Musical Genre Classification with Convolutional Neural Networks

Zheng Xinyuan
1023041003
Nanjing University of Posts and Telecommunications
School of Computer Science
Nanjing, China

***Abstract*—In view of the shortcomings of misjudgment, omission, misclassification, slow processing speed and low efficiency caused by the music genre classification model established by a single feature, this paper proposes a music genre classification method based on convolutional neural network. Firstly, the cepstrum coefficient is used to extract the audio MFCC feature matrix, and its feature value is used as the input of CNN neural network to train the audio signal, and the optimal classifier is obtained and used as the trainer.**

The accuracy of the network is assessed by dividing the ratio between the original training data and the test data. The model outperforms baseline implementations based on mel frequency cepstrum coefficients and achieves results comparable to other recent methods.

Keywords—GTZAN, classification, neural network

I. Introduction

With the rapid development of multimedia and digital technology, there are more and more digital music resources on the Internet, and consumers' music consumption habits have shifted from physical music to online music platforms. With the change of music carrier, music data presents the status of expansion. Massive music resources and huge online music library stimulate users' needs for a variety of complex music retrieval. For example, when users are eager to listen to songs of a certain genre or with certain emotions at a certain moment, music labels are crucial to the quality of music retrieval. In addition to music retrieval, many recommendation and subscription scenarios also require song category information to provide users with more precise content.

In speech, face and image recognition, machine learning and deep learning are gradually stepping into these fields. Various scholars and researchers try to use computer intelligence technology incisively and vividly in the scope of music generation. Relatively speaking, in the storage and initial processing of massive data sets, deep learning is significantly more powerful than machine learning, and neural networks are more active in music analysis and processing, among which convolutional neural networks are more active. CNN is a very famous network model of deep learning. In recent years, its influence has spread to many fields such as speech system recognition, and its application value has been continuously improved. At the same time, it is also suitable for music classification and recognition.

Based on the above theoretical basis, I am familiar with the processing mechanism of neural network for massive audio data, and can produce significant effects on its classification efficiency. In this paper, a network architecture based on artificial neural network and machine learning algorithm is designed. After reading GTZAN music data and sorting and processing the data, Convert the pure playable audio form to the index matrix format acceptable to the computer; Explore the classification ability of the model according to the index variables of music. Furthermore, the convolutional neural network is used to embed the generation model to learn the music style, and the playable music file that is close to the original music is obtained through continuous training.

II. Problem Statement

A. Music Classification

Music information retrieval (MIR) is a large area of research dedicated to trying to extract useful information from music. Automatic genre classification is one of the tasks motivating this research. An automatic genre classification algorithm could greatly increase efficiency for music databases. The goal of this project is to use a convolutional neural network (CNN) to classifty a song by its genre. CNNs have additional layers for edge detection that make them well suited for image classification problems.

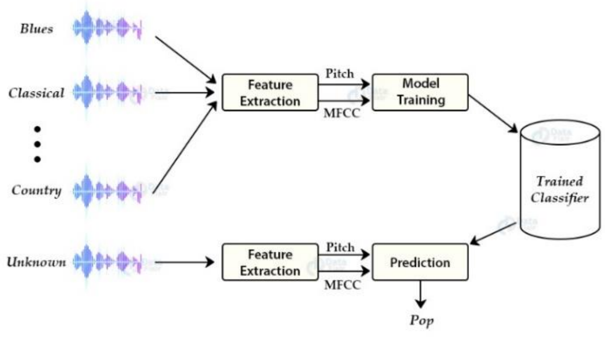


Fig. 1. Music classification process

B. Research Status at Home and Abroad

The earliest research on music classification can be traced back to the 1990s. Due to the copyright problem of music itself, it was difficult to collect and label music data. At that time, there was no large-scale public music labeling data set, so the research on music classification mostly used the data set collected by the authors themselves. And the research object is also limited to the genre, emotion and other fixed categories of music. Before the advent of deep learning technology, the research on music classification mainly focuses on the selection of special collection and classifier. Matityah et al. [1] proposed in 1995 the method of using neural network to classify music. The method first used Fourier transform [2] to process the original audio signal, then converted it to a logarithmic scale, and then input it into the neural network for classification. In 1997, Foote et al. [3] also used K-nearest neighbor algorithm to classify music, taking Mayer cepstrum coefficient and energy as selected audio features. Yang et al. [4] proposed a music classification method combining Gaussian mixture model and support vector machine in 2006.

