evolution of human beings. Under the influence of different periods, regions, nationalities, and cultures, it has gradually formed some unique musical classic characteristics in musical thought, creative principles, artistic personality, and means of expression and techniques, and music types with different styles appeared. These types can be called music schools. Popular music genres include classical, jazz, blues, hip-hop, rock, country, pop, and metal [19, 20]. There is no strict classification standard for the classification of music genres, which is subjective. Music works of the same genre have similar artistic styles.

3.2. Music Features. The features and characteristics of the music genre can be divided into three different types: (i) time domain characteristics, (ii) frequency domain characteristics, and (iii) cepstrum domain characteristics.

3.2.1. Time Domain Characteristics. Time domain features include zero crossing rate (ZC3) and short-time energy (STE). These features can be extracted directly from the waveform of the original signal. The processing process is simple and requires less mathematical calculation. They are widely used in the research of music classification tasks [20, 21]. The two common time domain features are described in detail below:

(1) Short-time energy: Short-time energy is the sum of energy in a small window, reflecting the change range of music signal over a period of time. It should be noted that it is generally used to judge the silence in a piece of music, carry out endpoint detection, and identify the beginning, transition, or end of music signal [22]. The calculation formula for the short-time energy is given by the following equation:

$$E_n = \sum_{k=-\infty}^{\infty} [x(k)\omega(n-k)]^2. \tag{12}$$

In formula (12), $\omega(n-k)$ represents "window function." The more popular window functions used to calculate short-time energy include "rectangular window" and an improved raised cosine window, "Hamming window" [23]. The calculation formula for window function is given by the following equation:

$$\omega(n) = \begin{cases} 1, & (0 \le n \le N-1), \\ 0, & \text{otherwise}, \end{cases}$$

$$g(n) = \begin{cases} 0.54 - 0.46\cos\left[\dfrac{2\pi n}{(N-1)}\right], & (0 \le n \le N-1), \\ 0, & \text{otherwise}. \end{cases} \tag{13}$$

In formula (13), N represents the length of the window.

(2) Short-time zero crossing rate: If the adjacent voice signal samples carry the opposite algebraic symbols, it is considered that zero crossing will be produced. The level of zero crossing rate directly reflects the number of high-frequency components of music signal. Short-time zero crossing rate is commonly used to detect silent frames in voice time domain analysis. The calculation method of this feature is given by the following equation:

$$\text{zerocross} = \frac{1}{2}\sum_{m=0}^{N-1} \left|\text{sgn}[x_n(m)] - \text{sgn}[x_n(m-1)]\right|. \tag{14}$$

In formula (14), $x_n(m)$ represents a discrete speech signal, and $\text{sgn}[\cdot]$ is a special function used to represent algebraic symbols. The definition of the function that denotes the algebraic symbols is given by the following equation:

$$\text{sgn}[x] = \begin{cases} 1, & (x \ge 0), \\ 0, & (x < 0). \end{cases} \tag{15}$$

3.2.2. Frequency Domain Characteristics. Common frequency domain features include spectrum centroid (SC), spectrum energy (SE), spectrum bandwidth (SB), and spectrum traffic (SF). The description and calculation

formulas of several common frequency domain features are listed below.

(1) Spectrum centroid (SC): The spectrum centroid is a commonly used measure. The size of this value represents the size of the frequency component of the music signal. The larger the value, the more high-frequency components and vice versa. The calculation formula is illustrated as follows:

$$SC = \frac{\sum_{\omega=l_0}^{h_0} \omega |F(\omega)|^2}{\sum_{\omega=l_0}^{h_0} |F(\omega)|^2}. \tag{16}$$

(2) Spectrum energy (SE): The frequency domain feature is used to characterize the frequency domain energy of a frame signal of music. The calculation formula for the spectrum energy is as follows:

$$SE = \sqrt{\frac{1}{h_0 - l_0} \sum_{\omega=l_0}^{h_0} |F(\omega)|^2}. \tag{17}$$

(3) Spectrum traffic (SF): The spectrum traffic is a dynamic feature that represents the spectrum of the music signal. In fact, it is the sum of the squares of the signal differences of all adjacent frames in a discrete frequency domain music signal. The calculation formula is given as follows:

$$SF = \frac{1}{h_0 - l_0} \sum_{\omega=l_0}^{h_0} |F(\omega + 1) - F(\omega)|^2. \tag{18}$$

In the three above formulas, $F(\omega)$ represents the Fourier transform of each frame of signal. Furthermore, l_0 and h_0 represent the maximum frequency and minimum frequency of a piece of music in the frequency domain signal, respectively.

