

MCA Assignment2

Kshitiz 2016051

March 2020

1 Question1

1.1 Feature Extraction

Spectrogram features are calculated as follows :

1. **Preamphasis** is done on the input signal, so as to boost the energy of high frequency part of the signal. (This is optional, can be done without this preprocessing)
2. Dividing the input signal in frames.

$$\text{WindowSize} = 25\text{milliseconds}$$

$$\text{Stride} = 10\text{milliseconds}$$

3. The **Hamming filter** is applied on each of the frames, so as the start and the end of the frames to avoid the sudden drop of amplitude near the edges.
4. Zero **Padding** is done on the signals which has different size than the other signals.
5. **DFT** on each frame is calculated, which is then used to calculate the **power spectrum** of the frame. DFT is implemented from scratch.

1.1.1 Results

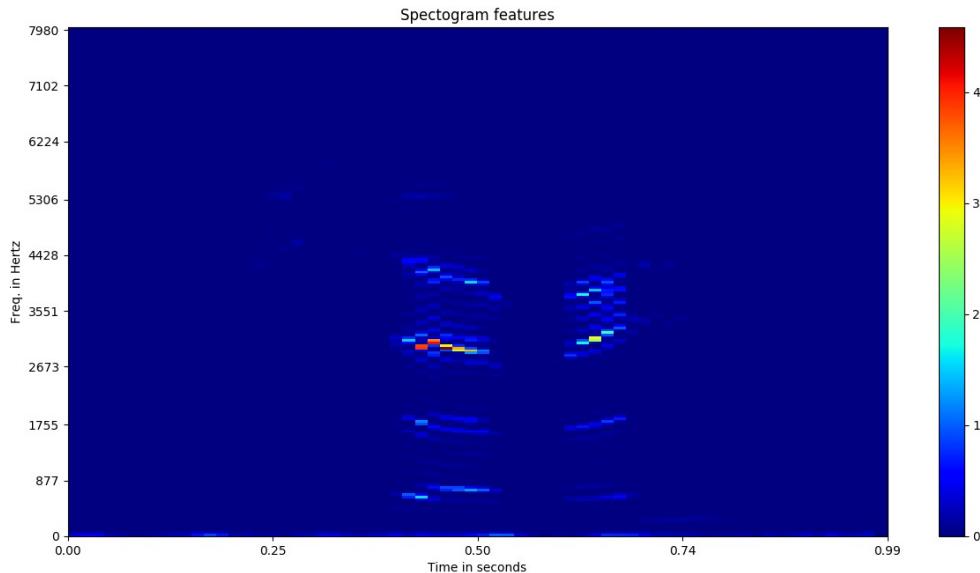
Classes	Spectograms									
	0	1	2	3	4	5	6	7	8	9
Precision	0.322	0.320	0.280	0.320	0.504	0.560	0.491	0.529	0.569	0.433
Recall	0.212	0.361	0.339	0.476	0.736	0.326	0.527	0.316	0.391	0.478
F1-Score	0.255	0.339	0.306	0.382	0.598	0.413	0.508	0.395	0.463	0.454

Spectograms with Noise										
Classes	0	1	2	3	4	5	6	7	8	9
Precision	0.4	0.287	0.282	0.314	0.411	0.444	0.424	0.304	0.392	0.267
Recall	0.407	0.243	0.233	0.258	0.461	0.392	0.298	0.357	0.346	0.465
F1-Score	0.404	0.264	0.255	0.283	0.434	0.417	0.350	0.329	0.368	0.340

Miscellaneous		
Type	Without Noise	With Noise
Average Precision	0.433	0.353
Average Recall	0.416	0.346
Average F1-Score	0.411	0.344
Accuracy	0.419	0.348035

1.2 Plotting of Spectrogram feature

The following spectrogram is of sound of seven.



2 Question2

2.1 Feature Extraction

MFCC features are calculated as follows :

1. **Preamphasis** is done on the input signal, so as to boost the energy of high frequency part of the signal.

$$x'[t_d] = x[t_d] - \alpha x[t_d-1] \quad 0.95 < \alpha < 0.99$$

2. Dividing the input signal in frames.

$$\text{WindowSize} = 25\text{milliseconds}$$

$$\text{Stride} = 10\text{milliseconds}$$

3. The **Hamming filter** is applied on each of the frames, so as the start and the end of the frames to avoid the sudden drop of amplitude near the edges.

