International Journal of Early Childhood Special Education (INT-JECSE)
DOI:10.9756/INTJECSE/V14I5.153 ISSN: 1308-5581 Vol 14, Issue 05 2022

MUSIC GENRE CLASSIFICATION USING CONVOLUTIONAL NEURAL NETWORKS

Mr. G M Subhani
Asst. Professor
Dept. of. CSE
St.Peter's Engineering College Hyderabad, TS, India
gm.subhani90@gmail.com

Perala Shravya
UG Scholar
Dept. of. CSE
St.Peter's Engineering College Hyderabad, TS, India
shravyaanju1234@gmail.com

Gorighe Akhil kumar
UG Scholar
Dept. of. CSE
St.Peter's Engineering College Hyderabad, TS, India
gorigheakhilkumar@gmail.com

Chitumalla Hrithika
UG Scholar
Dept. of. CSE
St.Peter's Engineering College Hyderabad, TS, India
ch.hrithika24@gmail.com

Chimalpade Ajay shrinivas
UG Scholar
Dept. of. CSE
St.Peter's Engineering College Hyderabad, TS, India
ajpatil999@gmail.com

***Abstract*—Musical genres are different classes created by humans to categorize pieces of music. A music is classed by some common characteristics shared by its members. These characteristics are associated with music instrumentation, rhythm, and harmonic content of the music. The music industry may find genre classification to be a crucial topic with many practical applications. The ability to quickly categorise songs in a playlist or library by genre is a crucial feature for any music streaming or purchasing business. We all know that there are hundreds of music genres and millions of songs in each genre. This Music Genre Classification uses Convolutional Neural Network techniques and gives an approach to classify the music automatically. Our approach for music genre classification explores both neural networks and algorithms such as KNN.**

Keywords— Convolutional Neural Network; GTZAN; genre; spectrogram; librosa; json;

I. INTRODUCTION

In this Music Genre Classification we used the GTZAN data set which is a collection of around thousand audio files with 30 seconds duration each and 10 classes where data is classified accordingly like Classical, Blues, Disco, Country, Metal, Pop, Hip-hop, Jazz, Rock, Reggae. We employed KNN, where the K-nearest neighbor's most common class is used to classify the music sample's genre. Mel-spectrograms are created from each audio file. You might imagine the Mel spectrogram as a visual representation of an audio signal. Specifically, it represents how the spectrum of frequencies vary over time. A CNN model is provided with these images as input and we predict genre using the mel-spectrograms. This work presents a deep learning approach for automatic genre classification of the audio signal. This system is developed using Convolutional Neural Network (CNN) to acknowledge the genres.

II. LITERATURE REVIEW

Artificial neural networks are capable of dealing with complex high level knowledge from raw inputs. Its hypothesis function has non-linear representation. Training a neural network with several hidden layers is challenging. We have compared several music datasets and selected GTZAN Music Genre Collection which contains 1000 sound records. All the records are 22050 Hz Mono 16bit audio files. When compared to other datasets with millions of songs, the dataset involved here is very small. In fact, most of the The GTZAN dataset, for example, is still the centre of most music research. Mel-frequency Mel-spectrogram and Cepstral Coefficient

(MFCC) are two tools frequently used to categorise music by genre. Because they can extract features from raw data for the learning process. Most of the research on genre classification focuses only on music features and not on lyrics due to the difficulty of collecting large-scale lyric data.

III. METHODOLOGY

A. *DataSet*

MARSYAS is a sound handling open source website with a focus on sound information data users. We used the GTZAN dataset, which contains a collection of 1000 audio tracks, to classify the musical genres. This dataset has ten classes, each of which has 100 tracks. Each track has a .wav file extension. There are ten different genres of audio files on it.

- Classical
- Blues
- Disco
- Country
- Metal
- Pop
- Hip-hop
- Jazz
- Rock
- Reggae

Many libraries are used to classify the music according to genre.