3.2.3. Cepstrum Domain Characteristics. The music signal is transformed into frequency domain through Fourier transform, and the frequency domain characteristics are obtained through mathematical calculation and analysis, as discussed in previous sections. Then, take the logarithm of the music spectrum signal and perform the inverse Fourier transform. The audio signal in the frequency domain will be converted to the cepstrum domain, so as to obtain the cepstrum domain characteristics [24, 25]. The most common cepstrum domain features and related formulas are listed below:

(1) Mel frequency cepstral coefficient (MFCC): It is one of the most commonly used cepstral domain features, which can well represent the audio signals. The Mel frequency cepstrum coefficient can transform nonlinear relationship into linear relationship. The calculation step of the MFCC is through pre-emphasis, framing, windowing, fast Fourier transform, and taking the absolute value or the square value. Through the triangular band-pass Mel frequency filter bank, the logarithm of the output energy of the filter is taken and DCT inverse transformation is performed to obtain the characteristics of the dynamic Mel frequency cepstrum coefficient [26]. The relationship between the mel frequency represented by $\mathrm{mel}(f)$ and the linear frequency represented by f is given by the following equation:

$$\mathrm{mel}(f) = 2595 \times \log_{10}\left(1 + \frac{f}{700}\right). \tag{19}$$

(2) Linear prediction and cepstrum: Combining the two principles of linear prediction and cepstrum, the all pole model function is defined as illustrated in the following equation [27]:

$$H(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}}. \tag{20}$$

In formula (20), a_k and p represent prediction coefficient and prediction order, respectively. Assuming that $h(n)$ represents the impulse response of the original music signal without preprocessing and $H(z)$ represents the system function, the process of obtaining the cepstrum is to calculate the logarithm of $H(z)$ first and then perform the inverse transformation. The calculation process is given by the following equation:

$$\lg H(z) = \widehat{H}(z) = \sum_{n=1}^{\infty} \widehat{h}(n) z^{-n}. \tag{21}$$

4. Classification of Music Genres

Grounded on the deep learning-based music genre classification method, in fact the music genre characteristics are extracted by preprocessing the musical signals. Furthermore, the music genre classification neural network model is planned according to the fully connected neural network structure. According to the characteristic sequence of the input music genre, the attention mechanism is researched, and the classification network of this article is designed using the attention mechanism to realize the classification of music genres.

4.1. Music Signal Preprocessing. Preprocessing the music signal is a very important stage in the music genre classification method. The preprocessing can make the next extracted features more effective. Moreover, less useful signals and noise can be removed to increase the prediction outcomes and accuracy. The following steps were carried out to preprocess the music signals.

(1) Preemphasis: In order to improve the high-frequency resolution of the music signal [28] and in order to perform overall spectrum analysis on the entire frequency band, the preemphasis is introduced. The preemphasis is generally achieved

through a first-order digital filter before the feature parameter extraction. The transfer function of the filter is expressed as given by the following equation:

$$H(z) = 1 - az^{-1}. \tag{22}$$

In formula (22), parameter a denotes the factor of preemphasis that is, in general, considered as a decimal digit nearby to 1. If we suppose that the worth of sample, related to the music genre signal, is $x(n)$ at time n, then the outcome after the preemphasis phase is as given by the following equation:

$$y(n) = x(n) - ax(n-1). \tag{23}$$

(2) Framing: In order to smoothly transition between the two frames of signals and to ensure that information is not lost, the framing phase needs to have an overlapping part of 1/3~1/2 frame length between the two frames. This overlapping fragment is entitled the frame shift. Then, the theoretical calculation formula for the number of frames of a music signal segment is computed as explained in the following equation:

$$N = \left[\frac{N_1 - N_0}{N_2 - N_0}\right]. \tag{24}$$

In formula (24), N_1 characterizes the entire span of the music signal, and N_2 symbolizes the length of the frame. Similarly, N signifies the total amount of frames, and N_0 exemplifies the frame shift.

