Department of Electronics and Communication Engineering, MNNIT Allahabad, Session 2021-2022

Music Style (Genre) Classification Using CNN

Under Guidance of

Prof. V K SRIVASTAVA

Project Presented by Group of

Divas Gupta (20185116)

Priyanka Soni (20185053)

Dipesh Sharma Poudel (20185105)

Contents

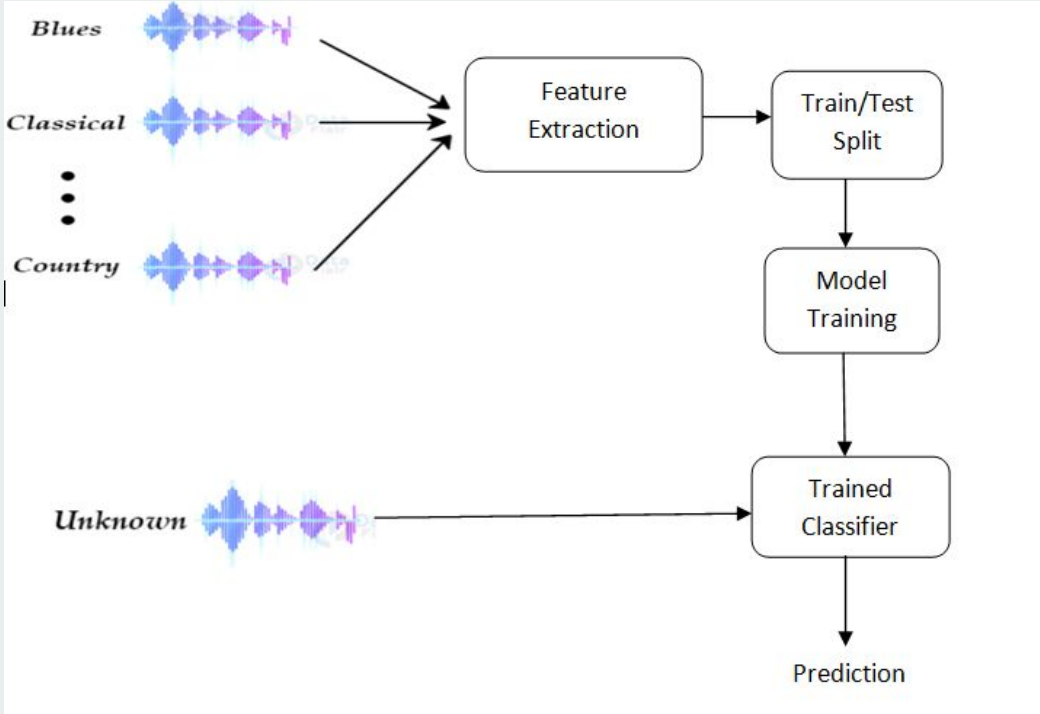
What is Genre?

- By definition , genre is a category of literary composition, determined by literary techniques, tone , content or even length.

- Genre is characterized by common features (of pieces belonging to it) such as Instrumentation, texture, dynamics, rhythmic characteristics, melodic gestures, and harmonic contents.

INTRODUCTION

- There are 1264 micro - genres of popular music.
- Genre recognition is a way to recognise/predict the genre with the help of the audio track and features of dataset.
- It is a machine learning model to automatically classify different musical genres from audio files.



Why Genre classification?

- Great utility to musical information retrieval systems.
- Genre is intrinsically built on the similarities between pieces of the same genre and differences between pieces of different genres.
- An automated genre recognition system would make it possible to classify and search large electronic music libraries.
- In order to generate better recommendation, user generated rating & reviews, genre or books metadata is used.

Dataset (GTZAN genre collection)

Contains 1000 music files. Dataset has 10 types of genres with uniform distribution. Dataset has the following genres: **blues, classical, country, disco, hiphop, jazz, reggae, rock, metal, and pop**. Each music file is 30 seconds long. Dataset has 2 folders:-

- **Genres original** - A collection of 10 genres with 100 audio files each, all having a length of 30 seconds.

- **Images original** - A visual representation for each audio file. One way to classify data is through neural networks. Because NNs (like CNN) usually take in some sort of image representation, the audio files were converted to Mel Spectrograms to make this possible.

Technology Used

- Python
- ML Algorithms
- Sklearn
- Matplotlib
- Numpy & pandas
- Scipy
- Librosa
- python_speech_features

Workflow

1.Preprocess the data using stft (to convert original data into frequency and time domain).
2.Feature Extraction using MFCC.
3.Training the classifier using various classification algorithms.
4.Testing the data and predicting the genre of our data files.
5.Compare performances of different classifiers using different benchmarks.

