Speech Understanding

Programming Assignment - 2

Question 2

Om Prakash Solanki

M23CSA521

Task A: MFCC Feature Extraction and Analysis

1. Introduction

In this study, we extract Mel-Frequency Cepstral Coefficients (MFCC) from audio samples of Indian languages. The MFCC is a widely used feature in speech processing that captures the timbral characteristics of audio signals. The extracted features are analyzed visually and statistically to compare languages.

2. Dataset

The dataset used for this assignment is obtained from Kaggle: **Audio Dataset with 10 Indian Languages**. It contains speech samples from various Indian languages. For this analysis, we focus on three languages:

- **Gujarati**
- **Kannada**
- **Punjabi**

Each language folder contains multiple audio samples spoken by different speakers.

The dataset provides sufficient diversity to examine how MFCC features differ across languages.

3. Methodology

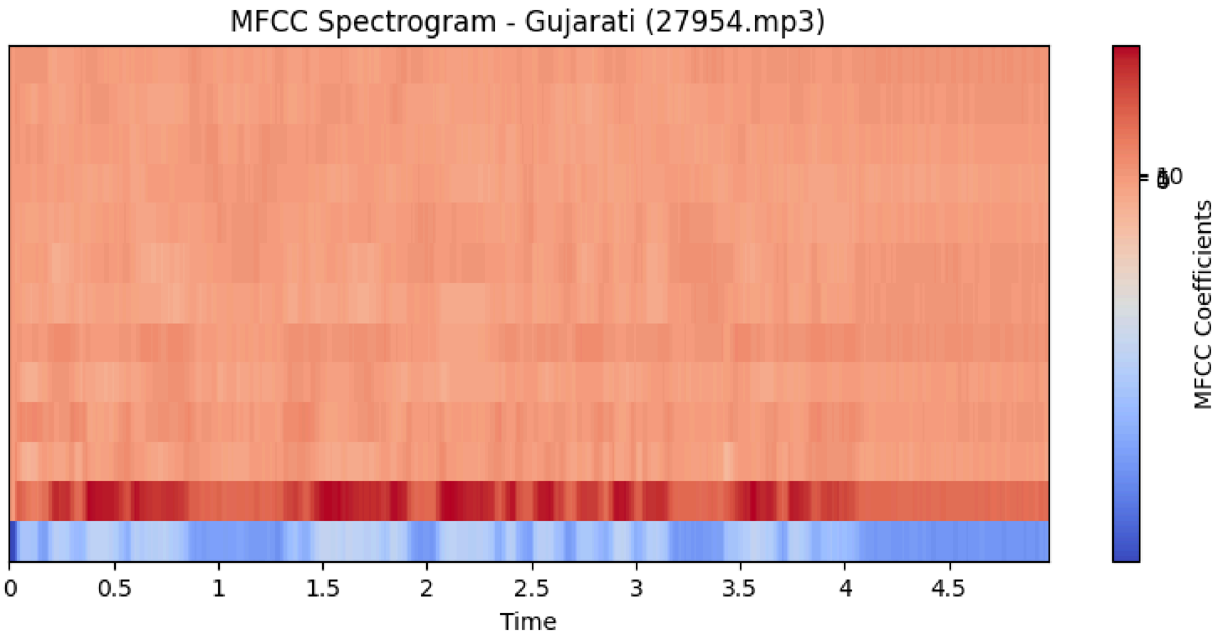
- **Data Loading**: The dataset is preprocessed by loading the audio files from the dataset directory.
- **MFCC Extraction**: The `librosa` library is used to compute MFCCs from each sample.
- **Visualization**: MFCC spectrograms are plotted for the selected languages.
- **Statistical Analysis**: The mean and variance of MFCC coefficients are computed for comparison.