(3) Windowing: After framing all music genre segments, in order to increase the continuity between frames, it is suggested to reduce edge effects and also reduce spectrum leakage. Furthermore, it is also essential and crucial to accomplish the process of windowing on the framed music signal. The commonly used window functions in audio signal processing include (i) Hamming window, (ii) rectangular window, and (iii) Hanning window. The three window functions are defined as follows:

$$\omega(n) = \begin{cases} 1, & 0 \le n \le M-1, \\ 0, & \text{otherwise}, \end{cases}$$

$$\omega(n) = \begin{cases} 0.5\left(1 - \cos\left(\dfrac{2\pi n}{(M-1)}\right)\right), & 0 \le n \le M-1, \\ 0, & \text{otherwise}, \end{cases}$$

$$\omega(n) = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi n}{(M-1)}\right) & 0 \le n \le M-1, \\ 0, & \text{otherwise}, \end{cases} \tag{25}$$

These three window functions all have low-pass characteristics, and the main performance is determined by the attenuation of the first side lobe and the width of the main lobe. Since the boundary of the window function of the Hamming window is smooth, the first side lobe attenuation is the most severe, which can meritoriously circumvent the phenomenon of leakage [29]. Consequently, this paper selects Hamming window as the window function.

4.2. Music Feature Extraction. After preprocessing the signal of each music genre, the characteristic of the music genre, namely, MPCC, is extracted. The specific steps for extracting MPCC characteristic parameters of music genre signals are illustrated in the following steps:

(1) Accomplish the FFT transformation on every frame of the music genre signal after preprocessing to acquire the spectrum of the frequency.

(2) Proceed with the square of the modulus for the FFT-transformed spectrum, computed in previous step, in order to acquire the discrete power spectrum, denoted by $|X(k)|^2$, of every music signal.

(3) In the third step, pass the power spectrum $|X(k)|^2$ for filtering through a set of Mel filters using the following equation:

$$S(i,m) = \sum_{k=0}^{N-1} |X(i,k)|^2 H_m(k), 0 \le m < N. \tag{26}$$

(4) Finally, calculate the natural logarithm to acquire the MPCC parameters for each and every music genre signal using the following equation:

$$\text{mpcc}(i,m) = \ln S(i,m). \tag{27}$$

Subsequently, the range of the frequency in the music signal changes from a little and few hertz to thousands or kilo of hertz, and the transformation is moderately very slow. Therefore, the MPCC parameters extracted from each frame of the music genre signal in this paper are 12-dimensional.

4.3. Design of Network Model for Music Genre Classification. The neural network learning process is listed in Figure 5(a). According to the neural network structure, the design and research of music classification model is shown in Figure 5(b) [15, 16].

The input of the input layer processes the music signal through preemphasis, framing, and windowing to extract music genre features. The music genre feature sequence, extracted from the input layer, is their features learned. Similarly, the influence on the current time state is calculated from the future and the past, respectively. The feature representation $H = \{H_1, H_2, \ldots, H_L\}$ is obtained and combined with the context semantic information, which is input into the attention mechanism network. The attention

