E9 261 - Speech Information Processing

Homework # 4

Due Date: April 18, 2022

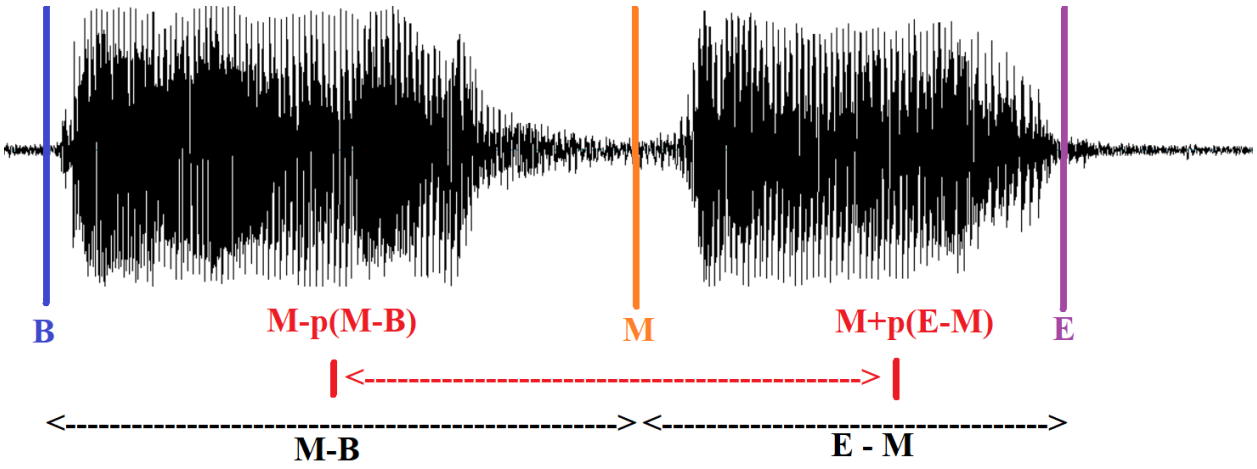
Please upload (in the course webpage) your codes as a zipped folder FirstName LastName HW4.zip (or in multiple zip files with filenames have part1 part2 etc. each not exceeding 100Mb). In the zipped folder the program names should be self explanatory and accompanied with a README file, as required. Filename of each program should contain the question number it is associated with.

In this homework we will carry out CT2 and CT3 from Midterm2 of Speech Information Processing course.

- Classification Task 2 (CT2): Three-class classification where C1: /f/, C2: /s/, and C3: /ch/
- Classification Task 3 (CT3): Three-class classification where C1: /v/, C2: /z/, and C3: /j/

This time, you will use the recordings from all students in the class which can downloaded from:
https://drive.google.com/drive/folders/1_s8F-0haQ9FSjE1B4rRHfvJtsg3wnKRt?usp=sharing
Also, unlike KNN with DTW distance measure in Midterm2, you will use discrete and continuous observation density hidden Markov model (HMM) [1] as well as time delay neural network (TDNN) [2] as discussed in the class.
For both the classification tasks, first THREE (*_1.wav, *_2.wav and *_3.wav) of the five repetitions from all students should be used in training and remaining TWO (*_4.wav and *_5.wav) for every vowel and consonant combination should be used for testing.

You need to experiment with various segment lengths from each VCV recording centered around the middle of the consonant as illustrated in the figure below. A segment (as shown by the red colored part in the figure below) starts from M-p(M-B) and ends at M+p(E-M). Thus, the segment length is parameterized by p (0<p<1). When p=1, the entire VCV from B to E is used. You need to experiment with four values of p, namely, 0.25, 0.50, 0.75 and 1.0.



For the classification task, you need to consider a 39-dim MFCC with delta and delta delta coefficients computed with a window size of 20msec with a 10msec shift. Thus, each VCV recording will be represented as a sequence of 39-dim features, i.e., a feature matrix, the length of which varies from one recording to another and also depending on the choice of p.

A) Perform K-means clustering to quantize the MFCC features from all training VCV recordings using the centers of K clusters. Vary K=16, 32, 64, 128. Use the quantized features as the observations and build a discrete observation density HMM separately for each class. The classification during the test phase is done using a maximum likelihood criterion. Vary the number of states N in HMM as 3, 5, 7. Report the unweighted average recall for both CT2 and CT3 for different values of K and N. For the best case, report the confusion matrices as well. Summarize your observations.

B) Repeat part A) using original MFCC features and continuous density HMM where the observation density is modeled using Gaussian mixture model (GMM). Use a diagonal covariance matrix in GMM and vary the number of mixtures (M) as 4, 8, 16, 32. Vary the number of states N in HMM as 3, 5, 7. Report the unweighted average recall for both CT2 and CT3 for different values of M and N. For the best case, report the confusion matrices as well. Summarize your observations.

C) Use TDNN [2] to carry out the classification tasks, CT2 and CT3. You can linearly interpolate the feature matrices to be of the same length, if required. Report the unweighted average recall for both CT2 and CT3

together with their confusion matrices. Vary the hyperparameters and report the results of your experiments. Alternatively, you can experiment with any other neural network architecture to improve the classification accuracy. Describe neural network you may use in detail.

D) Compare performance in part A), B), C) together with the KNN classifier with DTW distance measure from Midterm2. Summarize your observations.

E) Is there any other way you propose to further improve the classification accuracy for both CT2 and CT3?

For implementation of HMM and Neural Networks, you can choose suitable toolboxes (e.g. Kevin Murphy's toolbox [3] for HMM).

References:

[1] Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257-286.

[2] Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., & Lang, K. J. (1989). Phoneme recognition using time-delay neural networks. *IEEE transactions on acoustics, speech, and signal processing*, *37*(3), 328-339.

[3] https://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html