1)

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ utils/prepare\_lang.sh --position-dependent-phones false data/local/dict "<SIL>" data/local/lang data/lang

Checking data/local/dict/silence\_phones.txt ...

--> reading data/local/dict/silence\_phones.txt

--> data/local/dict/silence\_phones.txt is OK

Checking data/local/dict/optional\_silence.txt ...

--> reading data/local/dict/optional\_silence.txt

--> data/local/dict/optional\_silence.txt is OK

Checking data/local/dict/nonsilence\_phones.txt ...

--> reading data/local/dict/nonsilence\_phones.txt

--> data/local/dict/nonsilence\_phones.txt is OK

Checking disjoint: silence\_phones.txt, nonsilence\_phones.txt

--> disjoint property is OK.

Checking data/local/dict/lexicon.txt

--> reading data/local/dict/lexicon.txt

--> data/local/dict/lexicon.txt is OK

Checking data/local/dict/extra\_questions.txt ...

--> data/local/dict/extra\_questions.txt is empty (this is OK)

--> SUCCESS [validating dictionary directory data/local/dict]

\*\*Creating data/local/dict/lexiconp.txt from data/local/dict/lexicon.txt

fstaddselfloops data/lang/phones/wdisambig\_phones.int data/lang/phones/wdisambig\_words.int

prepare\_lang.sh: validating output directory

utils/validate\_lang.pl data/lang

Checking data/lang/phones.txt ...

--> data/lang/phones.txt is OK

Checking words.txt: #0 ...

--> data/lang/words.txt is OK

Checking disjoint: silence.txt, nonsilence.txt, disambig.txt ...

--> silence.txt and nonsilence.txt are disjoint

--> silence.txt and disambig.txt are disjoint

--> disambig.txt and nonsilence.txt are disjoint

--> disjoint property is OK

Checking sumation: silence.txt, nonsilence.txt, disambig.txt ...

--> summation property is OK

Checking data/lang/phones/context\_indep.{txt, int, csl} ...

--> 1 entry/entries in data/lang/phones/context\_indep.txt

--> data/lang/phones/context\_indep.int corresponds to data/lang/phones/context\_indep.txt

--> data/lang/phones/context\_indep.csl corresponds to data/lang/phones/context\_indep.txt

--> data/lang/phones/context\_indep.{txt, int, csl} are OK

Checking data/lang/phones/nonsilence.{txt, int, csl} ...

--> 2 entry/entries in data/lang/phones/nonsilence.txt

--> data/lang/phones/nonsilence.int corresponds to data/lang/phones/nonsilence.txt

--> data/lang/phones/nonsilence.csl corresponds to data/lang/phones/nonsilence.txt

--> data/lang/phones/nonsilence.{txt, int, csl} are OK

Checking data/lang/phones/silence.{txt, int, csl} ...

--> 1 entry/entries in data/lang/phones/silence.txt

--> data/lang/phones/silence.int corresponds to data/lang/phones/silence.txt

--> data/lang/phones/silence.csl corresponds to data/lang/phones/silence.txt

--> data/lang/phones/silence.{txt, int, csl} are OK

Checking data/lang/phones/optional\_silence.{txt, int, csl} ...

--> 1 entry/entries in data/lang/phones/optional\_silence.txt

--> data/lang/phones/optional\_silence.int corresponds to data/lang/phones/optional\_silence.txt

--> data/lang/phones/optional\_silence.csl corresponds to data/lang/phones/optional\_silence.txt

--> data/lang/phones/optional\_silence.{txt, int, csl} are OK

Checking data/lang/phones/disambig.{txt, int, csl} ...

--> 2 entry/entries in data/lang/phones/disambig.txt

--> data/lang/phones/disambig.int corresponds to data/lang/phones/disambig.txt

--> data/lang/phones/disambig.csl corresponds to data/lang/phones/disambig.txt

--> data/lang/phones/disambig.{txt, int, csl} are OK

Checking data/lang/phones/roots.{txt, int} ...

--> 3 entry/entries in data/lang/phones/roots.txt

--> data/lang/phones/roots.int corresponds to data/lang/phones/roots.txt

--> data/lang/phones/roots.{txt, int} are OK

Checking data/lang/phones/sets.{txt, int} ...

--> 3 entry/entries in data/lang/phones/sets.txt

--> data/lang/phones/sets.int corresponds to data/lang/phones/sets.txt

--> data/lang/phones/sets.{txt, int} are OK

Checking data/lang/phones/extra\_questions.{txt, int} ...

Checking optional\_silence.txt ...

--> reading data/lang/phones/optional\_silence.txt

--> data/lang/phones/optional\_silence.txt is OK

Checking disambiguation symbols: #0 and #1

--> data/lang/phones/disambig.txt has "#0" and "#1"

--> data/lang/phones/disambig.txt is OK

Checking topo ...

Checking word-level disambiguation symbols...

--> data/lang/phones/wdisambig.txt exists (newer prepare\_lang.sh)

Checking data/lang/oov.{txt, int} ...