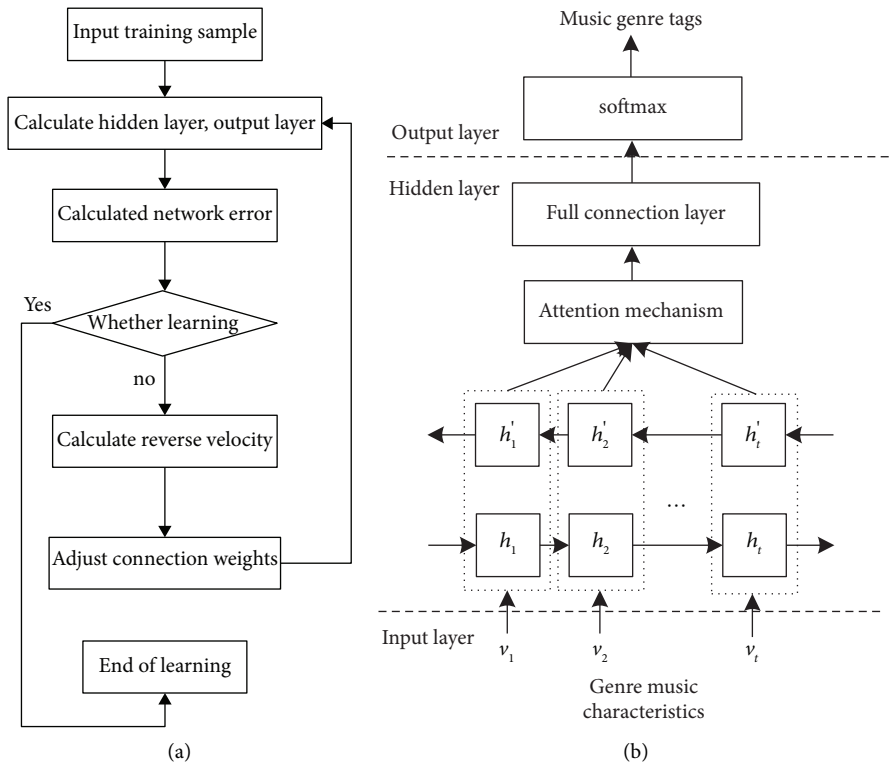

Figure 5: (a) Neural network learning flow chart; (b) structure diagram of music genre classification network model.

mechanism network learns the input feature representation H and obtains the corresponding attention probability distribution [14]. Subsequently, it multiplies each attention probability by its corresponding feature vector and finally obtains the music genre feature vector representation v. The attention process is given as follows:

$$e_t = \tanh\left(WH_t + b\right). \tag{28}$$

In the above formula, e_t is the attention score of the feature vector H_t at time t in the feature representation H. In the next phase, the activation function softmax is applied, as given by equation (28), to compute v as given by the following equation:

$$\alpha_i = \text{soft}\max\left(e_i\right) = \frac{\exp\left(e_i\right)}{\sum_{k=1}^{L}\exp\left(e_k\right)}, \tag{29}$$

$$v = \sum_{i=1}^{L}\alpha_i H_i. \tag{30}$$

The output layer of the network model is defined as follows by calculating the cross-entropy loss function:

$$C = \frac{1}{n}\sum_{x} y\,\ln\left(a\right) + \left(1-y\right)\ln\left(1-a\right). \tag{31}$$

In the above formula, C is the loss, n is the number of samples, x is the input sample, and y and a are the output predicted value and target expected value, respectively, of input x of the network model. Note that the classification of music genres is calculated using the following equation:

$$\Delta\theta = -\alpha\frac{v_t}{\sqrt{s_t+\varepsilon}}. \tag{32}$$

In the above formula, the classification of music genres is realized through the steps described above.

5. Experimental Analysis

5.1. Experimental Environment and Datasets. In order to verify the effectiveness of the music genre classification method based on deep learning, the MATLAB 2016a programming software was used to extract the features of music signals. We build a fully connected neural network based on Theano library using the Python language. Similarly, we model training that uses the Adam optimization method as the gradient descent optimization algorithm. The learning rate is set to 0.001, and the training rounds are set to 200 rounds. All experiments are carried out and verified on the GTZAN dataset. There are a total of 1000 audio files in the GTZAN dataset. These 1000 files contain 10 genres of music, and each genre has a total of 100 samples. Note that the experiments were carried out several times and the reported results are averaged over multiple runs. In the experiments, the method of nonrepetitive random sampling is adopted, and 80% of each music genre dataset is selected. Furthermore, the distribution of the number of music genres in each

Table 1: Distribution of music genres in each category.

Music genre	Rock	Metal	Country	Classical	Blues
Training set	320	308	268	320	320
Validation set	80	78	67	80	79

Table 2: Music genre classification effect of the proposed method.

		Forecast confusion matrix (%)				
		Rock	Metal	Country	Classical	Blues
Actual confusion matrix (%)	Rock	85.07	5.97	8.96	0	0
	Metal	3.80	94.94	0	0	1.27
	Country	15.38	0	82.05	0	2.56
	Classical	2.50	1.25	2.50	92.50	1.25
	Blues	1.25	0	0	3.75	95.00

category of the training set and validation set is as shown in Table 1.

