Indian Institute of Technology Jodhpur

Speech Understanding Programming Assignment - 2

Submitted by

MITESH KUMAR (M23MAC004)

https://github.com/mitesh-kr/Speech_Understanding_Assignment_2.git

Question 1:

1.2

wavlm base plus has been used for speaker verification and identification tasks, and the imported pre-trained model is fine-tuned on 100 identities OF the VoxCeleb2 dataset using LoRA (Low-Rank Adaptation) and ArcFace loss.

Trainable Parameters: 1,953,280
Non-Trainable Parameters: 94,381,936
Total Parameters: 96,335,216
Percentage of Trainable Parameters: **2.03%**

Speaker Identification:

There are two methods

1. If the model not trained on the dataset
   computes speaker centroids by extracting embeddings from a model for each speaker's audio samples and averaging them to obtain a single representative vector per speaker.This results in a compact speaker representation, enabling efficient speaker verification, identification, and clustering by comparing new voice samples with precomputed centroids.

2. If the model is trained on the dataset for this, we need to find the identity of the speaker, It's simple to find the label from the model like the classification task.

Speaker Verification:

Both audio pairs are fed in the model, and then the normalised embedding is used to get similarity. Based on the obtained similarity, it can we conclude that whether the both auios are of same speaker or different. Since the similarity score lies between -1 to 1. So zero is taken as the threshold value.

Model is fine-tuned for 10 epochs on

Speaker identification embeddings based accuracy: **85.76%**
Speaker identification logistics based accuracy: **78.09%**

Here's a comparison table of the evaluation results for the pre-trained model versus the fine-tuned model on list of trial pairs - VoxCeleb1 (cleaned)

Metric	Pre-trained Model	Fine-tuned Model
EER (Equal Error Rate)	44.32%	16.00%
TAR@1%FAR	2.12%	31.85%
Verification Accuracy	49.99%	62.17%
Identification Accuracy	55.72%	83.46%

Here are the improvements in different metrics after fine-tuning:

The improvement in Equal Error Rate (EER): **28.32%**
TAR@1%FAR Improvement: **29.73%**
Verification Accuracy Improvement: **12.18%**
Identification Accuracy Improvement: **27.74%**

1.3

Audio Mixing

Two audios are mixed randomly, the mixed audio is kept at the length equal to the minimum among two. They are samples at 16 KHz. For each pair of speakers two mixed audios are being generated. In this way ,there are 1250 speaker pairs and total of 2450 mixed audio are created both for train and test.

Audio Separation

pre-trained SepFormer model used for separation, the model generates the separated audio at 8 KHz. Now, to compare the separated audios with the source audios, the audios are trimmed to minimum length and downsampled to 8 KHz.

Evaluation of test data (50 speakers mixed audios)

Here are the evaluation metrics presented in a horizontal table:

Metric	SDR	SIR	SAR	PESQ
Value	3.82	19.34	5.19	1.22

Here's the updated comparison table with both accuracy metrics for pre-trained and fine-tuned WavLM on separated audio by SepFormer model

Model	Embedding Identification Accuracy	Classification Identification Accuracy
Pre-trained WavLM	17.73%	
Fine-tuned WavLM	38.50%	56.83%

The table shows significant improvement in embedding identification accuracy after fine-tuning (from 17.73% to 38.50%), while the classification identification accuracy for the fine-tuned model is 6.83%.

Rank 1 is Fine-tuned WavLM with an Embedding Identification Accuracy of **56.83%.**

1.4 The pre-trained SepFormer model mode is integrated with the WAVLM model to create a pipeline/algorithmic approach to combining the speaker identification model with the SepFromer model to perform speaker separation with the speaker identification model and speech enhancement. LoRA and arch facebook loss are already considered with the wavlm for classification loss. The separation loss in sepFormer consists of SI-SNR, L2 waveform loss and STFT-based perceptual loss. The whole model loss is the combination of separation loss and classification loss.

l2_loss_weight: 0.2
perceptual_loss_weigh: 0.3

Total loss = separation loss + 0.2 * claaasification loss

Total Number of Parameters: **124.1M**
Total Number of Trainable Parameters: **29.7M**
Trainable Parameters represent: **23.9202%**

