Music Genre Classification based on Deep Learning

Wang Jinyi
1023040927
Nanjing University of Posts
and Telecommunications

***Abstract*—With the popularization of digital music, music genre classification has become an important issue in the field of music information processing. This study aims to use neural network technology to classify music genres and improve training efficiency by using GPU on Google Colab. We chose the GTZAN dataset, which contains audio samples from various genres, providing rich and diverse data for music classification tasks. In terms of model design, we adopted a deep neural network structure to extract audio features through convolutional neural network (CNN) layers. By using GPU resources on Google Colab, we successfully accelerated the training process of the model and significantly shortened the training time. After training, our neural network model achieved satisfactory accuracy on the GTZAN dataset, providing good performance for music genre classification tasks. Specifically, the accuracy of the model on the test set reached 84%. Overall, this study demonstrates the effectiveness of using GPU for neural network training on Google Colab and combining it with the GTZAN dataset for music genre classification, providing useful information for the field of music information processing.**

***Index Terms*—music classification, deep learning, convolutional neural network**

I. Introduction

In the vast realm of music resources, the role of music classification tags is pivotal in aiding users to conveniently discover their desired musical content[1,2]. These tags not only differentiate various music styles and genres but also describe song characteristics such as emotion, rhythm, and lyrics. For example, when users seek relaxing music, they can effortlessly select songs tagged with labels like "calm" or "healing," eliminating the need for extensive searching and filtering. Additionally, music classification tags assist users in discovering music aligned with their preferences, enhancing personalized recommendations and customized services[3]. Therefore, the importance of music classification tags becomes increasingly prominent in the current digital music era[4].Traditionally, music researchers extract statistical features such as pitch differences, rhythm, and melody from WAV format audio based on accumulated experience[5]. The mean or standard deviation of these statistical features often serves as input variables in experiments. While manual recognition of musical attributes and signal feature extraction has the advantages of strong expressiveness and high distinctiveness, reducing the difficulty of algorithm model design and improving classification accuracy, it has drawbacks such as lengthy processing times, substantial manpower requirements, and limited applicability, making efficient extraction of deep musical features challenging.In the current exploration of music classification, the majority of research results are built upon traditional machine learning, employing classifiers like Support Vector Machine (SVM), Random Forest, and k-Nearest Neighbors[6]. The shallow structure of machine learning classifiers limits its deep exploration of audio data, hindering the extraction of representative music features as experimental indicators, thereby impacting classification performance[7]. Hence, the significance of extracting audio feature variables is evident. In recent years, the influence of deep learning technologies has been steadily increasing across various domains, especially in natural language processing and computer vision, achieving remarkable research outcomes[8]. The advantage lies in the ability of deep learning technologies to automatically extract deeper features from shallow-level features, reflecting the local correlations within input data[9]. Furthermore, this technology is poised to lead a new direction and vision in research on genre classification in current and future music studies.This paper focuses on extracting signal features from music audio signals and, leveraging deep learning tools, aims to perform automatic and efficient filtering, achieving effective music classification.

II. Data preprocessing

Music is a comprehensive art, and its ability to express and convey emotions stems from three core elements: rhythm, melody, and harmony. These three elements are intertwined to jointly construct the structure and beauty of musical works. Rhythm is the backbone of music, endowing it with an orderly structural pattern by reflecting the length and strength of notes[10]. It effectively combines an disordered rhythmic flow into an orderly whole, integrating and processing different lengths and repetitive patterns of the spectrogram[11]. Different rhythmic styles convey the unique emotions and attitudes of composers and performers, experiencing tension and panic from the rapid rhythm to the soothing rhythm and feeling relaxed and happy. Therefore, rhythm is not only a fundamental element of music, but also indirectly reveals the style to which music belongs. Melody, also known as melody, is the most important language medium in music and the soul of music[12]. It arranges songs according to a certain rhythmic hierarchy and creates music sequences composed of different notes. Similar to visual perception, the perception of auditory melodies involves important factors in the perception of musical movements and senses. The melody conveys the composer's unique creativity and emotions through a rich combination of notes. Harmony is one of the

main forms of expression in music, consisting of two or more different tones being pronounced simultaneously according to specific rules[13]. The harmony of mixed tones creates a rich and colorful sound effect, injecting new innovative inspiration into musical works. The wonderful harmony of harmony brings people a sense of joy and tranquility, making it easier for listeners to immerse themselves in the world of music[14]. These three elements together construct the multi-level structure of music, endowing the work with unique and profound artistic charm.

