

# 數位語音處理概論

## HW#2-1

## HMM Training and Testing

助教：盧宏宗

教授：李琳山

# Digit Recognizer

Construct a digit recognizer

ling | yi | er | san | si | wu | liu | qi | ba | jiu

Free tools of HMM: HTK

<http://htk.eng.cam.ac.uk/>

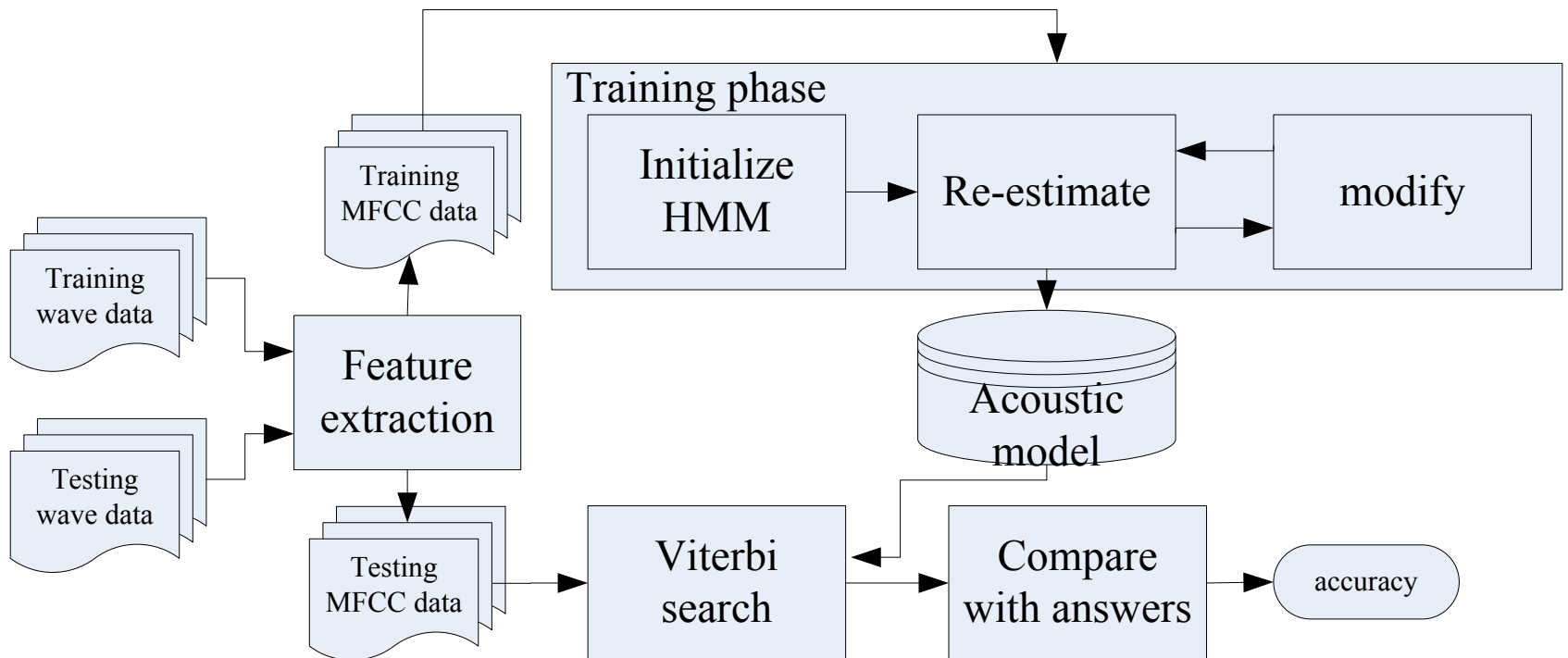
build it yourself, or use the compiled version

*htk341\_debian\_x86\_64.tar.gz*

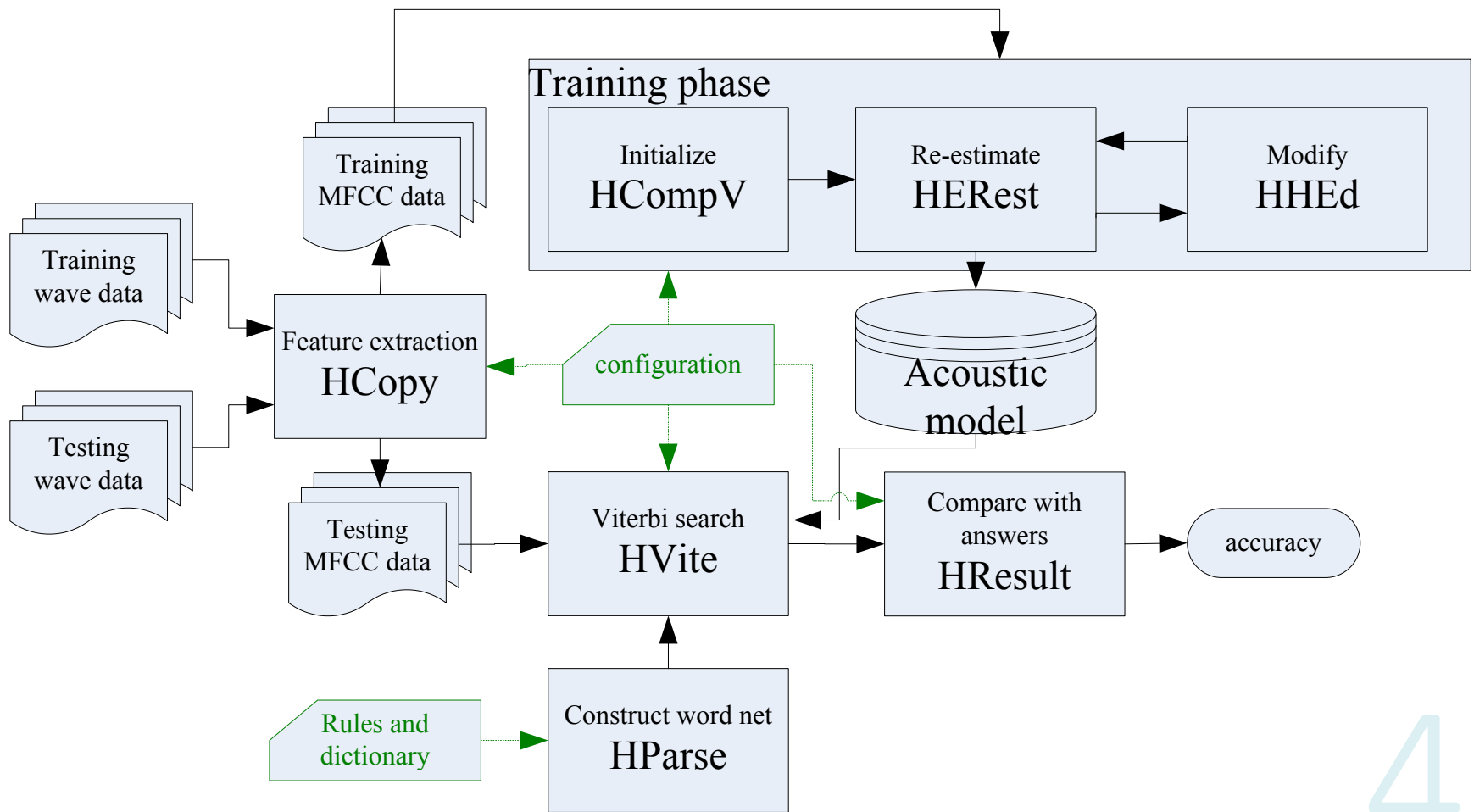
Training data, testing data, scripts, and other resources

