Music Genre Classification

Presented By:-Sanket Revadigar (801203510)

Problem Statement

- Classifying the music dataset into different genres using machine learning techniques
- Genres include blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock
- Music Genre Classification has been one of the most prolific areas in machine learning, specifically, and in computer science, generally.
- One of the most popular classification methods for this is the use of deep learning techniques, most notably the Neural Networks

Dataset

- The dataset is of GTZAN dataset, which contains 1000 music files
- Dataset has ten 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.
- http://opihi.cs.uvic.ca/sound/genres.tar.gz

Motivation

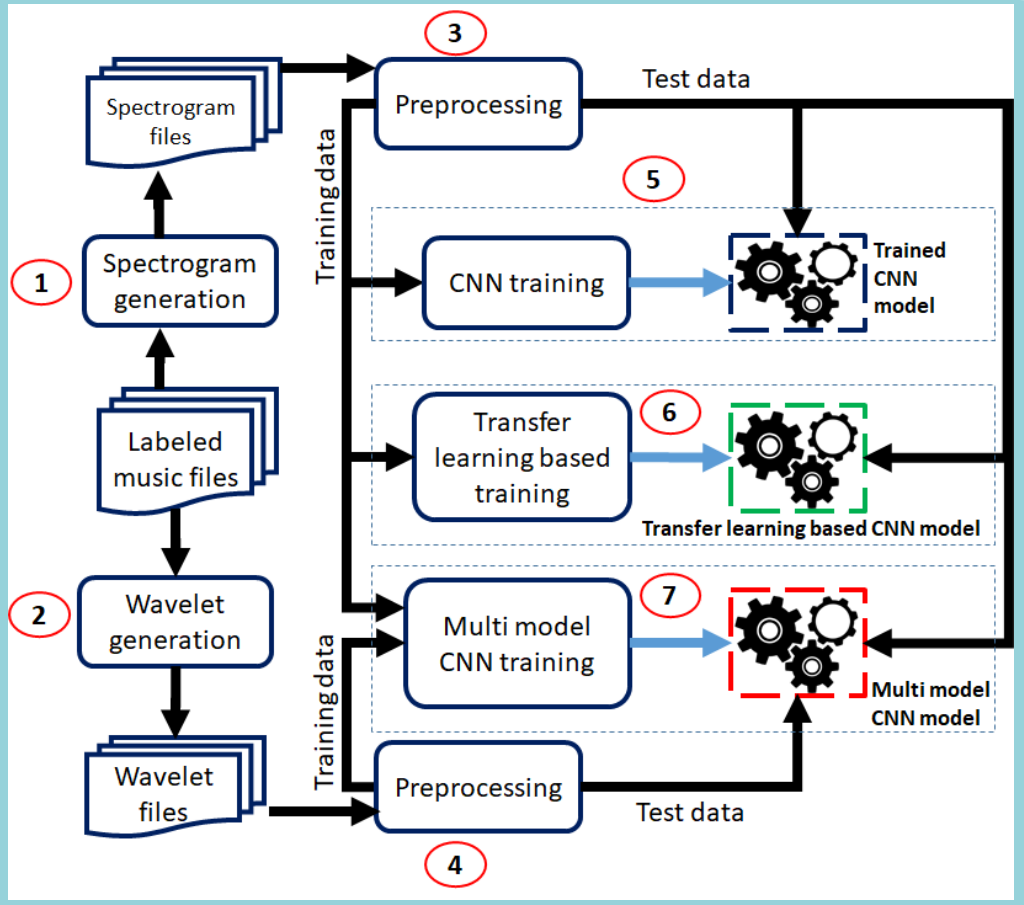
- In the field of music information retrieval, categorizing music files according to their genre is a difficult issue.
- People are finding it increasingly difficult to control the songs they listen to, thanks to the expansion of internet music databases and easy access to music content.
- One way to categorize and organize songs is based on the genre, which is identified by some characteristics of the music such as rhythmic structure, harmonic content and instrumentation.
- It would be helpful for audio streaming services like Spotify, iTunes and also for the user to be able to automatically classify and tag music in a user's library depending on genre.

Existing Related Approaches

- There are many ways through which the model can be trained. Some of these approaches are:-
- Multiclass Support Vector Machines
- K-Means Clustering
- K-Nearest Neighbors
- Convolutional Neural Networks
- In this project the **Convolutional Neural Networks** is used to train the model.

