Speech Recognition Spring 2025 Project 2

Due date: Jan 20th 2026 12pm Beijing time

Note: For this assignment, you will need to build HMM-based isolated word recognition systems. Specifically, you must train separate models for each digit from 0 to 9. All feature vectors should be 39-dimensional, consisting of cepstral coefficients, delta cepstra, and double-delta cepstra, obtained using the code you wrote for Assignment 1. Mean subtraction and variance normalization should be applied to these features. However, if you encounter difficulties with Assignment 1, you are permitted to use the following code to extract audio features instead.

```
import librosa
from python_speech_features import mfcc, delta

audio, sr = librosa.load(audio_path, sr=16000)
mfcc_features = mfcc(
        audio,
        samplerate=sr,
        winlen=0.025,
        winstep=0.01,
        numcep=13,
        nfilt=40,
        nfft=512,
        preemph=0.97,
    )

delta_features = delta(mfcc_features, 2)
double_delta_features = delta(delta_features, 2)

features = np.hstack([mfcc_features, delta_features, double_delta_features])
features = self._normalize_features(features)
```

We have provided the TISIGITS_LDC93S10 dataset on the DKUCC system along with prepared training and testing lists. The dataset is stored in the directory: /dkucc/group/courses/compsci304-2526-s3/cl688/TISIGITS_LDC93S10. You can use the audio samples of the ten digits (0–9) from this dataset to train and test your speech recognition model. We also provided the extracted training and testing files in the attachment on canvas for your convenience.

TASK 1

Use the **segmental K-means procedure** to train an HMM for each of the digits (0–9) using those **20 training recordings per digit in the training set**.

- Each HMM must have **5 states**.
- Each state must be modeled using a **single Gaussian distribution**.
- Use hard alignment (Viterbi-based segmental K-means).

Test the trained models on the test set and report the **recognition accuracy**.

TASK 2

Repeat the segmental K-means training procedure but now model each HMM state using a **mixture of 4 Gaussians**.

- Each HMM still has **5 states**.
- Hard alignment (timewise) is acceptable.
- K-means clustering may be used to replace full GMM training.

Test the trained models on the test set and report the **recognition accuracy**.

Please be careful that the model you implement in task 2 is a simplified HMM model. Although the timewise alignment is not soft, and the component-wise alignment in GMM is not soft, it is relatively simple and can achieve reasonable good performance.