5.2. Classification Evaluation Index. We performed the music genre classification experiments on five different music genre files of rock, metal, country, classical, and blues. In fact, this is a multiclassification task, and the categories are relatively balanced. The accuracy of the sample population accuracy is expressed follows:

$$\text{accuracy} = \frac{\sum_i M(i,i)}{\sum_i \sum_j M(i,j)} \times 100\%. \tag{33}$$

In the above formula, $M(i, j)$ is the number of samples in the population.

5.3. Music Genre Classification Effect. After the music genre classification network model is trained by the proposed method, the classification performance of the music genre classification network model is evaluated by using the verification set. The results and the forecast confusion matrix outcomes for 5 files are shown in Table 2.

Analyzing the results demonstrated in Table 2, we conclude that the metal music, classical music, and blues music all successfully fit into their appropriate classification categories, with accuracy rates of 94.94 percent, 92.50 percent, and 95.00 percent, respectively. Furthermore, the rock music and country music are sometimes mislabeled. Due to the fact that some country music can be used as an accompaniment to country dancing and that some rock music is mistakenly categorized as country music, country music is often confused with rock music. The distinction between rock music and metal music is somewhat erroneous. However, the possible reason is that they both pay more attention to rhythm and are similar. In general, the proposed method is used to effectively classify the music of the above five genres, and the proposed method has a better effect on the classification of music genres.

The total number of neurons in the BP neural network has a significant impact over the training and test error. For example, as shown in Table 3, when the number of neurons increases, the training error continues to decrease, and we observed that there is a certain correlation between them. After the analysis, we concluded that 7 as the number of neurons is the most ideal measurement for our experimental setup.

Table 3: Relationship between number of neurons in hidden layer and error.

Number of hidden layer neurons	The training error	Test error
3	1.385	1.11
4	0.805	0.81
5	0.706	0.72
6	0.629	0.71
7	0.621	0.70

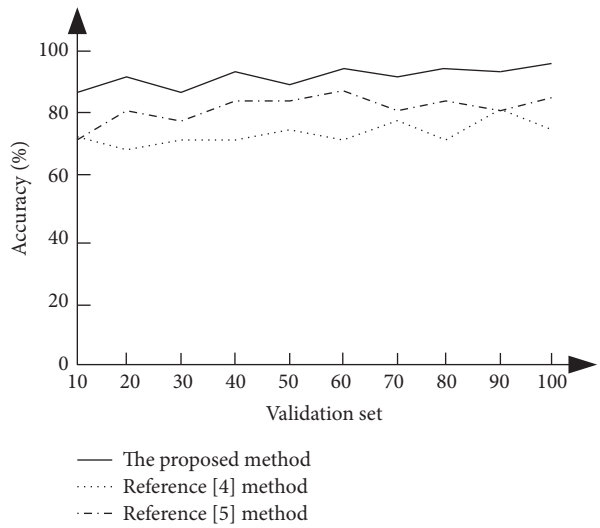

Figure 6: Comparison of outcomes of various techniques for music genre classification in terms of accuracy.

5.4. Classification Accuracy of Music Genres. The assessment outcomes and comparative study of classification precision of various music genre approaches are presented in Figure 6.

We can easily observe from Figure 6 that, under different validation sets, [4] is 73%, and [30] is 82%. The average music genre classification accuracy rate is 91%. Furthermore,

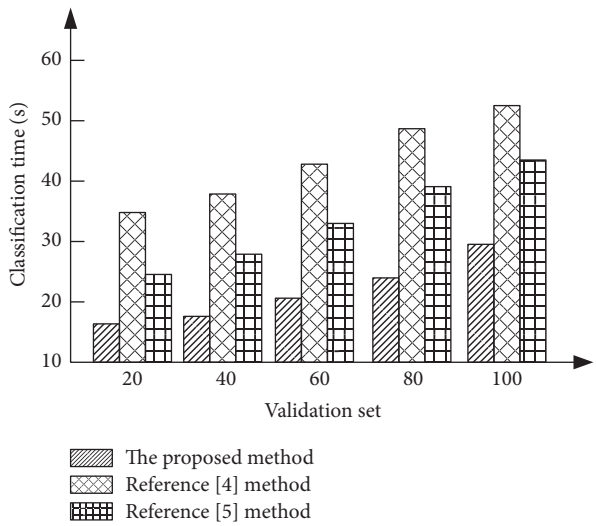

Figure 7: Comparison of outcomes of various approaches for music genre classification in terms of time.

we can also observe that, associated with the method demonstrated in [4] and the approach presented in [30], the correctness and accuracy of the proposed music genre classification method are significantly higher.