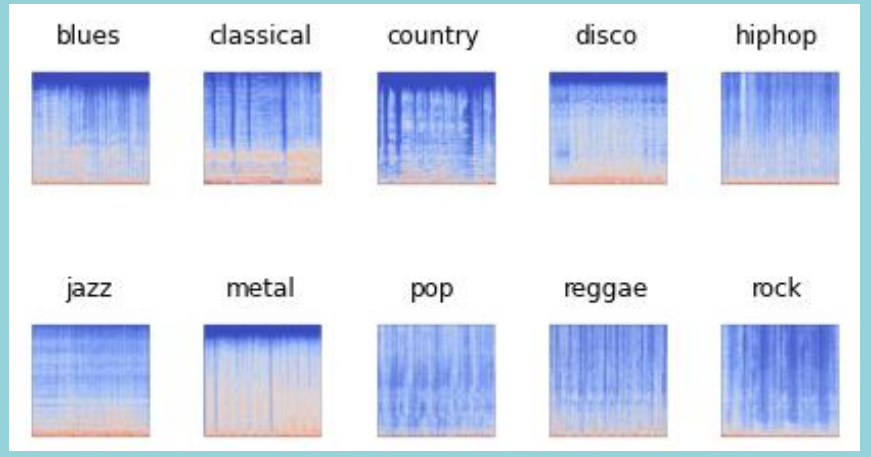
The Model

- Figure represents the overview of the methodology for the genre classification task
- 3 types of deep learning models were prepared
- But I have used the first CNN model and created the 1st model using Spectrograms.
- Used the Wavelets data to create the 2nd CNN model
- General image preprocessing steps to generate training and testing data.
- Each image is of size (256, 256, 32)

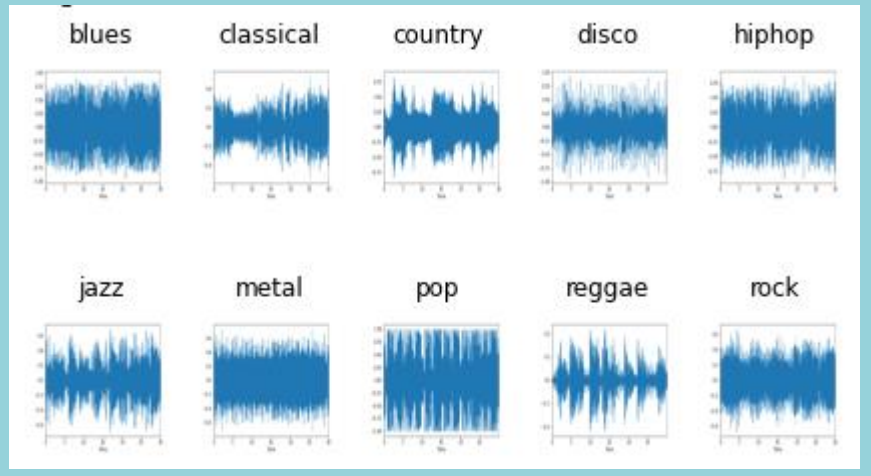


Spectrograms and Wavelets

- A spectrogram is a visual representation of the spectrum signal frequencies as it varies with time.
- Used librosa library to transform each audio file into a spectrogram.



- The Wavelet Transform is a transformation that can be used to analyze the spectral and temporal properties of non-stationary signals like audio.
- Used librosa library to generate wavelets of each audio file