✓ About NumPy:

NumPy could be a Python extension package written largely within the C programing language. it is a Python module for conducting multidimensional and single-dimensional array element processing and numerical computations. NumPy arrays outperform traditional Python arrays in terms of speed. it's extremely useful in Machine Learning for basic scientific computations. It's especially handy for algebra, the Fourier transform, and random number functions. Tensor could be a high-end library.

✓ About SciPy:

An open source, BSD-licensed library for mathematics, science, and engineering, SciPy is a scientific library for Python. The NumPy library, which offers simple and quick N-dimensional array manipulation, is a prerequisite for the SciPy library. The SciPy library was created primarily so that it would be compatible with NumPy arrays. It provides a wide range of simple and efficient numerical techniques, including methods for numerical integration and optimization. This is frequently an introductory tutorial that goes over the fundamentals of SciPy and explains how to accommodate all of its different modules.

✓ About JSON:

The JSON phrase describes the conversion of information into a string of bytes for storage or transmission over a network. The JSON library in Python employs the dump() function to convert Python objects into the appropriate JSON objects to handle the information flow within a file, making it simple to write data to files.

✓ About OS:

Python's OS module offers tools for working with the package. OS is included in the basic utility modules for Python. A portable method of using functionality that depends on software packages is provided by this module. There are many ways for the OS to interact with the file system.

✓ About Pickle module:

A Python object structure is serialised and deserialized using the Python pickle package. In Python, any object is frequently pickled in order to be saved on disc. Pickle "serialises" the object before sending it to a file, which is what it does. A Python object (list, dict, etc.) can be converted into a personality

stream by pickling. The idea is that everything needed to recreate the article in another Python script is contained in this character stream.

- ✓ About random module:

  The pseudo-random variables are produced using the built-in random module. It can be used to do random actions like generating a random number, choosing random items from a list, randomly shuffling items, etc. Add from random import * to the top of your code or type it into the Python shell to access the random module. Open the programme in a second terminal by running randOps.py in vim. Keep in mind that if you run the application again, the results are random.

- ✓ About Operator module:

  The operator module offers functions that are comparable to the operators in Python. These functions come in handy when it's necessary to store, pass as arguments, or return callables as function results. When it's required to store, pass as arguments, or return callables as function results, these functions come in useful. The special methods have the same names as the functions in operator. For instance, names with and without the preceding and trailing double underscores are both acceptable for the same function, as are names like operator.add(a,b) and operator. add _(a,b) return a+b.

- ✓ About math library:

  We have access to certain common arithmetic functions and constants in Python thanks to the Python Math Library, which we may use throughout our code to perform more intricate mathematical operations. You don't need to install the library to use it because it is a built-in Python module.

Three different modules are used for this classification.

- ✓ Data Gathering Module:

  The collection of training and test data via the internet is the topic of this subject. Then the extracted data set from the internet is assembled. Hence in this module initially the gathered audios consisting of both linguistic content and discarding noise. The audio recordings are then extracted to reveal their features and constituent parts, which are then utilised to train the model.

- ✓ Model Development module:

  This module deals with model building. After gathering and segregating the data in the gathering phase, a deep learning model is built. Here we use the K-nearest neighbor algorithm to build the model. After the model is built it is then trained in the training module by fitting dataon it.

- ✓ Training Module:

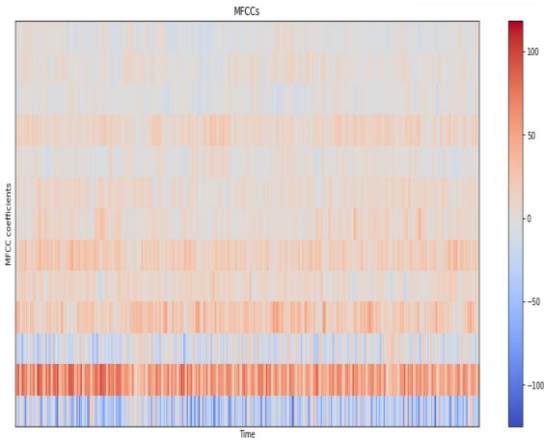
  This module includes creation of training data from the data set which is fed to the model. This module is used to train the model which is built earlier in the development module. The model is trained on the training data which contains different genre of audio files. Cross validation set is added to ensure the model doesn't overfit by running on extra epochs. The number of epochs is dependent on the cross validation set accuracy and cross validation set loss. Training is stopped at a certain epoch by looking at the training and cross validation accuracy and loss

B. Mel-Frequency Cepstral Coefficients (MFCC)

- The first step in any method for classifying music genres is to take its characteristics and identify those aspects of music that are helpful for identifying the linguistic content.
- The loudness of a signal at various frequencies present in a specific waveform over time is visually represented by a spectrogram. In addition to observing whether there is more or less energy at a

given moment, one may also observe how energy levels change over time.

- Spectrograms are also called as sonographs, voiceprints and voicegrams.
- To maintain the consistency of the audio signal, we divide the signal into a few brief frames. Typically, we divide them into 50 or 60-size slices.
- At that moment, we were able to distinguish the different frequencies present in each frame.
- We refer to this as the spectrum because it provides the amplitude at each frequency.
- Since frequency content frequently changes over time, we apply the Fourier transform to signal segments that overlap within windows. This enables us to track changes in the frequency spectrum over time. This is referred to as a spectrogram.
- We map the frequencies to the mel scale, which is a measure of pitch, in order to make equal distances in pitch sound similarly far to the human ear because humans do not perceive frequency on a linear scale. The result is the mel-spectrogram.
- Mel-frequency cepstral coefficients (MFCC) were investigated with the features using several modelling approaches, such as autoregressive models, over different time scales and compared the performance of the MFCCs as an independent feature set to that of the beat histogram and pitch histogram feature sets. Cepstral coefficients are a common audio feature set for applications including speech recognition, environmental sound recognition, and music genre classification.



The tonality system is the foundation of most music. Tonality places sounds into interconnected spatial and temporal patterns based on pitch connections. These structures are crucial for defining chords, keys, melody, themes, and even form. The pitch helix is a diagram that represents pitch connections by placing tones on a cylinder's surface. It simulates the unique interaction between octave intervals. It simulates the unique interaction between octave intervals.

C. *Feature Extraction*

Data must first be preprocessed before we can begin training it. With the aid of the LabelEncoder() method in sklearn.preprocessing, we will attempt to concentrate on the last column, which is label, and encode it. If we want to run any model on our data, text cannot be present in it. Therefore, this data needs to be preprocessed and prepared for the model before we run it. We need to utilise the Label Encoder class to translate this type of categorical text data into numerical data that the model can comprehend.

Scaling the features is the crucial next step after preprocessing. By eliminating the mean and scaling the feature to a unit variance, standard scalar is used to standardise features. The formula used to determine x(sample)'s score is:

$$Z = (x - u) / s$$

For the majority of machine learning estimators, a dataset must be standardised.

D. *k-Nearest Neighbors Algorithm*

The categorisation of music genres ends with this stage. After features were taken out of the raw data, the model needed to be trained. The model

can be trained using a variety of techniques. Some of these approaches are:

- K-Nearest Neighbors
- K-Means Clustering
- Multiclass Support Vector Machines

Because it is a well-known and straightforward procedure, we employed the k-nearest neighbours (K-NN) machine learning technique for this genre classification. KNN algorithm assumes that similar things belong to the class with close proximity. KNN algorithm calculates the distance between the current entity and all its neighbors. This method's main advantage is that it may be used to address both classification and regression problems. It considers the nearest class that has the most neighbours.