Evaluation of test data (50 speakers mixed audios)

Here are the evaluation metrics presented in a horizontal table:

Metric	**SDR**	**SIR**	**SAR**	**PESQ**
Value	4.10	20.03	5.37	1.35

Here's the updated comparison table with both accuracy metrics for pre-trained and fine-tuned WavLM on separated audio by SepFormer + WAVLM model

Model	**Classification Identification Accuracy**
Pre-trained WavLM	47 %
Fine-tuned WavLM	70 %

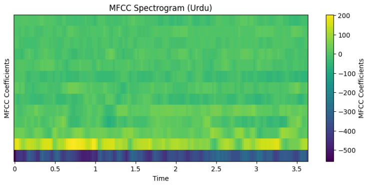
Overall, the sepFormer + wavlm separator and identificer improve et the quality of separated audio,by which the identification accuracy increases from **56.83% to 70 %.**

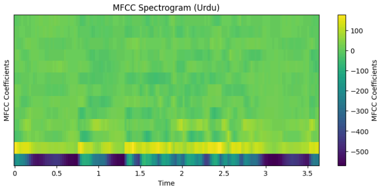
Question 2: MFCC Feature Extraction and Comparative Analysis of Indian Languages
Google Colab Link :

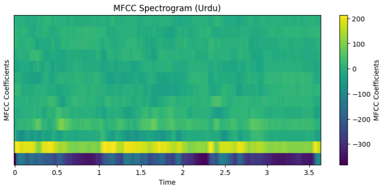
Task A

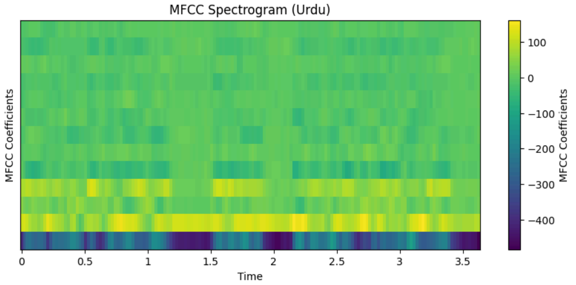
For the analysis, I have chosen the 13 MFCC coefficients. Below are the spectrogram plots for Urdu, Punjabi and Bengali.

