How to train automatic speech recognition in kaldi

First, you need to download kaldi package and make it.

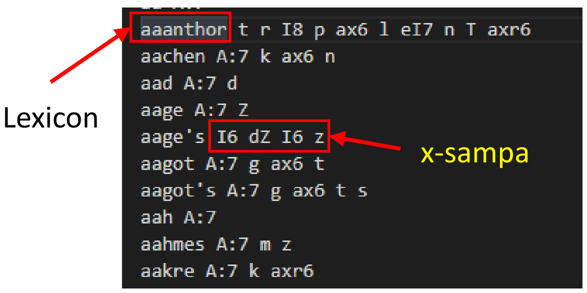
You can follow is website (http://jrmeyer.github.io/asr/2016/01/26/Installing-Kaldi.html) to install it.

Or go to here (https://github.com/yfliao/kaldi/tree/master/egs/formosa) to install it.

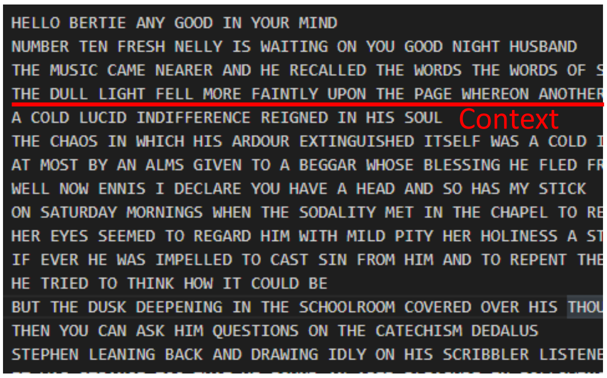
Second, prepare all of your data.

1. Prepare your lexicon by x-sampa and your context by sentence.

Note. The words of the context text file must be included in the lexicon. And lexicon needs to sort it.



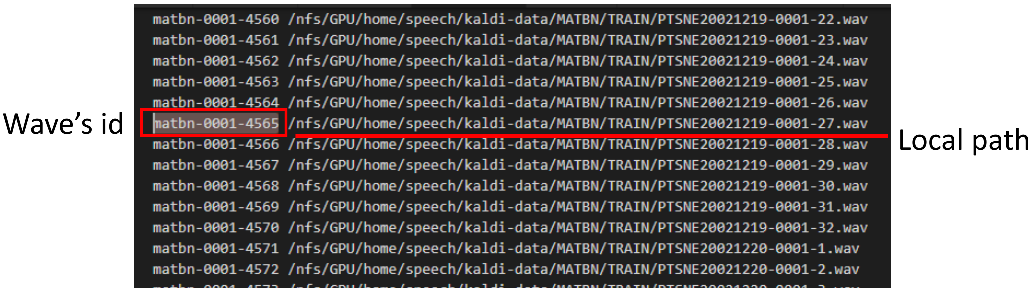
```
aaanthor t r I8 p ax6 l eI7 n T axr6
aachen A:7 k ax6 n
aad A:7 d
aage A:7 Z
aage's I6 dZ I6 z
aagot A:7 g ax6 t
aagot's A:7 g ax6 t s
aah A:7
aahmes A:7 m z
aakre A:7 k axr6
```



```
HELLO BERTIE ANY GOOD IN YOUR MIND
NUMBER TEN FRESH NELLY IS WAITING ON YOU GOOD NIGHT HUSBAND
THE MUSIC CAME NEARER AND HE RECALLED THE WORDS THE WORDS OF S
THE DULL LIGHT FELL MORE FAINTLY UPON THE PAGE WHEREON ANOTHER
A COLD LUCID INDIFFERENCE REIGNED IN HIS SOUL
THE CHAOS IN WHICH HIS ARDOUR EXTINGUISHED ITSELF WAS A COLD I
AT MOST BY AN ALMS GIVEN TO A BEGGAR WHOSE BLESSING HE FLED FR
WELL NOW ENNIS I DECLARE YOU HAVE A HEAD AND SO HAS MY STICK
ON SATURDAY MORNINGS WHEN THE SODALITY MET IN THE CHAPEL TO RE
HER EYES SEEMED TO REGARD HIM WITH MILD PITY HER HOLINESS A ST
IF EVER HE WAS IMPELLED TO CAST SIN FROM HIM AND TO REPENT THE
HE TRIED TO THINK HOW IT COULD BE
BUT THE DUSK DEEPENING IN THE SCHOOLROOM COVERED OVER HIS THOU
THEN YOU CAN ASK HIM QUESTIONS ON THE CATECHISM DEDALUS
STEPHEN LEANING BACK AND DRAWING IDLY ON HIS SCRIBBLER LISTENE
```

