ISSN 2356-5462
http://socj.telkomuniversity.ac.id/ijoict/

Intl. Journal on ICT
Vol. 10, No. 2, Dec 2024. pp. 152-160
doi:10.21108/ijoict.v10i2.978

Music Genre Classification Using Adam Algorithm of Convolutional Neural Network

Fadi Joubran[1], Jean Paul Chaiban[2], Jaafar Al Shall[3], Ramar Aoun[4], Gaby Abou Haidar[5, *], Roger Achkar[6]

[1,2,3,4,5,6]*Department of Computer and Communications Engineering, American University of Science and Technology*
Beirut - Lebanon

*gabouhaidar@aust.edu.lb

Abstract

Even though technology has been evolving rapidly lately, music classification is still definitely a major task in the Music Information Retrieval (MIR) domain. Music genre classification is a key challenge in Music Information Retrieval (MIR), aiming to identify the genre, style, and mood of audio tracks. This study explores the use of Convolutional Neural Networks (CNNs) with the Adam optimizer for music genre classification. We conducted experiments to evaluate the performance of our proposed model, which incorporates advanced machine learning techniques to improve classification accuracy. Our approach involves extracting features from audio files, converting them into Mel spectrograms, and training the CNN model using Python. The results demonstrate a high classification accuracy of 98.5%, significantly improving upon previous methods. Additionally, GPU acceleration enhanced the training speed by five times. Future work includes developing a mobile application for real-time classification and exploring integration with speech recognition technologies.

Keywords: Music Information Retrieval, Music Genre, Python, Music Classification, CNN, Neural Networks, Audio File, Adam Optimizer, Training Process

I. INTRODUCTION

Music classification is considered to be one of the most interesting subjects and a difficult task in the Music Information Retrieval (MIR) domain. It assigns genre, style, mood, and many other types for each track to facilitate managing the music data. Classifying music by genre, style, and mood is essential for managing and discovering songs, but it remains a challenging task due to the diverse nature of music. To improve this process, we use advanced Machine Learning techniques, particularly Convolutional Neural Networks (CNNs) and the Adam optimizer, which help our program learn from audio data to make accurate genre predictions. Our system supports various audio formats and is trained on a wide range of music genres, allowing it to classify songs quickly and accurately. This approach addresses the difficulties of music classification and enhances both speed and precision, making it more effective and user-friendly.

Lately, there has been a huge interest in this subject, with many applications trying to classify music having different and variable accuracies and many issues. These issues are such as selecting the most appropriate feature sets and choosing the best algorithm for the classification. The most effective way to make such application is to use Machine Learning with the usage of neural networks to predict the genre of a song or any audio file and train the program to handle any genre changing. One of the best optimizers in neural networks for such application is the Adam optimizer, and one of the best machine learning languages is Python. Therefore,

Received on 30 Jul 2024. Revised on 07 Aug 2024. Accepted and Published on 04 Dec 2024.

as a first step, Adam optimizer and Python language were used in order to create the music classification program. The proposed methods classify and specifies the genre of any audio file selected on the application using convolutional neural networks. The program supports many audio formats and it has a technique to train and pre-direct itself using a database of the classes and some training examples of each one. Many experiments were done to validate the results, and many modifications were added to maximize the accuracy of the program and its speed. A convolutional neural network was used and thus had great results with high classification accuracy and speed.

II. LITERATURE REVIEW

There has been many different proposals and projects concerning the music classification domain since year 2000 till now. were discussed and proposed earlier, and they all tried to classify music using its retrieved data according to different mechanisms. The story began from the 2000s, and one example is [1] [2] in 2002 when authors began the journey to investigate the music genre classification of audio signals. Authors started by stating the characteristics of music genre; these features are highly correlated to the instruments used, rhythm structure, and the harmonic content of any specific musical piece. And then they stated the importance of the automatic music genre classification and how it simplifies the humans' work and processes in this domain. They also proposed three feature sets: timbral texture, pitch content, and rhythm content. These features are tested and experimented to show how valuable and effective these features are for the automatic music genre classification process. These sets achieved arguably a good work at that time, which is 61% of accuracy [3].

