Final Project

Notes for Automatic Speech Recognition

Chia-Ping Chen

cpchen@cse.nsysu.edu.tw

Department of Computer Science and Engineering

National Sun Yat-Sen University

Kaohsiung, Taiwan ROC

Introduction

The final project asks you to build an ASR system to recognize connected Mandarin digit speech. In doing this project, you will learn how to

- extract speech features
- define your HMMs
- train the HMMs
- decode an unknown speech
- evaluate the performance of the system

HTK Software

The software toolkit is the HMM Toolkit (HTK). You should

- download HTK from http://htk.eng.cam.ac.uk/ and install it.
- have the documentation HTKBook available
- join the users group, *htk-users@eng.cam.ac.uk* to share your knowledge with the world

HTK is written in the “c” language. It is open-source, so you can modify the source code as you want. It is also straightforward to debug when something is not right.

Tools

Here we list some of the tools in HTK.

- HCopy: extracting speech features from speech
- HCompV: computing a global mean and variance
- HHEd: editing hmm definitions, such as when doing state-tying and increasing number of mixture components per state
- HLEd: editing hmm label files, such as changing from word labels to phone labels
- HERest: parameter re-estimating
- HVite: decoding with Viterbi algorithm
- HResults: evaluating recognition results

Data and Files

You will need the following data and files.

- Speech data files
- Label files, specifying the linguistic content of each speech file
- The prototype file, specifying the topology of the hmms
- The file that list hmms
- The dictionary file (lexicon)
- The word net, specifying the syntax or language model

Outline of Experimental Setup

- Extracting speech features
- Preparing the label files
- Parameter initialization
- Parameter re-estimation for several epochs
- Until the desired amount of training is finished
  - increase the number of mixture components
  - parameter re-estimation for several epochs
- Recognition
- Evaluation

Goals

- Use a script to automate the entire procedure
- Experiment with different features or models for comparison
- Get **word error rate**, the lower the better