all are available on <http://speech.ee.ntu.edu.tw/courses/DSP2015Spring/>

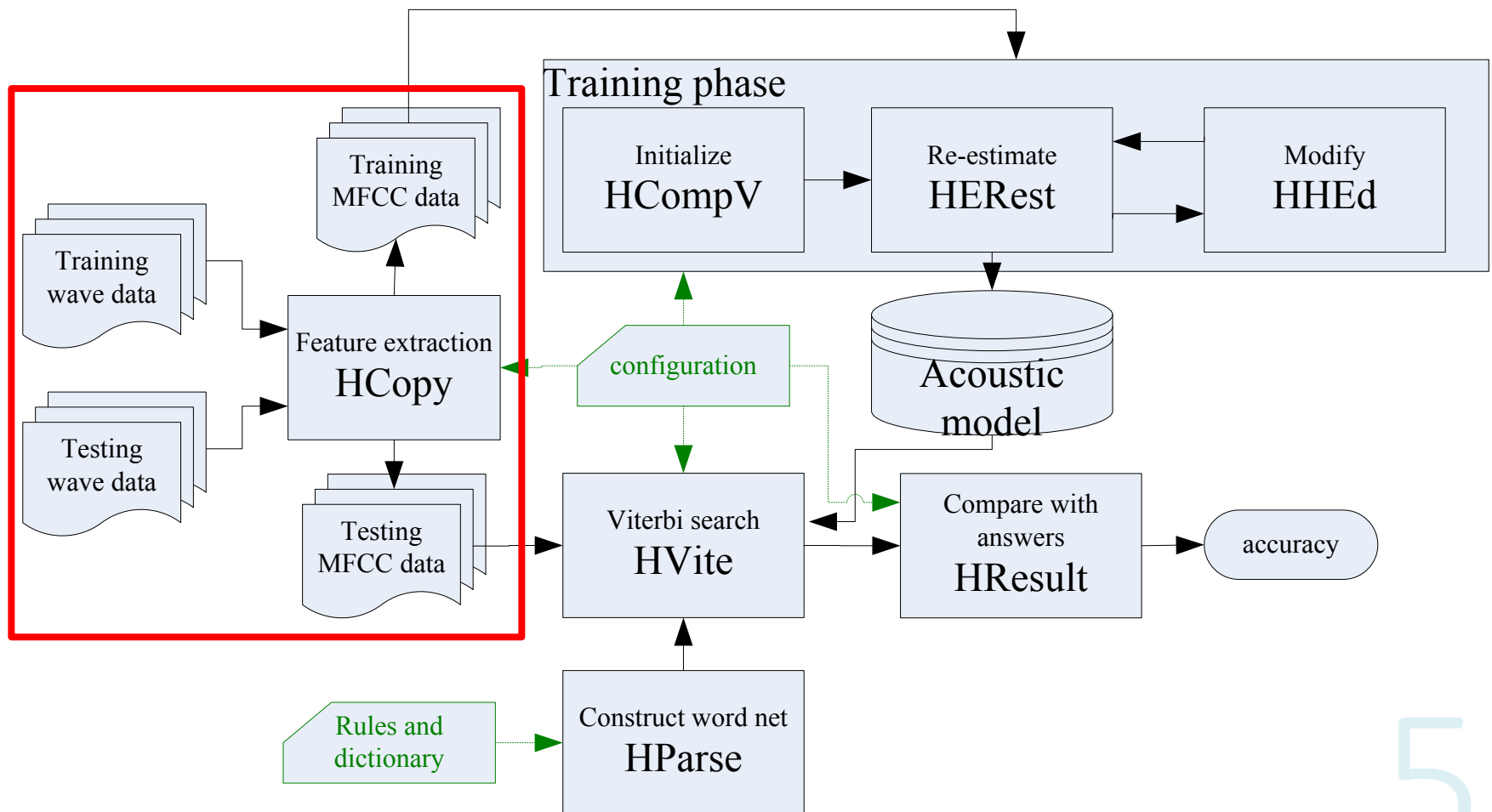
# Flowchart



# Thanks to HTK!



# Feature Extraction



# Feature Extraction - HCopy

```
HCopy -C lib/hcopy.cfg -S scripts/training_hcopy.scp
```

Convert wave to 39 dimension MFCC

-C lib/hcopy.cfg

- input and output format

e.g. wav -> MFCC\_Z\_E\_D\_A

- parameters of feature extraction
- Chapter 7 - Speech Signals and Front-end Processing