5.5. Music Genre Classification Time. The evaluation results, in terms of classification time, when the proposed approach is compared with other music genre classification techniques, are presented in Figure 7.

We can observe from Figure 7 that when the number of verification sets increases, the music type classification time of various techniques will also increase. The technique based on the deep learning algorithm, projected in this paper, has the benefits of refining the accurateness, precision, and effectiveness of the music classification.

6. Conclusions and Future Work

In this paper, a prediction method based on the deep learning algorithm was proposed, which has the advantages of refining the correctness, precision, and effectiveness of the music classification. The experimental outcomes demonstrated that the projected method has the ability to effectively improve the accuracy of the music classification and is helpful for music channel classification. Moreover, its music genre classification accuracy is high, which can effectively shorten the music genre classification time and has, therefore, a better music genre classification effect. However, because the research scope of this algorithm is not extended to the subject of finite element, the proposed method has some limitations. In the process of extracting music genre features, this paper ignores the accompaniment information of music. The main melody of the same piece of music, accompanied by different music, may present different genres and styles.

In subsequent research, we can consider combining the main melody and accompaniment of music to extract features to further improve the accuracy of classification. Moreover, advanced deep learning methods such as deep neural networks should be considered to improve the accuracy of the prediction outcome. In learning algorithms, the training is one of the activities that take significant time and can degrade the performance of the whole system. Therefore, we will consider dividing the training and prediction phases over the edge-cloud architecture so that the training may happen at the remote cloud that has usually bulk of resources. The prediction part of the algorithm should run on edge which will essentially increase the processing and response time of the system.

Data Availability

The data used to support the findings of this study are available from the author upon request.

Conflicts of Interest

The author declares that he has no conflicts of interest.

References

[1] M. Schedel, S. Yuditskaya, and S. E. Green, "New interfaces for musical expression (NIME) conference," *Computer Music Journal*, vol. 43, no. 2-3, pp. 159–166, 2019.

[2] J. U. Hou and H. K. Lee, "Layer thickness estimation of 3D printed model for digital multimedia forensics," *Electronics Letters*, vol. 55, no. 2, pp. 86–88, 2019.

[3] J. Nam, K. Choi, J. Lee, S. Y. Chou, and Y. H. Yang, "Deep learning for audio-based music classification and tagging: teaching computers to distinguish rock from Bach [J]," *IEEE Signal Processing Magazine*, vol. 34, no. 5, pp. 236–240, 2019.

[4] A. Elbir and N. Aydin, "Music genre classification and music recommendation by using deep learning," *Electronics Letters*, vol. 56, no. 12, pp. 627–629, 2020.

[5] A. Gopi, A. Selvaraj, and W. Jebarani, "Digital image steganalysis: a survey on paradigm shift from machine learning to deep learning based techniques [J]," *IET Image Processing*, vol. 15, no. 2, pp. 504–522, 2020.

[6] V. M. Sineglazov and O. I. Chumachenko, "Structural-parametric synthesis of deep learning neural networks [J]," *Artificial Intelligence*, vol. 25, no. 4, pp. 42–51, 2020.

[7] Y. Zhu, C. Yao, and X. Bai, "Scene text detection and recognition: Recent advances and future trends," *Frontiers of Computer Science*, vol. 10, no. 1, pp. 19–36, 2016.

[8] D. Chen, J.-M. Odobez, and H. Bourlard, "Text detection and recognition in images and video frames," *Pattern Recognition*, vol. 37, no. 3, pp. 595–608, 2004.

[9] W. Weihong and T. Jiaoyang, "Research on license plate recognition algorithms based on deep learning in complex environment," *IEEE Access*, vol. 8, pp. 91661–91675, 2020.