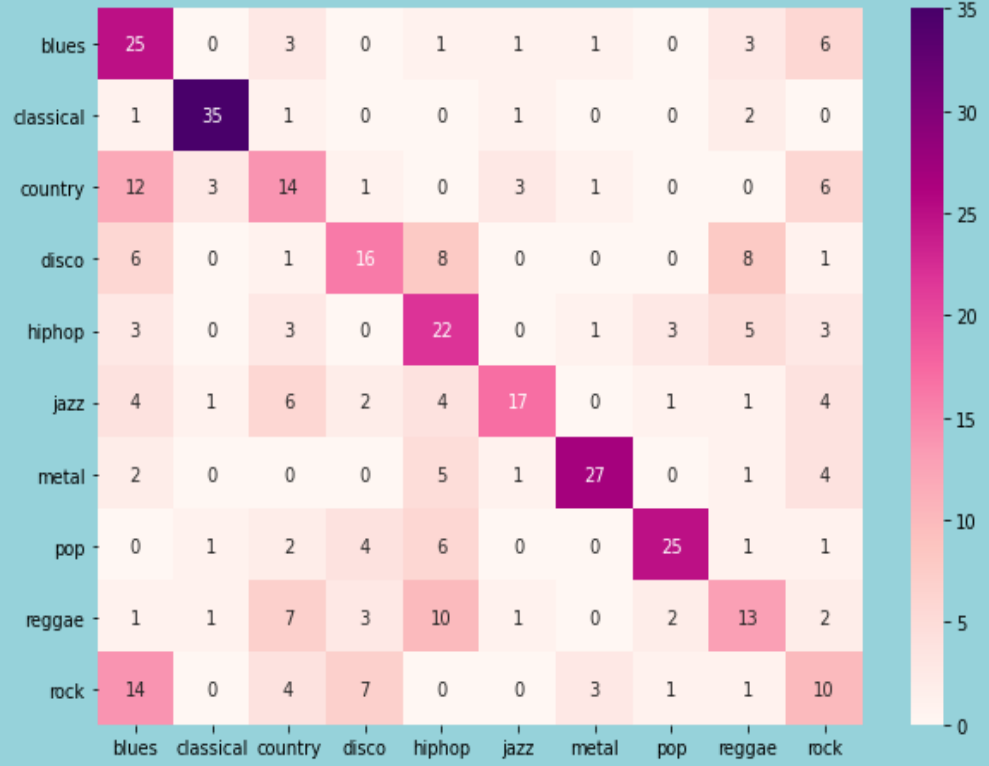


Basic CNN model training

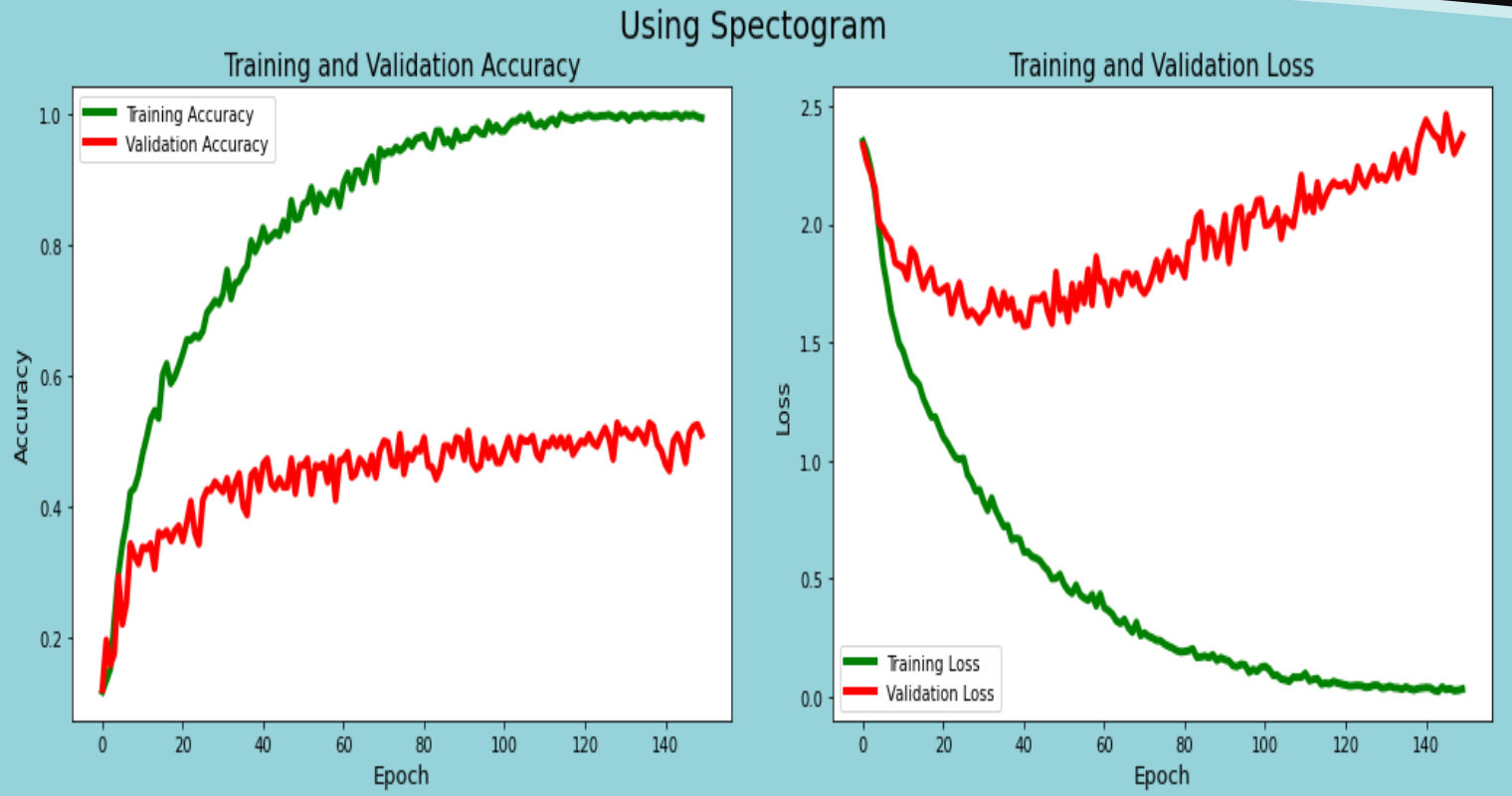
- After preprocessing the data, the first deep learning model.
- Constructed a Convolution Neural Network model with required input and output units.
- The model is trained for 150 epochs.
- Adam optimizer is used with learning rate of 0.0001
- Used categorical cross-entropy as the loss function.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 256, 256, 32)      896
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 128, 128, 32)      0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 128, 128, 32)      9248
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 64, 64, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 64, 64, 64)        18496
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 32, 32, 64)        0
_________________________________________________________________
dropout (Dropout)            (None, 32, 32, 64)        0
