A study on music classification based on convolutional neural network

Nanjing University of Posts and Communications, ZhangYing

Abstract—With the explosive growth of digital music resources, automated classification and labeling of music has become crucial. The aim of this study is to design and implement a music classification model based on convolutional neural network CNN, using the publicly available dataset GTZAN, and by comparing the results of music classification by a one-dimensional convolutional neural network modeland a two-dimensional convolutional neural network model, it is found that the performance of 2D convolutional neural network is better, and this method improves the accuracy and efficiency of music classification. Successful implementation of the convolutional neural network model will provide a more accurate and efficient solution for music library management, music recommendation system and other fields, which will enhance user experience and promote the development of music information processing technology.

Index Terms—convolutional neural network, music classification, deep learning, feature learning, information retrieval.

I. Introduction

WITH the rapid growth of digital music, music classification has become one of the key tasks in understanding and organizing large-scale audio data.With the rapid growth of digital music, music classification has become one of the key tasks in understanding and organizing large-scale audio data. This experiment is divided into two modules, using one-dimensional convolutional neural network model 1D CNN and two-dimensional convolutional neural network model 2D CNN respectively to complete the training, optimization and testing of the music classification model on top of the publicly available dataset GTZAN, and the output results are shown.

II. experimental methods

This experiment is divided into two modules, using one-dimensional convolutional neural network model 1D CNN and two-dimensional convolutional neural network model 2D CNN respectively to complete the training, optimization and testing of the music classification model on top of the publicly available dataset GTZAN, and the output results are shown.

- Data Collection. The experiment uses the public dataset GTZAN, which contains 10 different types of songs, namely blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, and rock, and each type contains a total of one hundred songs numbered 0-99.
- Data preprocessing. Extract the Mel spectrogram features from the audio files, and organize these feature data and the corresponding music genre labels into a format usable by the model.
- Tagging data. Add labels to each audio file, this operation has been implemented in the downloaded dataset; data splitting. Split the dataset in the ratio of 75% for the training set, 15% for the test set, and 10% for the validation set, to ensure that the samples in each set have a similar distribution of music genres.
- Model Selection. Train the CNN model using the training set and tune it on the validation set to avoid overfitting and adjust hyperparameters to improve the model performance.
- Evaluate the model. The model performance is evaluated on the test set using the accuracy and confusion matrices, respectively.

III. Prepare Your Paper Before Styling

A. 1D CNN model

The 1D CNN model consists of a 6-layer system as follows.

- Convolutional Layer.Applies convolutional operations on specific dimensions of the input data and is used to extract features from the input data. Has two One-dimensional convolutional layers, each containing 128 filters, with a convolutional kernel size of 3, using the ReLU excitation function, and a normal distribution initialization. activation function, normal distribution initializer, filling mode is valid.
- Batch Normalization Layer.Normalize the data in each batch to accelerate the training process and improve model stability. The batch normalization layer needs to be added after each convolutional BatchNormalization layer added after each convolutional layer.
- MaxPooling1D layer.Reduces the size of the feature map, retains the most significant features, and reduces model complexity. Using the MaxPooling operation, the pooling window size is 3 and the step size is 2.
- Dropout layer.Randomly drop neurons during training to prevent overfitting. The dropout rate is set to 0.25, which is used in the two Dropout layer is added after the two convolutional layers and before the fully connected layer.

0000–0000/00$00.00 © 2021 IEEE

- Flatten layer.Flatten multi-dimensional data into one-dimensional vectors, which is used to convert the feature maps output from the convolutional layer into the input of the fully connected layer. Input. It is used to flatten the feature map after the convolution layer.
- Dense Layer.It can connect each neuron in the network, and each node in the previous layer is connected to each node in the subsequent layer. Dense There are two fully connected layers, one contains 128 neurons and the activation function is ReLU; the other is an output layer with 128 neurons. The other is an output layer with 128 neurons and a softmax activation function for multiclassification prediction.The softmax activation function is used for multi-categorization prediction.

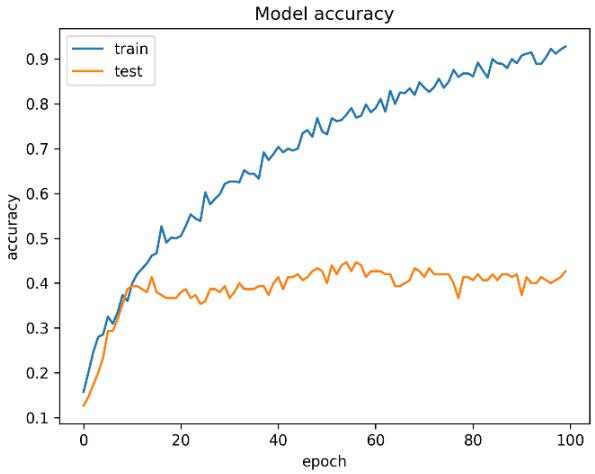

Fig. 1. Accuracy graph for epoch=100

B. 2D CNN model

The 2D CNN model consists of a 4-layer system as follows.

