AUDIO SIGNAL PROCESSING & GENRE CLASSIFICATION

Name: Kaushik Ray Chowdhury

Subject – Project (KCS554)

B.Tech Computer Science

Roll No: 1812710026

IIMT Engineering College (127)

Contents

INTRODUCTION

Audio signal processing is an engineering field that focuses on the computational methods for intentionally altering sounds, methods that are used in many musical applications.

Music Genre Classification –

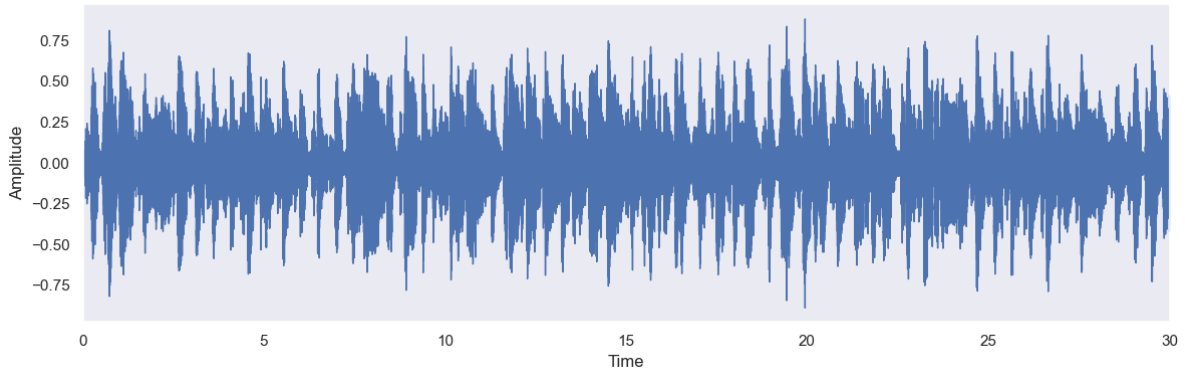
Music genre prediction is one of the topics that digital music processing is interested in. In this study, acoustic features of music have been extracted by using digital signal processing techniques and then music genre classification and music recommendations have been made by using machine learning methods. In addition, convolutional neural networks, which are deep learning methods, were used for genre classification and music recommendation and performance comparison of the obtained results.

This project is about how to analyse audio data and different methods to classify music genres & to find out which method is better for genre classification, Traditional Machine Learning method or using Deep Neural Network.

Chapter I

AUDIO SIGNAL PROCESSING (ASP) & AUDIO FEATURES

Audio signal processing is used to convert between analog and digital formats, to cut or boost selected frequency ranges, to remove unwanted noise, to add effects and to obtain many other desired results. Today, this process can be done on an ordinary PC or laptop, as well as specialized recording equipment.



Waveform

Feature groups of audio signals are as follows:

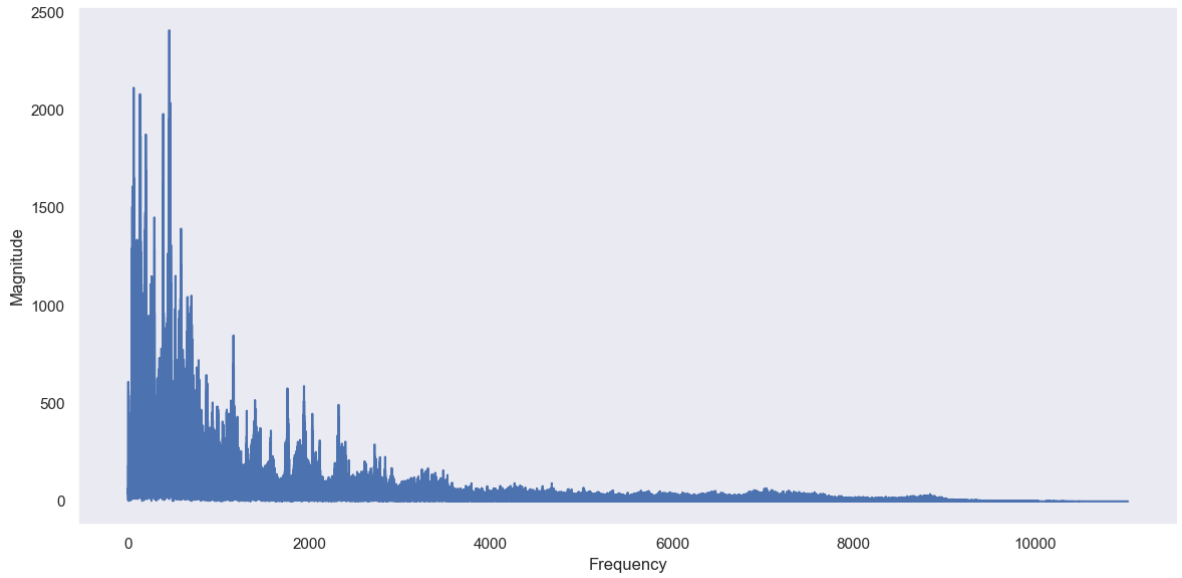
1. Temporal Shape
2. Temporal Features
    a. Zero-crossing rate
3. Energy Features
    a. Global
    b. Harmonic
    c. Noise
4. Spectral Shape (timbral texture) Features
    a. Centroid
    b. Spread
    c. Skewness
    d. Kurtosis
    e. Spectral roll-off
    f. MFCC (Mel-Frequency Cepstral Coefficients)

