Speech Understanding
Programming Assignment - 2

Question 2

Prateek (M24CSA022)

GitHub Link

1 Introduction

Speech recognition and language identification are critical applications in Natural Language Processing. Speech processing techniques enable language recognition by analyzing acoustic patterns in audio signals. In this assignment, we extract MFCC features from Indian language speech samples and use them for classification. The objectives of this study are:

I. Extract MFCC features from audio samples.

II. Visualize and compare MFCC spectrograms for different languages.

III. Perform statistical analysis on MFCC features.

IV. Train a neural network classifier for language prediction.

2 Dataset Description

The dataset, sourced from Kaggle [1], contains audio samples from 10 Indian languages: Marathi, Gujarati, Urdu, Malayalam, Punjabi, Kannada, Bengali, Hindi, Tamil, and Telugu. Each file is in `.mp3` format, representing spoken phrases in their respective languages.

3 MFCC Feature Extraction and Analysis

3.1 MFCC Extraction Methodology

MFCCs are computed using `torchaudio` in Python. The steps include:

I. Audio is loaded and converted to mono.

II. The MFCC transformation extracts 13 coefficients per frame.

3.2 MFCC Spectrograms

Figure 1 illustrates MFCC spectrograms for three languages: Hindi, Bengali, and Tamil.

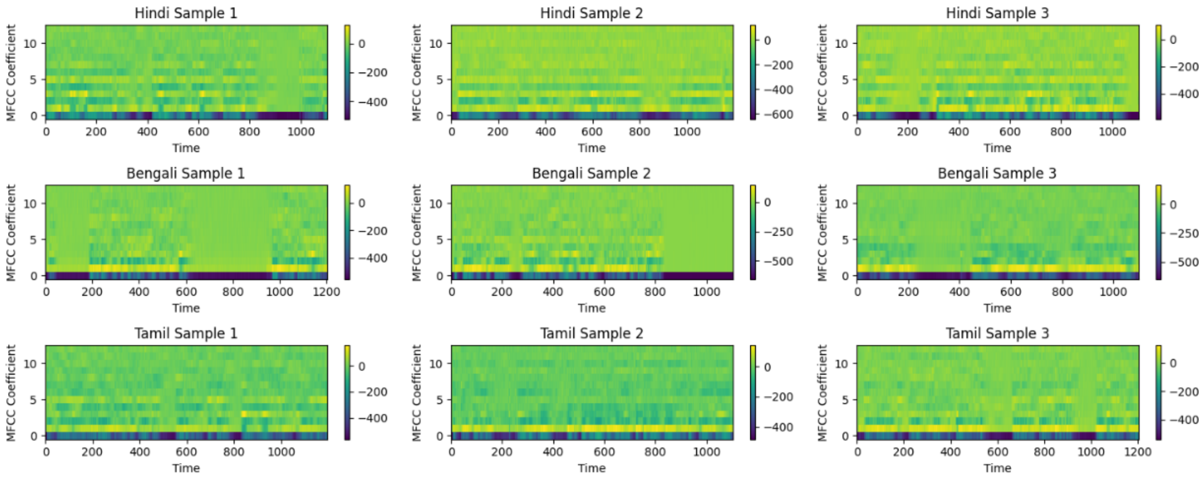


Figure 1: MFCC spectrograms for Hindi, Bengali, and Tamil samples.

3.3 Statistical Analysis of MFCC Features

We compute the mean and variance of MFCC coefficients for three languages:

Table 1: MFCC Statistics for Hindi, Bengali, and Tamil

Language	**Mean of MFCC**	**Variance of MFCC**
Hindi	[-356.65, 22.44, -53.65, ..., 1.99]	[13439.52, 1189.40, ..., 97.71]
Bengali	[-461.91, 59.69, -36.83, ..., -1.09]	[13219.81, 1898.08, ..., 97.25]
Tamil	[-322.66, 66.64, -47.60, ..., 0.41]	[7832.18, 1044.34, ..., 126.11]

4 Language Classification Using MFCC Features

4.1 Preprocessing

We extract mean MFCC values for each audio file and apply:

I. Standardization using `StandardScaler`.

II. Label encoding for categorical language labels.

III. 80% training and 20% test split.

4.2 Neural Network Model

A feedforward neural network is implemented with:

I. **Input:** 13 MFCC features.

II. **Hidden Layers:** Two layers (64 and 32 neurons).

III. **Output:** 10 language classes.

4.3 Training and Evaluation

The model is trained for 20 epochs using Adam optimizer and CrossEntropy loss. It achieves a test accuracy of **87.68%**.