Feature Extraction Techniques for audio files

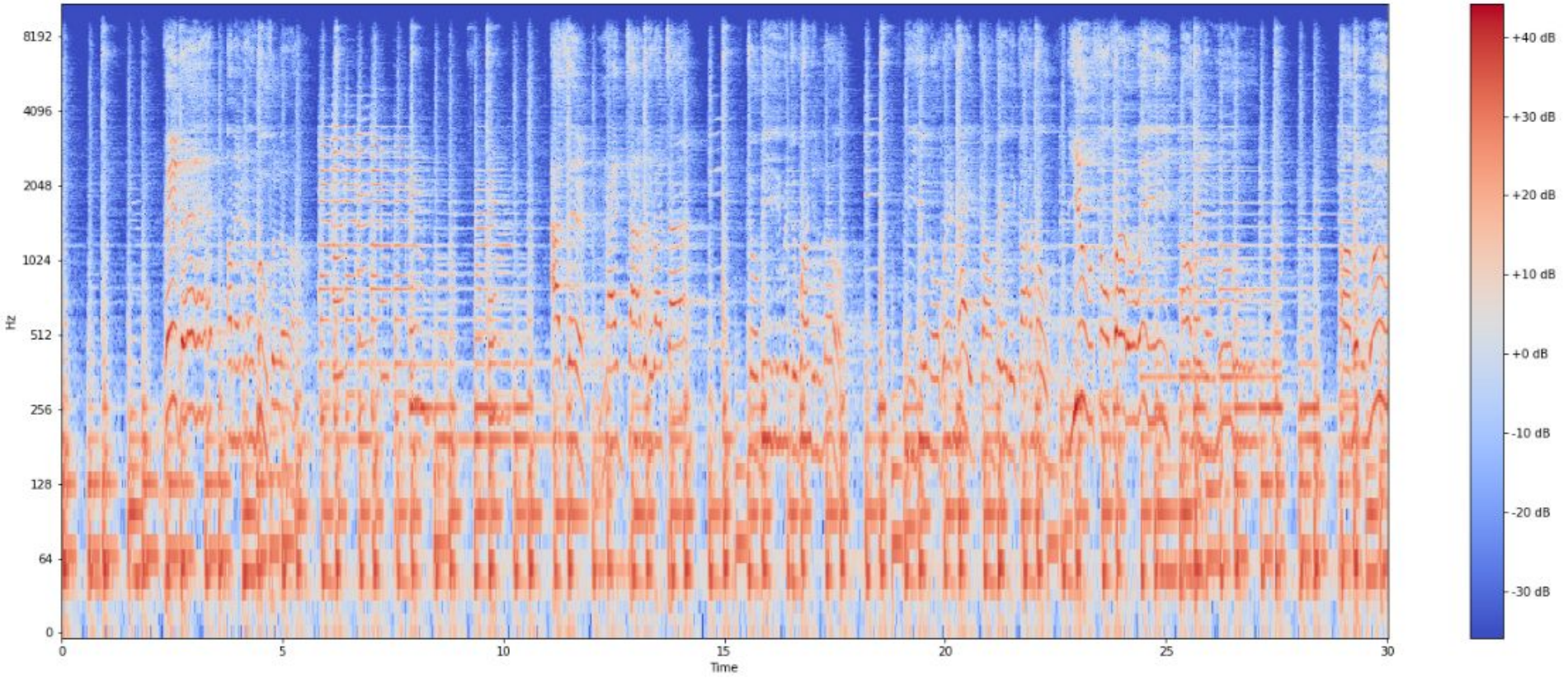
- **Time domain:** Extracted from waveforms of the raw audio. **Zero crossing rate, amplitude envelope, and RMS energy** are examples.

- **Frequency domain:** Focus on the frequency components of the audio signal. Signals are generally converted from the time domain to the frequency domain using the *Fourier Transform*. **Band energy ratio, spectral centroid, and spectral spread** are examples.