I have used these features in my experiment and also used to analyse the audio signals.

Traditional Analog – Digital Conversion Algorithms

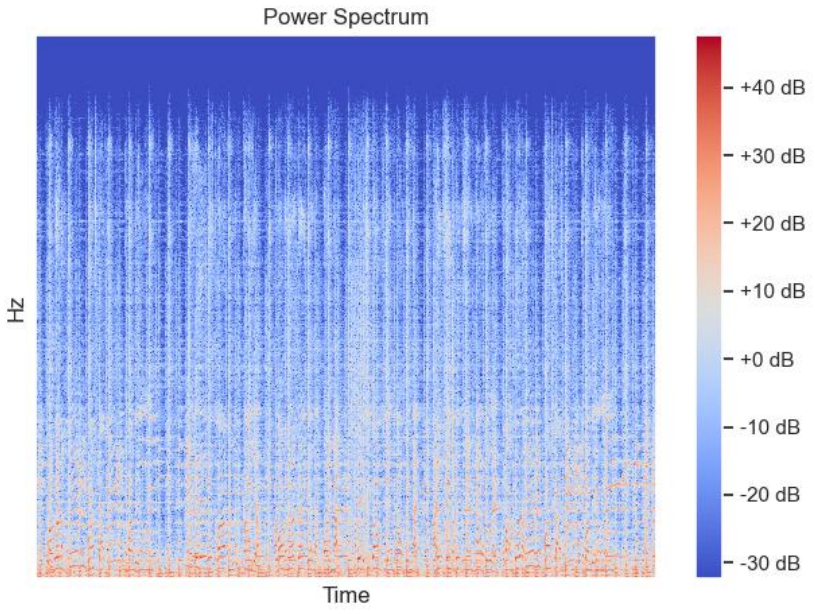
1. Fast Fourier Transform

A fast Fourier transform is an algorithm that computes the discrete Fourier transform of a sequence, or its inverse. Fourier analysis converts a signal from its original domain to a representation in the frequency domain and vice versa.