_________________________________________________________________
flatten (Flatten)            (None, 65536)             0
_________________________________________________________________
dense (Dense)                (None, 128)               8388736
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1290
=================================================================
Total params: 8,418,666
Trainable params: 8,418,666
Non-trainable params: 0
_________________________________________________________________
```

Result using Spectrograms

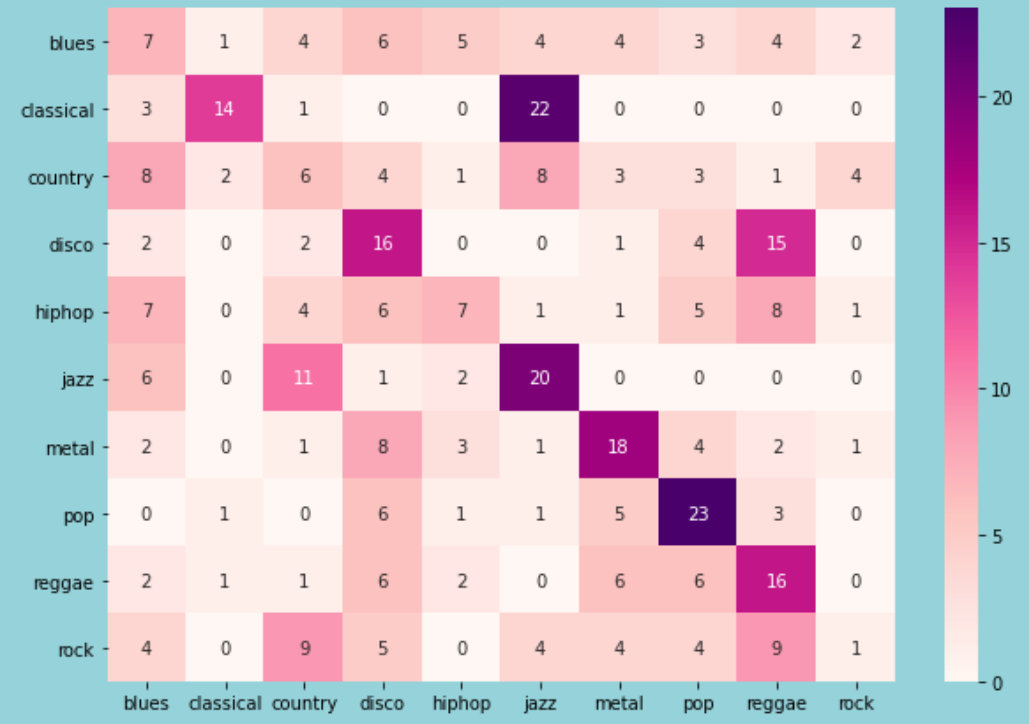


Confusion Matrix

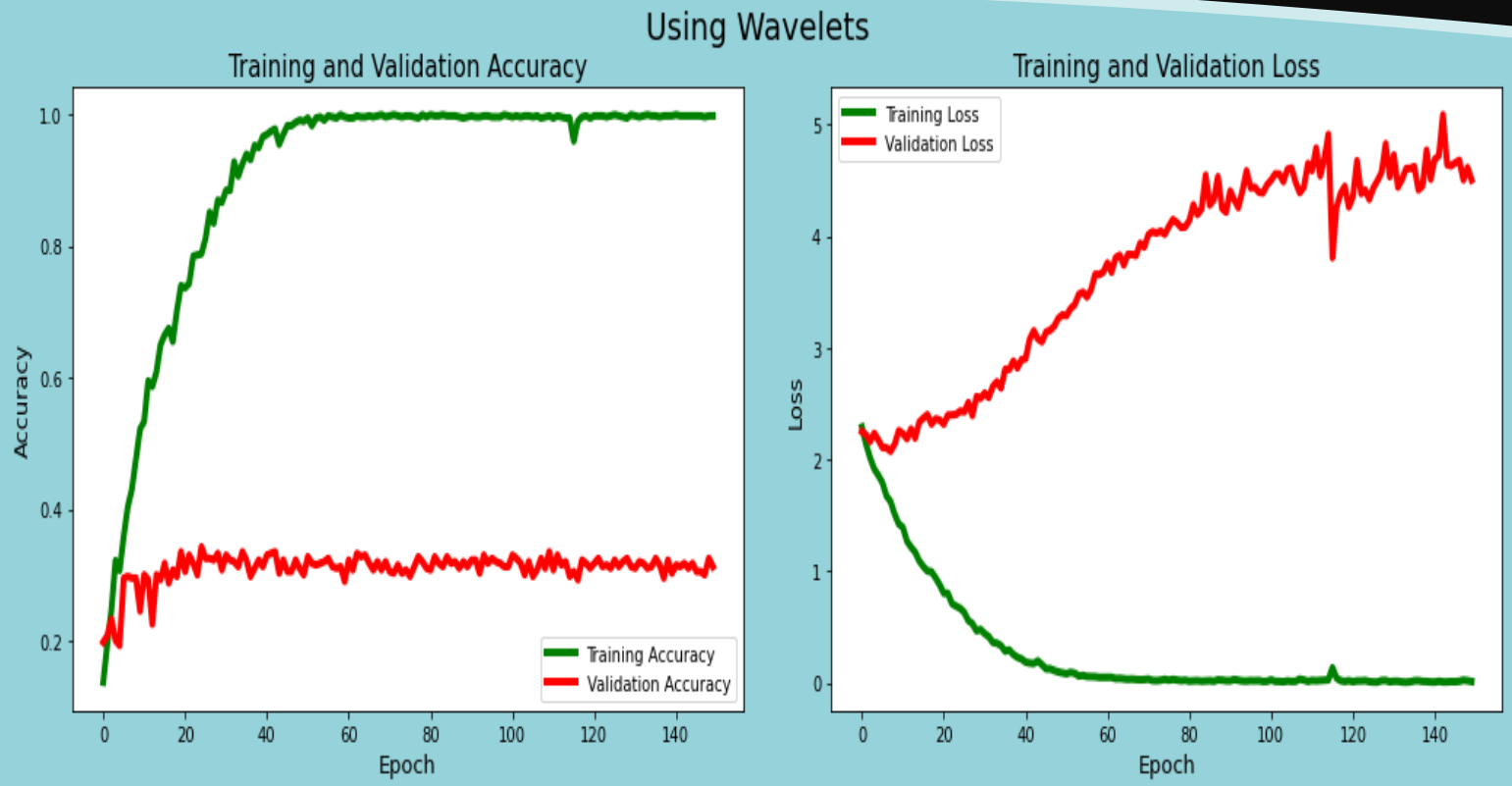


Accuracy and Loss Graphs

Result using Wavelets

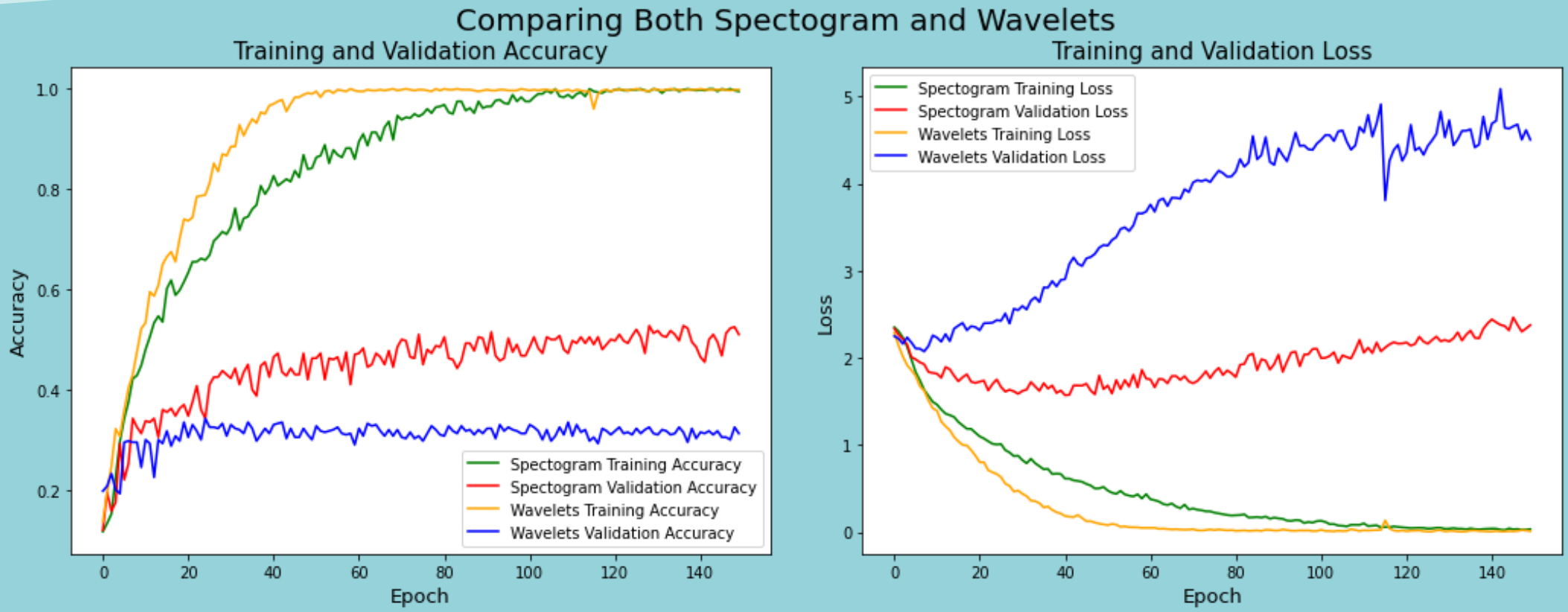


Confusion Matrix



Accuracy and Loss Graphs

Result of both models



Accuracy and Loss Graphs

Contribution