Another work is [4] which was published in 2006. Authors made a survey concerning automatic music genre sorting of music content and proposed three major criteria: specialist systems, unsupervised, and supervised classification. They then discussed the significance of these paradigms and their differences on automatic music genre classification. Experiments and test were done and results were achieved making this proposal a valuable proposal with good results. A third work [5] was published in 2014 proposing the detection of music genre boundaries to achieve music genre classification. The importance of music genre borders was listed, and there has been a discussion about multi-label genre classification task and the single-label genre classification function. So, they proposed dividing the multi-label genre classification task into the single-label genre classification task. This proposal can allow the software to find boundary lines of different music genres. Added to that, single-label genre classification can be used to detect music segments.

In 2017, a paper [6] entitled by "Music Genre Classification with Machine Learning Techniques" was published. It came up with an innovative proposal using a very new technology which is Machine Learning Techniques. And to achieve their goal, they used signal processing techniques to extract the features from the audio file. Then these features are teamed up with the Machine Learning Techniques to make a multiclass classification for music genres. In 2018, another paper [7] was published. It aimed to automatically classify the music genre using the Gaussian Mixture Model (GMM). They tried to use latest technologies that enhance feature sets: Mel-Frequency Cepstral Coefficients (MFCCs), Spectral Roll off, Time Domain Zero Crossings, and Flux. And then GMM is used for the final classification task. The experiment results proved that as number of features used increases, the percentage of classification is significantly enhanced [8] [9] [10].

III. RESEARCH METHOD

A. Steps and Procedure

The steps adopted to bring this study to life are outlined in the following manner, as depicted in the flowchart in Fig. 1, where the program follows four main steps to achieve its objective:

1. The program allows the user to choose an audio file from any browsing location. The program supports many audio formats such as: wav, mp3, au, aac, and many other audio formats that are supported by ffmpeg.

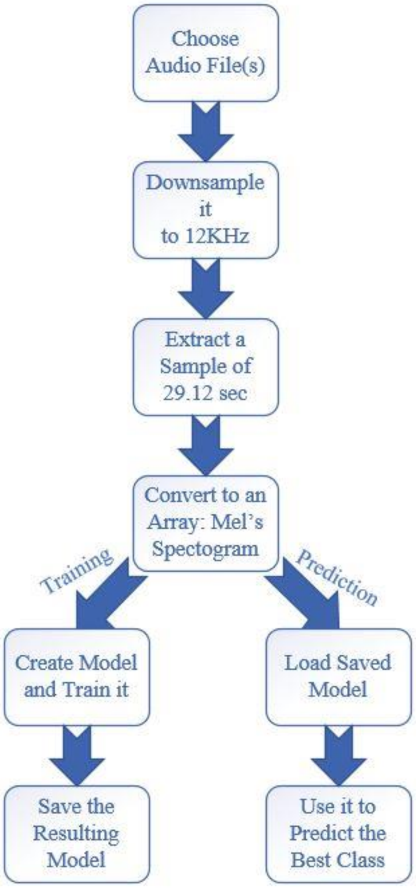
Choose Audio File(s)
Downsample it to 12KHz
Extract a Sample of 29.12 sec
Convert to an Array: Mel's Spectogram
Training
Prediction
Create Model and Train it
Load Saved Model
Save the Resulting Model
Use it to Predict the Best Class

Fig. 1. Flow Chart Steps used in Training and Prediction

2. The audio file is then down sampled to 12KHz. This down sampling is to preserve the information present in this audio signal as much as possible. This helps the program to be more accurate by taking into consideration every detail and every piece of information from this audio signal. Any lost information may result in false classification because a part of the signal was not taken into consideration, and this part may contain the most important features that would help in genre classification. The down sampling also helps the program with the machine learning process. It needs every detail and piece of information present in all audio files to be able to learn and improve. It is improved by being able to judge more correctly and more accurately about the genre of the selected audio files.
3. A sample is extracted from this audio file and it was chosen to be 29.12 seconds.
4. This sample is converted to an array: Mel's Spectrogram. This spectrogram is done by applying Fast Fourier Transform on each sampled signal to have the spectrum, and then the spectrum will pass through Mel-filters [11], [12], [13]. Mel-filters are based on analysis on human sensitivity experiments in which it is noticed that human ears have filters. There is a higher number of filters in the low frequency pass band, while there are less number of filters in the high frequency band. The same applies to the Mel-filters; more in the low frequency scope, and less in the high frequency scope.