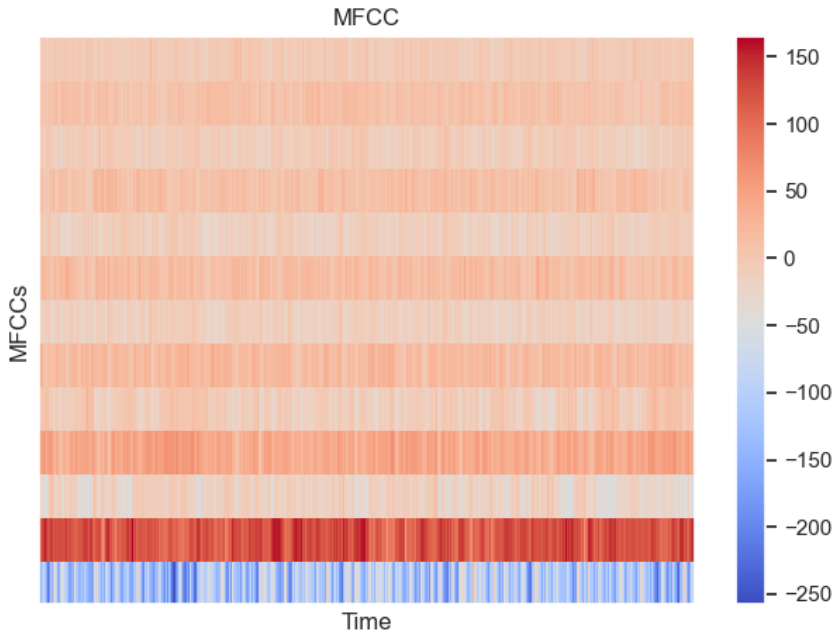
2. Short Time Fourier Transform

The Short-time Fourier transform, is a Fourier-related transform used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time.



3. **MFCC (Mel-Frequency Cepstral Coefficients)**

   In sound processing, the mel-frequency cepstrum is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. Mel-frequency cepstral coefficients are coefficients that collectively make up an MFC.



About the Dataset

The dataset consists of 1000 audio tracks each 30 seconds long. It contains 10 genres, each represented by 100 tracks. The tracks are all 22050Hz Mono 16-bit audio files in .wav format.

http://marsyas.info/downloads/datasets.html

https://www.kaggle.com/andradaolteanu/gtzan-dataset-music-genre-classification

From this dataset, various features are extracted and converted into .csv file. Features consist of Spectral Shape, RMS, STFT, MFCCs, etc. These features are extracted for traditional ML pipeline.

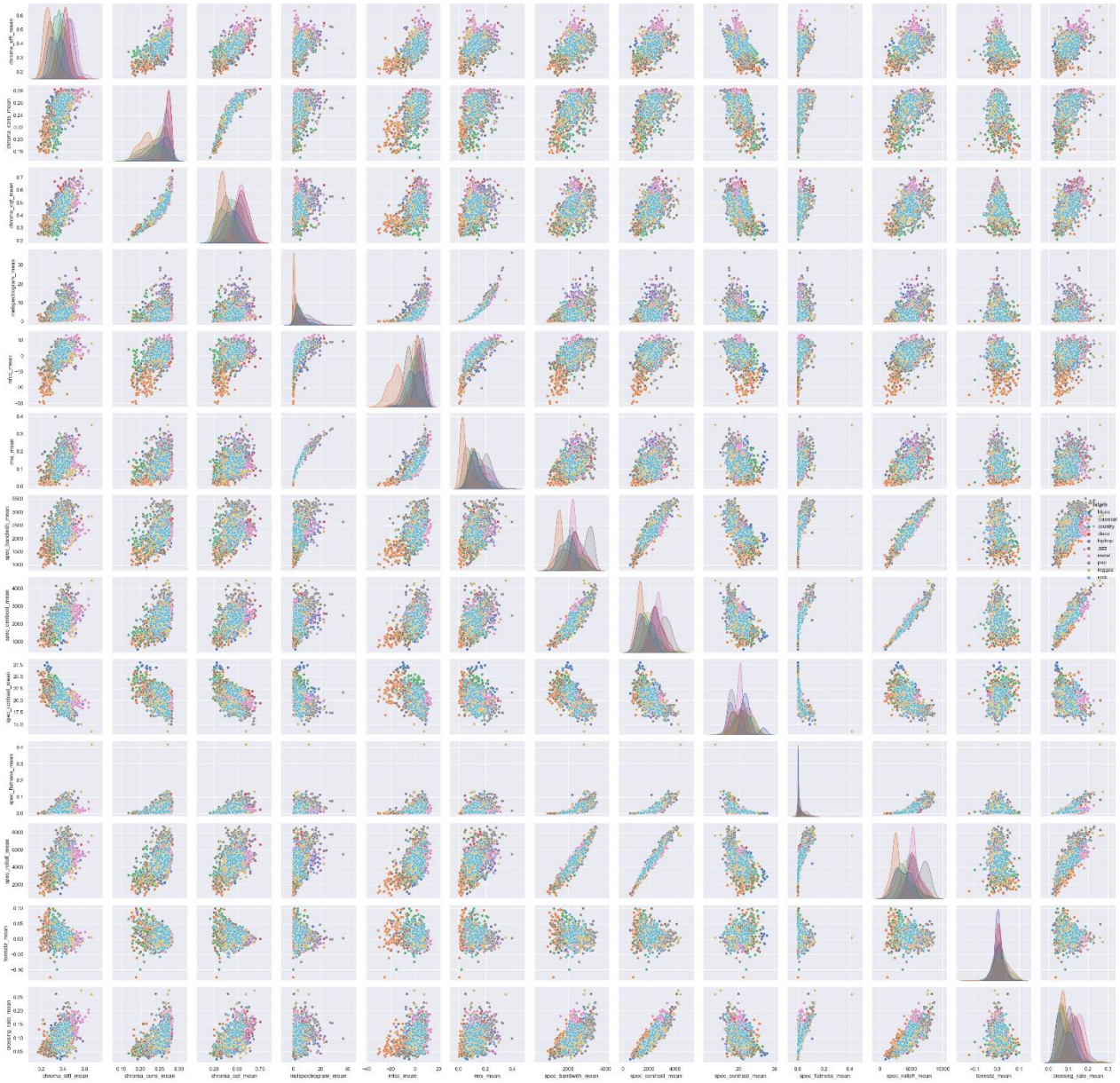
For Deep Learning, all the 30 sec audio track is segmented into 5 parts and then extracted their MFCCs with their target labels, in this case genres. All of this information is saved in .json file.