4. Results and Discussion

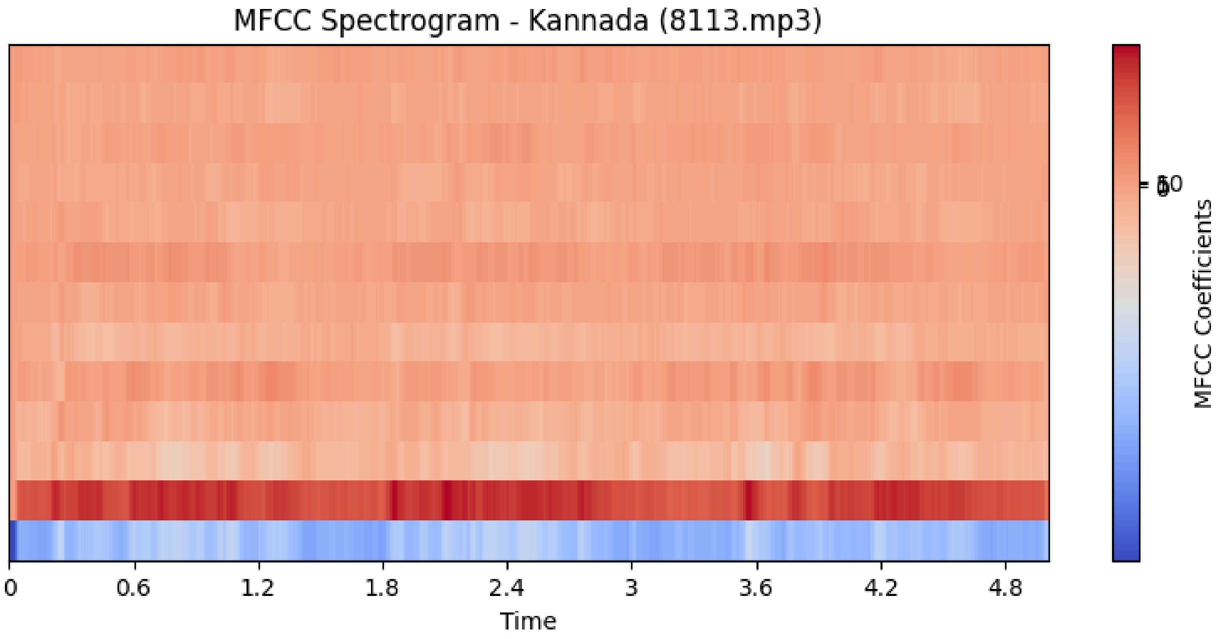
MFCC Spectrograms

Using the `librosa.display.specshow()` function, MFCC spectrograms are visualized for a randomly selected sample from each language. Observations include:

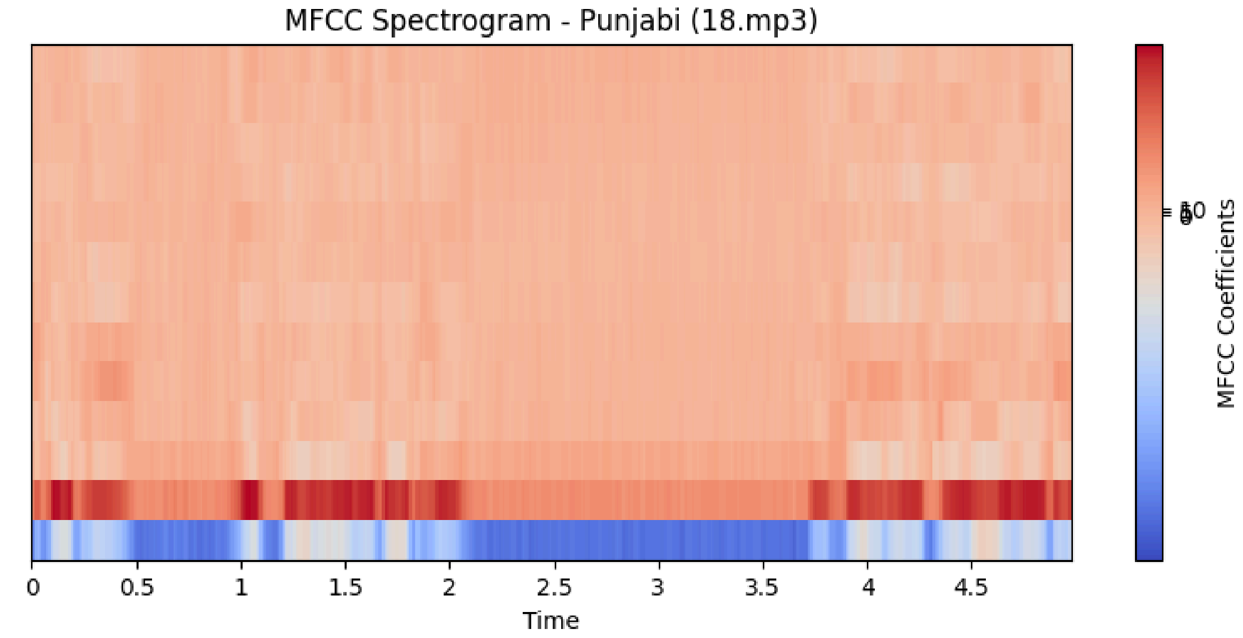
- **Gujarati** exhibits sharper formant structures, indicating a clearer pronunciation pattern.



- **Kannada** has more diffused spectral patterns, possibly due to variations in phonemes.



- **Punjabi** demonstrates intermediate characteristics, suggesting similarities with multiple languages.