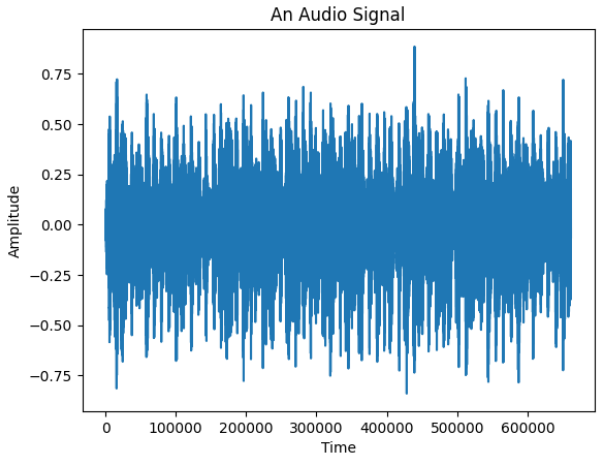

Fig. 1. Music waveform.

A. Audio Signal Features

In the research domain of audio feature processing, Mel-Frequency Cepstral Coefficients (MFCCs) stand out as a widely utilized indicator for describing audio features. MFCCs find primary applications in the field of sound processing, founded on Fourier and cepstral analysis[15]. The process involves sampling the audio, followed by applying a Fourier transform to the sampled points, resulting in the energy distribution of audio frames in the frequency domain. It represents a logarithmic transformation of the nonlinear Mel-scaled energy spectrum based on sound frequencies. To obtain MFCC coefficients, the input speech signal undergoes windowing and discrete Fourier transform to be converted into the frequency domain[16]. The Mel scale approximately maintains a linear frequency interval below 1000 Hz and a logarithmic interval above 1000 Hz. As a reference, the pitch of a 1 kHz tone with a perceived loudness above the auditory threshold of 40 dB is defined as 1000 Mel. A spectrogram is a heatmap used to visualize the relationship between time and frequency, displaying the results of time-frequency analysis. The following is a Mel spectrogram of an audio segment:

Utilizing MFCC as features for music classification can enhance the accuracy of music categorization.To input Mel spectrogram data into a neural network, their sizes must be consistent. If they are not of the same size, we need to use zero-padding on the smaller arrays to make their sizes uniform.

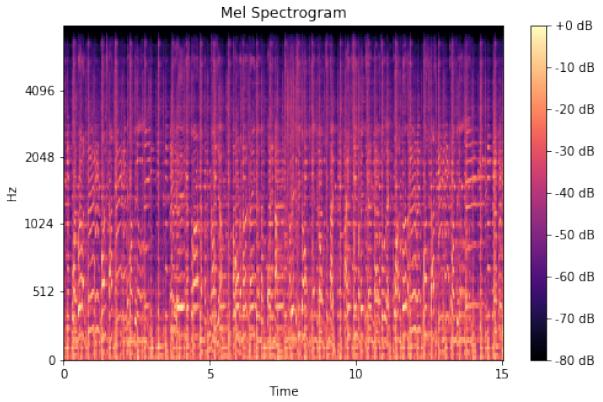

Fig. 2. Mel-spectrogram.

B. Convolutional Neural Network Algorithm

Convolutional Neural Networks (CNNs), proposed by the Japanese scholar Fukushima in 1984, have now been widely applied in various fields such as image and speech processing, achieving breakthrough results. CNNs have two main characteristics: local perception and weight sharing[17]. In each convolutional operation, local perception is carried out, and the obtained feature map shares parameters. Through multiple convolutions, the receptive field expands, gradually forming global characteristics and eventually becoming a high-level representation. CNNs typically include convolutional layers, pooling layers, and fully connected layers. Convolution and pooling layers are used for input and feature extraction, while fully connected layers map the features into the dimensional space.