Augmented data and code can be accessed from this repository
https://github.com/kaushikroychowdhury/Audio-Exploration

Chapter II

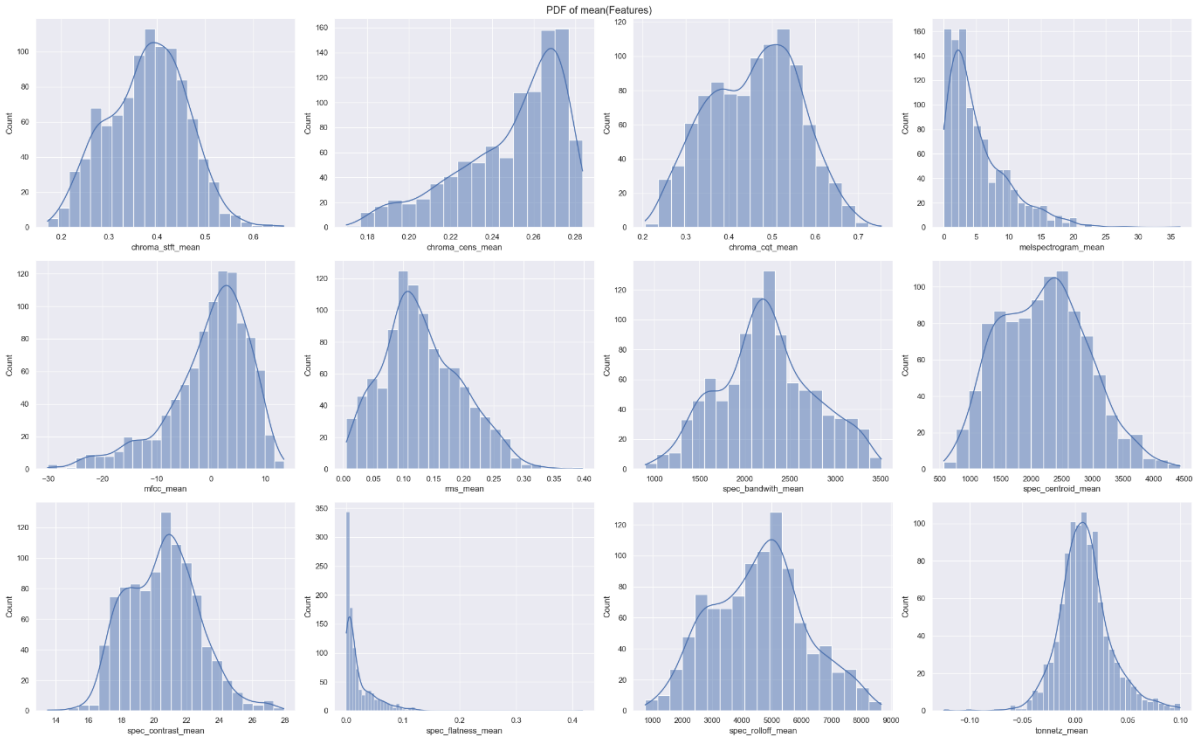
Statistical Analysis & Visualization

We explored the augmented dataset and here are some insights about features distribution. (Every feature's mean and variance are recorded)

