Practical Exercise - Task 1
MFCCs
Speech Technology - COM4511/6511

10th February 2020

1 Introduction

The aim of this task is to learn how to use Python for speech processing. There are several online Python tutorials, and particularly on numpy and how to use it for signal processing. In this task you will implement a Python program to convert a speech signal to Mel Frequency Cepstral Coefficients (MFCCs).

2 Infrastructure

A speech audio file from the TIMIT corpus, a Python helper script and example output are provided. These can be found on MOLE in the folder named `task1`. It is also available for download at https://www.dcs.shef.ac.uk/∼th/campus_only

3 Task

To help you start the task, the provided Python script contains all the necessary steps and a few utility functions for this task. You are asked to fill in the missing parts indicated by ####.

1. From the link above, download the audio file “SA1.wav” and the Python script “ task1_compute_mfcc.py ”
2. Take a look at the script and find out which parts need to be implemented
3. Convert the complete audio file into a sequence of MFCC vectors using the following baseline configuration.
   (a) A pre-emphasis filter with coefficient 0.97
   (b) 10ms frame step, 25ms frame length (Hamming window)
   (c) 26 Mel-filters
   (d) 12 cepstral coefficients (C1-C12, omitting C0)
   (e) 22-order liftering.
   (f) Cepstral mean and variance normalisation (optional)
   (g) Plot different coefficients and consider the implications of what you see
   (h) Adopt any good coding style with meaningful comments. Use functions when possible.

You can (and should) use NumPy and SciPy for basic numerical computing and the FFT/DCT computation, but not any other packages (i.e. the helper script has provided all the necessary imports). If you are not sure, just raise your hand and ask.

4 Assessment:

In order to report the task as complete 3 elements have to be submitted, in gzup form. The file name should have the following format:

<lastname>_<firstname>_task1.gz

1. The MFCC encoded reference audio file in text format.
   Each line contains one MFCC vector, the values are separated by single space.
2. A compiled Latex document (pdf) that contains 4 plots (matplotlib) of the MFFCs in the following configurations (modification to the baseline) and brief comments on the variation to the baseline configuration as outlined above. Also briefly comment on the reasons for the effects.
   (a) No Hamming window
   (b) 40 MFCCs
   (c) 80 Filterbanks, 40 MFCCs
   (d) No Pre-emphasis.
3. The completed source code for the baseline configuration as outlined in 3.