- Convolutional Layer:The convolutional layer, also referred to as a filter, serves as a feature extractor, allowing it to flexibly extract relevant features from the input signal. Each convolutional layer comprises multiple convolutional kernels, and the features extracted from each audio input through the convolutional layer can be considered as a category of extracted audio features. Each convolutional kernel typically corresponds to a specific feature category, and its size represents the scope of attention to local details. During the convolution process, the convolutional kernel moves according to a pre-defined stride, conducting matrix multiplication operations on the scanned region, and after adding bias values, the corresponding output value is obtained through weighted summation.
- Pooling Layer:Pooling Layer is a common layer in convolutional neural networks (CNNs). Its main function is to reduce the spatial size of feature maps, reduce computational complexity, and retain important feature information. The pooling layer usually follows the convolutional layer and downsamples the features extracted by the convolutional layer. The two main types of pooling layers are Max Pooling and Average Pooling. In max pooling, for each pooling window, only the maximum

value in the window is retained; In average pooling, the average of all values in the window is taken. Pooling operation is performed by sliding a window on the input feature map, and the values within the window are pooled to generate an output value. The key parameters of the pooling layer include: pooling window size, stride, and padding. Through pooling layers, neural networks can reduce data dimensions, improve computational efficiency, and to some extent extract more significant features. This helps to reduce the complexity of the model, prevent overfitting, and make the network more robust. The method used here is maximum pooling.

- Fully Connected Layer:The fully connected layer not only serves the purpose of feature extraction but is primarily designed to achieve the goal of classification. For layers n-1 and layer n, any neuron node in layer n-1 is connected to all nodes in layer n. Simultaneously, the computation integrates each node in layer n, and when calculating the activation function, the input features consist of the weighted sum of all nodes in layer n-1. Therefore, the fully connected layer exemplifies a comprehensive connection between layers, demonstrating a full-range connectivity between them.

The final architecture of the Convolutional Neural Network (CNN) is as follows:

- Input layer:The dimensions of this layer are 128 x 660 neurons. It receives input data containing 128 mel scales and 660 time windows, laying the foundation for the extraction of subsequent features.
- Convolutional Layer 1:The first convolutional layer employs 16 unique 3x3 filters to effectively extract essential features from the input data.
- Max Pooling Layer 1: This layer reduces the spatial size of the feature maps by 2x4 max pooling, enhancing computational efficiency and preserving essential features.
- Convolutional Layer 2:The second convolutional layer employs 32 unique 3x3 filters to enhance the extraction of refined features.
- Max Pooling Layer 2: This layer further reduces the spatial size of the feature maps using a 2x4 window size, enhancing the saliency of important information.
- Fully Connected Layer:The fully connected layer with 64 neurons aids in capturing the intricate relationships between features extracted from the convolutional layers.
- Output Layer:The output layer consists of 10 neurons, which predict the classification of 10 different music genres.

All of the hidden layers used the RELU activation function and the output layer used the softmax function. The loss was calculated using the categorical crossentropy function. Dropout was also used to prevent overfitting.

III. Experimental Analysis

Before running CNN, I plan to train a Feedforward Neural Network (FFNN) for comparison. CNNs, with additional layers for edge detection, are well-suited for image classification, but their computational costs are often higher than FFNNs. If the performance of the FFNN is equally good, there may be no need to use CNN.During the experiments on deep learning music classification, we initially attempted various model architectures, including Feedforward Neural Networks (FFNN).The best model (based on test accuracy) achieved a training score of 69% and a test score of 45%.As shown in Figure 3.

```
Epoch 33/40
800/800 [==============================] - 2s 2ms/sample - loss: 1.0939 - accuracy: 0.5962 - val_loss: 2.1285 - val_accuracy: 0.3950
Epoch 34/40
800/800 [==============================] - 2s 2ms/sample - loss: 0.8789 - accuracy: 0.6988 - val_loss: 1.8983 - val_accuracy: 0.4400
Epoch 35/40
800/800 [==============================] - 2s 2ms/sample - loss: 0.9724 - accuracy: 0.6275 - val_loss: 1.7103 - val_accuracy: 0.4300
Epoch 36/40
800/800 [==============================] - 2s 2ms/sample - loss: 0.8706 - accuracy: 0.6837 - val_loss: 2.1170 - val_accuracy: 0.4200
Epoch 37/40
800/800 [==============================] - 2s 2ms/sample - loss: 0.9016 - accuracy: 0.6775 - val_loss: 2.0173 - val_accuracy: 0.3700
Epoch 38/40
800/800 [==============================] - 2s 2ms/sample - loss: 0.8616 - accuracy: 0.6862 - val_loss: 1.9092 - val_accuracy: 0.4000
Epoch 39/40
800/800 [==============================] - 2s 2ms/sample - loss: 0.6099 - accuracy: 0.7738 - val_loss: 1.6381 - val_accuracy: 0.4300
Epoch 40/40
800/800 [==============================] - 2s 2ms/sample - loss: 0.8407 - accuracy: 0.6850 - val_loss: 1.7282 - val_accuracy: 0.4500
```