The majority of the time, it is accurate to the tune of 40% to 80%. The most fundamental machine learning algorithm, KNN, uses the supervised learning method as its foundation. KNN stores the dataset and performs action on it but it does not learn from the data set i.e. training set. So, it is also called lazy learner algorithm. The method of "windowing" separates an audio recording into brief, 20–40 ms-long frames. A time series that represents the entire audio file is the resulting vector.

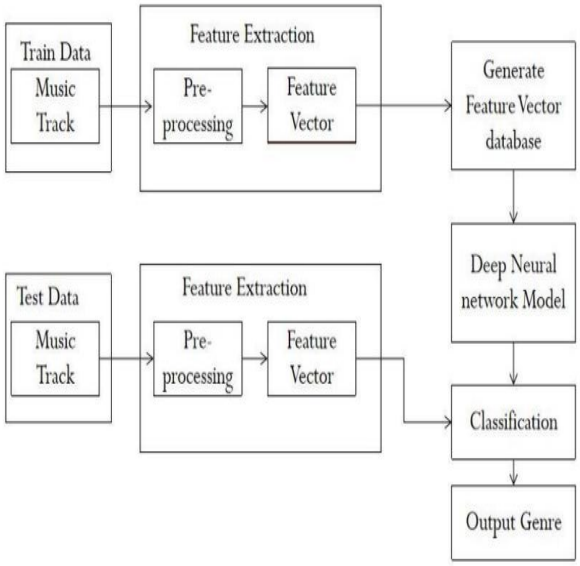
Feature similarity is used by the K-Nearest Neighbors (KNN) algorithm to forecast the values of fresh data points. Additionally, it implies that the nearest neighbour class will be chosen for the new data point depending on how eagerly it cooperates with the points in the training set.

- Load the data set, import the resources, and then explore it.
- Create a function that measures the separation between the feature vectors and the neighbours that were discovered.
- Identify the nearest neighbors.
- Create a function that will be used to evaluate the model.
- Create a binary file called "my.dat" to save the features you've extracted from the data set.
- Separate the training and testing portions of the data set.
- Use KNN to make a prediction and calculate the accuracy on test data.

E. Model Evaluation

- • To train the CNN model, we used the Adam optimizer. A 100-epoch was assigned to the training model.
- The loss computation uses the sparse categorical crossentropy function, the output layer uses the RELU activation function, and the hidden layers employ the softmax function.
- Overfitting is avoided by using dropout.
- Having contrasted numerous optimizers, we settled on the Adam optimizer because the findings were really the best.
- Expanding the number of epochs can raise the model's accuracy. However, after a particular amount of time, we might reach a threshold, thus the value should be chosen properly.
- Our test set accuracy rate of 92.90 percent is rather respectable.
- Thus, we conclude that Convolutional Neural Network (CNN), which aids in the classification process, can be implemented using TensorFlow and that Neural Networks are particularly effective in machine learning models.
- There are many applications of music genre classification.
  - ✓ Mall: In the malls, music is played continuously, and choosing the appropriate music to play is a busy and time-consuming task. As a result, our approach aids in selecting the appropriate song for each occasion or event.
  - ✓ Restaurant: In a restaurant, selecting the appropriate music for different occasions based on customer demand is a crucial duty; our system will assist in selecting a specific genre song for the same.
  - ✓ Airport: In order to keep passengers entertained while they wait for hours

on end for various reasons, music is played in airports. Our system will assist in selecting the appropriate tune.