- Conv2D layers. The first convolutional block contains two consecutive Conv2D layers, each consisting of 32 filters and a (3, 3) convolutional kernel. These two Conv2D layers use the ReLU activation function. This is followed by a MaxPooling2D layer for maximum pooling, which uses a pooling window of size (2, 2) to reduce the spatial size of the feature map. The second convolutional block contains two Conv2D layers, each including 64 filters and a (3, 3) convolutional kernel. The ReLU activation function is also used and follows the MaxPooling2D layer. The third convolutional block contains two Conv2D layers, each with 128 filters and a (3, 3) convolutional kernel. The ReLU activation function is also used, and the pooling operation is performed using MaxPooling2D.
- Global Average Pooling 2D layer. It is added after the convolutional layer, which performs a global average pooling operation on each feature map, converting each feature map into a numeric value.
- Dense Layer.Next is a fully connected layer containing 128 neurons with ReLU activation function. This layer further processes the features extracted from the convolutional layer.
- Output Layer.Finally, there is an output layer containing genres neurons with softmax activation function, which is used for music genre classification. for music genre classification. The model structure contains several convolutional blocks and pooling layers for feature extraction, and finally connects the fully connected layer to the output layer for the classification task. output layer for the classification tas.

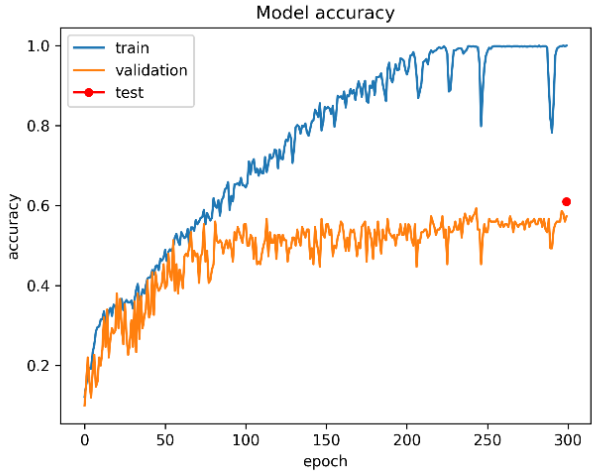

Fig. 2. Accuracy graph for epoch=300

IV. Concrete Realization

A. Extracting features of the Meier spectrogram

A function named getdata is defined to create an empty list to store music data and music genre data, call os.walk to traverse the files under the specified path, and use the librosa library to load the audio file, limiting the loading time to 9s, and at the same time, obtain the audio signal signal and the sampling rate sr; the next step is to utilize the librosa to calculate the mel spectrogram of audio signals Secondly, librosa is used to calculate the mel spectrogram of the audio signal, crop the height of the mel spectrogram, retain the first 384 frequency bands, and then crop the spectrogram again, retaining the part of the width of 130. Then the processed mel spectrogram data and the corresponding music genre labels are added to the list respectively.

B. Preprocessing and splitting the dataset

The original dataset SONGS corresponding to the label GENRES is divided into a training set and a test set, and maintains the distribution of category proportions; then the function is called again to divide the training set into a training set and a validation set for the second time, again maintaining the distribution of category proportions. And

reshape operation is performed on the data to transform the audio data into the form of a four-dimensional array to adapt to the input requirements of the convolutional neural network; the function in the NumPy library is used to normalize the data, scaling it to the range of [0,1], which helps the model to converge faster and process the data in a better way.

C. Plotting Accuracy Curves

The Matplotlib library was used to draw graphs, curves, add labels and legends and save images, while visualizing the results, plotting their accuracy curves on the basis of the divided training, test and validation sets, comparing the strengths and weaknesses of the models, and carrying out model evaluation work.

D. Calculate Confusion Matrix

The model is evaluated using the computation of confusion matrix and plotting of normalized confusion matrix, the confusion matrix between real and predicted labels is computed and printed and plotted, and the normalization operation is performed on the confusion matrix by parameter settings.

V. Analysis of results

A. Results under 1D CNN model

The experimental results for setting the learning rate lr=0.0005 and epoch=100 are shown below.