Fig. 3. FFNN training accuracy.

In response to this observation, we decided to shift to training with Convolutional Neural Networks (CNN). CNNs, equipped with additional layers for edge detection, have demonstrated excellent performance in image classification problems. We anticipate that CNNs will better capture and learn features in the music classification task, enhancing the model's generalization capabilities.However, it was observed that most models started to exhibit overfitting after approximately 15-20 epochs, and further increasing the number of epochs did not significantly improve the model's performance.As shown in Figure 4. To look deeper into what was

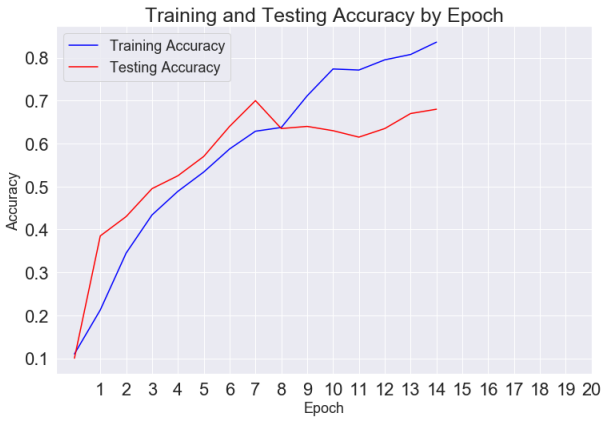

Fig. 4. Overfitting.

happening with the model, I computed a confusion matrix to visualize the model's predictions against the actual values. Since the confusion matrix function from sklearn does not return the labels for predicted values and actual values, I checked how many predicted and actual values there were for each genre to be able to figure it out.The final experimental results are shown in Figure 5. The score of the best CNN model (based on test score accuracy) is 68%. The training score is 84%, indicating overfitting of the model. This means that the model has adapted well to the training data but fails to

generalize to new data. Nonetheless, this is certainly a learning experience.

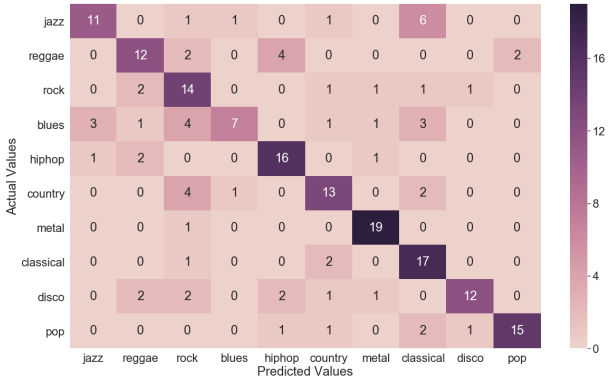

Fig. 5. Music classification diagram.

