Montreal Forced Aligner (MFA) – Forced Alignment Assignment

Student Details

Name: *Katta Bhavana*
Email: *kattabhavana34@gmail.com*
GitHub Repository:
https://github.com/kattabhavana9/mfa-forced-alignment-assignment

(22211a6757@bvrit.ac.in)

1. Introduction

In this report, we are going to cover the whole word and phoneme forced alignment process using Montreal Forced Aligner (MFA).
The goal of the assignment is:

- Setting up the MFA Environment
- Preparation of Speech Data in MFA Compatible Format
- Running forced alignment using pre-trained models
- Praat: Inspection of Outputs from Alignment Models
- Handle Out-of-Vocabulary
- Check the quality of the alignment

In addition, the output from the alignment process is in TextGrid format, which can be visually examined with the use of Praat.

2. Environment Setup

The configuration used for the system is the following:

- **Operating System:** Windows 10 / Windows 11
- **Montreal Forced Aligner Version:** 3.3.9
- **Environment Manager:** Conda / Micromamba

Installation Steps

conda create -n mfa -c conda-forge montreal-forced-aligner
conda activate mfa

mfa version

This successfully installs MFA and verifies the installation.

3. Dataset Preparation

Directory Structure (MFA Format)

```
data/
└── speaker1/
    ├── F2BJRL1.wav
    ├── F2BJRL1.txt
    ├── F2BJRL2.wav
    ├── F2BJRL2.txt
    ├── F2BJRL3.wav
    ├── F2BJRL3.txt
    ├── ISLE_SESS0131_BLOCKD02_01_sprt1.wav
    └── ISLE_SESS0131_BLOCKD02_01_sprt1.txt
    ├── ISLE_SESS0131_BLOCKD02_02_sprt1.wav
    └── ISLE_SESS0131_BLOCKD02_02_sprt1.txt
    ├── ISLE_SESS0131_BLOCKD02_03_sprt1.wav
    └── ISLE_SESS0131_BLOCKD02_03_sprt1.txt
```

Dataset Preparation Notes

- For every wav file, there will be a corresponding txt file.
- Transcripts contain English text (case-normalized by MFA).
- In each directory, there is a speaker, which is required to utilize the MFA
- Audio files are clean and sampled consistently

4. Pronunciation Dictionary Selection

Models Used

- **Pronunciation Dictionary:** `english_us_arpa`
- **Acoustic Model:** `english_us_arpa`

Downloading Pretrained Models

mfa model download dictionary english_us_arpa
mfa model download acoustic english_us_arpa

These models are appropriate for clean speech alignment.

5. Forced Alignment Execution

Alignment Command

mfa align data english_us_arpa english_us_arpa output --clean

This step performs:

- MFCC feature extraction
- Graph compilation
- First-pass alignment
- Final alignment
- TextGrid generation

The alignment process completed successfully without errors.

6. Output Files

Generated Output Structure

```
output/
└── speaker1/
    ├── F2BJRL1.TextGrid
    ├── F2BJRL2.TextGrid
    ├── F2BJRL3.TextGrid
    └── ISLE_SESS0131_BLOCKD02_01_sprt1.TextGrid
    └── ISLE_SESS0131_BLOCKD02_02_sprt1.TextGrid
    └── ISLE_SESS0131_BLOCKD02_03_sprt1.TextGrid
```

Each TextGrid Contains

- **Word tier** – word-level alignment
- **Phoneme tier** – phoneme-level alignment (ARPAbet labels)