-S scripts/training\_hcopy.scp

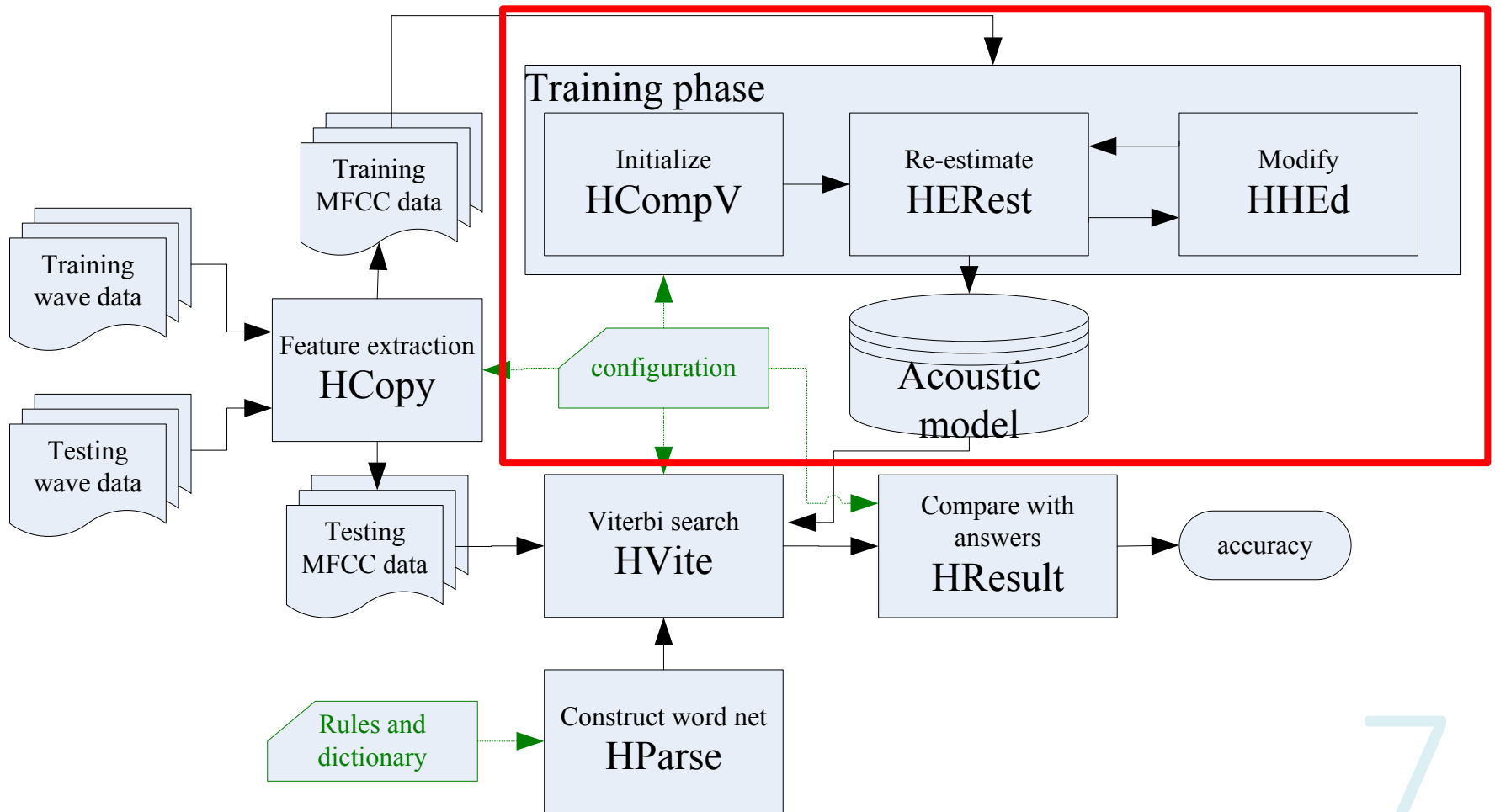
- a mapping from Input file name to output file name