IV. Experimental Results And Analysis

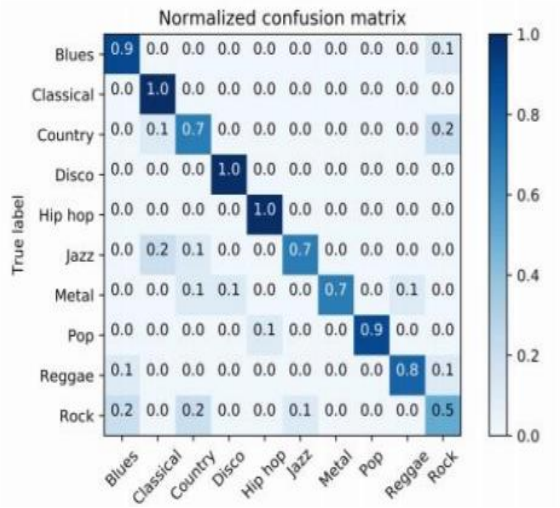
The PC with TensorFlow 1.12.1, sklearn 0.3.2, Keras 2.2.4, and OpenCV 3.4.2 the libraries installed in Python 3.5.2, an Intel Xeon 3.4GHz processor, and 32 GB RAM is used to create the dataset and system settings for our suggested models. The GTZAN dataset is used to gauge how well our deep neural network performs. There are 1000 audio recordings total in the dataset, 100 tracks for each class. Our model took about 35 minutes to complete. We have 44100 characteristics for the raw audio input because the data was sampled at 22050HZ. To keep the amount of features to a minimum, we limited our windows to two seconds. The optimal balance of audio sample time and feature space dimension was found to be 44100 features. As a result of the pre-processing, our input has the shape (8000, 44100), with each feature denoting the amplitude at a particular timestep out of 44100. In each of our cross validation and test sets, we also included 100 samples of unaugmented data. Pre-processing our data by converting the raw audio into mel spectrograms was another experiment. This resulted in significant performance enhancements for all models. Mel- spectograms are a popular way to visualise audio because they closely resemble how humans perceive frequency in log form.

Python is the language used for this. The code is created in the Python language. Python offers a large range of libraries for use in computation and science. Python provides code that is clear and readable. Developers may build reliable systems with Python's simplicity, but machine learning and artificial intelligence are driven by complex algorithms and flexible workflows. Developers get to focus exclusively on solving an ML problem, rather than on the language's technical specifics. Many developers find Python appealing because it is simple to learn. It is simpler to create machine learning models since Python code is human readable.

This literary work has several different ways that are suggested. Different dataset types have been used, and some people have even produced their own datasets. Additionally, the authors' consideration of distinct genres varies throughout works. Some authors have taken into account five genres. But when they built their models, the majority of them took the range of 5 to 10 into account. As a result, not all of their findings can be matched to this model. We have only taken into account the works that make use of all 10 genres found in the GTZAN dataset. As a result, we created a music genre detector that can identify the genre of any audio file. Graphical User Interface The user's input is received as an mp3 file through the Graphical User Interface GUI. Librosa converts music files into pitch, frequency, and many spectrogram characteristics. The SVM model, which can be altered for a number of reasons, was the one we used to predict the music. It provides a wide selection of open source libraries and modules that can be used in several applications. Using all of the aforementioned tools, we were able to successfully create our search engine. We have offered a way for categorising audio files based on their genre and automatically extracting musical elements from audio recordings. We preprocess the data first, followed by feature extraction, selection and lastly classification. Here, we focused our spectrum of features onto just mfcc features as these act as a useful metric for the human perception of music. For the task of classification, we have used machine learning algorithm(knn) classifier we got the maximum accuracy of 80%.

V. CONCLUSION

To achieve this objective of identifying the musical genre, the suggested research work used the GTZAN dataset and generated various models. The proposed model communicated this information to our CNN by using a range of inputs for various models, including audio and mel-spectrogram. Additionally, it utilised numerous sound file characteristics that were archived in ANN, SVM, MLP, and Decision Tree archives. Its accuracy rate of 91 percent is comparable to the maximum accuracy rate of human perception of genre. Frequency-based mel-spectrograms produced data that was more precise than amplitude-based mel-spectrograms. In contrast to amplitude, which only offers information on strength, or how "loud," a sound is, frequency distribution over time includes information on the substance of the sound. Mel-spectrograms also have a pleasant aesthetic. The best results came from CNN. The training process takes the longest, but the improved precision justifies the added expense of the computer. However, the SVM, KNN, and feed-forward neural network's accuracy similarities are encouraging. While traditional and blues were simple to distinguish, certain genres, like country and the rock genre, were highly unique and others, like blues, were rather distinctive. As they performed the best, we intend to experiment with various deep learning techniques in the future. An RNN model might be suitable because the data are time series (for example GRU, LSTM). In the same vein as generative adversarial networks that recreate photographs in the manner of Van Gogh, but specifically for music, we are also interested in the project's generative features, such as any genre conversion. Additionally, we think that there can be chances for transfer learning, like when categorising music by decade or artist.