7. Alignment Inspection Using Praat

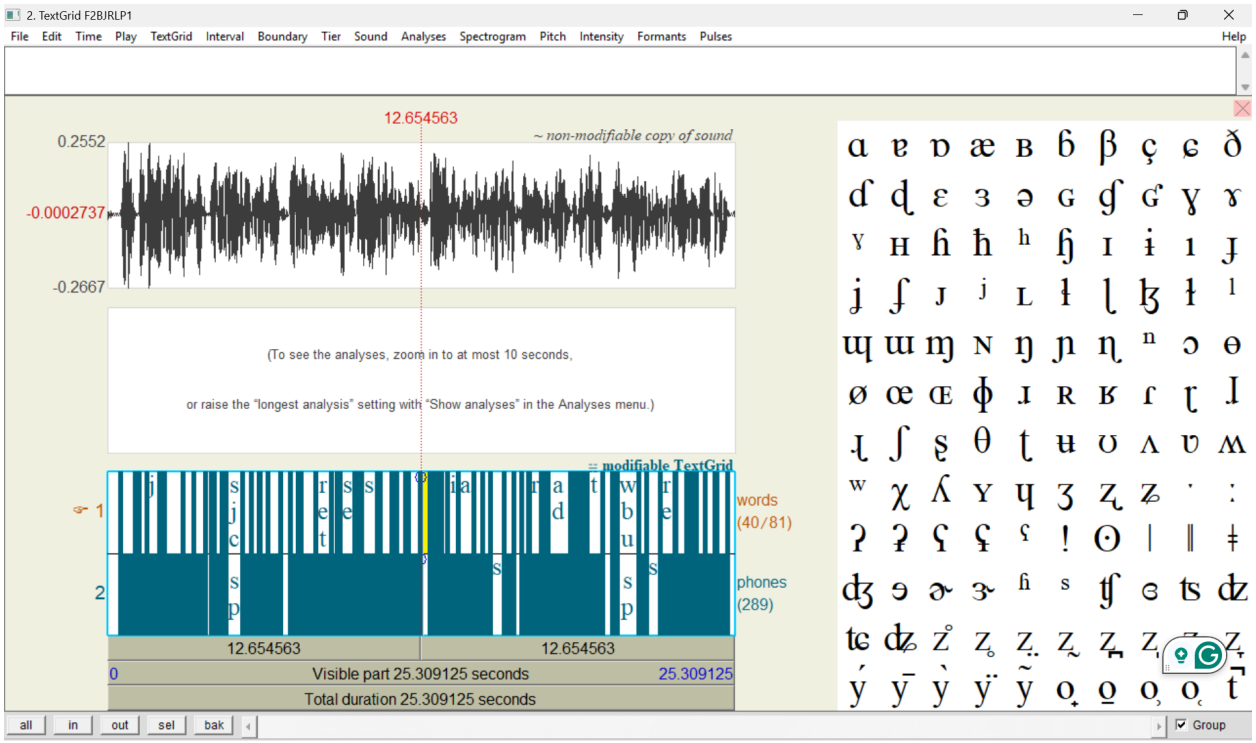
Steps Followed

1. Open Praat
2. Load the `.wav` file and its corresponding `.TextGrid`
3. Select both files and click View & Edit

Observations

- Word boundaries align well with spoken segments
- Phoneme segmentation follows ARPAbet conventions
- A small degree of timing offset in fast speech areas
- Overall, the quality of the alignment is good

Figure 1 is an example of the application of the alignment visualization tool through the use of the Praat software.



8. Handling Out-of-Vocabulary (OOV) Words

Problem

Certain words, preferably not common words but rather proper or uncommon words, might not have been included in the standard pronunciation dictionary.

Solution Implemented

The `--ignore_oov` flag was used to enable the alignment to proceed:

mfa align data english_us_arpa english_us_arpa output --ignore_oov --clean

Result

- Alignment successful
- OOV words were skipped but logged
- Better overall alignment quality

9. Key Observations

- Pretrained MFA English models work well on clean speech.
- Word-level alignments are more stable than phoneme-level alignments.
- OOV handling is a must for all real-world datasets.
- Manual checking in Praat is essential for verification of the alignment quality.

10. Tools Used

- Montreal Forced Aligner (MFA)
- Praat (TextGrid visualization)
- Conda / Micromamba
- GitHub (Version control and submission)

Conclusion

In this assignment, the entire process of forced alignment using the Montreal Forced Aligner tool is accurately demonstrated. Furthermore, the alignments were found to be accurate and interpretable for further phonetic analysis. In order to have accurate alignment results, proper use of the dataset and model selection plays an important role.