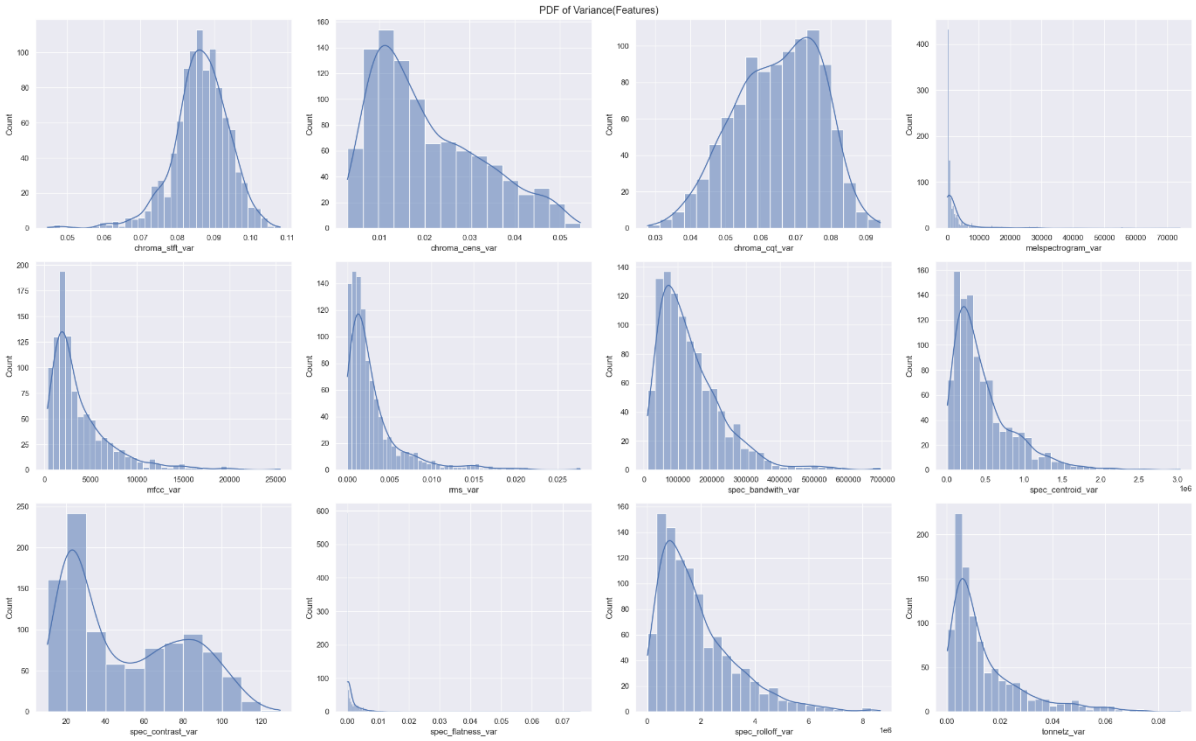


From this we can see the " melspectogram_mean " and " spec_flatness_mean " is too much left skewed, so we can transform these dimensions, such as Log Transform.

Now let’s see the different PDFs of various feature’s mean and variance.



PDFs of feature’s mean



PDFs of feature’s variance

Chapter III

Genre Classification

Traditional Machine Learning Methods

To use traditional machine learning algorithms, I first extracted features which are as follows: Chromagram, Constant-Q Transform, Normalise Chroma Energy, mel-scaled spectrogram, MFCCs, root-mean-Square value for each frames, Spectral Features (centroid, bandwidth, fatness, contrast, roll-off), tonal centroid, zero-crossing rate.

After extracting features, I computed mean and variance of each features and stored in a .csv file. Then after segmenting the data into input (features) and output (label) I scaled the input data and split the data into train-test ratio of 80:20.