--> 1 entry/entries in data/lang/oov.txt

--> data/lang/oov.int corresponds to data/lang/oov.txt

--> data/lang/oov.{txt, int} are OK

--> data/lang/L.fst is olabel sorted

--> data/lang/L\_disambig.fst is olabel sorted

--> SUCCESS [validating lang directory data/lang]

2)

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ lm/prepare\_lm.sh

Preparing language models for test

arpa2fst --disambig-symbol=#0 --read-symbol-table=data/lang\_test\_tg/words.txt lm/yesno-unigram.arpabo data/lang\_test\_tg/G.fst

LOG (arpa2fst[5.5.901~1-a401]:Read():arpa-file-parser.cc:94) Reading \data\ section.

LOG (arpa2fst[5.5.901~1-a401]:Read():arpa-file-parser.cc:149) Reading \1-grams: section.

LOG (arpa2fst[5.5.901~1-a401]:RemoveRedundantStates():arpa-lm-compiler.cc:359) Reduced num-states from 1 to 1

fstisstochastic data/lang\_test\_tg/G.fst

1.20397 1.20397

Succeeded in formatting data.

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ steps/make\_mfcc.sh --nj 4 data/train\_yesno exp/log/train\_yesno

steps/make\_mfcc.sh --nj 4 data/train\_yesno exp/log/train\_yesno

utils/validate\_data\_dir.sh: utt2spk is not in sorted order when sorted first on speaker-id

(fix this by making speaker-ids prefixes of utt-ids)

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ steps/make\_mfcc.sh --nj 4 data/train\_yesno exp/log/train\_yesno

steps/make\_mfcc.sh --nj 4 data/train\_yesno exp/log/train\_yesno

utils/validate\_data\_dir.sh: file data/train\_yesno/utt2spk is not in sorted order or has duplicates

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ steps/make\_mfcc.sh --nj 4 data/train\_yesno exp/log/train\_yesno

steps/make\_mfcc.sh --nj 4 data/train\_yesno exp/log/train\_yesno

utils/validate\_data\_dir.sh: Successfully validated data-directory data/train\_yesno

steps/make\_mfcc.sh: [info]: no segments file exists: assuming wav.scp indexed by utterance.

Succeeded creating MFCC features for train\_yesno

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ steps/compute\_cmvn\_stats.sh data/train\_yesno exp/log/train\_yesno

steps/compute\_cmvn\_stats.sh data/train\_yesno exp/log/train\_yesno

steps/compute\_cmvn\_stats.sh: warning: it seems not all of the speakers got cmvn stats (1 != 0);

Succeeded creating CMVN stats for train\_yesno

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ steps/train\_mono.sh --nj 1 --cmd utils/run.pl data data/lang exp/mono

steps/train\_mono.sh --nj 1 --cmd utils/run.pl data data/lang exp/mono

cat: data/utt2spk: No such file or directory

Failed to open utt2spk file data/utt2spk at utils/split\_scp.pl line 85.

yasin@XJU-YI-PC:~/GitHub/kaldi-yesno-tutorial$ steps/train\_mono.sh --nj 1 --cmd utils/run.pl data/train\_yesno data/lang exp/mono

steps/train\_mono.sh --nj 1 --cmd utils/run.pl data/train\_yesno data/lang exp/mono

\*\* split\_data.sh: warning, #lines is (utt2spk,text) is (31,30); you can

\*\* use utils/fix\_data\_dir.sh to fix this.

steps/train\_mono.sh: Initializing monophone system.

steps/train\_mono.sh: Compiling training graphs

steps/train\_mono.sh: Aligning data equally (pass 0)

steps/train\_mono.sh: Pass 1

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 2

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 3

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 4

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 5

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 6

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 7

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 8

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 9

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 10

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 11

steps/train\_mono.sh: Pass 12

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 13

steps/train\_mono.sh: Pass 14

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 15

steps/train\_mono.sh: Pass 16

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 17

steps/train\_mono.sh: Pass 18

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 19

steps/train\_mono.sh: Pass 20

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 21

steps/train\_mono.sh: Pass 22

steps/train\_mono.sh: Pass 23

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 24

steps/train\_mono.sh: Pass 25

steps/train\_mono.sh: Pass 26

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 27

steps/train\_mono.sh: Pass 28

steps/train\_mono.sh: Pass 29

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 30

steps/train\_mono.sh: Pass 31

steps/train\_mono.sh: Pass 32

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 33

steps/train\_mono.sh: Pass 34

steps/train\_mono.sh: Pass 35

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 36

steps/train\_mono.sh: Pass 37

steps/train\_mono.sh: Pass 38

steps/train\_mono.sh: Aligning data

steps/train\_mono.sh: Pass 39

steps/diagnostic/analyze\_alignments.sh --cmd utils/run.pl data/lang exp/mono

steps/diagnostic/analyze\_alignments.sh: see stats in exp/mono/log/analyze\_alignments.log

28 warnings in exp/mono/log/update.\*.log

exp/mono: nj=1 align prob=-79.02 over 0.05h [retry=0.0%, fail=0.0%] states=11 gauss=943

steps/train\_mono.sh: Done training monophone system in exp/mono