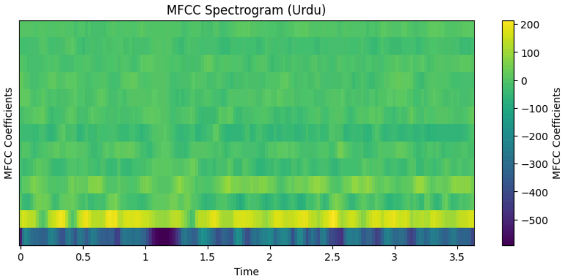
Urdu



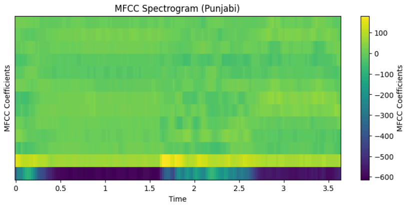


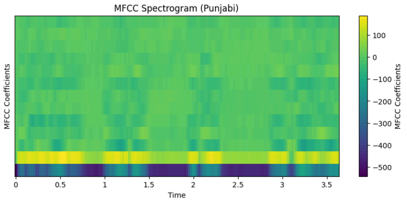


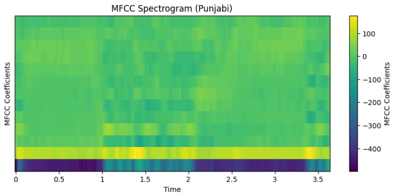


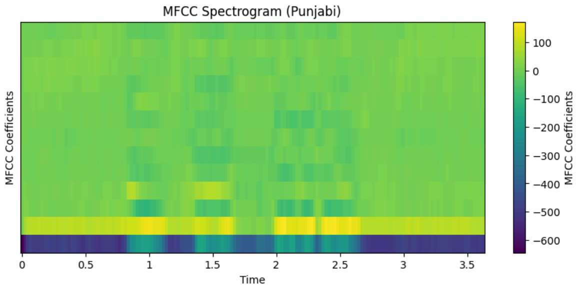


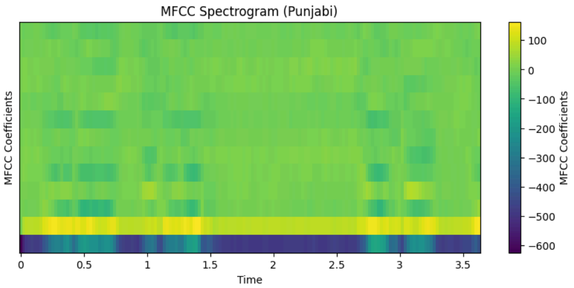
Punjabi



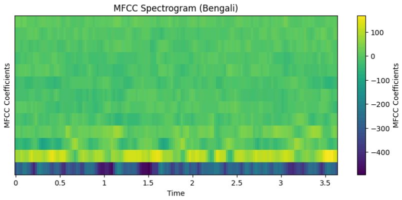


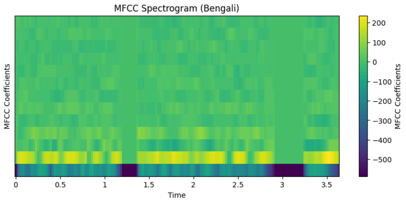


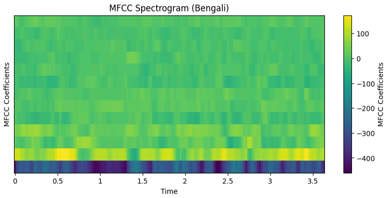


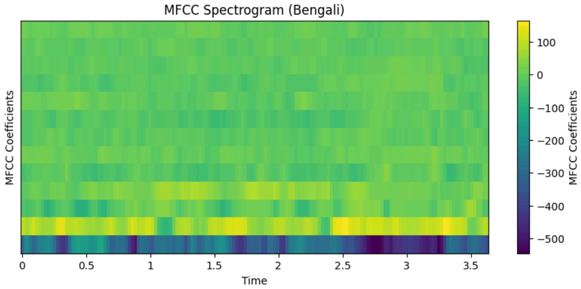


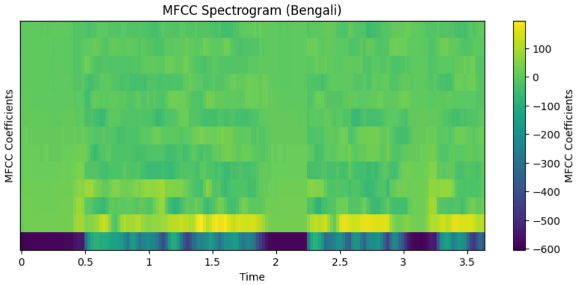
Bengali











Analysis

4.

Urdu:

The Urdu spectrogram often shows very sharp and sudden bursts of energy. This means that, when someone speaks Urdu, there are quick and clear changes in the sound. These bursts can be seen as sudden spikes on the spectrogram. Such spikes usually occur when the speaker makes strong, clear sounds—especially consonants that need a burst of energy. This results in a pattern where some parts are very busy and full of energy over a short time.