To get the best results, we intend to test out alternative deep learning methodologies in the future.

- Recording the Music

  Features that are used for music recording and genre detection can be included. This can be usedto determine the genre without having to choose a file.

- Song Detection

  We may also use other characteristics to detect the lyrics in order to implement music recognition.

- Android Application

  We may also create an Android app that can be downloaded from the Google Play Store.

REFERENCES

[1] McKinney, Martin, and Jeroen Breebaart. "Features for audio and music classification." (2003).

[2] O'Shea, Keiron, and Ryan Nash. "An introduction to convolutional neural networks." arXiv preprint arXiv:1511.08458 (2015).

[3] Bisharad, Dipjyoti, and Rabul Hussain Laskar. "Music Genre Recognition Using Residual Neural Networks." In TENCON 2019-2019 IEEE Region 10 Conference (TENCON), pp. 2063-2068. IEEE, 2019.

[4] Zhang, Scott, Huaping Gu, and Rongbin Li. "MUSIC GENRE CLASSIFICATION: NEAR-REALTIME VS SEQUENTIAL APPROACH." (2019).

[5] Chillara, Snigdha, A. S. Kavitha, Shwetha A. Neginhal, Shreya Haldia, and K. S. Vidyullatha. "Music Genre Classification using Machine Learning Algorithms: A comparison." (2019).

[6] Bahuleyan, Hareesh. "Music genre classification using machine learning techniques." arXiv preprint arXiv:1804.01149 (2018).

[7] Yang, Hansi, and Wei-Qiang Zhang. "Music Genre Classification Using Duplicated Convolutional Layers in Neural Networks." In INTERSPEECH, pp. 3382-3386. 2019.

[8] Gessle, Gabriel, and Simon Åkesson. "A comparative analysis of CNN and LSTM for music genre classification." (2019).

[9] Defferrard, Michaël, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. "Fma: A dataset for music analysis." arXiv preprint arXiv:1612.01840 (2016).

[10] George, Tzanetakis, Essl Georg, and Cook Perry. "Automatic musical genre classification of audio signals." In Proceedings of the 2nd international symposium on music information retrieval, Indiana. 2001.

[11] Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. "Dropout: a simple way to prevent neural networks from overfitting." The journal of machine learning research 15, no. 1 (2014): 1929-1958.

[12] Grossi, Enzo, and Massimo Buscema. "Introduction to artificial neural networks." European journal of gastroenterology & hepatology 19, no. 12 (2007): 1046-1054.

[13] Weston, Jason, and Chris Watkins. "Support vector machines for multi class pattern recognition." In Esann, vol. 99, pp. 219-224. 1999.

[14] Chang, Kaichun K., Jyh-Shing Roger Jang, and Costas S. Iliopoulos. "Music Genre Classification via Compressive Sampling." In ISMIR, pp. 387- 392. 2010.

[15] Hamel, Philippe, and Douglas Eck. "Learning features from music audio with deep belief networks." In ISMIR, vol. 10, pp. 339-344. 2010.

[16] Zlatintsi, Athanasia, and Petros Maragos. "Comparison of different representations based on nonlinear features for music genre classification." In 2014 22nd European Signal Processing Conference (EUSIPCO), pp. 1547