To support the program give the best results, there are few things that should have been done. Fig. 2 shows the steps taken to allow the program to know the existing classes that it should choose from. And then the Machine Learning Techniques start to let the program know carefully the specifications of each class alone.

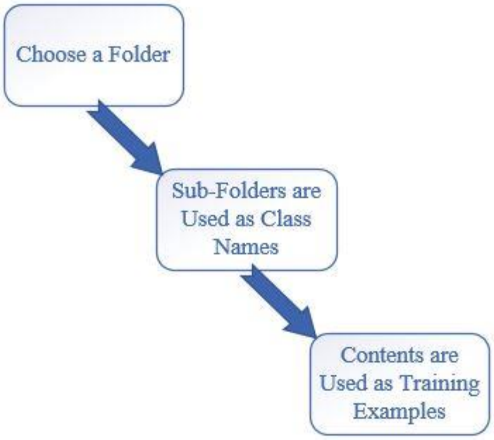
Choose a Folder
Sub-Folders are Used as Class Names
Contents are Used as Training Examples

Fig. 2. Division of Training Data into Classes

This experience in having training examples to study from, will allow the program to adapt with any audio file it faces or challenged to examine. And it will be able to classify its genre and gain more and more experience to improve. The steps are:

1. Choose a specific folder to be the program database and the core of the Machine Learning training process.
2. This folder must have as many sub-folders as the number of classes. Each sub-folder must have a name of one class. And by that, the program will be able to know all the classes and the genres of the music files in general. And from these classes/genres, the program will judge the genre of the audio file the user chooses.
3. In each sub-folder, there will be a number of audio files (more than 100) belonging to the same class/genre. So, if the sub-folder is named “pop”, then it should contain a number of audio/music files whose genre is pop. And the next sub-folder if its name is “jazz”, it should contain a number of audio files whose genre is jazz. And so on.
4. And by that, the program will be able to extract the specifications and features of every class/genre of music. And the audio files in the database will be as training examples in the Machine Learning process that will make the program able to adapt and study all different genres of music and will handle any audio file chosen to be classified. As the number of existing training examples in the sub-folders increase, the accuracy of classifying the genre of audio files increase.

B. System Improvements

There have been some improvements that were done on the system/program to make it more functional and has a better performance. Performance in such programs is characterized by the speed and the accuracy, and both have been paid attention to, in order to improve them and make the program better. In order to boost the accuracy, there was the usage of Machine Learning techniques that helps the program study the genres of music files carefully and apply the training experience on new audio files. Machine Learning techniques were used along with convolutional neural network to be able to stay up-to-date and have the best result, and thus the highest accuracy, hoping to have a perfect program which is a program that has 100% accuracy. Increasing the number of training examples may also increase the accuracy of the program by having more examples the program should study. This will allow the program to train on various and different audio files having many features, and therefore be capable of knowing all the features of every music genre.

But the high increase in training examples may also have a drawback. The high increase in training examples will make the database/dataset of the program go bigger and bigger and thus making the Machine Learning process harder to be implemented since it has to search and study all the dataset more than once to be fully ready. The increase in dataset will increase the time the program needs to classify the genre of the selected audio file. And this increase in time is another term of making the program slower. So, there must be an optimal number of training examples that will make the program accurate enough and yet fast.

Another improvement was done on the speed of the program. Making the program faster will make it more powerful and capable of handling larger datasets. This improvement was the ability to use the GPU instead of the CPU. And as it is widely known, the GPU is much faster than the CPU in rendering and mathematical and geometrical calculations. The GPU has much more cores than the CPU and thus more Arithmetic Logical Units that allow the GPU to run multiple calculations in parallel, while the CPU does them in sequential order. So, the intention was to make the program run over the GPU instead of the CPU to speed up the Machine Learning process and make the program more functional using tensorflow GPU library, Nvidia CUDA, and Nvidia CUDNN.