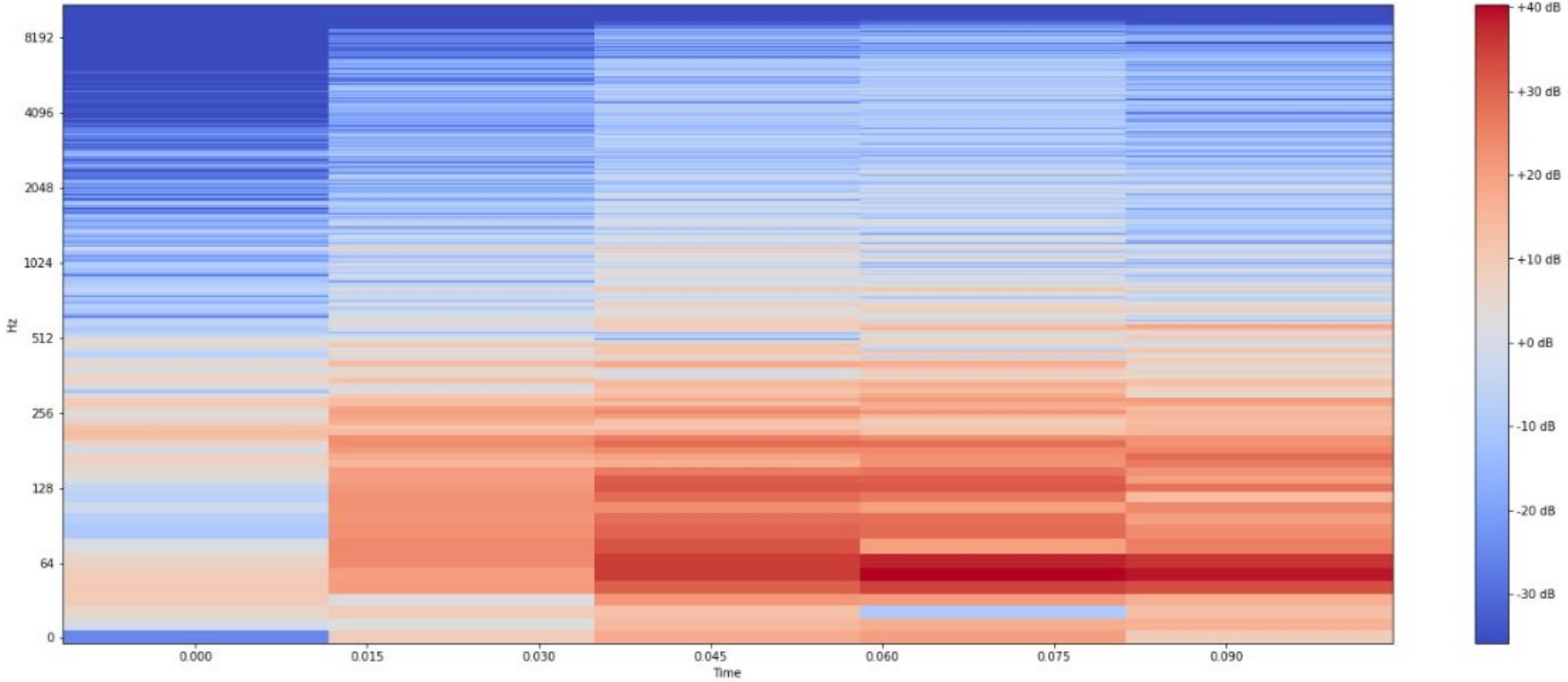
- **Time-frequency representation:** Combine both the time and frequency components of the audio signal. Obtained by applying the Short-Time Fourier Transform (STFT) on the time domain waveform. **Spectrogram, mel-spectrogram, and constant-Q transform** are examples.

Spectrograms

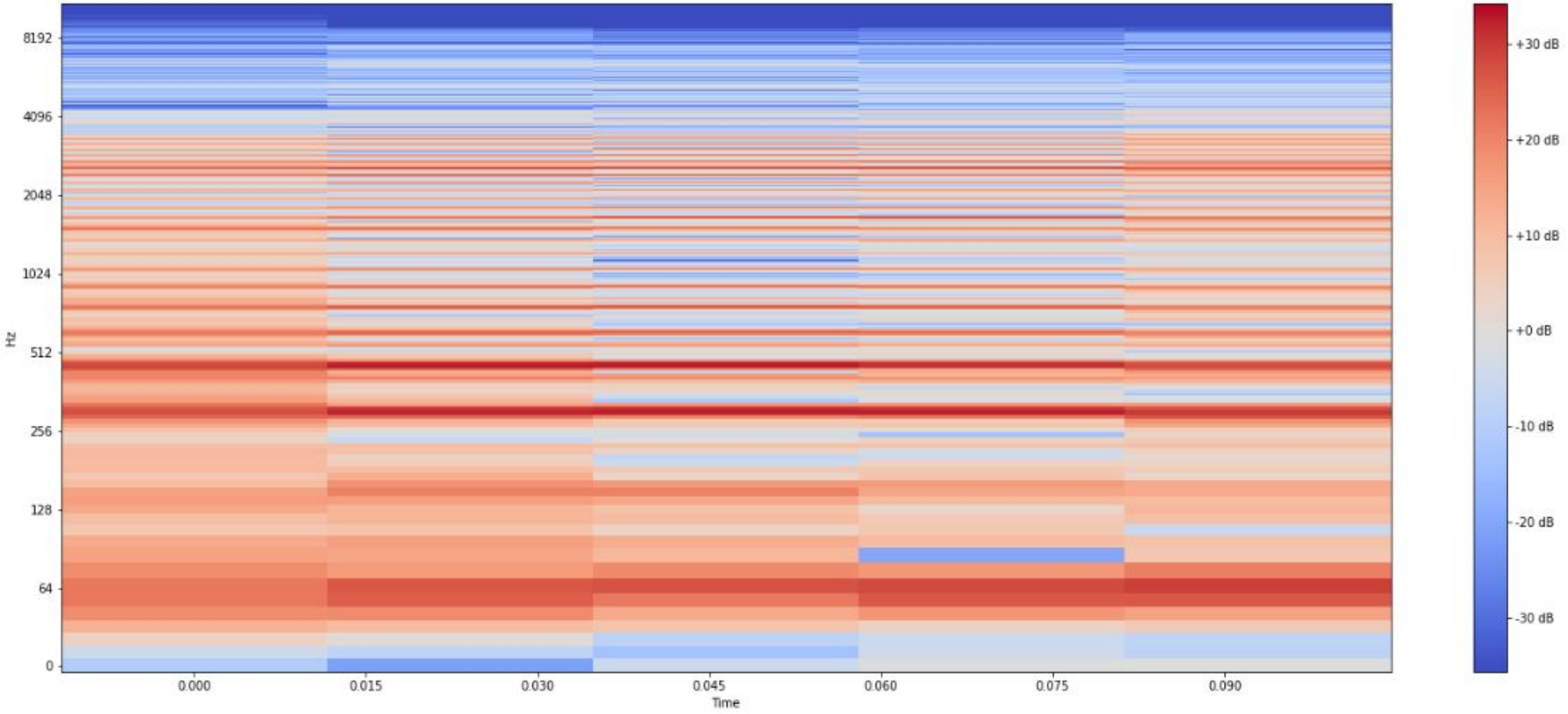
- Visual depiction of the spectrum of frequencies of an audio signal as it varies with time.
- Obtained by applying the Short-Time Fourier Transform (STFT) on the signal.