After pre-processing the dataset, I fit the data into several classifier to see which performs better.

1. **Random Forest Classifier**
2. **Support Vector Classifier (svc)**
3. **Decision Tree Classifier**
4. **K-Nearest Neighbour Classifier (KNN)**
5. **Gaussian Naive Byes Algorithm**

After fitting the data into these classifiers, **Random Classifier & Support Vector Classifier** works better than other classifiers.

As the traditional machine learning method takes much more time to prepare the dataset, so I decided to use deep learning methods.

Deep Learning Method

Deep learning is an AI function that mimics the workings of the human brain in processing data for use in detecting objects, recognizing speech, translating languages, and making decisions. **Deep learning** AI is able to **learn** without human supervision, drawing from data that is both unstructured and un-labeled. Deep learning is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.

Dense Neural Network (Basic Net)

The name suggests that layers are fully connected (dense) by the neurons in a network layer. Each neuron in a layer receives an input from all the neurons present in the previous layer—thus, they're densely connected.

In other words, the dense layer is a fully connected layer, meaning all the neurons in a layer are connected to those in the next layer.

Model specification for the dense net that I used is as follows:

Model: "sequential"

_____

Layer (type) Output Shape Param #

==========

flatten (Flatten) (None, 3367) 0

_____

dense (Dense) (None, 512) 1724416

_____

dropout (Dropout) (None, 512) 0

_____

dense_1 (Dense) (None, 256) 131328

_____

dropout_1 (Dropout) (None, 256) 0

_____

dense_2 (Dense) (None, 64) 16448

_____

dropout_2 (Dropout) (None, 64) 0

_____

dense_3 (Dense) (None, 10) 650

==========

Total params: 1,872,842

Trainable params: 1,872,842

Non-trainable params: 0

Convolutional Neural Network (CNN)

In deep learning, a convolutional neural network is a class of deep neural networks, most commonly applied to analysing visual imagery. They are also known as shift invariant or space invariant artificial neural networks, based on their shared-weights architecture and translation invariance characteristics.

The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex.

Model specification for the convolutional neural net that I used is as follows:

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 257, 11, 32)	320
max_pooling2d (MaxPooling2D)	(None, 129, 6, 32)	0
batch_normalization (BatchNo	(None, 129, 6, 32)	128
conv2d_1 (Conv2D)	(None, 127, 4, 32)	9248
max_pooling2d_1 (MaxPooling2	(None, 64, 2, 32)	0
batch_normalization_1 (Batch	(None, 64, 2, 32)	128
conv2d_2 (Conv2D)	(None, 63, 1, 32)	4128
max_pooling2d_2 (MaxPooling2	(None, 32, 1, 32)	0
batch_normalization_2 (Batch	(None, 32, 1, 32)	128
flatten (Flatten)	(None, 1024)	0
dense (Dense)	(None, 64)	65600
dropout (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 10)	650

Total params: 80,330
Trainable params: 80,138
Non-trainable params: 192

Recurrent Neural Network (RNN)

A recurrent neural network is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behaviour.

Long – Short Term Memory (LSTM)

Long short-term memory is an artificial recurrent neural network architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can not only process single data points, but also entire sequences of data.

Model specification for the LSTM that I used is as follows:

Model: "sequential"