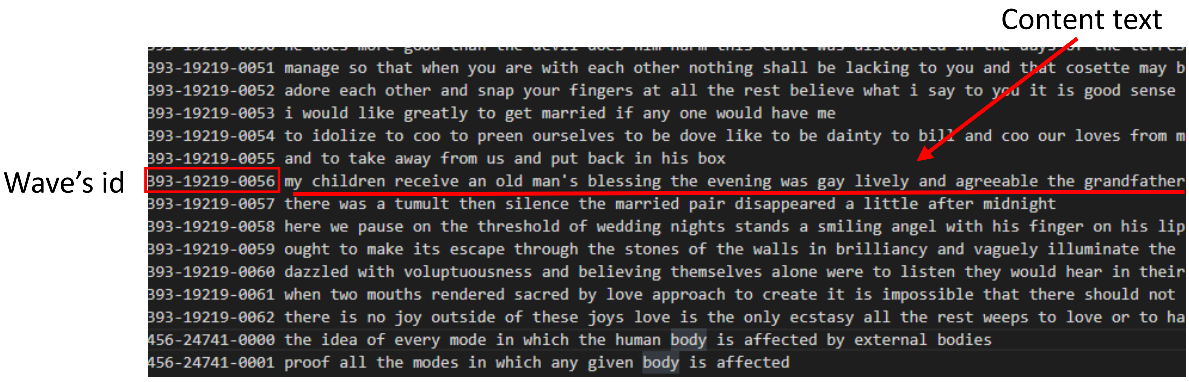
2. Prepare your training audio data and testing audio data.
   A. The audio of data is human voice audio file with
      - Sampling format :
        - mono (1 channel)
        - sample rate 16kHz
        - 16bits PCM.
      - Audio format :
        - Waveform Audio File Format (*.wav)

3. Prepare 3 file is about information of training data. The testing data also has to prepare it.
   A. wav.scp (format : wave's id and wave's local path)



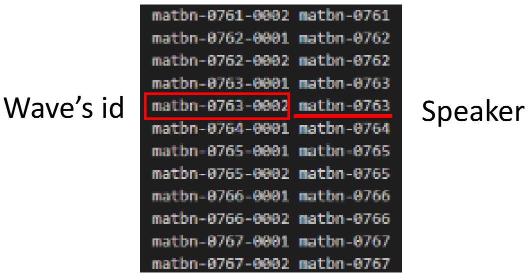
```
matbn-0001-4560 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-22.wav
matbn-0001-4561 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-23.wav
matbn-0001-4562 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-24.wav
matbn-0001-4563 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-25.wav
matbn-0001-4564 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-26.wav
matbn-0001-4565 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-27.wav
matbn-0001-4566 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-28.wav
matbn-0001-4567 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-29.wav
matbn-0001-4568 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-30.wav
matbn-0001-4569 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-31.wav
matbn-0001-4570 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021219-0001-32.wav
matbn-0001-4571 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021220-0001-1.wav
matbn-0001-4572 /nfs/GPU/home/speech/kaldi-data/MATBN/TRAIN/PTSNE20021220-0001-2.wav
```

B. text (format : wave's id and the text of data is content text of the human voice audio.)



```
393-19219-0051 manage so that when you are with each other nothing shall be lacking to you and that cosette may b
393-19219-0052 adore each other and snap your fingers at all the rest believe what i say to you it is good sense
393-19219-0053 i would like greatly to get married if any one would have me
393-19219-0054 to idolize to coo to preen ourselves to be dove like to be dainty to bill and coo our loves from m
393-19219-0055 and to take away from us and put back in his box
393-19219-0056 my children receive an old man's blessing the evening was gay lively and agreeable the grandfather
393-19219-0057 there was a tumult then silence the married pair disappeared a little after midnight
393-19219-0058 here we pause on the threshold of wedding nights stands a smiling angel with his finger on his lip
393-19219-0059 ought to make its escape through the stones of the walls in brilliancy and vaguely illuminate the
393-19219-0060 dazzled with voluptuousness and believing themselves alone were to listen they would hear in their
393-19219-0061 when two mouths rendered sacred by love approach to create it is impossible that there should not
393-19219-0062 there is no joy outside of these joys love is the only ecstasy all the rest weeps to love or to ha
456-24741-0000 the idea of every mode in which the human body is affected by external bodies
456-24741-0001 proof all the modes in which any given body is affected
```

C. utt2spk (format : wave's id and speaker)



```
matbn-0761-0002 matbn-0761
matbn-0762-0001 matbn-0762
matbn-0762-0002 matbn-0762
matbn-0763-0001 matbn-0763
matbn-0763-0002 matbn-0763
matbn-0764-0001 matbn-0764
matbn-0765-0001 matbn-0765
matbn-0765-0002 matbn-0765
matbn-0766-0001 matbn-0766
matbn-0766-0002 matbn-0766
matbn-0767-0001 matbn-0767
matbn-0767-0002 matbn-0767
```

Note. The id must be consistent.