speechdata/training/  
N110022.wav

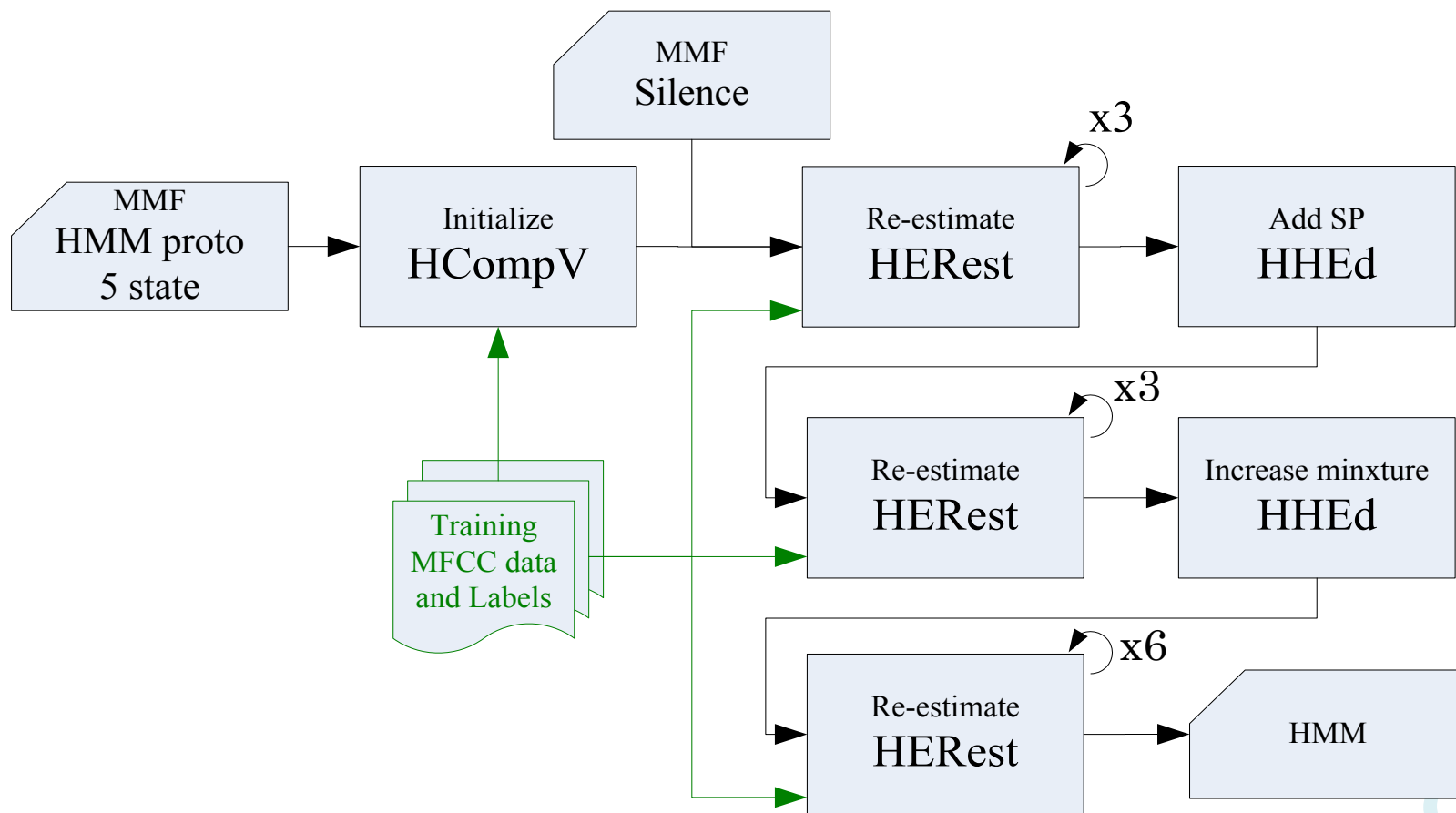


MFCC/training/  
N110022.mfc

# Training Flowchart

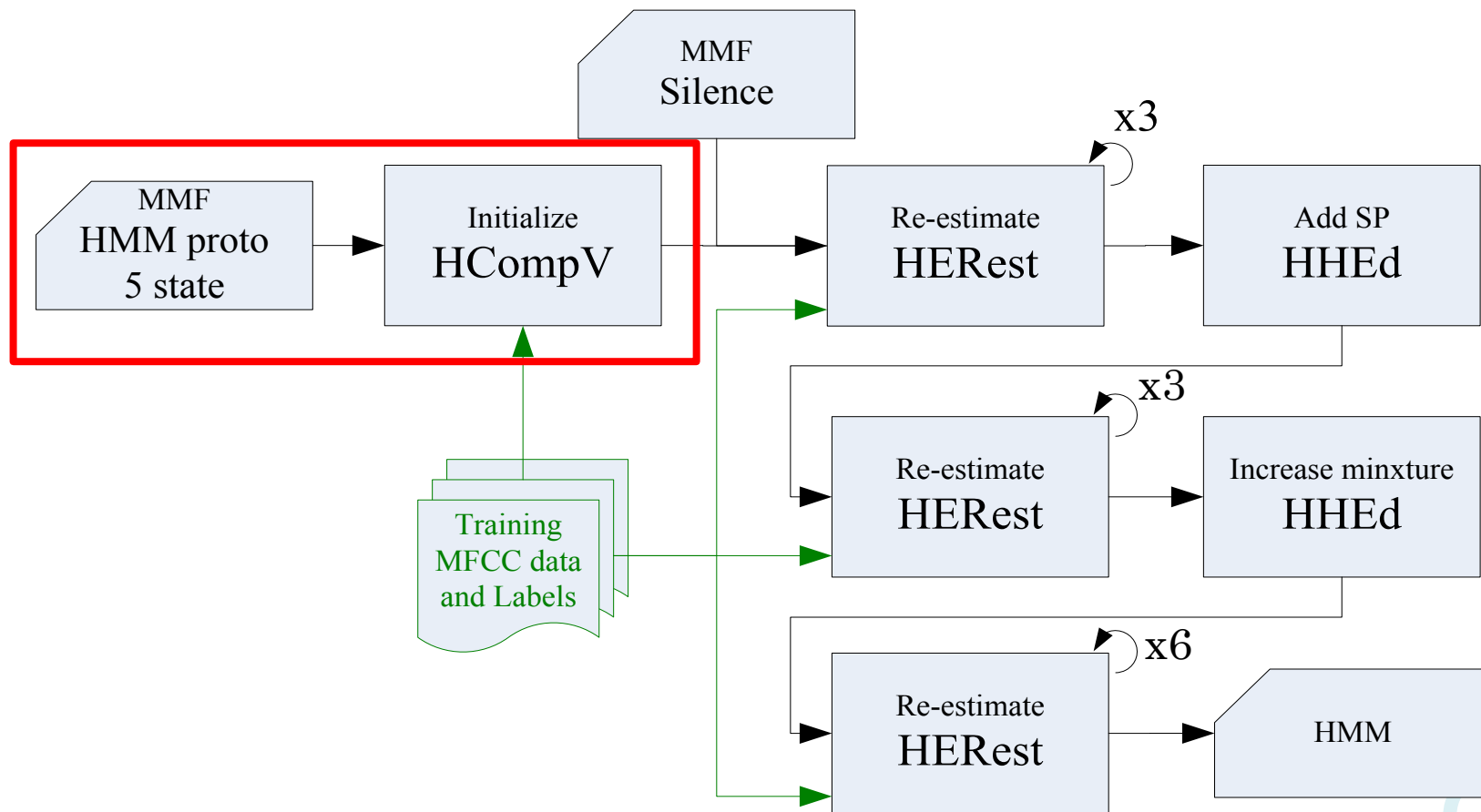


# Training Flowchart



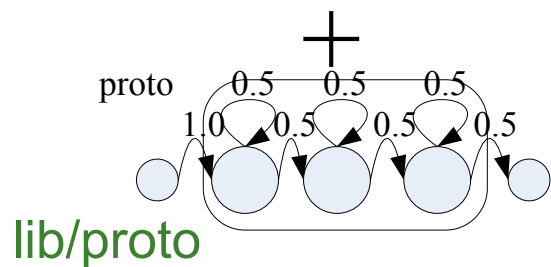
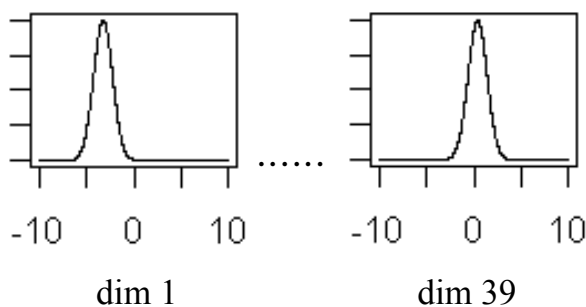
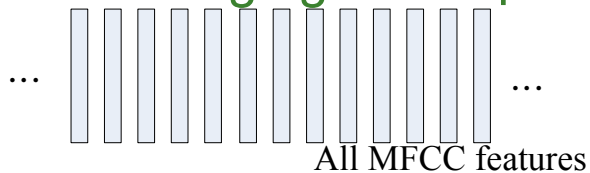