_____

Layer (type) Output Shape Param #

==========

lstm (LSTM) (None, 259, 64) 19968

_____

lstm_1 (LSTM) (None, 259, 64) 33024

_____

lstm_2 (LSTM) (None, 64) 33024

_____

dense (Dense) (None, 64) 4160

_____

dropout (Dropout) (None, 64) 0

_____

dense_1 (Dense) (None, 10) 650

==========

Total params: 90,826

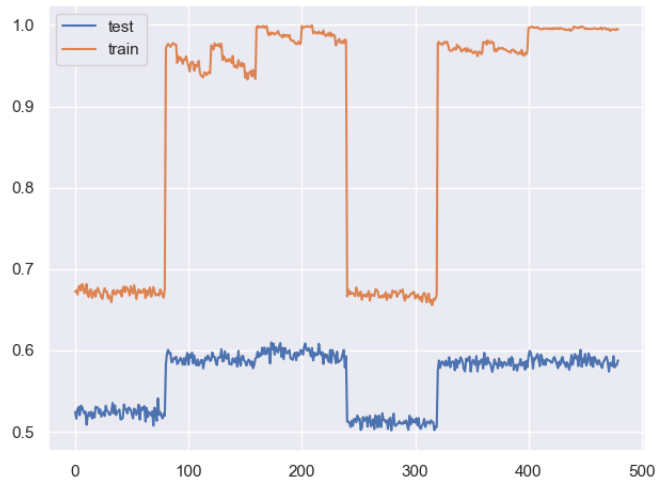
Trainable params: 90,826

Non-trainable params: 0

Experiments & Results

Traditional Machine Learning Approach

Random Forest Classifier



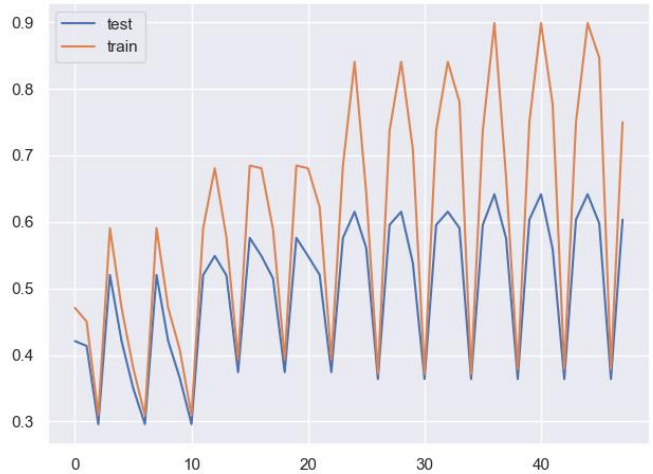
Hyper-params :-

'bootstrap': True,
'max_depth': 10,
'max_features': 'auto',
'min_samples_leaf': 1,
'min_samples_split': 5,
'n_estimators': 233

Train Accuracy : 0.983
Test Accuracy : 0.588

As from the plot we can see that Random Forest doesn't perform well, this is due to the dataset. The dataset is not balanced and also due to the curse of dimensionality.

Support Vector Classifier (svc)



Hyper-params : -

'C': 10,

'degree': 1,

'kernel': 'rbf'

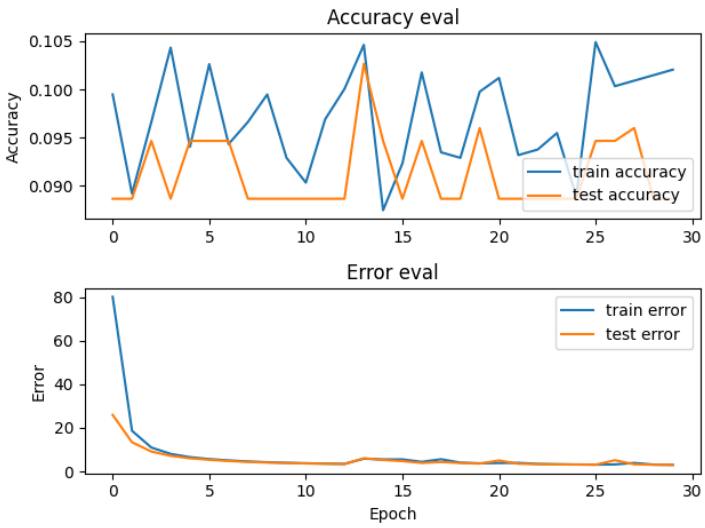
Train Accuracy : 0.882

Test Accuracy : 0.667

As from the plot we can see that svc (support vector classifier) doesn't perform well, as the same reason as random forest classifier.