Hamming ($\alpha = 0.46164$) or Hanning ($\alpha = 0.5$) window

$$w[n] = (1-\alpha) - \alpha \cos\left(\frac{2\pi n}{L-1}\right) \quad L : \text{window width}$$

4. Zero **Padding** is done on the signals which has different size than the other signals.
5. **Energy of each frame** is calculated by summing all the values for a frame.
6. **DFT** on each frame is calculated, which is then used to calculate the **power spectrum** of the frame. "np.fft.rfft" library is used to calculate the same.

$$X[k] = \sum_{n=0}^{N-1} x[n] \exp\left(-j\frac{2\pi}{N}kn\right)$$

7. **Mel Filters** are calculated now. This helps the signal to align with the human perception of sound, giving the high priority to lower frequencies. This filter bank is now multiplied to the frames.

$$Y_t[m] = \sum_{k=1}^N W_m[k] |X_t[k]|^2$$

where k : DFT bin number ($1, \dots, N$)
 m : mel-filter bank number ($1, \dots, M$)

8. **DCT** is applied on each of the frames and top/first 12 values(excluding 1st) are used for the features.
9. **Energy element** is now append to the above calculated features, giving a vector of length 13 for each frame.
10. Now **delta features** are calculated giving the vector of 39 for each frame.

2.1.1 Results

MFCC										
Classes	0	1	2	3	4	5	6	7	8	9
Precision	0.727	0.306	0.298	0.494	0.665	0.493	0.761	0.645	0.295	0.396
Recall	0.4612	0.509	0.275	0.359	0.546	0.455	0.584	0.304	0.749	0.265
F1-Score	0.565	0.382	0.286	0.416	0.6	0.473	0.661	0.413	0.423	0.318

MFCC with Noise										
Classes	0	1	2	3	4	5	6	7	8	9
Precision	0.765	0.276	0.436	0.460	0.754	0.675	0.693	0.422	0.234	0.593
Recall	0.35	0.591	0.072	0.258	0.296	0.215	0.630	0.558	0.839	0.152
F1-Score	0.480	0.377	0.1234	0.331	0.426	0.326	0.66	0.481	0.366	0.242

Miscellaneous		
Type	Without Noise	With Noise
Average Precision	0.508	0.531
Average Recall	0.451	0.396
Average F1-Score	0.454	0.381
Accuracy	0.453	0.398

3 Question3

3.1 SVM training

Following steps are followed for training the data.

1. The feature data is stored previously in pickle file, where each data point has a feature matrix.
2. The data is loaded and is ravelled and then min-max scaling is done on the data.
3. LinearSVC is used from sklearn library for learning the data.
4. Now on the trained model : Class-wise Precision, Recall, F1 Score is calculated. Average Precision, average recall, average F1 score, and accuracy is also reported for the same.

3.2 Noise Augment

Noise is added according to the following steps.

1. Similar to data, noise features are also calculated and stored in a pickle file.
2. Now for added the noise in the data, random 50% of the datapoints are selected and random noise is added in that datapoint. Weighted average of noise and datapoint is calculated for adding the noise in data.
3. Now when the data is calculated, similar steps like above are followed.
4. Same results are reported as in the previous section.

3.3 Inferences

- MFCC has greater accuracy, precision and recall than compared to the Spectral features.
- Min-Max scaling increases the accuracy, hence there is need of normalisation after calculating the speech features.
- As the noise is added in the data, the accuracy of the model decreases.
- If we use the model which is trained on the non-noisy features and then asked to predict the noisy validation set, the accuracy is decreased.
- MFCC being lower in the dimension (39), is easier to store and trains faster than the Spectrograms. But Spectrograms are faster to compute than MFCC, hence there is trade between them.

References

- [1] MFCC
- [2] DFT and FFT
- [3] SVM
- [4] Spectrograms