

Homework 2 - Speech Classification

Due: 2359 Hrs on 20 March 2020

Instructions:

- The instruction plagiarism policy applies.
 - You have to implement the assignment in Python.
 - You can use libraries for reading and perform image operations. However, the core of the retrieval algorithm has to be implemented from scratch. In case of a doubt make sure you confirm with the TA's.
 - For any doubts make a comment on google classroom or email TA with Subject like [MCA HW-2] Doubt.
 - Submission Instructions:
 - All submissions must be inside a zip file named `a2_{name}_{rollnumber}.zip` containing `report.pdf` and a folder named `src` with all your scripts.
 - All the scripts must be uploaded in `.py` format. Make a separate script for each subquestion. Naming convention example `question1_1.py` for the first subquestion of question 1.
 - Write all the scores and your analysis of it in the `report.pdf`
-

Dataset description

This assignment is going to give you a birds-eye view of the methods used in Speech and Signal Processing.

The given dataset contains 10 classes of spoken commands, collected from users with various accents, with noise files given in the folder names '`_background_noise_`' and validation data given in the '`validation`' folder. Each class will have 1000 training files, and the folders of each class are present in the '`training`' folder. We have a held-out test dataset, which will have files from speakers not in the training or the validation set, on which we will calculate your F1 score. Handle variable dimensions.

Question 1 - 50 Points

We will be using spectrograms based features for further exercises. As a first step, implement the spectrogram computation and plotting method from scratch using Python. Make it a generic method so that later it can be computed for various sampling rates and window lengths.

Question 2 - 50 Points

MFCC Feature sets are also very widely used for speech recognition tasks. Implement the MFCC features computation method from scratch. For this part, you are allowed to use FFT utilities from Numpy and DCT utility from Scipy.

Question 3 - 100 Points

Now, use both of the feature sets computed in the above parts individually with SVM to create a classifier for a given spoken command. Use noise files provided to perform random noise augmentation on audio files, and report the changes in Precision and Recall. Compare the performance of SVMs trained on spectrograms and MFCCs. Also, perform analysis of predictions done by both models.