Spectrogram showing first 5 frames of ‘blues’ Genre



Spectrogram showing first 5 frames of ‘rock’ genre

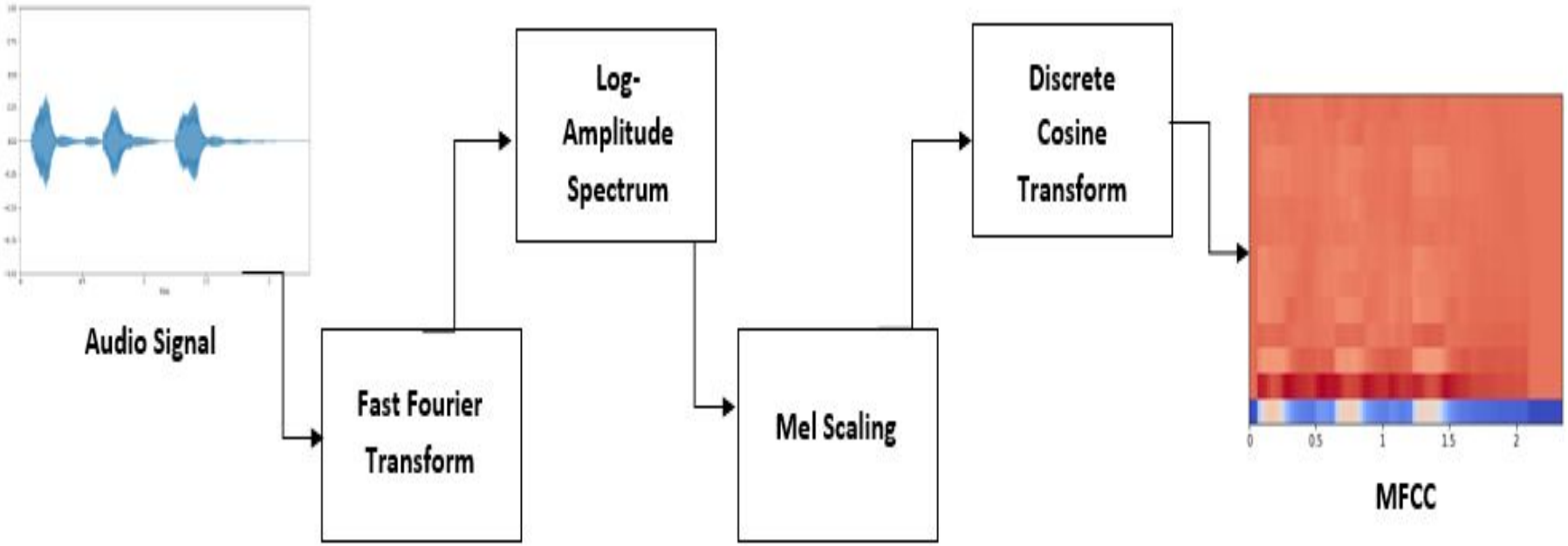


Mel Spectrograms

- Humans perceive sound logarithmically.
- Better at detecting differences in lower frequencies than higher frequencies.
- Can easily tell the difference between 500 and 1000 Hz, but hardly be able to tell a difference between 10,000 and 10,500 Hz, even though the distance is same between two.
- The **mel scale** conversion from frequency (f) to mel scale (m) is given by:- $m=2595.\log(1+f/500)$.

