Project Overview
By Noopur R K and Ruchika Mishra

Github : https://github.com/noopurrkalawatia/Dhvani

1. **Problem Statement:** Urban Sound classification using the deep learning techniques like Multilayer Neural Network, Convolutional Neural Network and Recurrent Neural Network. We are going to evaluate the best model and compare the performance metrics for each model.

2. **Proposed System Model:**

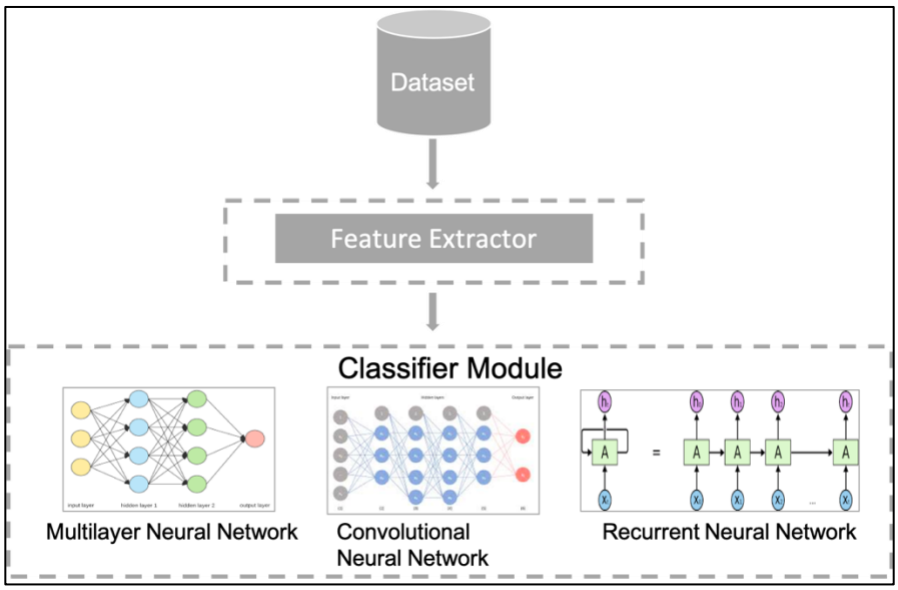

Fig – 1 System Model

The proposed system model mainly consists of three modules:

2.1 Dataset / Input Module: The dataset we are using for the models/project is the UltraSound8k dataset. The dataset is made by Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. The dataset consists of 8753 samples of .wav sounds. The entire dataset is a folder structure consisting of sounds in .wav form and the metadata of each waveform is also provided. There is a total of 10 categories of sound samples present in the dataset.

The following are the ten categories,

1. Air conditioner
2. Children playing
3. Dog barking
4. Siren
5. Engine Idling
6. Gun Shot
7. Jackhammer
8. Drilling
9. Car horn

10. Street Music

The preprocessing done to the dataset includes padding the shorter waveform with zeroes and setting the frame size to a fixed value for all the waveforms in order to maintain the uniformity of the data to be fed to the models.

2.2 Feature Extractor:

The features extracted from the dataset are as shown below,

- ***Melspectrogram*:** Acoustic time-frequency representation of sound
- ***MFCC***: Mel-frequency cepstral coefficients.
- ***Chorma-stft*:** Entire spectrum is divided into 12 bins representing 12 semitones of the musical octave.
- ***Spectral_contrast*:** Decibel difference between peaks and valleys in the spectrum.
- ***Tonnetz*:** Tonnetz (German: tone-network) is a conceptual lattice diagram representing tonal space
- ***Filterbanks and log filterbanks***

2.3 Classifier Module :

For the classifier model we aim on implementing atleast the three techniques,

1. Multilayer Neural Network
2. Convolutional Neural Network
3. Recurrent Neural Network

A short introduction to the same is already described in the report submitted by us.

3. Preliminary Results:

We have completed the preprocessing of data and also the extraction of the features. Currently we are working on the multilayer neural network and optimizing the same with the help of the hyperparameters. We have been having problems with issues like overfitting and underfitting. Given below are the visualization of the features extracted from the dataset.