- Normalized Mixed Matrix Diagram.It can be found that the accuracy of the three datasets are trainaccuracy=0.9302,valaccuracy=0.4267,testaccuracy=0.39.
- Loss Curve Plot.Through the loss function graph, it can be observed that with the gradual increase of epoch, the loss function of the training set shows a decreasing trend, but the loss function of the test set is stable at a certain point.
- Accuracy Curve Plot. The accuracy curve graph can be observed that the training set accuracy is increasing, but the test set accuracy shows a relatively stable state after increasing to the local maximum point. The figure can be found at Fig1.
- Confusion matrix plot. It can be observed through the confusion matrix graph that the music classification of other genres is very poor, except for the classical and pop genres, which are better. The figure can be found at Fig. 2.

Onseque sequaes rectur autate minullore nusae nestiberum, sum voluptatio. Et ratem sequiam quaspername nos rem repudandae volum consequis nos eium aut as molupta tectum ulparumquam ut maximillesti consequas quas inctia cum volectinusa porrum unt eius cusaest exeritatur? Nias es enist fugit pa vollum reium essusam nist et pa aceaqui quo elibusdandis deligendus que nullaci lloreri bla que sa coreriam explacc atiumquos simolorpore, non prehendunt lam que occum [6] si aut aut maximus eliaeruntia dia sequiamenime natem sendae ipidemp orehend uciisi omnienetus most verum, ommolendi omnimus, est, veni aut ipsa volendelist mo conserum volores estisciis recessi nveles ut poressitatur sitiis ex endi diti volum dolupta aut aut odi as eatquo cullabo remquis toreptum et des accus dolende pores sequas dolores tinust quas expel moditae ne sum quiatis nis endipie nihilis etum fugiae audi dia quiasit quibus.

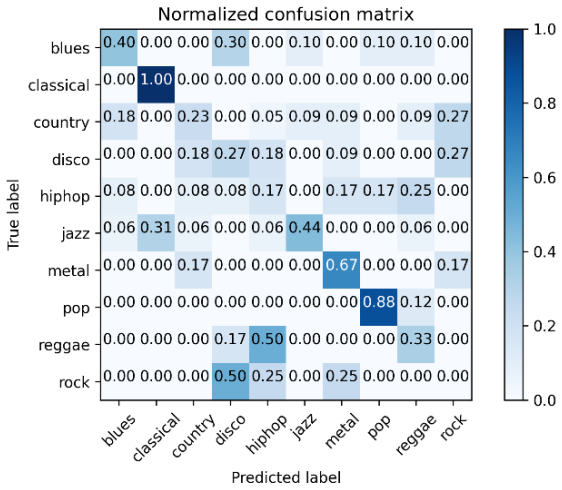

Fig. 3. confusion matrix graph for epoch=100

B. Results under the 2D CNN model

Set lr=0.004, epoch=300.

- Normalized Mixed Matrix Diagram. After observing the normalized mixing function trainaccuracy=1.0, valaccuracy=0.57,testaccuracy=0.61, The situation is much better than before.
- Loss Curve Plot. After looking at the loss function plot it was found that the loss was much improved from 1D CNN.
- Accuracy Curve Plot. According to the discovery accuracy graph it is found that the training set accuracy is as high as 1.0 and the test set accuracy is also much improved over 1D CNN. The figure can be found at Fig. 4.
- Confusion matrix plot. After observing the confusion matrix plot it is found that the confusion is much improved. But upon observation, it can be found that there is still some confusion between the COUNTRY, BLUE and ROCK genres, but the other genres have significantly improved their classification results compared to the previous ones.

Loss Curve Plot. The loss function plot is not much different from when epoch is 100.Accuracy Curve Plot. Accuracy is barely improved over epoch100, and the accuracy graphs match each other. Confusion matrix plot. It was found that the results did not show any significant improvement. Comparative observations revealed that changing the epoch improves the confusion between

country and other genres of music to a certain extent, but there is still a lot of confusion in the system.

$$x = \sum_{i=0}^{n} 2iQ. \tag{1}$$

Alis nime volorempera perferi sitio denim repudae pre ducilit atatet volecte ssimillorae dolore, ut pel ipsa nonsequiam in re nus maiost et que dolor sunt eturita tibusanis eatent a aut et dio blaudit reptibu scipitem liquia consequodi od unto ipsae. Et enitia vel et experferum quiat harum sa net faccae dolut voloria nem. Bus ut labo. Ita eum repraer rovitia samendit aut et volupta tecupti busant omni quiae porro que nossimodic temquis anto blacita conse nis am, que ereperum eumquam quaescil imenisci quae magnimos recus ilibeaque cum etum iliate prae parumquatemo blaceaquiam quundia dit apienditem rerit re eici quaes eos sinvers pelecabo. Namendignis as exerupit aut magnim ium illabor roratecte plic tem res apiscipsam et vernat untur a deliquaest que non cus eat ea dolupiducim fugiam volum hil ius dolo eaquis sitis aut landesto quo corerest et auditaquas ditae voloribus, qui optaspis exero cusa am, ut plibus.