C. Implementation

As for the implementation, the steps were followed and the improvements were done. The sub-folders were named as the names of the classes. Popular classes were chosen, like 11: Arab, Blues, Classical, Country, Disco, Hip-Hop, Jazz, Metal, Pop, Reggae, and Rock. When booting the program, it requests the dataset, so we choose the folder that contains the classes as sub-folders. And then we choose the audio file we want to classify its

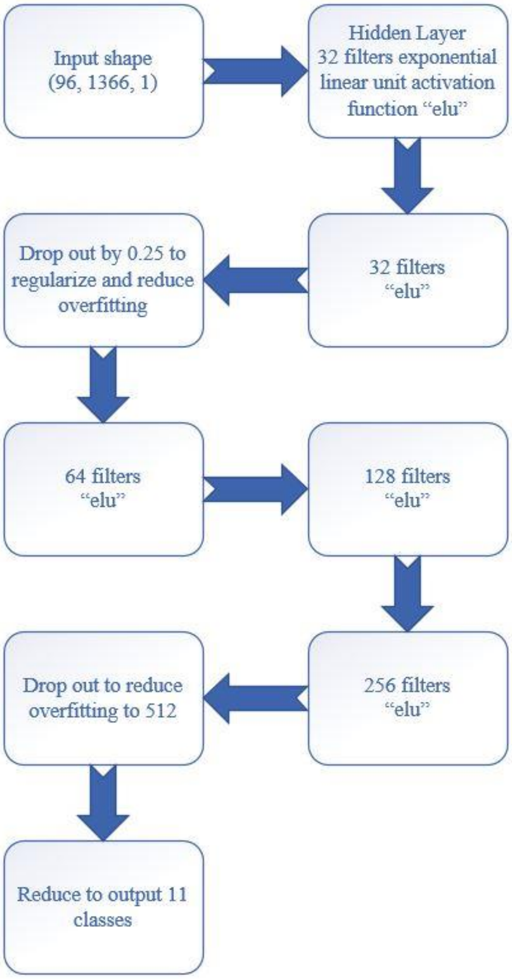

Fig. 3. Model Design

genre. The program decompiles the audio file from audio to binary. The program then loads and converts the audio file using ffmpeg library with a sampling rate of 12KHz. Then it selects a part of the audio file of around 29.19 seconds and converts it into 1366 frames. After that, the audio file is converted to Mel's Spectogram which converts the data into integers and normalizes them using Log function of the amplitude. This process is called the Mel computation. In addition to that, Librosa library was used:

- Normalization: log(a) = Librosa.core.amplitudetodB
- Melgram: Librosa.feature.melspectogram

From the above process, the features of the audio file are plucked out and compared with the features of the classes in the dataset. Here comes the benefit of Machine Learning Techniques that will allow the program to make the right comparison and come up with the correct results. The training process uses two main libraries:

- Tensorflow
- Keras

These libraries are used to create the Machine Learning models and they are mostly used in the deep neural networks and in Machine Learning Techniques. The training process uses input model shape size 96 (Mel Frequency Cepstral Coefficients MFCC) as duration, and 1366 frames for the script file (sample of 29.12 seconds). Adam optimizer then creates the model and saves it. The dataset of the program now contains 20 epochs and a batch size of 64 according to the Python time understanding. A GPU is instead of the CPU as an improvement, which made the program five times faster using a 6GB Nvidia GTX 1060. Training uses the famous seikit-learn library to properly choose examples to train and validate the model and determine training accuracy. Keras library is used to create the convolutional neural network model.

Fig. 3 shows the model design which was used in the program to make things more precise. The input shape sized (96, 1366, 1) passes through a sequence of several layer filters varying between 32, 64, 128, 256, and 512 including three drop out layers that reduce overfitting. And finally, the output is reduced to eleven classes. This model is designed similar to the cifar-10 photo processing model [14], [15]. Cifar-10 datasets are used basically to classify images and pictures. It helps in processing the image and training/prediction of the convolutional neural network model. It is often used in training machine learning and computer vision algorithms and is one of the most widely used datasets in such dataset domain.

