Introduction to Kaldi
Lab 3

[Objective]

1. Know how to use github to download open source project
2. Install and compile kaldi
3. Learn how to upload files to server

[Getting started]

Kaldi is a toolkit for speech recognition, intended for use by speech recognition researchers and professionals. The first step is to download and install Kaldi. Go to the kaldi source page on github https://github.com/kaldi-asr/kaldi and look for the clone URL which should be

https://github.com/kaldi-asr/kaldi.git

On the linux server, you can directly clone the project to your local folder by running

```
git clone https://github.com/kaldi-asr/kaldi.git
```

Then "cd" to the kaldi folder. Look at the INSTALL file and follow the instructions (it points you to two subdirectories). Look carefully at the output of the installation scripts, as they try to guide you what to do. Some installation errors are non-fatal, and the installation scripts will tell you so (i.e. there are some things it installs which are nice to have but are not really needed). The "best-case" scenario is that you do:

```
cd kaldi/tools/; make; cd ../src; ./configure; make
```

then everything will just work; however, if this does not happen there are fallback plans (e.g. you may have to install some package on your machine, or run install_atlas.sh in tools/, or run some steps in tools/INSTALL manually, or provide options to the configure script in src/). If there are problems, there may be some information in The build process (how Kaldi is compiled) that will help you.

[Overview of the distribution]

Before we jump into the example scripts, let us take a few minutes to look at what else is included in the Kaldi distribution. Go to the kaldi directory and list it. There are a few files and subdirectories. The important subdirectories are "tools/", "src/", and "egs/" which we will look at in the next section. We will give an overview of "tools/" and "src/".

The tools/ directory

The directory "tools/' is where we install things that Kaldi depends on in various ways. Change directory to tools/ and list it. You will see various files and subdirectories, mostly things that have been installed by the make command. Look very quickly at the file INSTALL. This file gives instructions on how to install the tools.

The src/ directory

Change directory back up to the top level (kaldi) and into src/. List the directory. You will see a few files and a large number of subdirectories. Look at the Makefile. At the top it sets the variable SUBDIRS. This is a list of the subdirectories containing code. Notice that some of them end in "bin". These are the ones that contain executables (the code and executables are in the same directory). The other directories contain internal code.

Change directory to ../featbin and type

```
./compute-mfcc-feats
```

It prints out the usage, which should give you a generic idea of how Kaldi programs are called. Note that while there is a – config option that can be used to pass a configuration file.

Then look at featbin/compute-mfcc-feats.cc, try to go through the code and see how it extracts the features from the audio.

[Running an experiment in Kaldi]

Go to the experiment folder of YESNO with the following command:

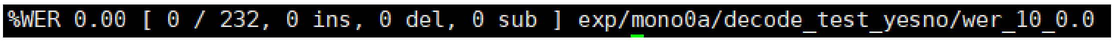
```
cd kaldi/egs/yesno/s5
```

execute the run script with

```
./run.sh
```

Observe whether the experiment can be run smoothly till the evaluation part at the end.

If so, you can see the following output:



```
%WER 0.00 [ 0 / 232, 0 ins, 0 del, 0 sub ] exp/mono0a/decode_test_yesno/wer_10_0.0
```

Please read the script file run.sh. It is actually a collection of script.
For a detailed explanation of each step in the script, please refer to
https://github.com/keighrim/kaldi-yesno-tutorial

[Exercise: to evaluate the system with your own voice]
Now, please test the system with your own voice.
Speak any number of "yes" and "no" in a random order.
You can use any device to record your voice, e.g. your mobile or your laptop.
Please use winscp to upload the audio files to the server from your PC. You can download winscp from
https://winscp.net/eng/download.php

After uploading the file, you can either replace the current test files with your own audio files and rerun the decoding or create another test folder of your own. Remember that the original data set is 8k sampling rate .wav file. You should convert your own audio files into the same format.
If your audio files are in .mp3 format, you should use ffmpeg to convert it into .wav. For the sampling rate conversion, you can use sox.

The decoding result of the system should be located in the decoding folder.