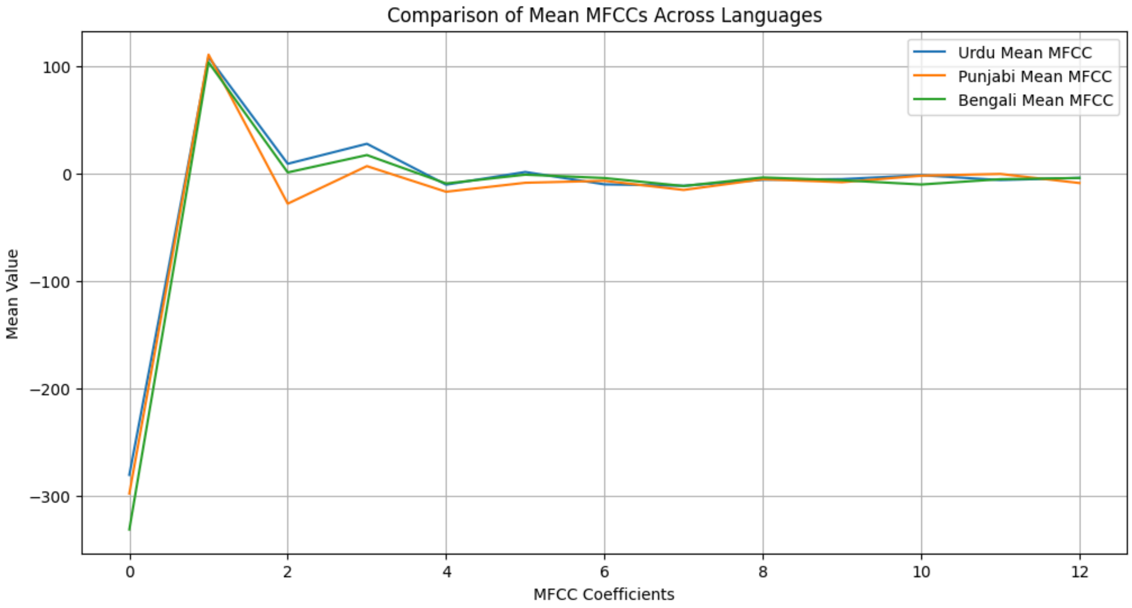
Punjabi:

For Punjabi, the spectrogram looks more even and smoother overall. The energy does not spike as sharply as in Urdu. Instead, the sound changes gradually, which creates a steady flow on the spectrogram. This smooth, continuous pattern suggests that the speech has a regular rhythm with fewer sudden changes. The overall look is calm and balanced, making it easier to see the flow of the language.

Bengali:

The Bengali spectrogram appears softer and more blended. The energy transitions here are gentle, meaning that changes in the sound are gradual. This blending of energy gives the spectrogram a less harsh and more flowing appearance compared to Urdu. It indicates that the speaker might use a smoother style, where sounds merge together without abrupt shifts.

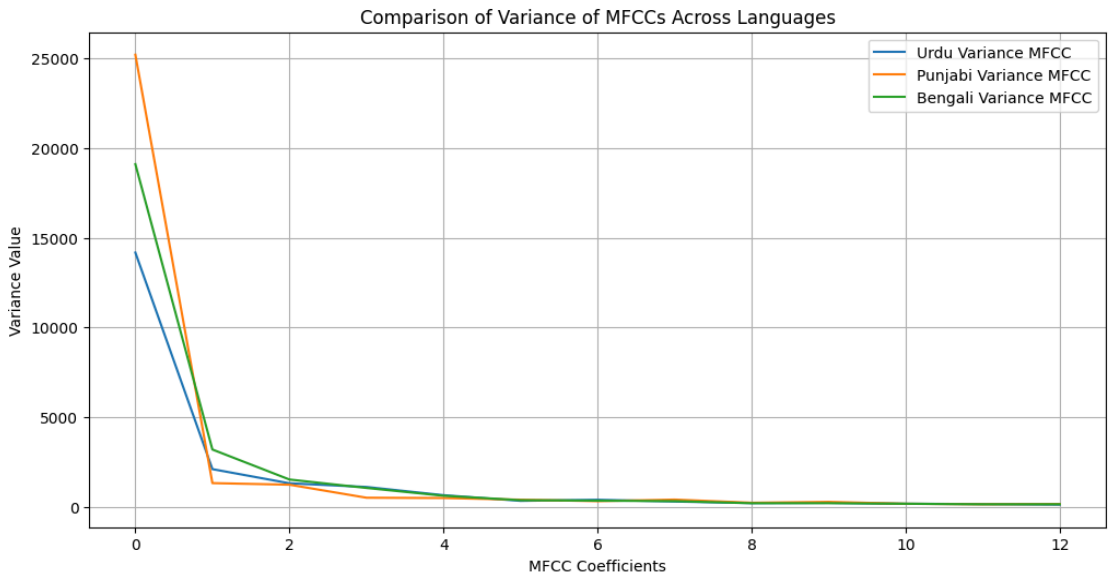
Overall, while all three languages are processed with the same method, the visual differences in the spectrograms reflect the unique ways each language is spoken. These differences in energy distribution and intensity help us see how the pronunciation and rhythm can vary from one language to another.