Third, check the run.sh script in your file.

Into file and open the run.sh. It's start from stage -2.

1. **Parameter setting**
   - *A. stage*
     - If you want to start from other stage, you can change it.
   - *B. num_jobs*
     - Use how many cpu core for compute in this script.

       Note. You need to think about your server's specification. And this values can't not over your total speaker.

2. **Each stage of script run.sh**
   - *A. In stage -2 is prepare lexicon and training language model.*
     - a. First command is prepare lexicon. It create some prepare file into the folder which are in data/local/dict. [local/prepare_dict.sh]
     - b. Second command is fix wav.scp, text and utt2spk of your training and testing data. It create the spk2utt and backup your old files. [utils/fix_data_dir.sh

data/train]

c. Third command is prepare training language model needs file. It create some prepare file into the folder which are in data/local/lang. [utils/prepare_lang.sh]
d. Fourth command is use Markov model to training language model. The “<SIL>” is silence phone. [local/train_lms.sh]

B. *In stage -1 is extracting features.*

a. This stage is make MFCC features from training and testing audio data, those data will create into the folder exp/mfcc.

C. *In stage 0 is training mono model.*

a. First command make some small data subsets for early system-build. [utils/subset_data_dir.sh]
b. Second command is training mono-phone model. The parameter *--boost-silence 1.25* is amplify the silence phone 1.25 times. [steps/train_mono.sh]
c. Third command is get alignments from mono-phone system. [steps/align_si.sh]
d. Fourth command is make HCLG.fst into exp/mono/graph. [utils/mkgraph.sh]
e. Fifth command is decode the testing data. And result will create into the folder exp/mono/decode_test. [utils/decode.sh]

D. *In stage 1 is training tri1 model.*

a. First command is training tri1 model (first tri-phone pass). The parameter 2500 is leaves numbers and the parameter 20000 is total of gauss. [steps/train_deltas.sh]
b. Second command is get alignments from tri1-phone system. [steps/align_si.sh]
c. Third command is make HCLG.fst into exp/tri1/graph. [utils/mkgraph.sh]
d. Fourth command is decode the testing data. And result will create into the folder exp/tri1/decode_test. [utils/decode.sh]

E. *In stage 2 is training tri2 model.*

a. First command is training tri2 model (delta+delta-deltas).
b. Second command is get alignments from tri2-phone system. [steps/align_si.sh]
c. Third command is make HCLG.fst into exp/tri2/graph. [utils/mkgraph.sh]
d. Fourth command is decode the testing data. And result will create into the folder exp/tri2/decode_test. [utils/decode.sh]

F. *In stage 3 is training tri3 model.*

a. First command is training tri3 model (LDA+MLLT).

b. Second command is make HCLG.fst into exp/tri3a/graph. [utils/mkgraph.sh]

c. Third command is decode the testing data. And result will create into the folder exp/tri3a/decode_test. [utils/decode.sh]

G. In stage 4 is training tri4 model. From now, we start building a more serious system (with SAT).

a. First command is do the alignment with fMLLR. [steps/align_fmllr.sh]

b. Second command is building serious system with SAT. [steps/train_sat.sh]

c. Third command is do the alignment with fMLLR again. [steps/align_fmllr.sh]

d. Fourth command is make HCLG.fst into exp/tri4a/graph. [utils/mkgraph.sh]

e. Fifth command is decode the testing data. And result will create into the folder exp/tri4a/decode_test. [utils/decode.sh]

H. In stage 5 is training tri5 model.

a. First command is building larger SAT system. [steps/train_sat.sh]
The parameter 3500 is leaves numbers and the parameter 100000 is total of gauss.

b. Second command is do the alignment with fMLLR. [steps/align_fmllr.sh]

c. Third command is make HCLG.fst into exp/tri5a/graph. [utils/mkgraph.sh]

d. Fourth command is decode the testing data. And result will create into the folder exp/tri5a/decode_test. [utils/decode.sh]

I. In stage 7 is training neural network

a. This stage is call local/chain/run_tdnn.sh to training.