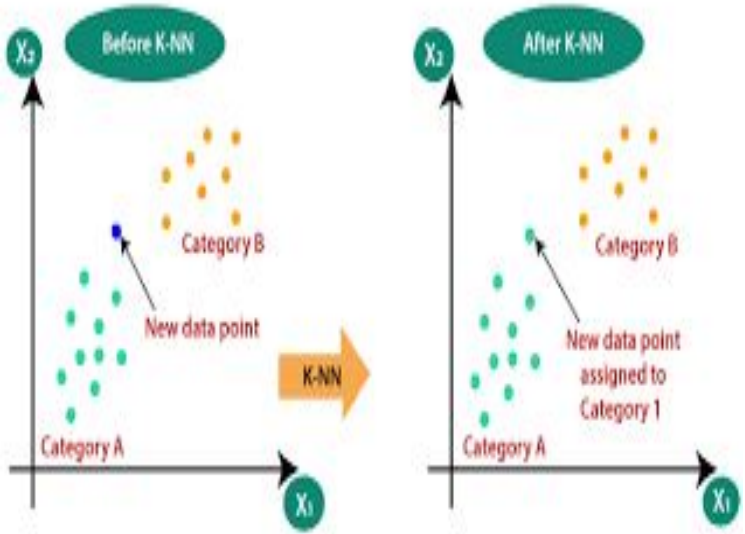
What are MFCC's(Mel Frequency Cepstral Coefficient)

Emphasize on obtaining the exact structure of the audio signal to extract linguistic features and discard the background noise.



Training using K-Nearest Neighbor (KNN)

1. Load the training data.
2. Intialised with feature vector.
3. Prepare data by scaling, missing value treatment, and dimensionality reduction as required.
4. Find the optimal value for K.
5. Predict a class value for new data:
    - Calculate distance(X, Xi) from i=1,2,3,….,n. where X= new data point, Xi= training data.
    - Sort these distances in increasing order with corresponding train data.
    - From this sorted list, select the top ‘K’ rows.
    - Find the most frequent class from these chosen ‘K’ rows.



Training using Random Forest Classifier

Steps of Random Forest Classifier:-

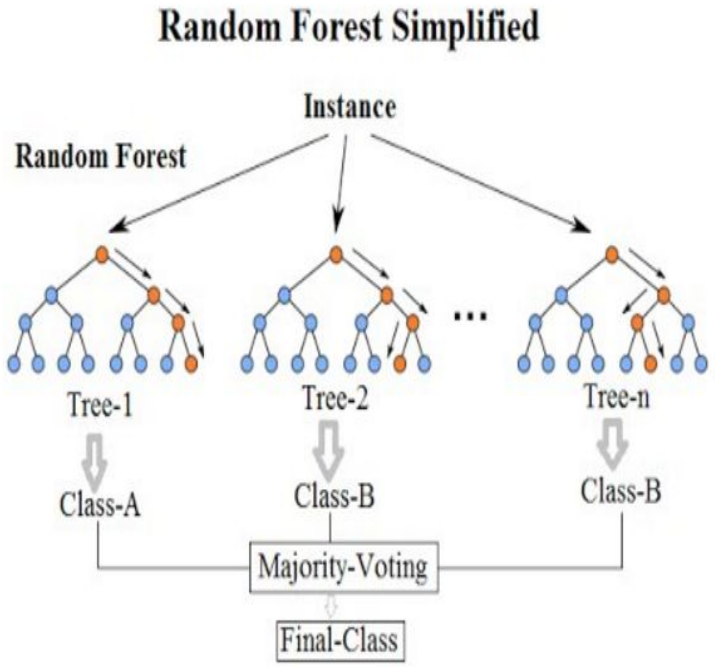
1. n number of random records are taken from the data set having k number of feature knows as row sampling and feature sampling.

2.Individual decision trees are constructed for each sample.

3.Each decision tree will generate an output.