4.a

Mean of MFCC Coefficients

The **Mean** plot shows the average (mean) values of the MFCCs, the 0th coefficient (often linked to overall signal energy) stands out for all three languages. In the provided plot, you can see that this coefficient has a relatively high mean compared to the others. This suggests differences in the average energy or loudness of the speech samples. Beyond the 0th coefficient, most of the mean MFCC values for Urdu, Punjabi, and Bengali stay close to zero, indicating that the average spectral shapes (once overall energy is removed) do not deviate too much. Still, small differences in these means can hint at unique resonances or articulatory characteristics in each language.



Variance of MFCC Coefficients

The variance plot shows how much each MFCC coefficient changes over the speech samples. Again, the 0th coefficient has the highest variance across all three languages, meaning that the overall energy level fluctuates the most. Urdu, for instance, may show a higher variance in c0 than Punjabi or Bengali, pointing to more pronounced shifts in loudness. After c0, the variance drops sharply for coefficients 1 through 12, suggesting that the detailed spectral shapes within each language are relatively stable. Nonetheless, any small differences in variance among these higher-order coefficients could reflect subtle variations in how each language's sounds are produced.

Analysis on 50 spectrograms and below is results

```
Statistics for Urdu:

Mean MFCC Coefficients:

 [-26.127768 -21.34419  -23.645468 ... -15.357951  -9.380239  -9.865012]

Variance of MFCC Coefficients:

 [14178.093     2094.632     1305.2477    1099.7206     641.56635

   325.06418     386.8762      288.017      200.23186    213.66309

   162.55269     144.74312     111.902824]
```

```
Statistics for Punjabi:

Mean MFCC Coefficients:

 [-32.67228  -26.470266 -25.106089 ... -22.87138  -13.582006 -12.727427]

Variance of MFCC Coefficients:

 [25218.072    1314.8843   1226.1779    499.504     482.32062   389.63205

   307.78128   395.54987   219.50655   268.84705   161.71593   134.27835

   151.21587]


Statistics for Bengali:

Mean MFCC Coefficients:

 [-38.901363 -22.223969  -8.166681 ... -25.634403 -20.414656 -21.167189]

Variance of MFCC Coefficients:

 [19109.377    3189.3079   1515.9521   1044.3712    609.6966    362.6595

   335.10684   309.18436   193.98122   196.3765    163.67429   134.59758

   137.64032]
```

The mean values of the MFCC coefficients tell us the average feature levels in the speech. For example, if the first MFCC has a mean of –26 for Urdu, –33 for Punjabi, and –39 for Bengali, this shows that, on average, the overall energy or frequency pattern of the speech differs between these languages. Each language has its own “spectral fingerprint” based on these average values.

The variance shows how much the MFCC values change over time. A higher variance means the feature is changing a lot during speech, while a lower variance means it stays more consistent. For instance, if the first coefficient has a variance of about 14,178 for Urdu, 25,218 for Punjabi, and 19,109 for Bengali, it suggests that Punjabi speech has more fluctuations in that feature, pointing to a more dynamic use of energy compared to Urdu and Bengali.

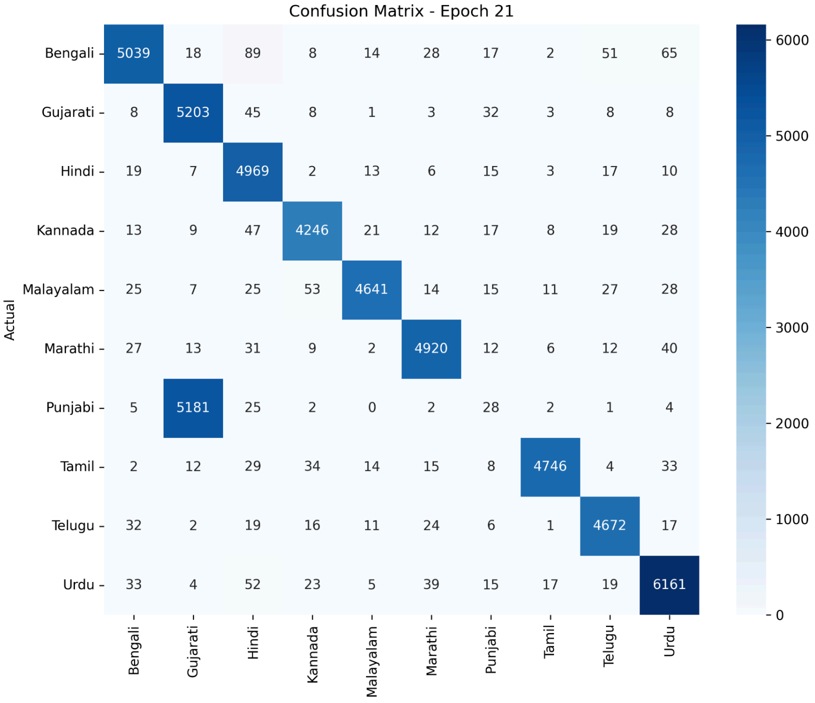
Task B.