IV. RESULTS AND DISCUSSION

After implementation, the program was tested and examined on many different audio files to make sure everything is working perfectly. The improvements that were discussed were also applied and tested. The results that were given before the improvements were compared with the results given after the improvements. One of the examinations / tests that were done was the following:

```
C:/Users/user/Desktop/Maroon_5_-
_Girls_Like_You_ft._Cardi_B-
aJOT1E1K90k.wav

Predicted: pop

Advanced Probability Prediction: pop at
76.87010765075684%, hiphop at 18%,
```

Fig. 4. Result of an Audio File Predictor

As shown in Fig. 4, the chosen audio file is the song "*Girls Like You*" for "*Maroon 5*". The program predicted the genre of the audio file to be "pop" which is correct. But the addition part which was added to the program is the probability prediction which gives the user the probability of each prediction the program did, and then it takes the prediction with the highest probability. In this example, the program predicted the audio file's genre to be "pop" with a probability of 76.87%, and "hiphop" with a probability of 18.56%, and "metal" with a probability of 3.56%. These different predictions and probabilities are because of the various music types in modern audio files. That means that a single audio file of a specific music genre may have in a piece of it some common features of another music genre. This causes the program to have predictions of multiple genres but the highest with no comparison should be to the correct music genre, which in this case is "pop".

GPU usage was also tested successfully and had a great impact and improvement on the program. GPU usage improved the speed of training by five times. And most importantly, the accuracy that was recorded for the program with only 20 epochs was 98.5% in our datasets. And this accuracy percentage is perfect having it very close to the optimal / ideal accuracy which is 100%. All of this was done after the extraction of each music genre features and specifications. The results are:

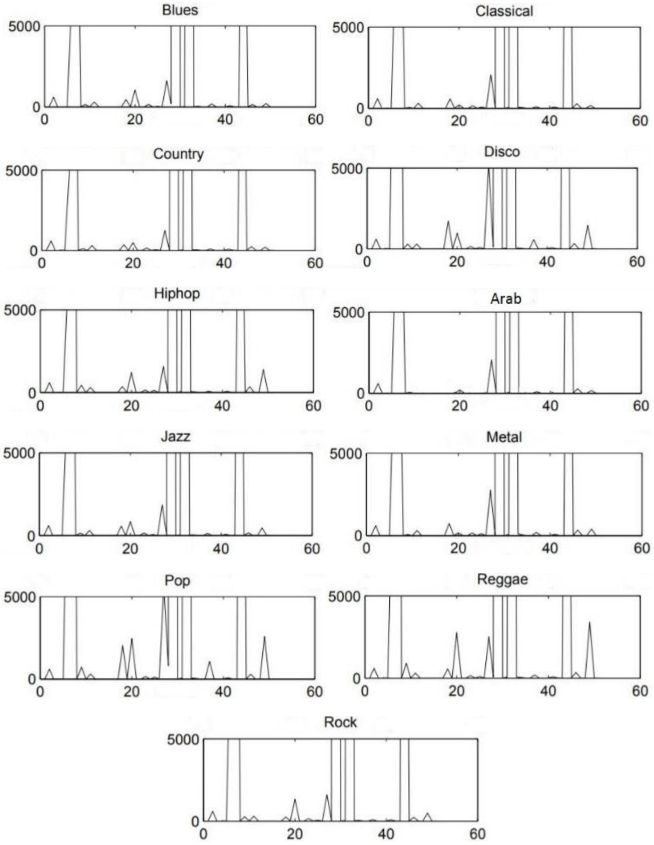

Fig. 5. Difference between each music genre

Fig. 5 shows the difference between each music genre in terms of signal and spectrum. This difference in spectrum identifies the features of each music genre, and as it is shown in the figure, each music genre has special and unique features and characteristics. These unique characteristics identifies the music genre and help the program specify whether this audio file is pop, rock, or classical, etc.

V. CONCLUSION AND FUTURE WORK

The major contribution of this paper is to stay up-to-date and make a music genre classification using the latest and promising technologies and having the best accuracy we can get. This led to the usage of Machine Learning Techniques along with the convolutional neural networks. Python was chosen to be the programming language to facilitate machine learning process including the datasets. The program was given training examples to identify the difference between each music genre and to help within the Machine Learning process. These training examples led to the extraction of the features and characteristics of each music genre through their spectrum. The program compares the audio file (after a sequence of sampling and filtering) with the music genres' features. The most probable music genre is the one chosen and taken to be the result. The accuracy of the program in guessing the right music genre reached 98.5% which is considered as a great accuracy in this domain. In addition to that, the speed of the training and identifying processes was improved by five times with the usage of the GPU instead of the CPU.