# Training Flowchart

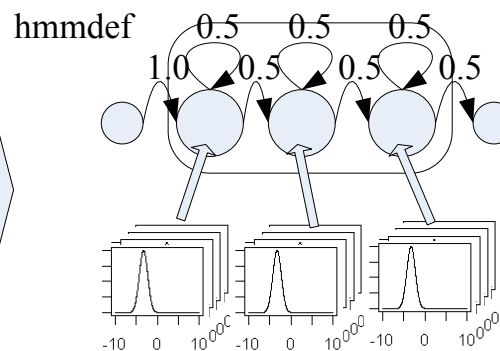


# HCompV - Initialize

**-C** lib/config.fig **-S** scripts/training.scp



**-o** hmmdef **-M** hmm



# HCompV - Initialize

```
HCompV -C lib/config.fig -o hmmdef  
-M hmm -S scripts/training.scp lib/proto
```

Compute global mean and variance of features

-C lib/config.fig

- set format of input feature (MFCC\_Z\_E\_D\_A)

-o hmmdef -M hmm

- set output name: hmm/hmmdef

-S scripts/training.scp

- a list of training data

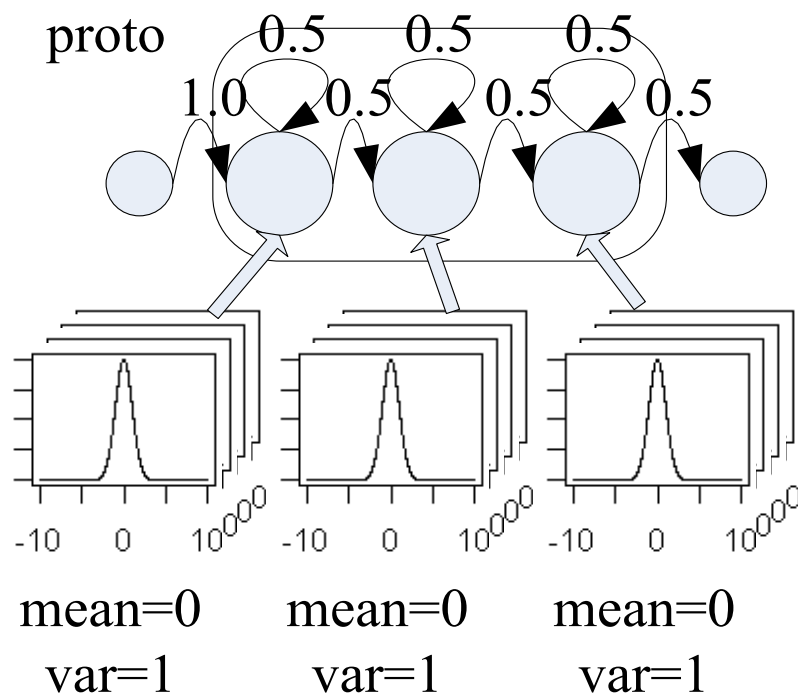
lib/proto

- a description of a HMM model, HTK MMF format

You can modify the Model  
Format here. (# states)

# Initial MMF Prototype

## MMF: HTKBook chapter 7

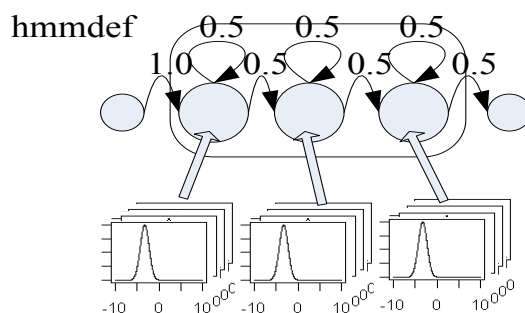


```

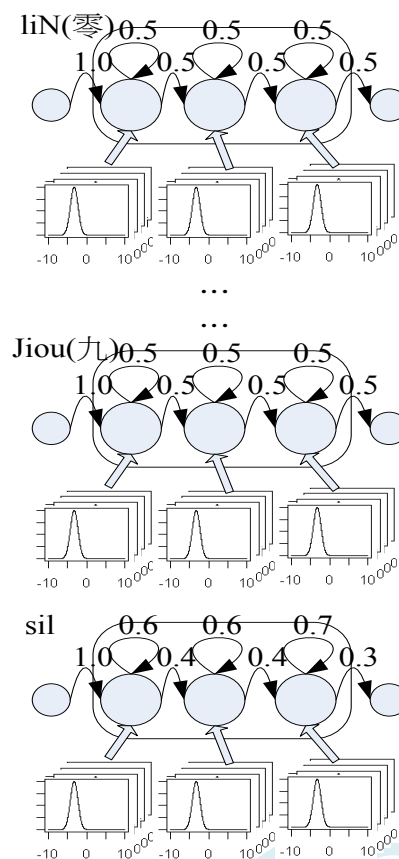
~o <VECSIZE> 39 <MFCC_Z_E_D_A>
~h "proto"
<BeginHMM>
<NumStates> 5
<State> 2
<Mean> 39
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ...
<Variance> 39
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 ...
<State> 3
<Mean> 39
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ...
<Variance> 39
...
<TransP> 5
0.0 1.0 0.0 0.0 0.0
0.0 0.5 0.5 0.0 0.0
0.0 0.0 0.5 0.5 0.0
0.0 0.0 0.0 0.5 0.5
0.0 0.0 0.0 0.0 0.0
<EndHMM>
    
```