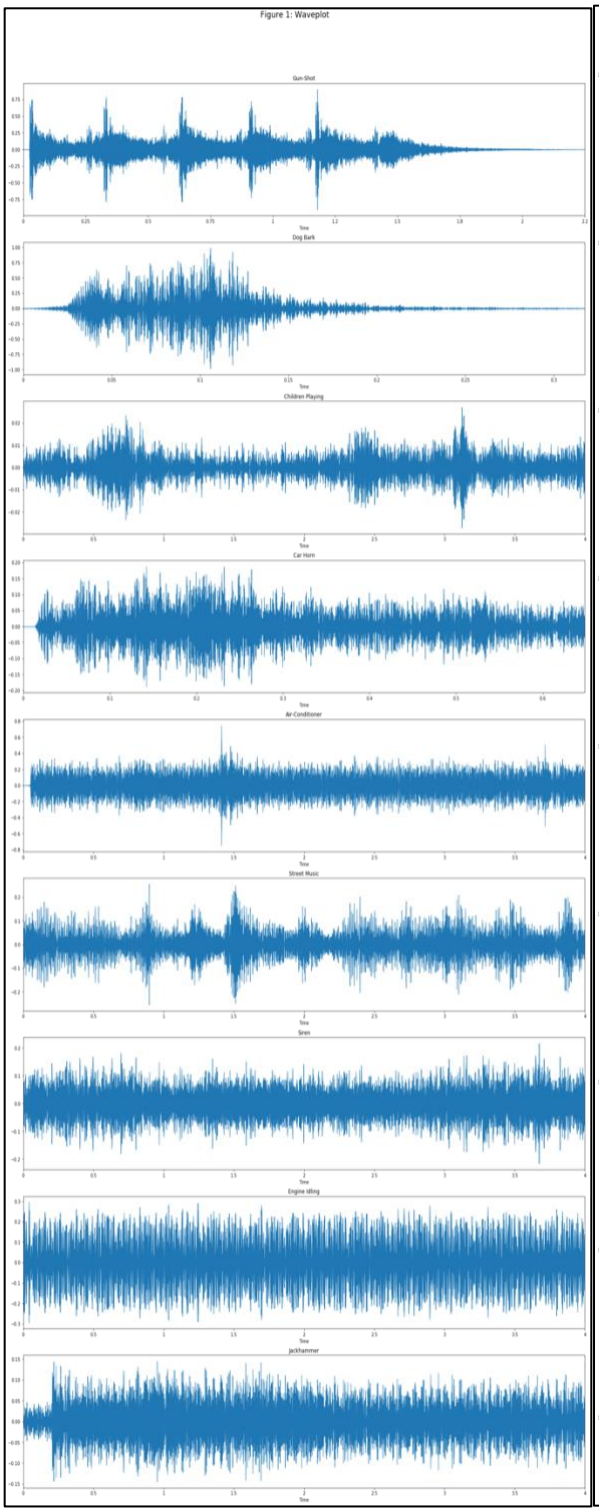

Fig – 2 : Raw waveforms of the data

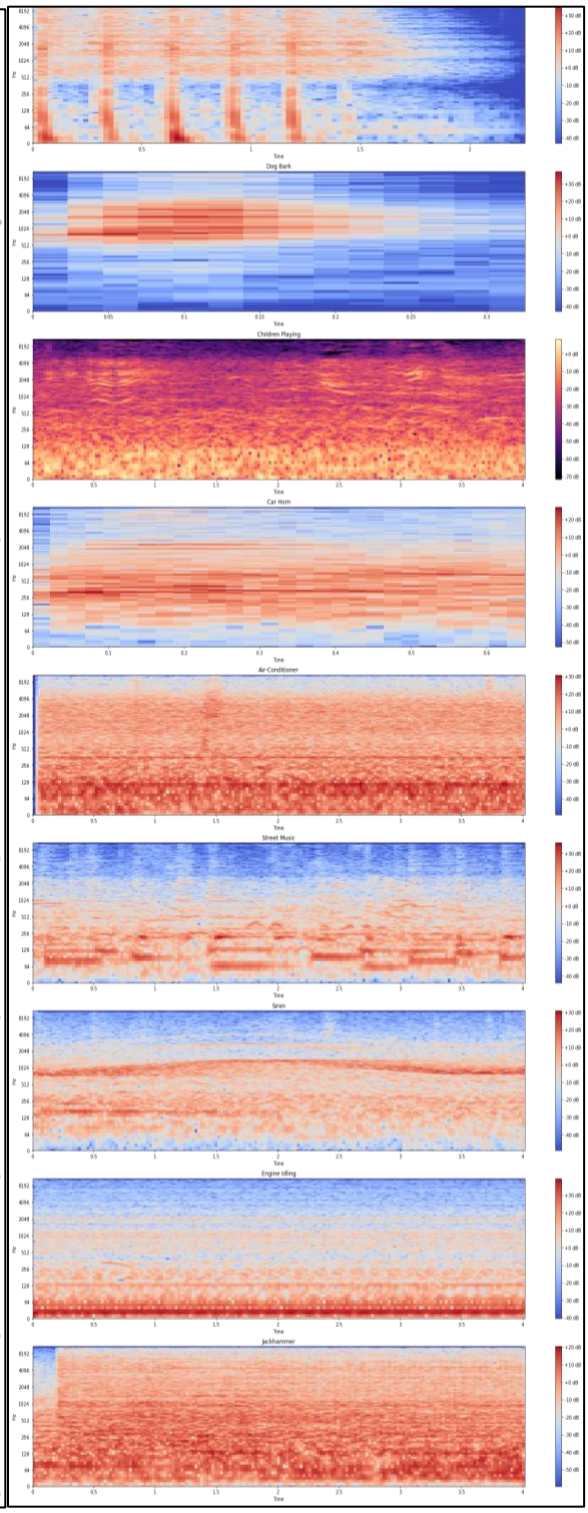

Fig – 3: Log Power Spectrogram

The power spectrogram is mentioned in the report furnished. The initial results from training the multilayer neural network are presented in the graph below. The multilayer neural network that we trained has three hidden layers and has 280, 300, 270 neurons respectively. We have achieved a test accuracy of about 0.893 for our dataset. The model was run on i5 8gb RAM processor.

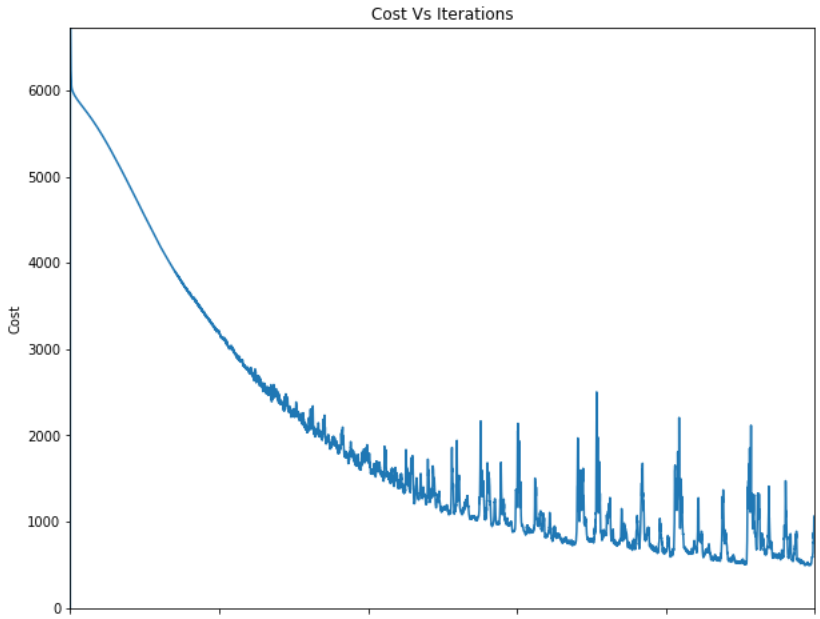

Fig – 4 Cost vs Iterations for MLP.

4.Performance Metrics: The following performance metrics for used for our project,

- **Accuracy**
  Total no. of correct results vs Total size
- **Precision**
  talks about how precise/accurate you model is out of those predicted positives. It is a good measure to determine when cost of false positives is high.
- **Recall**
  calculates how many of false negatives was our model able to capture. It is important in cases when cost associated with false negatives are high.
- **F1 score**
  Might be a better measure to use if we need to seek a balance between Precision and Recall and there is an uneven class distribution.

****