With the emergence of deep learning technology, classification methods based on deep neural networks began to emerge. For example, Weninger et al. [5] extracted the underlying features from the sound spectrum at the interval of one second, and calculated the regression coefficient, percentile and other statistical features based on these underlying features as the input of the recurrent neural network. The experiment showed that the model based on the recurrent neural network was superior to support vector machine and multi-layer perceptron. Li [6] et al. designed a network with three one-dimensional convolution layers for music genre classification using the Meir frequency cepstrum coefficient as the input of the network, and confirmed that one-dimensional convolution has excellent audio feature extraction capability.

III. Related Works

A signal is a variation in a quantity over time. For audio, the quantity that varies is air pressure. We can represent a signal digitally by taking samples of the air pressure over time. We are left with a waveform for the signal.

- Librosa

Librosa is a python library that allows us to extract waveforms from audio files along with several other features. This is the primary package that will be used for this project.A signal is a variation in a quantity over time. For audio, the quantity that varies is air pressure. We can represent a signal digitally by taking samples of the air pressure over time. We are left with a waveform for the signal. Librosa is a python library that allows us to extract waveforms from audio files along with several other features. This is the primary package that will be used for this project.

- Fast Fourier Transform (FFT)

An audio signal is comprised of several single-frequency sound waves. When taking samples of the signal over time, we only capture the resulting amplitudes. The Fourier transform is a mathematical formula that allows us to decompose a signal into it's individual frequencies and the frequency's amplitude. In other words, it converts the signal from the time domain into the frequency domain. The result is called a spectrum. The fast Fourier transform is an efficient way to compute the Fourier transform.

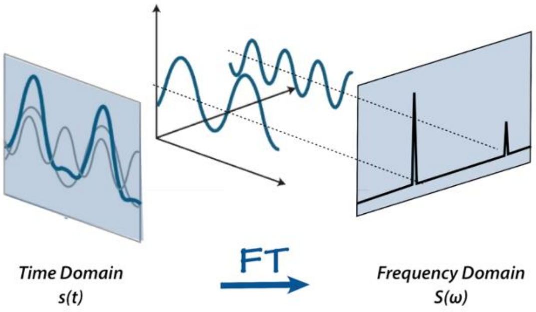


Fig. 2. Fourier transform (FT)

The figure shown is a Fourier transform (FFT) after the signal spectrum diagram

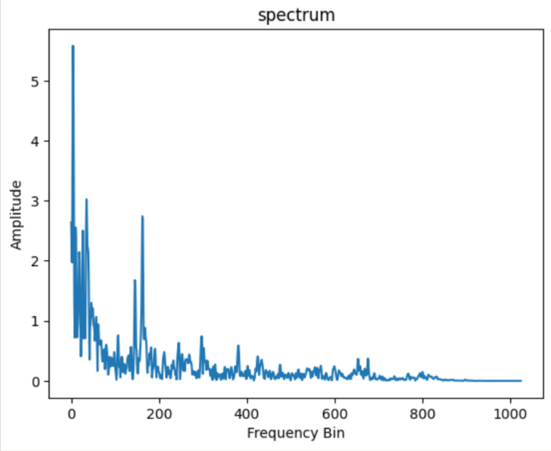


Fig. 3. Fast Fourier Transform (FFT)

- Mel Frequency Cepstral Coefficients (MFCC)

MMCCs are commonly used features in the field of music information retrieval (MIR). They are tyically used to measure timbre.

