Building a Speech Recognizer using Open-source Software

Dan Povey

http://sites.google.com/site/dpovey/TidigitsTutorial.pdf

What we'll do

- I have created a speech recognition setup that recognizes connected digits.
- Your goal is to reduce the error rate as much as possible by tuning and tweaking the setup.
  - I'll suggest ways to do this.
- You will work in groups of 3-4; members of group with lowest error rate at end get $20 each.
- You can tune on the test set; near the end I'll tell you how we'll "really" measure performance.

Logging in

```
$ ssh <your-username>@login.clsp.jhu.edu
Linux login 2.6.26-2-amd64 #1 SMP Mon Jun 13 16:29:33 UTC 2011 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Jun 16 12:29:02 2012 from arnab-laptop.clsp.jhu.edu

login:$ qlogin -q all.q@a*.clsp.jhu.edu
Your job 4156604 ("QLOGIN") has been submitted
waiting for interactive job to be scheduled ...
Your interactive job 4156604 has been successfully scheduled.
Establishing builtin session to host a07.clsp.jhu.edu ...
a07:$
###### note: you can also ssh directly to a{01,02,03,04,05,07,08,09,10}
```

The setup

- The speech-reco system is based on the open-source software "Kaldi" (which uses OpenFst).
- This is designed for large-vocabulary speech recognition, but here we'll use it for a toy task.
- The database is "TIDIGITS"-- very old, very easy task, clean recording, people saying digits (connected digits, i.e. without pauses).
- Train and test sets each have ~8k utterances, from various speakers including children.

Getting Started

```
a07:$ cp -r ~dpovey/tutorial_skeleton .
a07:$ cd tutorial_skeleton/egs/tidigits/s5
a07:$ # look at run.sh
```

- If I had not set it up for you, you'd have to
  - Order the TIDIGITS data from the Linguistic Data Consortium
  - download and compile Kaldi as described at http://kaldi.sf.net
  - cd to <kaldi-root>/egs/tidigits/s5
  - edit run.sh to have correct TIDIGITS path, cmd.sh to have correct queue name.

The results

- TIDIGITS is typically evaluated in terms of sentence error rate.
- The SER at the monophone stage is 3.67%, at triphone is 2.64%.
- The command at the end of the run.sh with "diff" shows you the errors.
- Seems to mostly be dominated by insertions of "o".

Things to try (1)

- Tune the command-line parameters (I never tuned #states, #Gaussians)
- Modify the dictionary (see local/tidigits_prepare_lang.sh)
  - e.g. make "oh" a two-phone word to make it harder to insert it
  - change the silence-insertion probability, currently 0.5 [this is in L.fst]

Things to try (2)

- Modify the language-model G.fst, which is currently a simple phone loop with constant costs.
  - E.g. change the cost of "o" [which is frequently inserted], or use unigram likelihoods estimated from the training data.
  - Create an FST that only allows sequences of 1, 2, 3, 4, 5, or 7 digits (all TIDIGITS sequences are of this form).

Things to try (3)

- Try out more advanced types of model.
  - Look at egs/rm/s5/run.sh for examples.
  - Typical sequence of model-building (for LVCSR, anyway) is:
    - MFCC+delta+accel, monophone
    - MFCC+delta+accel, triphone
    - MFCC+splice+LDA+MLLT, triphone
    - +Speaker Adapted Training
      - +discriminative training (BMMI)

Things to try (4)

- Best results in the RM ("Resource Management" setup are:
  - After LDA+MLLT+SAT stage, build Subspace Gaussian Mixture Model (SGMM), then do discriminative training on this.
- Caution: when numbers appear on the command line in the RM setup (e.g. 2500, 10000, 400), you'll typically want smaller numbers for TIDIGITS
  - These are things like number of clustered states; number of Gaussians in total systems; number of Gaussians in "background model"

Ask!

- A lot of things will be unclear; I will be around so ask me.
- If you can find a Hopkins student or even faculty who is willing to help you, that is allowable too.
  - This makes the competition more like real life, where asking for help is allowed.
- Introduction to Kaldi is available at http://kaldi.sf.net but it's aimed at speech experts, and you won't be able to read it in one afternoon.

The queue

- The scripts are currently configured so they'll run training on the machine you are logged into, and testing using the queue.
  - See "cmd.sh"-- you can comment or un-comment things to change this.
- The training scripts use 4 CPUs, which is really against the rules for our queue (but it's faster).
- If everyone is assigned the same machine it may be a problem. You might have to try to find a freer "a" machine.
- Try to avoid increasing the num-jobs "--nj" option

The end

- [will assign groups at this stage]
- Around 4:30 I'll tell you how we'll "really" measure the error rate, and will ask the groups with reasonable error rates to do this procedure.
  - Depending how long this takes, the best group will be selected, and prize awarded, either around 5:00, or the next day.
- Have fun!