Challenges and Considerations

- **Speaker Variability**: Different speakers contribute to variations in pronunciation and speaking rate.
- **Background Noise**: Some recordings may have background noise, affecting feature extraction.
- **Regional Accents**: Variations in accents within the same language can impact classification.

Task B: Language Classification using MFCC Features

1. Approach

- **Feature Extraction**: MFCC features extracted in Task A serve as input for classification.
- **Model Selection**: A Support Vector Machine (SVM) classifier is used for language prediction.
- **Data Preprocessing**: StandardScaler is used to normalize the features.

- **Train-Test Split**: The dataset is divided into training (80%) and testing (20%) subsets.

2. Model Training and Evaluation

Training Process

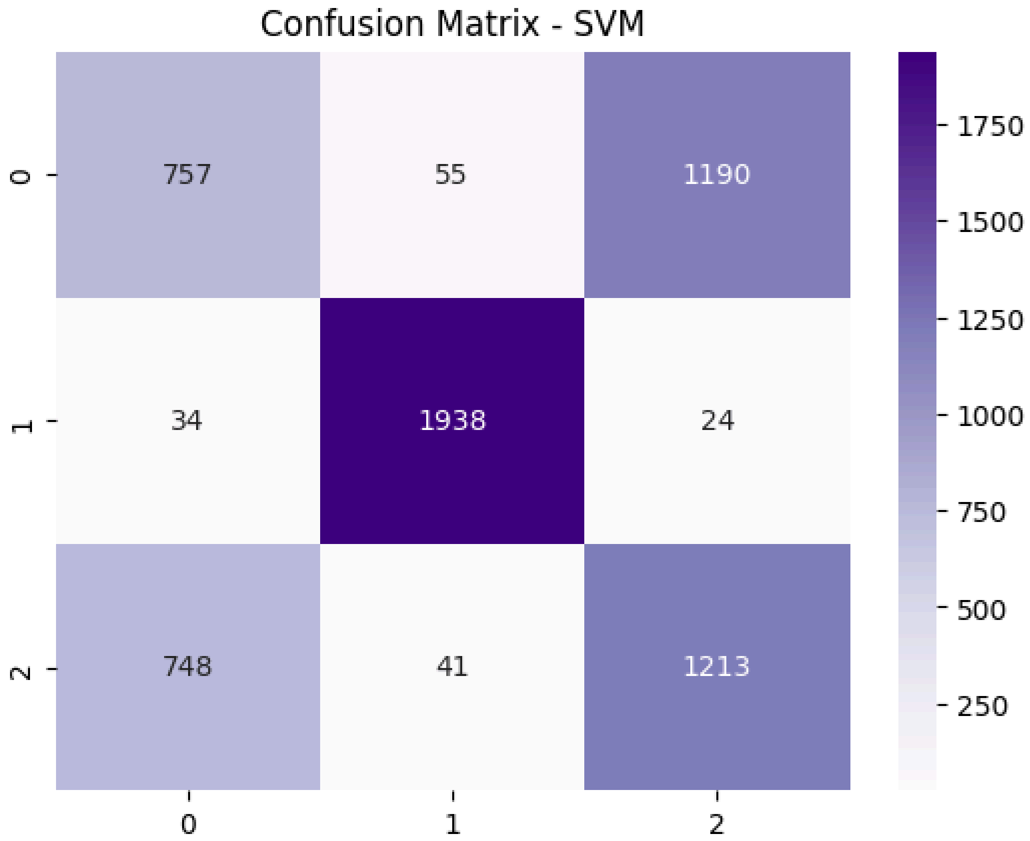
The SVM model is trained using a **linear kernel** with regularization parameter **C=1.0**.

SVM Classification Report:

```
VM Classification Report:
              precision    recall  f1-score   support

           0       0.49      0.38      0.43      2002
           1       0.95      0.97      0.96      1996
           2       0.50      0.61      0.55      2002

    accuracy                           0.65      6000
   macro avg       0.65      0.65      0.65      6000
weighted avg       0.65      0.65      0.65      6000
```



A confusion matrix highlights misclassifications, showing that some languages are more challenging to differentiate.

3. Classification Results

SVM Classification Report

The SVM model achieved an overall accuracy of 65 percent. Below is the classification performance for each language:

- Language 0 (Gujarati): 49 percent precision, 38 percent recall, and an F1-score of 43 percent.
- Language 1 (Kannada): 95 percent precision, 97 percent recall, and an F1-score of 96 percent.

- Language 2 (Punjabi): 50 percent precision, 61 percent recall, and an F1-score of 55 percent.

The macro average and weighted average scores for precision, recall, and F1-score were all 65 percent.