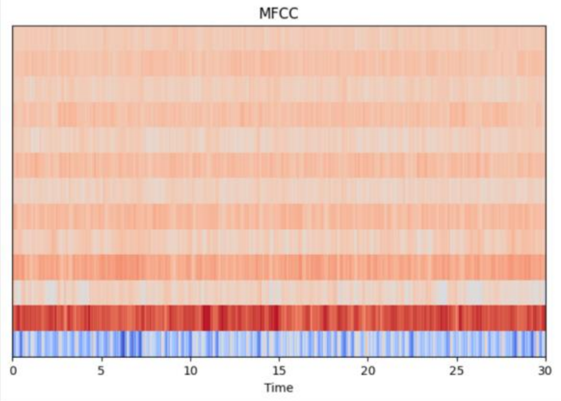


Fig. 4. Mel Frequency Cepstral Coefficients (MFCC)

- Convolutional Neural Networks (CNN)

A neural network is a model structure used to simulate the human brain, for which it is also known as an artificial neural network (ANN) [7]. ANN is made up of a large number of interconnected neurons. Each layer has multiple neurons and no connection between each neuron, while the neurons of the adjacent layer are fully connected, that is, the information received by the neurons of each layer is related to all the neurons of the previous layer. Although this structure has a strong learning ability, due to the fully connected mode between neurons, large-scale parameters in the model have to be considered. In the 1960s, Hubel and Wiesel found that neurons in the cat cerebral cortex could carry out local perception information, which was different from the existing neural network structure in that neurons in adjacent layers only needed to be partially connected, which greatly reduced the scale of parameters and effectively reduced the complexity of the neural network structure. Subsequently, Convolutional Neural Networks (CNNS) are proposed on this basis [8]. At first, CNN was used by researchers in the field of image recognition and obtained a lot of achievements. Especially, the successful application of classical structure LeNet5 in handwritten character recognition has attracted the attention of many scholars [9]. Later, CNN gradually developed rapidly in the fields of speech recognition and face recognition. In recent years, CNN has gone deep into the field of text analysis and has also gained recognition from researchers.

The CNN structure mainly consists of four layers, which are input layer, convolutional layer, pooling layer and fully connected layer in turn. As shown in the figure below, a CNN structure can contain multiple convolutional layers and pooling layers, and the figure includes two convolutional layers and two pooling layers. From the input layer to the fully connected layer, the output of each layer is used as the input of the next layer.

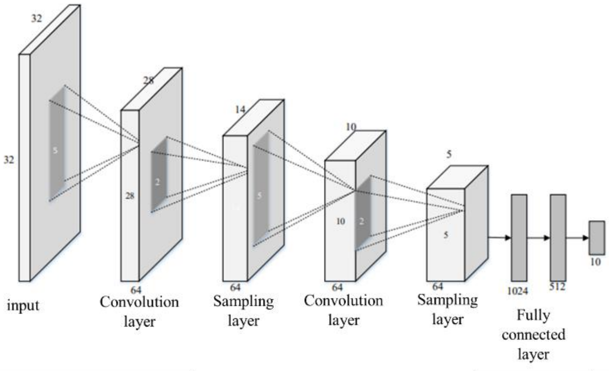


Fig. 5. Convolutional neural network structure

(1) The convolution layer belongs to the feature extraction layer, and the local features of the input layer are extracted through the filter. The specific convolution operation process is shown in formula. The filter size is n*n, b is a bias of the filter, and the filter size is 5*5 in the figure. After extracting local features, it is necessary to use activation functions to normalize the extracted features, even if the final value ranges from 0 to 1. Activation functions include Sigmoid, ReLu, etc.

$$\mathbf{y} = \sum_{\mathbf{i=1}}^{\mathbf{n \times n}} \mathbf{x_i} \times \mathbf{w_i} + \mathbf{b}$$

(2) The features extracted by the convolution layer are prone to overfitting, so it is necessary to use the pooling layer for sampling. The main function is to convert the high-dimensional features into low-dimensional medium. Generally, there are two types of maximum sampling and average sampling. As shown in the figure, the sampling is in the 2*2 region, xi represents the value of the input pixel in the region, and yi represents the corresponding output value after sampling.

$$\mathbf{y_i} = \frac{\mathbf{1}}{\mathbf{4}} \sum_{\mathbf{i=0}}^{\mathbf{3}} \mathbf{x_i}$$

$$\mathbf{y_i} = \max\{\mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3}, \mathbf{x_4}\}$$

(3) The fully connected layer is the last layer of the CNN structure, and the feature vector output through the fully connected layer is used as the basis for classification or retrieval.