J. Final stage is print your system words error rate (WER)

a. This stage will compute the best WER by your ASR system every each stages model.

Fourth, check the run_tdnn.sh script in local/chain

1. Parameter setting

A. Stage

- If you want to start from other stage, you can change it.

B. train_stage

- This is control the neural network's start stage. If you want to change your stage please check your folder in exp/chain/tdnn* which have some cache.* file you can start from that stage. Or you can just start from stage 0.

C. num_epochs

- This is set what epochs do you want to needs.
- Usually, you can set a small number like 4 or 6. If this model is well and you can set bigger like 15.

D. Inintal_effective_lrate

- The size used to update the network weight for each iter.

E. num_jobs_initial

- You need to consider the number of GPUs in your server.

2. Each stage of run_tdnn.sh script

A. First, need to make i-vector.

B. In stage 10 is neural network architecture

a. You can change your architecture here.

C. In stage 11 is neural network setting

D. In stage 12 is make graph with neural network model

a. This command is make HCLG.fst into exp/chain/tdnn*/graph. [utils/mkgraph.sh]

E. Final stage is decode testing data.

a. All decode result will save to exp/chain/tdnn*/graph.

Fifth, install and use kaldi offline system

1. Install kaldi offline system

A. go to kaldi/src

B. ./configure --shared

C. make depend -j 40

D. make -j 40

E. make ext

Note. If your system is already install please make clean it and start to stage C.

2. use offline ASR system

A. Go to your project and make a folder.

- mkdir nnet_online

B. make online decoding prepare data

- steps/online/nnet3/prepare_online_decoding.sh lm_build/lang/ exp/nnet3/extractor/exp/chain/tdnn _1a_sp / nnet_online/

C. copy nnet_online to src/online2bin

- cp -rf nnet_online/ /home/speech/kaldi/src/online2bin/

D. copy words.txt to src/online2bin/nnet_online

- cp data/local/lang_self/words.txt /home/speech/kaldi/src/online2bin/ nnet_online/

E. *copy HCLG.fst to src/online2bin/nnet_online*

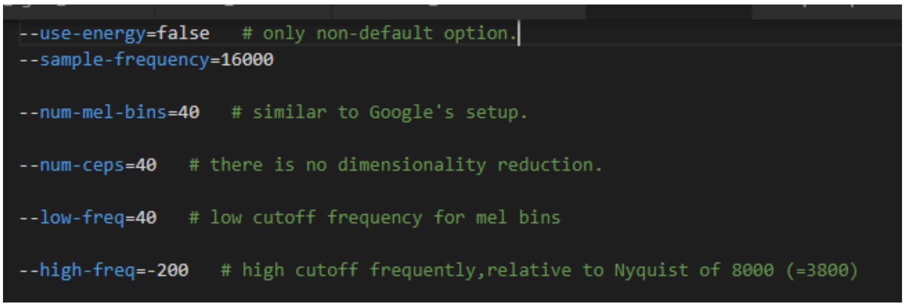
- cp exp/chain/tdnn*/HCLG.fst /home/speech/kaldi/src/online2bin/nnet_online/

F. *copy utils to src/online2bin*

- cp -rf utils /home/speech/kaldi/src/online2bin/

G. *fix config file in nnet_online/conf*

- Fix config’s path to new position in every config file.
- Fix mfcc.conf file.



```
--use-energy=false   # only non-default option.
--sample-frequency=16000

--num-mel-bins=40   # similar to Google's setup.

--num-ceps=40   # there is no dimensionality reduction.

--low-freq=40   # low cutoff frequency for mel bins

--high-freq=-200   # high cutoff frequently,relative to Nyquist of 8000 (=3800)
```

H. *Start to decode the wav file*

- *./online2-wav-nnet3-latgen-faster --add-pitch=true --do-endpointing=false --frames-per-chunk=20 --extra-left-context-initial=0 --online=true --frame-subsampling-factor=3 --config=nnet_online/conf/online.conf --min-active=200 --max-active=7000 --beam=15.0 --lattice-beam=6.0 --acoustic-scale=1.0 --word-symbol-table=nnet_online/words.txt nnet_online/final.mdl nnet_online/HCLG.fst 'ark:echo utter1 utter1|' 'scp:echo utter1 /nfs/GPU/home/speech/kaldi-data/SNR_Setting/SNR_Setting1/C0000001.wav|' ark:/dev/null*

You can change your own wav file

Sixth, Tutorials

https://sites.google.com/speech.ntut.edu.tw/fsw/home/tutorials?authuser=0