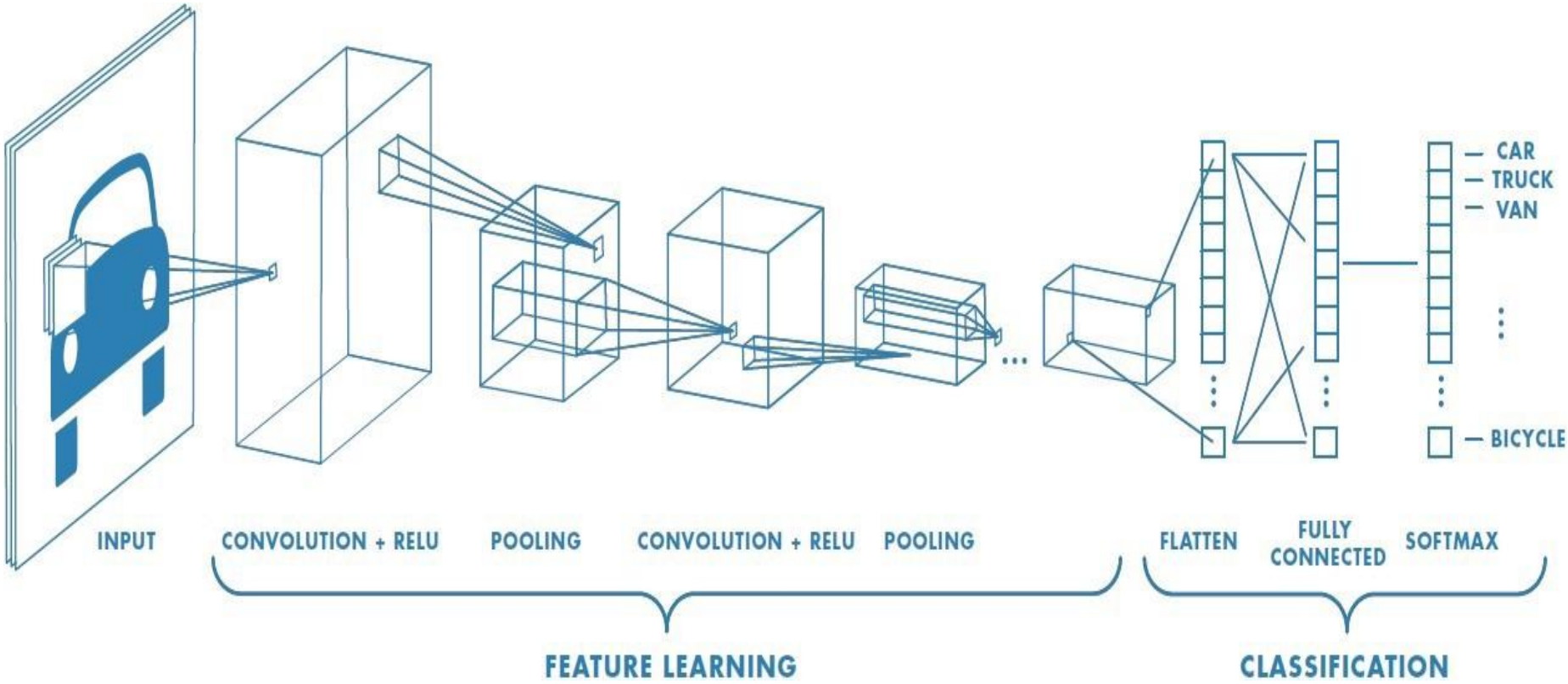
4.Final output is considered based on Majority Voting or Averaging for Classification and regression respectively.

Takes less time than KNN and works well with high dimensional features.



CNN Model

There are 3 Convolutional layers followed by 2 flattened and then finally output layer having 10 nodes representing 10 different genres.



Evaluation of Model

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Accuracy:how often is the classifier correct?
i.e.(TP+TN)/total = (100+50)/165

Precision: When it predicts yes, how often is it correct? i.e.TP/(TP+FP) = 100/110

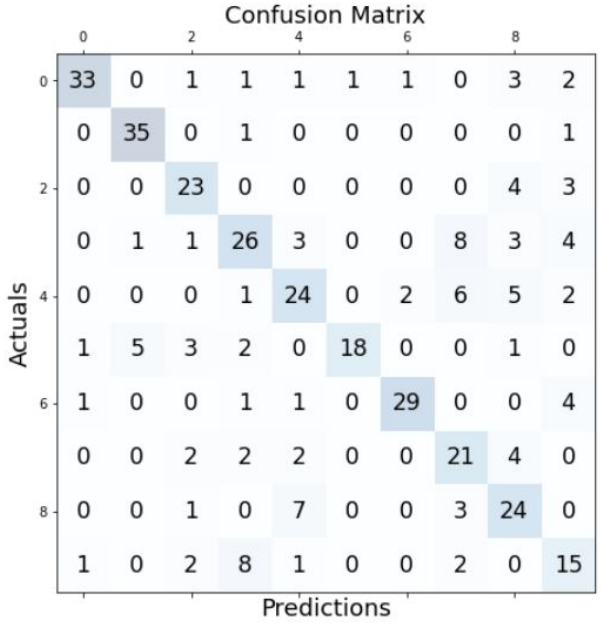
Recall: from all the positive classes, how many we predicted correctly? i.e. TP/(TP+FN)=100/105