CNN has the following obvious advantages: (1) The alternate appearance of multiple convolution layers and sampling layers in the structure is helpful to extract fine-grained features; (2) Feature extraction and classification can be performed simultaneously in the training process; (3) Extraction of local features and weight sharing between layers can reduce the number of parameters, thus reducing the computational complexity.

IV. MUSIC CLASSIFICATION

A. *The GTZAN dataset*

The dataset I used was the GTZAN [10] Genre Collection (linked above) which was used in a well known paper on genre classification in 2002. Due to copyright issues in music itself, more than half of the research on music classification uses data sets that are privately owned by the authors. [11] These data sets have never been published. The experiment in this chapter adopts GTZAN data set, which is widely used to verify the performance of music classification methods and is the most popular music classification data set. The GTZAN dataset has 10 music genre categories, each genre category has 100 audio samples, the sample duration is 30 seconds, and the sample rate is 22050Hz. The format of the files were .wavs, so I was able to use the librosa library to read them into the notebook.

B. *Data Preprocessing*

mel spectrograms can be thought of as visual representations of audio signals. Specifically, it represents how the frequency spectrum changes over time. Some of the special differences in genres can be represented in the mel spectrogram, which means that they can be excellent features.

In this paper, Meir sound spectrum is selected as the sound spectrum feature. The extraction process of Meir sound spectrum mainly includes the following three steps:

1) The sound signal is divided into frames and added Windows.

2) The sound spectrum is obtained by Short time Fourier Transform (STFT).

3) The sound spectrum is then generated by the Mayer filter bank.

First of all, the music sound signal is performed by short-time Fourier transform, and then the frequency on the amplitude spectrum is transformed by the Meir scale, and then the amplitude is converted by the Meir filter, and the result is the Meir spectrum representation of each frame, and then the sound spectrum within the analysis window length is pieced together to get the corresponding Meir spectrum.

What I did was basically turn the problem into an image classification task. Convolutional neural network (CNN) has a good performance in this aspect. This leads to the main problem of my project: using mel spectrum map to identify music genres for image classification task based on convolutional neural network.

Figure 6 shows the Mayer spectra of six songs with different genre categories. It can be seen that there are obvious differences in the spectra of different categories of music.

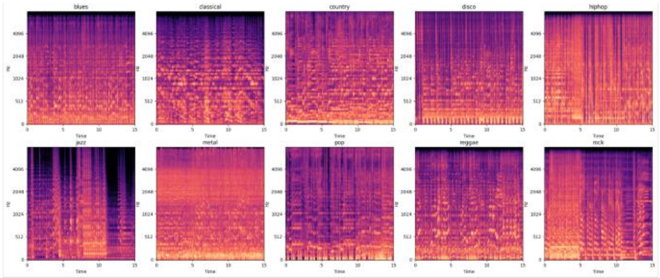


Fig. 6. Snippets of Mayer's vocal spectrum for songs of different genres

Figure 7 shows heat map colors: the color of each cell represents the correlation between two variables. Dark purple indicates a low or negative correlation, while bright red indicates a high positive correlation.

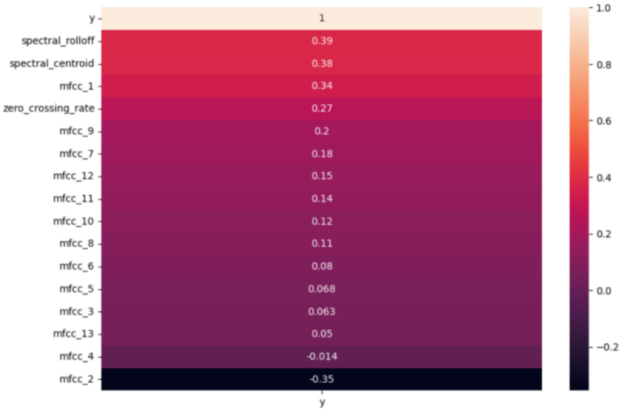


Fig. 7. Map colors

For the CNN model, I wrote a function that calculates the mel spectrum of each audio file in a given directory, reshapes them so that they all have the same size, and stores them in a numpy array. It also creates a set of corresponding targets that redefaults the type label. This function returns both feature array and target array.