As future work, and with this fast-growing technological world, there may be more upcoming and flourishing techniques and methods for music genre classification which may facilitate and improve the process more and more. But with the presence of machine learning, this program/system can be sustainable and functional for a good period of time with continuous improvements and modifications before the arrival and replacement of totally different methods. In the future, a mobile application can be done, for android and iOS devices. Such applications can allow the usage of such service on mobile phones. And widely known, smart phones are almost available for everyone nowadays, so it is a lot easier for them to have an application that can classify their audio files without the need to transfer these audio files to a computer to test them using the program. Another future contribution may be the addition of speech recognition to the program and the application. It can give the user the lyrics of the song, or in other words, convert the spoken language in the audio file into a text.

ACKNOWLEDGMENT

We would like to thank Dr. Roger Achkar - Dean of the Faculty of Engineering and Computer Science at AUST and Dr. Gaby Abou Haidar – Director – AUST Zahle Branck and Coordinator of the Faculty of Engineering and Computer Science for their valuable input and support.

REFERENCES

[1] G. Tzanetakis, P. Cook, "Musical genre classification of audio signals", https://ieeexplore.ieee.org/abstract/document/1021072, 07 November 2002.

[2] J. Saunders, "Real time discrimination of broadcast speech/music," inProc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP), 1996,pp. 993–996

[3] F. Pachet and D. Cazaly, "A classification of musical genre," in Proc.RIAO Content-Based Multimedia Information Access Conf., Paris,France, Mar. 2000

[4] N. Scaringella, G. Zoia, D. Mlynek, "Automatic genre classification of music content: a survey", https://ieeexplore.ieee.org/abstract/document/1598089, 24 April 2006.

[5] Hiroki Nakamura, Hung-Hsuan Huang, Kyoji Kawagoe, "Detecting Musical Genre Borders for Multi-label Genre Classification", https://ieeexplore.ieee.org/document/6746860, 24 February 2014.

[6] Ali Karatana, Oktay Yildiz, "Music genre classification with machine learning techniques", https://ieeexplore.ieee.org/document/7960694, 29 June 2017.

[7] Chandanpreet Kaur, Ravi Kumar, "Study and analysis of feature based automatic music genre classification using Gaussian mixture model", https://ieeexplore.ieee.org/document/8365395, 28 May 2018.

[8] R. Duda, P. Hart, and D. Stork, Pattern Classification. New York:Wiley, 2000

[9] D. Perrot and R. Gjerdigen, "Scanning the dial: An exploration of fac-tors in identification of musical style," in Proc. Soc. Music PerceptionCognition, 1999, p. 88, (abstract).

[10] D. Pye, "Content-based methods for the management of digital music,"in Proc. Int. Conf Acoustics, Speech, Signal Processing (ICASSP), 2000.

[11] Y. He, H. Chen, D. Liu, and L. Zhang, "A Framework of Structural Damage Detection for Civil Structures Using Fast Fourier Transform and Deep Convolutional Neural Networks," *Appl. Sci.*, vol. 11, no. 19, p. 9345, Oct. 2021, doi: 10.3390/app11199345.

[12] E. Rajaby and S. M. Sayedi, "A structured review of sparse fast Fourier transform algorithms," *Digit. Signal Process.*, vol. 123, p. 103403, Apr. 2022, doi: 10.1016/j.dsp.2022.103403.

[13] A. Ustubioglu, B. Ustubioglu, and G. Ulutas, "Mel spectrogram-based audio forgery detection using CNN," *Signal Image Video Process.*, vol. 17, no. 5, pp. 2211–2219, Jul. 2023, doi: 10.1007/s11760-022-02436-4.

[14] C. Jiang and G. Goldsztein, "Convolutional Neural Network Approach to Classifying the CIFAR-10 Dataset: How can supervised machine learning be applied as a technique on a convolutional neural network to solve the image classification problem of recognizing and classifying images in the CIFAR-10 dataset?," *J. Stud. Res.*, vol. 12, no. 2, May 2023, doi: 10.47611/jsrhs.v12i2.4388.

[15] College of Electrical & Information Engineering, Southwest Minzu University, Chengdu 610041, China and X. Lv, "CIFAR-10 Image Classification Based on Convolutional Neural Network," *Front. Signal Process.*, vol. 4, no. 4, Oct. 2020, doi: 10.22606/fsp.2020.44004.