DISCUSSION

Research in the field of deep learning for music classification has made significant progress; however, there are still important points of discussion and challenges.Deep learning models have demonstrated potent capabilities in music classification tasks by learning complex audio features, enabling accurate classification of different genres and styles. Nevertheless, it's essential to acknowledge the disparity between the model's performance on the training set and its generalization ability on the test set. Addressing the issue of model overfitting may require techniques such as regularization and data augmentation.Diversity is intrinsic to the world of music, encompassing different genres, cultures, and historical periods. Current deep learning models may have limitations in handling this diversity, making it a focal point of discussion on how to better capture the multifaceted nature of music.The success of music classification relies not only on the design of deep learning models but also on feature extraction. Discussing how to effectively integrate deep learning with traditional music feature extraction methods to optimize classification performance is a crucial direction for future research.In terms of experimental design and evaluation standards, careful consideration of dataset construction and selection of evaluation metrics is necessary to ensure the reliability of experimental results. Establishing more representative datasets and comprehensive evaluation metrics will contribute to a more thorough assessment of the performance of deep learning models in practical applications.Model interpretability has been a long-standing issue in the field of deep learning, and it is equally crucial for music classification. Discussing ways to enhance the interpretability of models to improve users' understanding and trust in classification results is a research direction that requires further exploration.Finally, future research can focus on enhancing models' recognition capabilities for different music genres, emotions, and cultural features. Leveraging techniques like transfer learning and reinforcement learning to adapt to diverse music classification requirements across various domains and scenarios holds promise for the continued development of deep learning in music classification.

SUMMARY

This paper has effectively designed the structure of a convolutional neural network (CNN) to automatically learn genre-related features from spectrograms. Each audio file is transformed into a spectrogram and used as input data for the network model. The model is trained for classification using pre-defined label vectors on Google Colab. Through experimental comparisons, it was found that the trained model achieves an average accuracy of 84%. Future efforts will focus on increasing the variety of classifiable music genres, improving the model's classification efficiency and accuracy, optimizing model performance, exploring different model structures and training methods to enhance speed and accuracy, and reducing the demand for storage and computational resources.

REFERENCES

[1] Zhou J., Cui G., Zhang Z., et al. Graph neural networks: A review of methods and applications[J]. Al Open, 2018, 1: 57-81.

[2] Li T., Ogihara M., Li Q. A comparative study on content-based music genre classification. Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2003, (01): 282-289.

[3] Järveläinen H. Algorithmic musical composition[C]. Helsinki University of Technology Telecommunications Software and Multimedia Laboratory, 2000.

[4] Dieleman S., Zen H., et al. WaveNet: A generative model for raw audio, 2016. 46arXiv:1609.03499v2.

[5] Mehri S., Kumar K., Gulrajani I., et al. SampleRNN: An unconditional end-to-end neual audio generation model, 2017.

[6] Donahue C., McAuley J., Puckette M. Synthesizing audio with generative adcersarial networks[J]. Tenth International Conference on Learning Representations conference, 2019. arXiv:1802.04208.

[7] Vaswani A., Shazeer N., Parmar N., et al. Attention is all you need[C]. Advances in Neural Information Processing Systems, 2017, (1): 5998-6008.

[8] Bogdanov D., Porter A., Herrera P., et al. Cross-collection evaluation for music classification tasks[C]. 17th International Society for Music Information Retrieval Conference, 2016.

[9] Dannenberg R. B., Thom B., Watson D. A machine learning approach to musical style recognition[C]. International Computer Music Conference, 1997, (1): 344-347.

[10] Mc K. C. Automatic genre classification of midi recordings[D]. Canada: Mc Gill University, 2004.

[11] Şimşekli U. Automatic Music Genre Classification Using Bass Lines[C]. International Conference on Pattern Recognition, 2010.

[12] Armentano M. G., Noni W. A. D. Genre classification of symbolic pieces of music[M]. Kluwer Academic Publishers, 2017.

[13] Valverde-Rebaza J., Soriano A., Berton L., et al. Music genre classification using traditional and relational approaches[C]. 2014 Brazilian Conference on Intelligent Systems, 2014.

[14] Luong M. T., Pham H., Manning C. D. Effective approaches to attention-based neural machine translation, 2015. arXiv:1508.04025v5.

[15] Sarikaya R., Hinton G. E., Deoras A. Application of deep belief networks for natural language understanding. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2014, 22(4): 778-784.

[16] He K., Zhang X., Ren S., et al. Deep residual learning for image recognition[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016: 770-778.

[17] Donahue J., Hendricks L. A., Guadarrama S., et al. Long-term recurrent convolutional networks for visual recognition and description[C]. 2015 IEEE Conference on Computer Vision and Pattern Recognition, 2015: 2625-2634.