- I have used the 1 of the 3 models that were presented in this repo, and created 1 more model using different dataset (Wavelet) as a standalone dataset.
- The previous work used 60:40 split in the data for training and testing, so I created a new dataset with a split of 70:30 but it turns out, the accuracy fell down by 7-8% with this new split so I reverted back to old 60:40 split.
- I observed that the model was set to run for 500 epochs which was actual not yielding any valuable results, so I set the new Epoch value to 150.
- This repo is 2-3 years old, many of the syntax were changed, so I updated the code so that is was supporting the newer versions of the libraries.
- The comparison chart.

Conclusion

- We can clearly see from the results that using spectrograms for classifying the music genre is far better.
- Its still difficult to get better accuracies as the topic is still new and there are no well known models for audio transforming.
- Many new techniques might arrive in future to classify the audio in original form and create a better model.
- It’s even difficult for normal humans to classify the genres by just listening a 30 second audio clip.
- Multiple models can be clubbed together to achieve a better accuracy in future iterations.

References

- Hareesh Bahuleyan. 2018. Music Genre Classification using Machine Learning Techniques. arXiv:1804.01149v1 [cs.SD]
- George Tzanetakis, Perry Cook. 2002. Musical genre classification of audio signals. IEEE Transactions on speech and audio processing 10(5):293– 302.
- Lonce Wyse. 2017. Audio spectrogram representations for processing with convolutional neural networks. arXiv preprint arXiv:1706.09559
- https://www.analyticsvidhya.com/blog/2021/06/music-genres-classification-using-deep-learning-techniques/
- https://github.com/sawan16/Genre-Classification-using-Deep-learning

Thank You