F1 Score: (2*recall*precision)/(recall+precision)
Where TP=**true positives,** TN=**true negatives**

FP=**false positives** (Type I error)
FN=**false negatives** (Type II error)

Evaluation of K-Nearest Neighbour

	precision	recall	f1-score	support
1	0.92	0.77	0.84	43
2	0.85	0.95	0.90	37
3	0.70	0.77	0.73	30
4	0.62	0.57	0.59	46
5	0.62	0.60	0.61	40
6	0.95	0.60	0.73	30
7	0.91	0.81	0.85	36
8	0.53	0.68	0.59	31
9	0.55	0.69	0.61	35
10	0.48	0.52	0.50	29
accuracy			0.69	357
macro avg	0.71	0.69	0.69	357
weighted avg	0.72	0.69	0.70	357



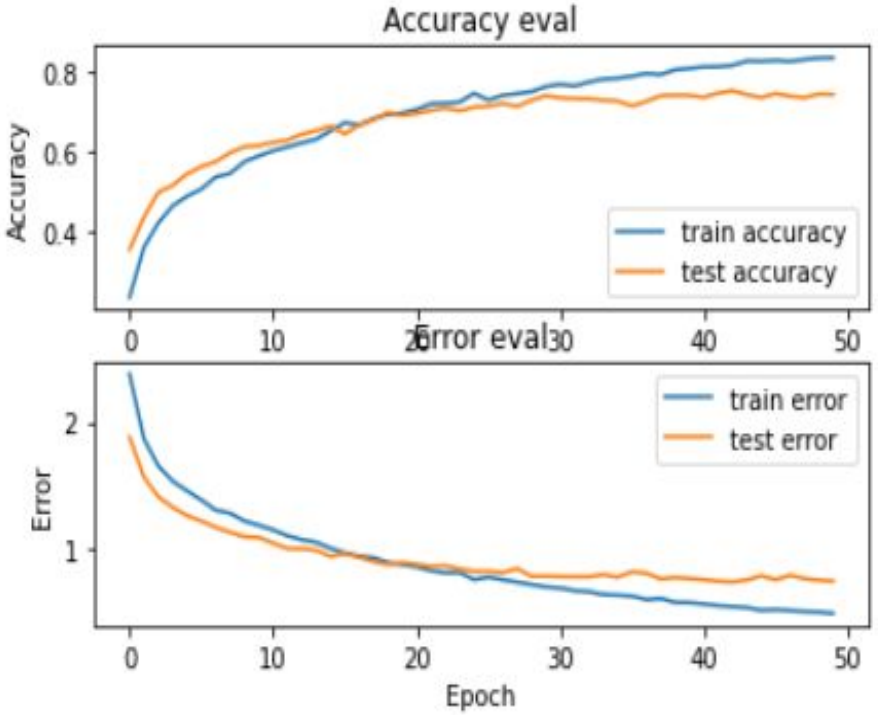
Evaluation of Random Forest Classifier

CLASSIFICATION REPORT

	precision	recall	f1-score	support
1	0.45	0.32	0.38	28
2	0.59	0.76	0.67	17
3	0.41	0.39	0.40	23
4	0.43	0.40	0.41	30
5	0.44	0.65	0.52	17
6	0.65	0.54	0.59	28
7	0.53	0.57	0.55	30
8	0.46	0.48	0.47	27
avg / total	0.50	0.49	0.49	200

CNN Model history

First graph is plotted between accuracy vs epoch showing it is increasing every iteration. & Second graph if between error vs epoch showing error is decreasing during each iteration.



Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 128, 11, 32)	320
max_pooling2d_3 (MaxPooling2	(None, 64, 6, 32)	0
batch_normalization_3 (Batch	(None, 64, 6, 32)	128
conv2d_4 (Conv2D)	(None, 62, 4, 32)	9248
max_pooling2d_4 (MaxPooling2	(None, 31, 2, 32)	0
batch_normalization_4 (Batch	(None, 31, 2, 32)	128
conv2d_5 (Conv2D)	(None, 30, 1, 32)	4128
max_pooling2d_5 (MaxPooling2	(None, 15, 1, 32)	0
batch_normalization_5 (Batch	(None, 15, 1, 32)	128
flatten_7 (Flatten)	(None, 480)	0
dense_26 (Dense)	(None, 64)	30784
dropout_10 (Dropout)	(None, 64)	0
dense_27 (Dense)	(None, 10)	650