For classification tasks I have used mfcc, which are stored in .npy files.
80 % for training and 20 % for validation and testing. Scaling has been done on the dataset.

A 7-layer simple ANN model having a Relu function and a dropout feature are added to avoid overfitting,

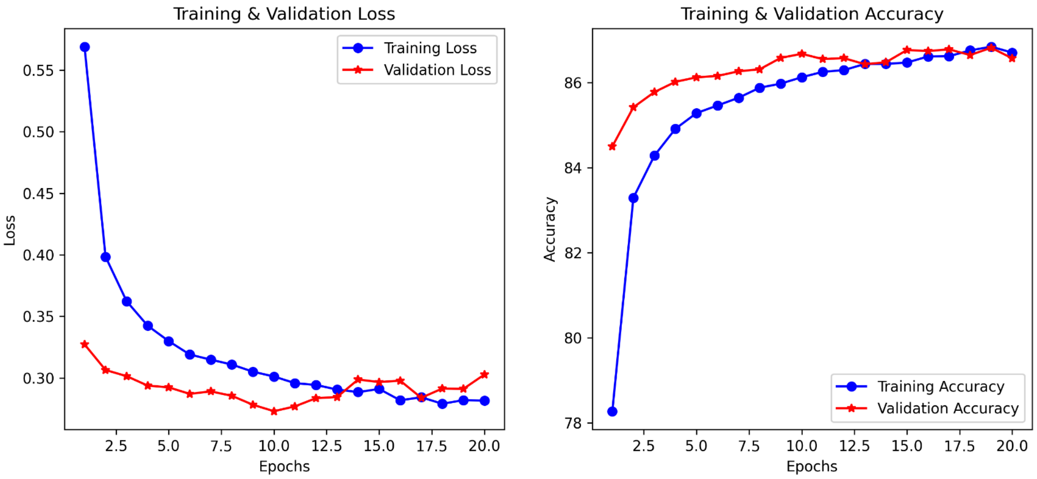
Training samples: 205454
Validation samples: 51364

```
Best result occurs after 17 epoche
✅ Train Loss: 0.2872,
✅ Train Acc: 86.65%,
✅ Validation Loss: 0.2814
✅ Validation Accuracy: 86.89%
```



Confusion Matrix - Epoch 21

Actual	Bengali	Gujarati	Hindi	Kannada	Malayalam	Marathi	Punjabi	Tamil	Telugu	Urdu
Bengali	5039	18	89	8	14	28	17	2	51	65
Gujarati	8	5203	45	8	1	3	32	3	8	8
Hindi	19	7	4969	2	13	6	15	3	17	10
Kannada	13	9	47	4246	21	12	17	8	19	28
Malayalam	25	7	25	53	4641	14	15	11	27	28
Marathi	27	13	31	9	2	4920	12	6	12	40
Punjabi	5	5181	25	2	0	2	28	2	1	4
Tamil	2	12	29	34	14	15	8	4746	4	33
Telugu	32	2	19	16	11	24	6	1	4672	17
Urdu	33	4	52	23	5	39	15	17	19	6161



Acknowledgements

- HuggingFace Transformers for providing WavLM implementation
- PEFT for LoRA implementation
- VoxCeleb datasets
- Audio mixing
- SepFormer