C. Data Cleaning

There wasn't much data cleaning to be done. Thankfully, the audio files were all the same length, so I didn't have to deal with different song lengths. Really the only data cleaning that had to be done was mapping the genre labels to numeric values. For some exploraory analysis, I also graphed one mel spectrogram for each genre to see what they looked like. Seeing the differences gave me more confidence that a CNN would perform rather well.

I chose 20% for testing. Before building a model, you must perform several steps:

1.The values of the mel spectrogram should be scaled so that they are between 0 and 1 to improve computational efficiency.

2. The target value must be one-hot encoded before it can be fed to the neural network.

D. Modeling

The final CNN achieved a training score of 84% and a testing score of 68%. I tried several different architectures to improve the model, most of which achieved 55 to 65 percent accuracy, but I couldn't do better. Increasing the number of epochs likely wouldn't help because it became increasingly overfit after about 15 epochs.

Here is a summary of the final architecture for the CNN:

Input layer: 128 x 660 neurons (128 mel scales and 660 time windows)

Convolutional layer: 16 different 3 x 3 filters

Max pooling layer: 2 x 4

Convolutional layer: 32 different 3 x 3 filters

Max pooling layer: 2 x 4

Dense layer: 64 neurons

Output layer: 10 neurons for the 10 different 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.

E. Result

The performance of the convolutional neural network in music classification in this paper is shown in Figure 8 and 9, where the blue line represents the accuracy and loss function values on the training set, and the red line represents the accuracy and loss function values on the verification set.

It's not hard to see from the picture. The convergence rate of this model is relatively slow. In the experiment, the convergence state is basically achieved on the basis of Epoch 15, and the classification accuracy of about 90% is achieved on the verification set. With the increase of the number of iteration rounds, the model may reach more than 90% classification accuracy. Finally, the model can achieve a classification accuracy of about 68% on the test set, which is still a good result compared with some simple convolutional neural networks.