Total params: 45,514
Trainable params: 45,322
Non-trainable params: 192

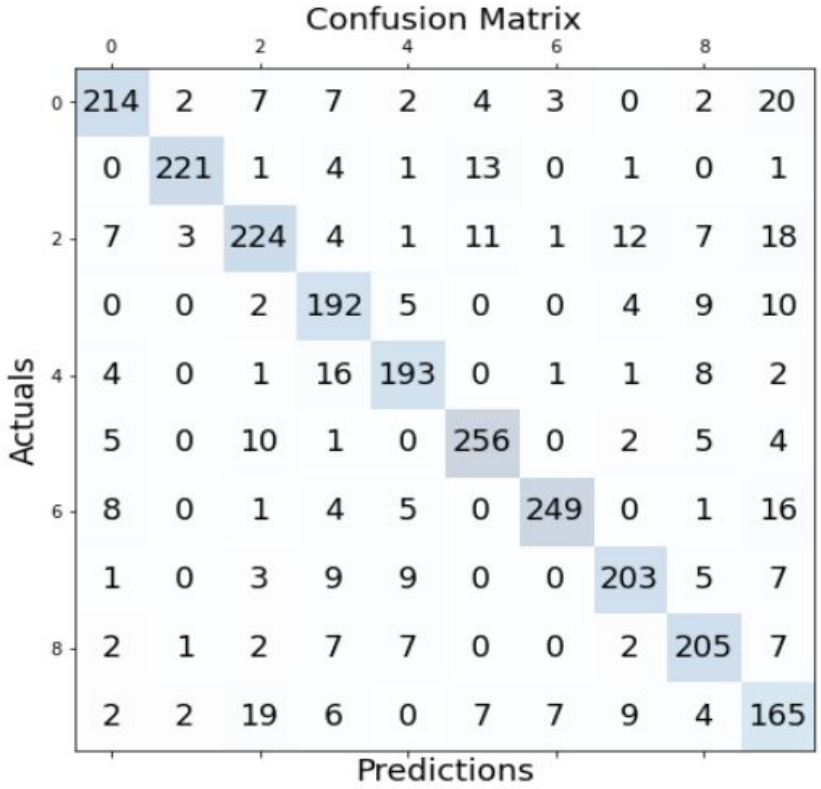
Evaluation of CNN Model

{0:'blues', 1:'classical', 2:'country', 3:'disco' , 4: 'hiphop', 5:'jazz', 6:'metal', 7:'pop', 8:'reggae', 9:'rock'}

First row represent out of 261 examples of class '0' 214 are predicted correct.

CONFUSION MATRIX

	precision	recall	f1-score	support
0	0.88	0.82	0.85	261
1	0.97	0.91	0.94	242
2	0.83	0.78	0.80	288
3	0.77	0.86	0.81	222
4	0.87	0.85	0.86	226
5	0.88	0.90	0.89	283
6	0.95	0.88	0.91	284
7	0.87	0.86	0.86	237
8	0.83	0.88	0.86	233
9	0.66	0.75	0.70	221
accuracy			0.85	2497
macro avg	0.85	0.85	0.85	2497
weighted avg	0.85	0.85	0.85	2497



Result and Conclusion

- We see that our CNN struggled most with the rock genre.
- It only managed to correctly classify 70% of rock audio as rock, labeling the others as mainly country or pop.
- Additionally, it incorrectly classified some country, disco and metal as rock music.
- Rock was a challenging genre – a qualitative inspection of rock mel-spectrograms implies that many rock music excerpts lack the easily visible beats that other genres such as hip-hop and disco possess.

Model	Accuracy(Training)	Accuracy(Testing)
Random Forest	47%	49%
KNN	70%	69%
CNN	75%	85%

Future Scope of work

- Can use more genres of music.
- Can use wavelet transform for feature extraction.
- Since, our dataset is a bit small we can combine different dataset to generate a big data which can cover as many variations of music as possible.
- Can extend our work to Speech recognition of different species and also to human’s mood recognition.

References

[1] J.W. Picone, Signal modeling techniques in speech recognition. Proc. IEEE 81, 1215–1247 (1993)

[2] J. Volkmann, S. Stevens, E. Newman, A scale for the measurement of the psychological magnitude pitch. J. Acoust. Soc. Am. 8, 185–190 (1937)

[3] Z. Fang, Z. Guoliang, S. Zhanjiang, Comparison of different implementations of MFCC. J. Comput. Sci. Technol. 16, 582–589 (2000)

[4] G.K.T. Ganchev, N. Fakotakis, Comparative evaluation of various MFCC implementations on the speaker verification task, in Proceedings of International Conference on Speech and Computer (SPECOM) (2005), pp. 191–194

[5] J.S. Mason, X. Zhang, Velocity and acceleration features in speaker recognition, in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) (1991), pp. 3673–3676

THANK YOU!