Deep Leaning Approach

Dense Neural Network (Basic)



Hyper-params : -

Train-Test Ratio – 80:20

Optimiser : ADAM (learning Rate = 0.01)

Loss : Sparse Categorical Cross-entropy

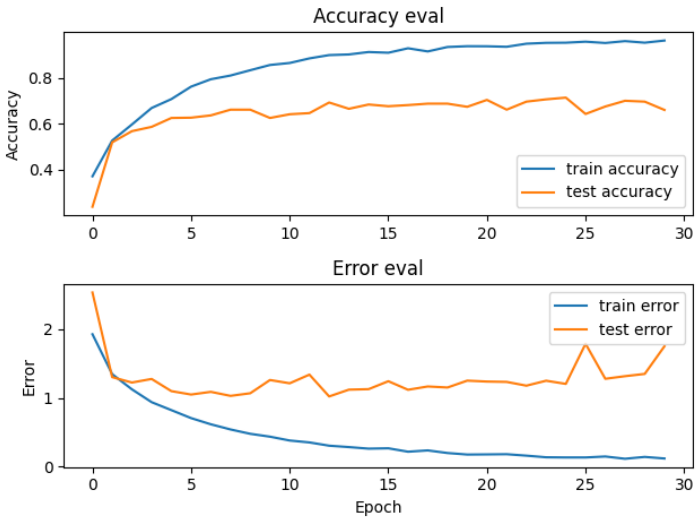
Batch Size : 32

Epoch : 30

Kernel Regularizer : L2 (0.01)

Test Accuracy : 0.0887

Convolutional Neural Network (CNN)



Hyper-params : -

Train-Test Ratio – 80:20

Optimiser : ADAM (learning Rate = 0.0001)

Loss : Sparse Categorical Cross-entropy

Batch Size : 32

Epoch : 30

Test Accuracy : 62%

Convolutional Neural Networks are better than basic neural networks consists of only dense layers for recognising patterns in audio signals.

Long-Short Term Memory Neural Networks (LSTM)

Hyper-params : -

Train-Test Ratio – 80:20

Optimiser : ADAM (learning Rate = 0.0001)

Loss : Sparse Categorical Cross-entropy

Batch Size : 64

Epoch : 30

Test Accuracy : 68%

Recurrent Neural networks are better for classifying audio signal data.

An LSTM has a similar control flow as a recurrent neural network. It processes data passing on information as it propagates forward. The differences are the operations within the LSTM's cells.

The core concept of LSTM's are the cell state, and it's various gates. The cell state act as a transport highway that transfers relative information all the way down the sequence chain. You can think of it as the "memory" of the network. The cell state, in theory, can carry relevant information throughout the processing of the sequence. So even information from the earlier time steps can make it's way to later time steps, reducing the effects of short-term memory. As the cell state goes on its journey, information get's added or removed to the cell state via gates. The gates are different neural networks that decide which information is allowed on the cell state. The gates can learn what information is relevant to keep or forget during training.

Implementation

The model is deployed in the form of a web application. The web application is developed by using Flask framework (Backend).

It takes input as a 30 sec audio (.wav format) and then the audio is pre-processed and fed into the model. The model predicts its genre and sends the output to a web variable & it reflects the genre of that audio (.wav).

Conclusion

As performing the Experiment, I conclude that using deep learning over traditional machine learning method is better, because for traditional method we need spend much more time to pre-process the data. Traditional ML pipeline is not good for recognising the important patterns of the sequential data. Neural networks work better in this situation. But all neural networks are not suitable for this problem, basic neural networks fail for this problem.

Convolutional Neural Network (CNN) and Recurrent Neural network (RNN) performs well. Test accuracy of both the networks are closer, this is because of less amount of data. The dataset contains only 1000 tracks which are then segmented in 5 parts of each track, so it makes 5000 segments, which is very small data compared to the dataset which can achieve above 75% accuracy.

RNN_LSTM (Long-Short Term Memory) is better of all because it can memorize and the power of memorizing gives better performance for recognising patterns in sequential data.

The next phase is to develop and optimise the model, so that it can be much accurate.

Train the model with large amount of dataset can also give better accuracy and it also prevents the model to overfit.

References

http://w2.mat.ucsb.edu/240/E/static/notes/Audio_features.html

https://en.wikipedia.org/wiki/Fast_Fourier_transform

https://en.wikipedia.org/wiki/Short-time_Fourier_transform

https://en.wikipedia.org/wiki/Mel-frequency_cepstrum

https://en.wikipedia.org/wiki/Convolutional_neural_network

https://en.wikipedia.org/wiki/Long_short-term_memory

https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21

https://www.tensorflow.org/

https://librosa.org/doc/

https://pythonise.com/series/learning-flask/flask-uploading-files