# Initial HMM

hmm/hmmdef



hmm/models



bin/macro

- construct each HMM

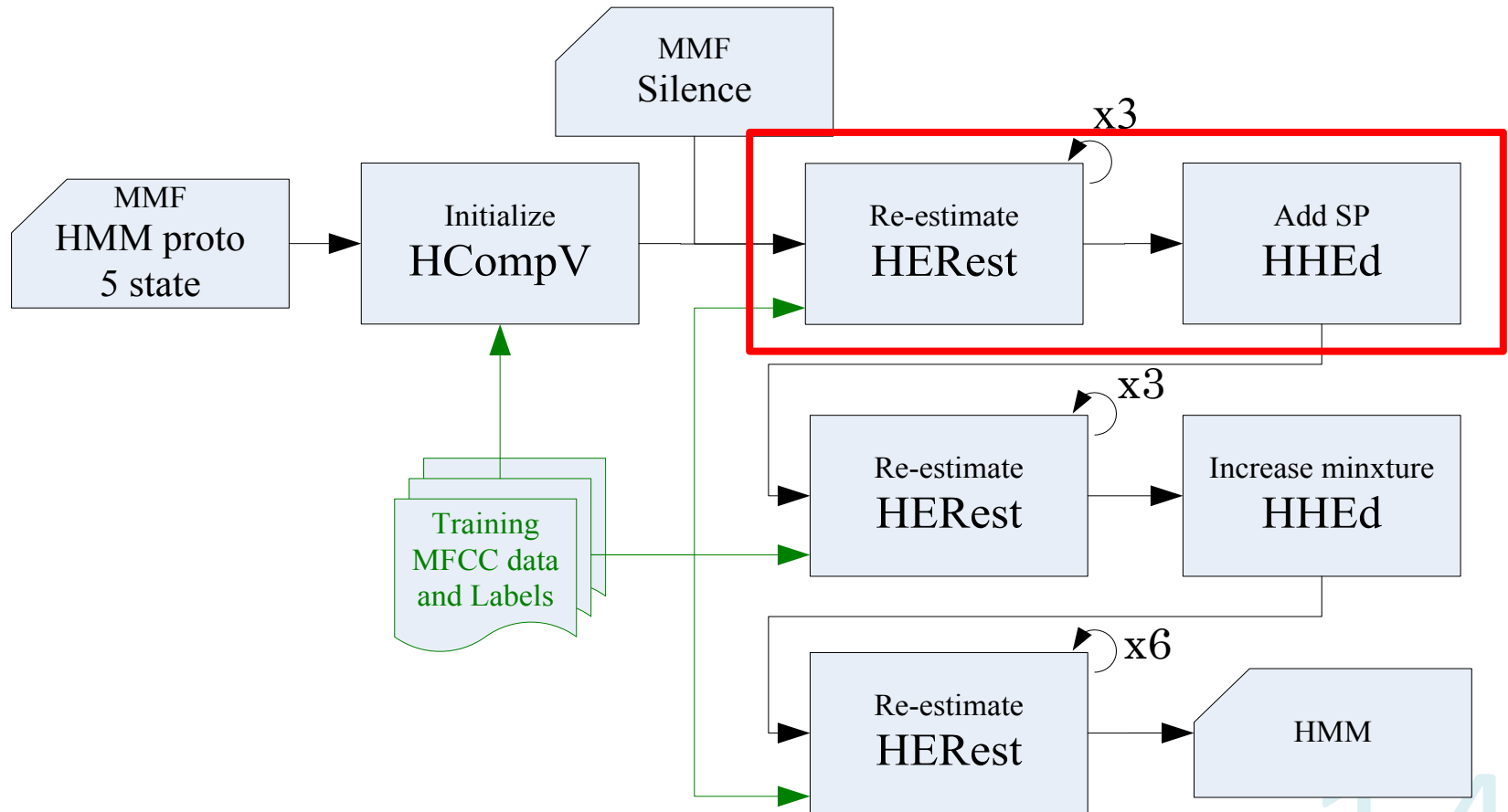
bin/models\_1mixsil

- add silence HMM

These are written in C

- But you can do these by a text editor

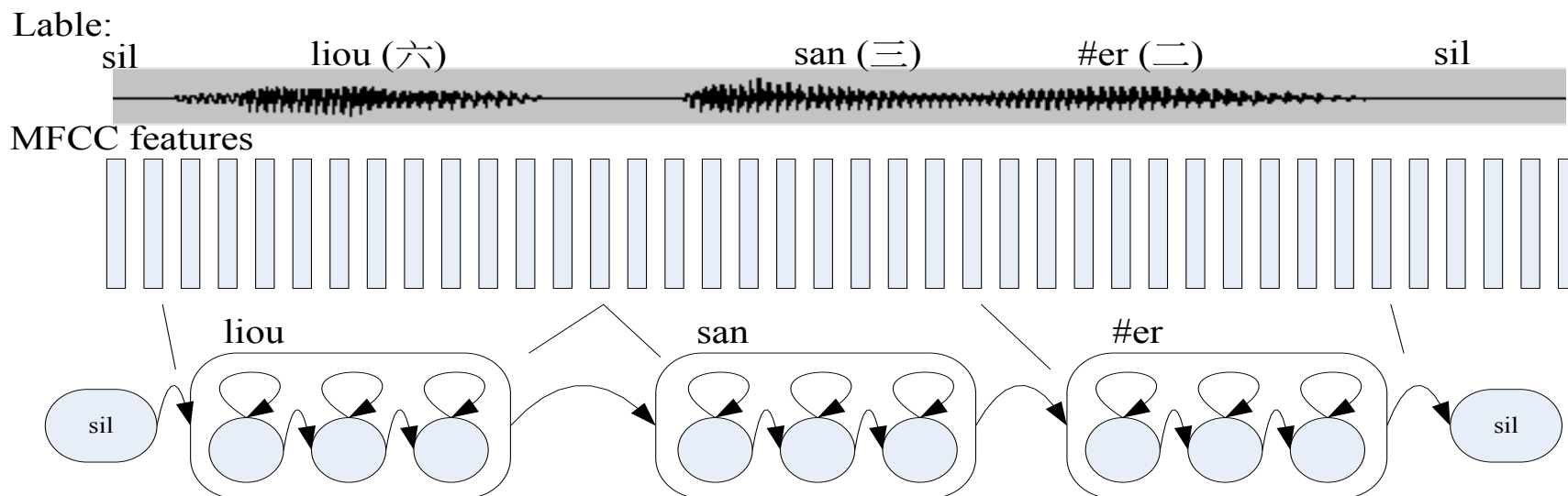
# Training Flowchart



# HERest - Adjust HMMs

## Basic problem 3 for HMM

- Given  $O$  and an initial model  $\lambda=(A,B, \pi)$ , adjust  $\lambda$  to maximize  $P(O|\lambda)$



# HERest - Adjust HMMs

```
HERest -C lib/config.cfg -S scripts/training.scp -I labels/Cleano8TR.mlf  
-H hmm/macros -H hmm/models -M hmm lib/models.lst
```