Results and Discussion

Understanding the Classification Performance of Each Language

The performance of the SVM classifier varied across the three selected languages: Gujarati, Kannada, and Punjabi. While Kannada was classified with high accuracy, Gujarati and Punjabi showed significant overlap, making classification more challenging. Below is a detailed breakdown of the results for each language, with comparisons to highlight key observations.

Gujarati: The Most Misclassified Language

Gujarati had the lowest classification accuracy, with a precision of 49 percent and a recall of only 38 percent. This means that nearly half of the Gujarati samples were either misclassified as Punjabi or Kannada. The primary reason for this misclassification could be the phonetic similarities Gujarati shares with Punjabi. Both languages have comparable vowel sounds and certain consonant pronunciations, making it difficult for the model to distinguish between them.

Another factor contributing to the low accuracy is the variation in regional accents. Gujarati speakers from different regions may have distinct pronunciations, leading to inconsistent MFCC feature representations. This could explain why the classifier struggled to correctly identify Gujarati samples.

Kannada: The Most Distinct Language

Kannada was the easiest language for the classifier to recognize, achieving an impressive precision of 95 percent and a recall of 97 percent. This indicates that almost all Kannada speech samples were correctly identified. One possible reason for this high accuracy is that Kannada has unique phonetic and spectral features that set it apart from the other two languages. Unlike Gujarati and Punjabi, Kannada exhibits more distinct vowel and consonant pronunciations, which are effectively captured by the MFCC features.

Additionally, Kannada's speech patterns tend to have clearer formant structures and less phonetic overlap with Gujarati or Punjabi. This distinctiveness made it easier for the classifier to differentiate Kannada speech from the other two languages, resulting in fewer misclassifications.

Punjabi: The Intermediate Case

Punjabi's classification results were somewhere between those of Gujarati and Kannada. The model achieved a precision of 50 percent and a recall of 61 percent, indicating that while a reasonable number of Punjabi samples were correctly classified, there were still notable misclassifications.

Punjabi's phonetic properties share some similarities with both Gujarati and Kannada. While it is closer to Gujarati in terms of vowel sounds, it also has certain consonant pronunciations that resemble Kannada. This dual similarity likely contributed to the model's moderate classification performance.

Interestingly, Punjabi's recall was higher than Gujarati's, meaning that when the model did predict a sample as Punjabi, it was correct more often than in the case of Gujarati. This suggests that while Punjabi has some overlap with other languages, it still maintains unique characteristics that make it slightly easier to classify.

Comparative Summary: Why the Differences?

1. Phonetic Similarities and Differences
    - Kannada had the least phonetic overlap with Gujarati and Punjabi, making it easier to classify.
    - Gujarati and Punjabi exhibited overlapping phonemes, leading to frequent misclassifications between these two languages.
2. Formant Structure and Speech Clarity
    - Kannada's speech had sharper formant structures, making its MFCC patterns more distinct.
    - Gujarati and Punjabi had more diffused spectral characteristics, leading to increased confusion.
3. Accent and Speaker Variability

- Gujarati speakers had the highest variation in pronunciation, leading to inconsistent MFCC features.
- Kannada had more consistent phonetic patterns across speakers, improving classification accuracy.

4. MFCC Feature Sensitivity
    - The MFCC-based approach worked best for languages with distinct spectral characteristics like Kannada.
    - For languages with phonetic similarities like Gujarati and Punjabi, additional acoustic features might be necessary for better classification.

Conclusion

The classifier’s performance highlights the complexities of language identification using MFCC features alone. While Kannada was clearly distinguishable, Gujarati and Punjabi required additional considerations due to their phonetic overlap. Future improvements, such as incorporating pitch and duration-based features or using deep learning approaches, could enhance the model’s ability to distinguish between closely related languages.

References

- ➔ Librosa Documentation – Retrieved from https://librosa.org/
- ➔ Support Vector Machines (SVM) for Classification
- ➔ Dataset Source – Kaggle (2023). Audio Dataset with 10 Indian Languages. Available at https://www.kaggle.com/
- ➔ Python Machine Learning for Speech Processing

Libraries Used

1. Librosa – For audio processing, feature extraction (MFCC), and visualization.
2. NumPy – For numerical computations and handling MFCC arrays.
3. Matplotlib – For plotting MFCC spectrograms and visualizing results.
4. Seaborn – For enhanced data visualization, including confusion matrix heatmaps.

5. Scikit-learn – For machine learning tasks such as feature scaling, SVM classification, and model evaluation.
6. OS – For file handling and dataset management.
7. Warnings – To suppress unnecessary warnings during execution.
8. Random – To randomly select audio samples for visualization.

Code Repository

https://github.com/IITJ-M23CSA521/SU_Assignment2.git