5 Results and Discussion

5.1 Classification Performance

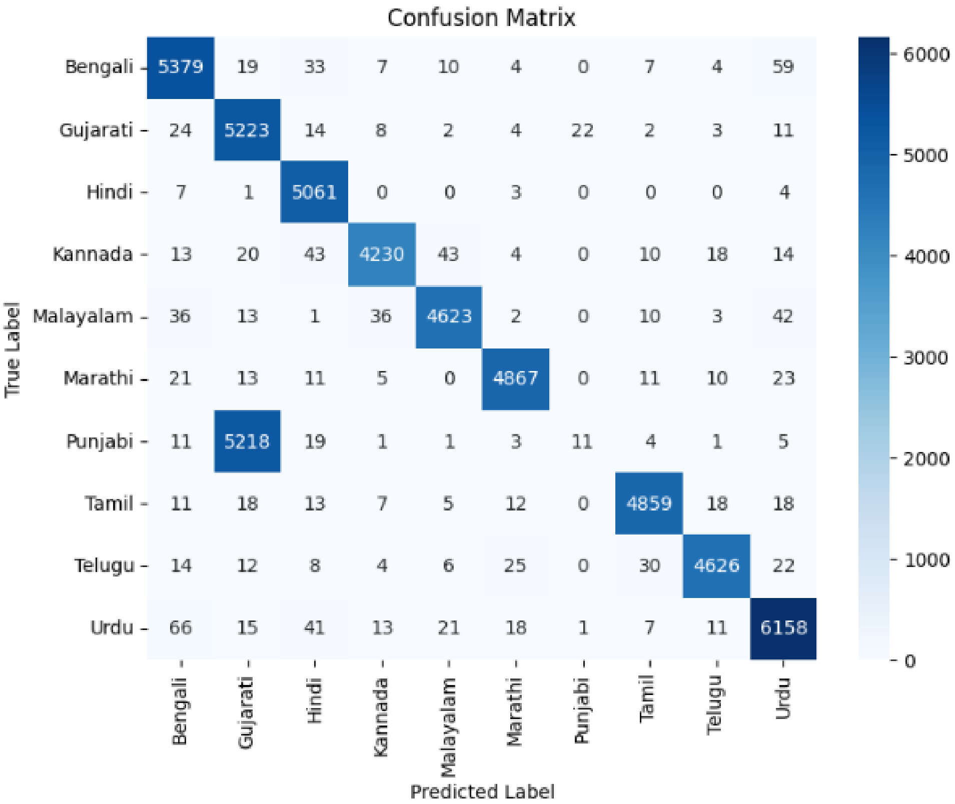
Table 2 presents the classification report.

Table 2: Classification Report

Language	Precision	Recall	F1-Score	Support
Bengali	0.96	0.97	0.97	5522
Gujarati	0.49	0.98	0.66	5313
Hindi	0.97	1.00	0.98	5076
Tamil	0.98	0.98	0.98	4961
Telugu	0.99	0.97	0.98	4747

5.2 Confusion Matrix

Figure 2 presents the confusion matrix, showing high misclassifications in Punjabi.



	Bengali	Gujarati	Hindi	Kannada	Malayalam	Marathi	Punjabi	Tamil	Telugu	Urdu
Bengali	5379	19	33	7	10	4	0	7	4	59
Gujarati	24	5223	14	8	2	4	22	2	3	11
Hindi	7	1	5061	0	0	3	0	0	0	4
Kannada	13	20	43	4230	43	4	0	10	18	14
Malayalam	36	13	1	36	4623	2	0	10	3	42
Marathi	21	13	11	5	0	4867	0	11	10	23
Punjabi	11	5218	19	1	1	3	11	4	1	5
Tamil	11	18	13	7	5	12	0	4859	18	18
Telugu	14	12	8	4	6	25	0	30	4626	22
Urdu	66	15	41	13	21	18	1	7	11	6158

Figure 2: Confusion Matrix of Language Classification.

5.3 Challenges

Challenges include:

I. Speaker variability affecting classification.

II. Background noise distorting MFCC features.

III. Accent and dialect differences.

6 Conclusion

This report demonstrates the effectiveness of MFCC features for Indian language classification. The neural network model achieved an accuracy of 87.68%, with some challenges in differentiating similar languages. Future improvements can focus on deeper learning architectures for better performance.

References

[1] Chaitanya Bharadwaj, “Audio Dataset with 10 Indian Languages,” Kaggle. Available: https://www.kaggle.com/datasets/hbchaitanyabharadwaj/audio-dataset-with-10-indian-languages

[2] Torchaudio Documentation. Available: https://pytorch.org/audio/stable/index.html

[3] Scikit-learn Documentation. Available: https://scikit-learn.org/stable/