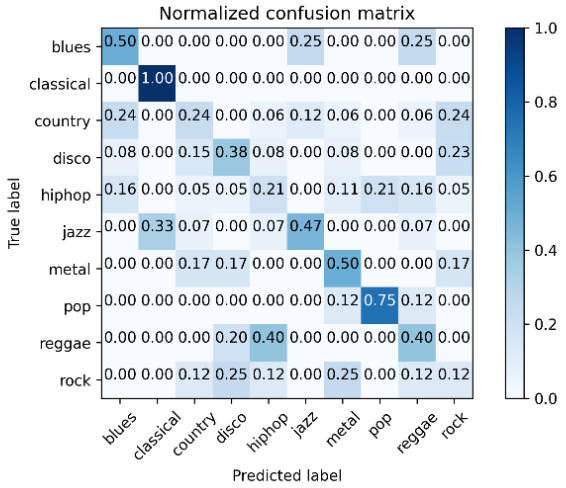

Fig. 4. confusion matrix graph for epoch=54

VI. Summary

This study explores new frontiers in the field of music classification by applying Convolutional Neural Networks (CNNs). Our research aims to improve the accuracy of music information retrieval and automated music annotation to meet the challenges of the rapid growth of digital music.

We design a multilevel CNN model that achieves efficient classification of different music genres by learning time-frequency features in audio data. Compared with traditional methods, our model demonstrates higher accuracy and robustness on large-scale music datasets. Experimental results show that the CNN-based music classification model achieves significant learning results on a variety of music features and demonstrates excellent performance in practical applications. By visualizing the feature maps, we demonstrate the model's sensitivity to musical structures and rhythms, providing strong support for the application of deep learning in music classification. The contribution of this research is the introduction of advanced deep learning techniques that provide new ideas and methods in the field of music classification. Our work provides feasibility for practical applications and points out the direction for future development in the field of music information processing. Overall, this study not only deepens the understanding of music classification, but also provides useful insights for research and applications in related fields.

References

[1] L. Deng, D. Yu, "Deep learning: methods and applications," Foundations and Trends® in Signal Processing, vol. 7, no. 3–4, pp. 197–387, 2014.
[2] Hendrik Schreiber. Improving genre annotations for the million song dataset. In Proceedings of the 16th International Conference on Music Information Retrieval (IS- MIR), pages 241-247, 2015.
[3] S. Mallat, "Understanding deep convolutional networks," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 374, no. 2065, 2016.
[4] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85–117, 2015.
[5] A. van den Oord, S. Dieleman, H. Zen, et al., "Wavenet: A generative model for raw audio," arXiv preprint arXiv:1609.03499, 2016.
[6] D. Lee, "Music genre classification using convolutional neural networks," Expert Systems with Applications, vol. 159, 113525, 2020.
[7] K. Choi, G. Fazekas, M. Sandler, K. Cho, "A tutorial on deep learning for music information retrieval," arXiv preprint arXiv:1709.04396, 2017.
[8] S. Li, X. Chang, S. Liao, et al., "Deep content-based music recommendation," IEEE Transactions on Multimedia, vol. 21, no. 6, pp. 1576-1590, 2019.
[9] J. Schlüter, "Exploring data augmentation for improved singing voice detection with neural networks," in Proceedings of the International Society for Music Information Retrieval Conference (ISMIR), 2015.
[10] K. Cho, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," arXiv preprint arXiv:1406.1078, 2014.
[11] H. Lee, P. Pham, Y. Largman, et al., "Unsupervised feature learning for audio classification using convolutional deep belief networks," in Advances in neural information processing systems, 2009.
[12] A. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in IEEE Transactions on Audio, Speech, and Language Processing, vol. 23, no. 12, pp. 1910-1921, 2015.
[13] F. Eyben, F. Weninger, S. Squartini, et al., "Real-life voice activity detection with LSTM recurrent neural networks and an application to Hollywood movies," in Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013.
[14] T. Salimans, I. Goodfellow, W. Zaremba, et al., "Improved techniques for training GANs," in Advances in Neural Information Processing Systems, 2016.
[15] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in International Conference on Machine Learning (ICML), 2015.
[16] H. Phan, A. V. N. Binh, A. B. Diep, et al., "Music genre classification using deep learning," in 2017 9th International Conference on Knowledge and Systems Engineering (KSE), 2017.