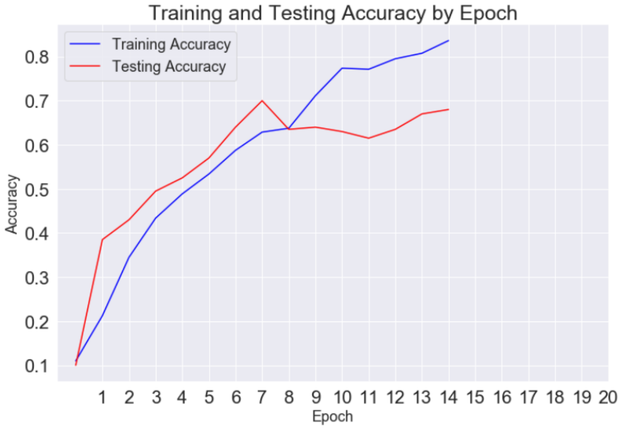


Fig. 8. Accuracy on training set and validation set

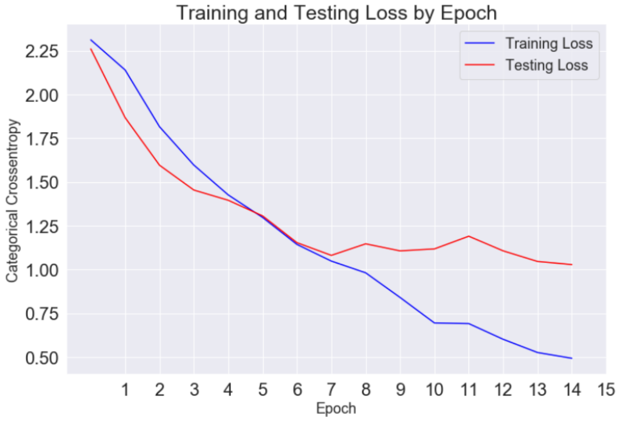
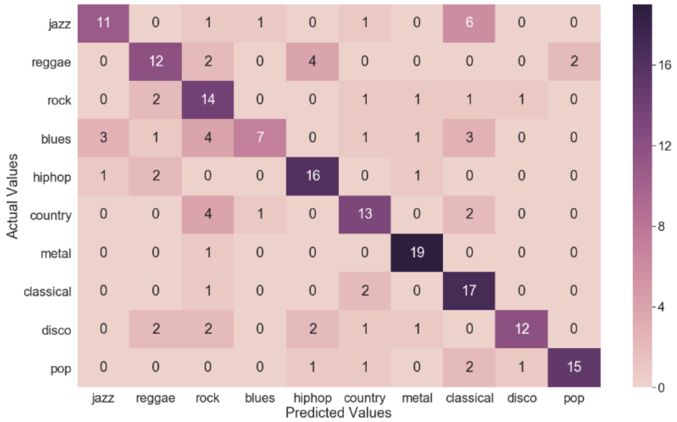


Fig. 9. Change in the loss function

To look deeper into what was happening with the model, I computed a confusion matrix to visualize the model's predictions against the actual values. What I found was really interesting!



	jazz	reggae	rock	blues	hiphop	country	metal	classical	disco	pop
jazz	11	0	1	1	0	1	0	6	0	0
reggae	0	12	2	0	4	0	0	0	0	2
rock	0	2	14	0	0	1	1	1	1	0
blues	3	1	4	7	0	1	1	3	0	0
hiphop	1	2	0	0	16	0	1	0	0	0
country	0	0	4	1	0	13	0	2	0	0
metal	0	0	1	0	0	0	19	0	0	0
classical	0	0	1	0	0	2	0	17	0	0
disco	0	2	2	0	2	1	1	0	12	0
pop	0	0	0	0	1	1	0	2	1	15

Fig. 10. Confusion matrix

V. DISCUSSION

Overall, with an accuracy of 68% for a 10-genre classification task (10% baseline), I can confidently say that the computer was able to learn some distinguishing factors between the different musical genres using a CNN. After diving deeper into the confusion matrix, I found that the computer's mistakes were similar to those that a human might make. This gives me even more confidence that the computer was actually learning the genres to at least some extent.

REFERENCES

[1] Matityaho B, Furst M. Neural network based model for classification of music type[A]. Eighteenth Convention of Electrical and Electronics Engineers in Israel[C]. 1995: 4.3. 4/1-4.3. 4/5.

[2] Pillay P, Bhattacharjee A. Application of wavelets to model short-term power system disturbances[J]. IEEE Transactions on Power Systems, 1996, 11(4): 2031-2037.

[3] Foote J T. Content-based retrieval of music and audio[A]. Multimedia Storage and Archiving Systems II. International Society for Optics and Photonics[C]. 1997: 138-147.

[4] Yang Y H, Liu C C, Chen H H. Music emotion classification: A fuzzy approach[A]. Proceedings of the 14th ACM international conference on Multimedia[C]. 2006: 81-84.

[5] Weninger F, Eyben F, Schuller B. On-line continuous-time music mood regression with deep recurrent neural networks[A]. 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)[C]. 2014: 5412-5416.

[6] Li T L H, Chan A B, Chun A H W. Automatic musical pattern feature extraction using convolutional neural network[A]. International MultiConference of Engineers and Computer Scientists 2010[C]. 2010: 546-550.

[7] Yegnanarayana B. Artificial neural networks [M]. New Delhi,India :Prentice-Hall ,2009.

[8] Hubel D H, Wiesel T N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex.[J]. J Physiol, 1962, 160(1):106-154.

[9] Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.

[10] Tzanetakis G, Cook P. Musical genre classification of audio signals[J]. IEEE Transactions on speech and audio processing, 2002, 10(5): 293-302.

[11] Sturm B L. A Survey of Evaluation in Music Genre Recognition[A]. adaptive multimedia retrieval[C]. 2012: 29-66