Adjust parameters  $\lambda$  to maximize  $P(O|\lambda)$

- one iteration of EM algorithm
- run this command three times => three iterations

**-I labels/Cleano8TR.mlf**

- set label file to “labels/Cleano8TR.mlf”

**lib/models.lst**

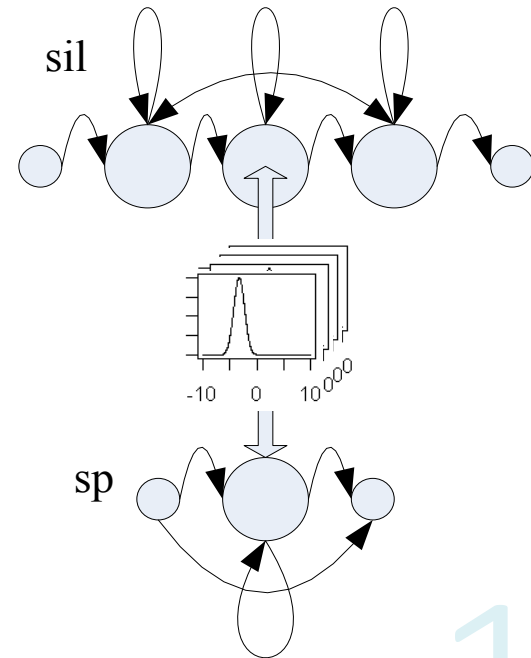
- a list of word models  
( liN ( 零 ), #i ( 一 ), #er ( 二 ), ... jiou ( 九 ), sil )



# Add SP Model

```
bin/spmodel_gen hmm/models hmm/models
```

Add “sp” (short pause) HMM definition to MMF file  
“hmm/hmmdef”



# HHEd - Modify HMMs

```
HHEd -H hmm/macros -H hmm/models  
-M hmm lib/sil1.hed lib/models_sp.lst
```

## lib/sil1.hed

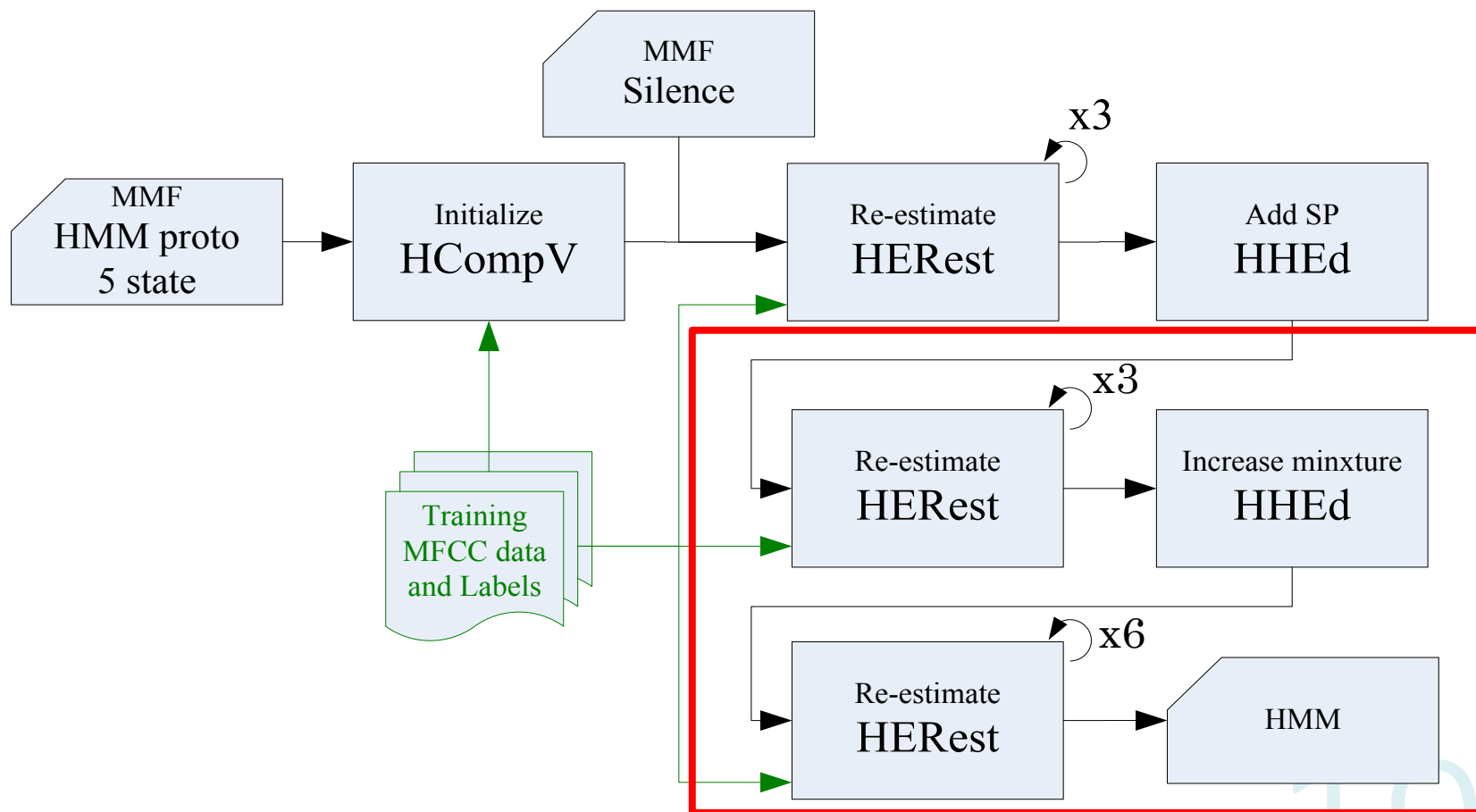
- a list of command to modify HMM definitions

## lib/models\_sp.lst

- a new list of model  
( liN ( 零 ), #i ( 一 ), #er ( 二 ), ... jiou ( 九 ), sil, sp )