[10] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.

[11] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 50–70, 2022.

[12] S. Pouyanfar, S. Sadiq, Y. Yan et al., "A survey on deep learning: algorithms, techniques, and applications," *ACM Computing Surveys*, vol. 51, no. 5, pp. 1–36, 2019.

[13] A. Gurhanli, "Accelerating convolutional neural network training using ProMoD backpropagation algorithm [J]," *IET Image Processing*, vol. 14, no. 13, pp. 761–769, 2020.

[14] A. Ali, Y. Zhu, and M. Zakarya, "Exploiting dynamic spatio-temporal correlations for citywide traffic flow prediction using attention based neural networks," *Information Sciences*, vol. 577, pp. 852–870, 2021.

[15] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2298–2304, 2017.

[16] O. Shamir, "Discussion of: "Nonparametric regression using deep neural networks with ReLU activation function" [J]," *Annals of Statistics*, vol. 48, no. 4, pp. 1911–1915, 2020.

[17] H. Zhang, Q. Zheng, B. Dong, and B. Feng, "A financial ticket image intelligent recognition system based on deep learning," *Knowledge-Based Systems*, vol. 222, Article ID 106955, 2021.

[18] A. Ali, Y. Zhu, and M. Zakarya, "Exploiting dynamic spatio-temporal graph convolutional neural networks for citywide traffic flows prediction," *Neural Networks*, vol. 145, pp. 233–247, 2022.

[19] G. J. Dimitrakopoulos and I. E. Panagiotopoulos, "In-vehicle Infotainment systems: using Bayesian networks to model cognitive selection of music genres," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 1–10, 2020.

[20] A. Ali, Y. Zhu, and M. Zakarya, "A data aggregation based approach to exploit dynamic spatio-temporal correlations for citywide crowd flows prediction in fog computing," *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 31401–31433, 2021.

[21] S. Langer, "Analysis of the rate of convergence of fully connected deep neural network regression estimates with smooth activation function," *Journal of Multivariate Analysis*, vol. 182, p. 104695, 2021.

[22] T. Fischer, M. Caversaccio, and W. Wimmer, "Speech signal enhancement in cocktail party scenarios by deep learning based virtual sensing of head-mounted microphones [J]," *Hearing Research*, vol. 408, Article ID 108294, 2021.

[23] S. Dong, P. Wang, and K. Abbas, "A survey on deep learning and its applications," *Computer Science Review*, vol. 40, Article ID 100379, 2021.

[24] S. Gannot and P. A. Naylor, "Highlights from the audio and acoustic signal processing technical committee [in the spotlight]," *IEEE Signal Processing Magazine*, vol. 36, no. 2, pp. 136–134, 2019.

[25] A. Khurshid, A. N. Khan, F. G. Khan, M. Ali, J. Shuja, and A. Khan, "Secure-CamFlow: a device-oriented security model to assist information flow control systems in cloud environments for IoTs," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 8, Article ID e4729, 2019.

[26] A. N. Khan, M. Ali, AuR. Khan et al., "A comparative study and workload distribution model for re-encryption schemes in a mobile cloud computing environment," *International Journal of Communication Systems*, vol. 30, no. 16, Article ID e3308, 2017.

[27] H. Bao and X. Fan, "Simulation of dynamic classification for Unbalanced big data in cloud computing environment," *Computer Simulation*, vol. 37, no. 08, pp. 311–314+461, 2020.

[28] W. Guo, S. Piao, T. C. Yang, J. Guo, and K. Iqbal, "High-resolution power spectral estimation method using deconvolution [J]," *IEEE Journal of Oceanic Engineering*, vol. 43, no. 2, pp. 1–11, 2020.

[29] S. Mustafa, B. Nazir, A. Hayat, A. Khan, and S. A. Madani, "Resource management in cloud computing: taxonomy, prospects, and challenges," *Computers & Electrical Engineering*, vol. 47, pp. 186–203, 2015.

[30] H. G. Kim, G. J. Jang, Y. H. Oh, and H. J. Choi, "Speech and music pitch trajectory classification using recurrent neural networks for monaural speech segregation," *The Journal of Supercomputing*, vol. 76, no. 10, pp. 8193–8213, 2019.