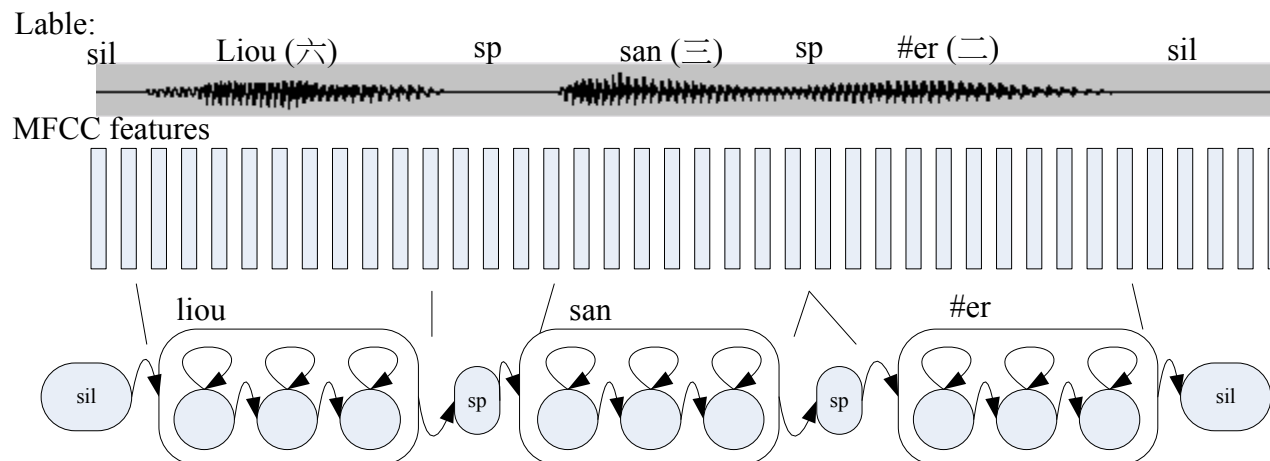
See HTK book 3.2.2 (p. 33)

# Training Flowchart



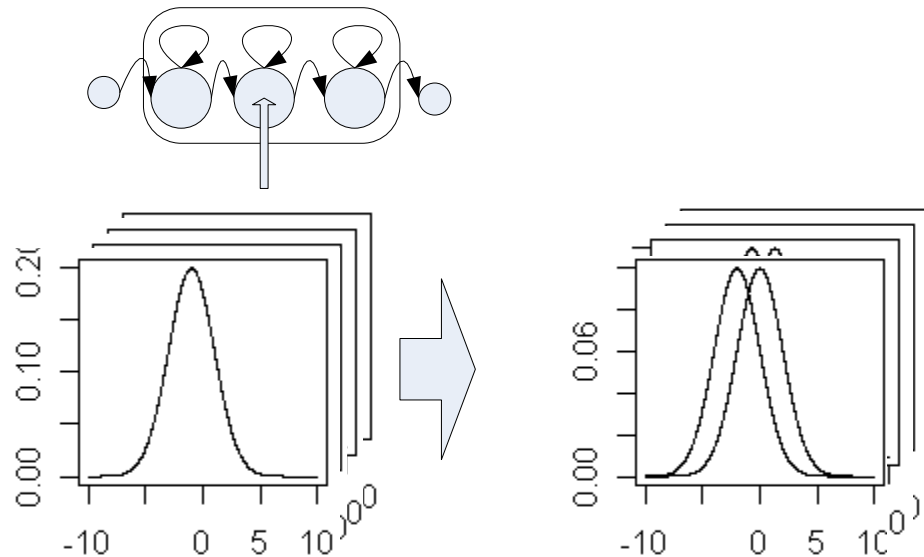
# HERest - Adjust HMMs Again

```
HERest -C lib/config.cfg -S scripts/training.scp  
-I labels/Clean08TR_sp.mlf -H hmm/macros  
-H hmm/models -M hmm lib/models_sp.lst
```



# HHEd – Increase Number of Mixtures

```
HHEd -H hmm/macros -H hmm/models  
-M hmm lib/mix2_10.hed lib/models_sp.lst
```



# Modification of Models

lib/mix2\_10.hed

MU 2 {liN.state[2-4].mix}

MU 2 {#l.state[2-4].mix}

MU 2 {#er.state[2-4].mix}

MU 2 {san.state[2-4].mix}

MU 2 {sy.state[2-4].mix}

...

MU 3 {sil.state[2-4].mix}

MU +2 {san.state[2-9].mix}

Check HTKBook 17.8 HHed for more details

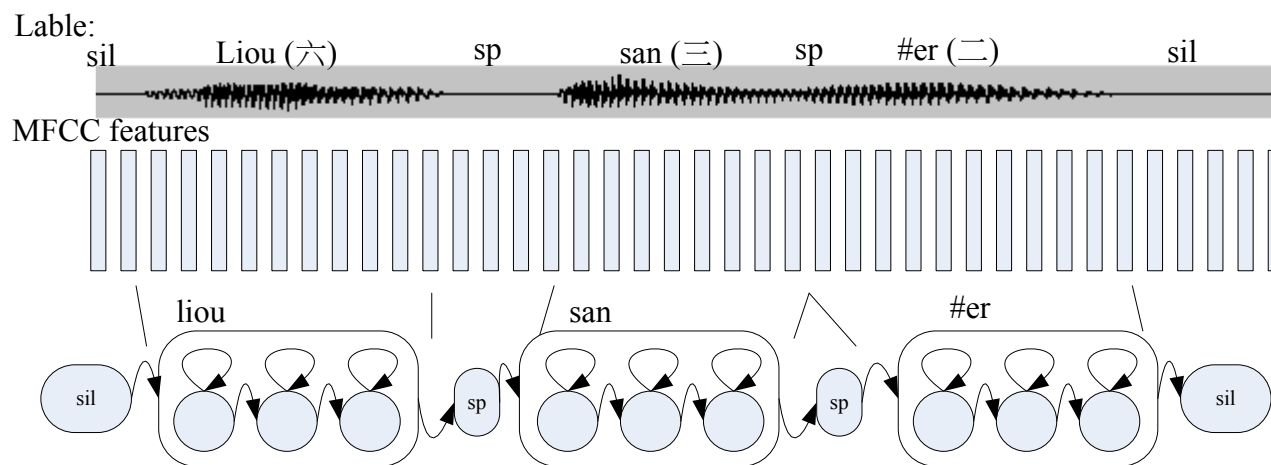
You can modify # of Gaussian mixture here.

This value tells HTK to change the mixture number from state 2 to state 4. If you want to change # state, check lib/proto.

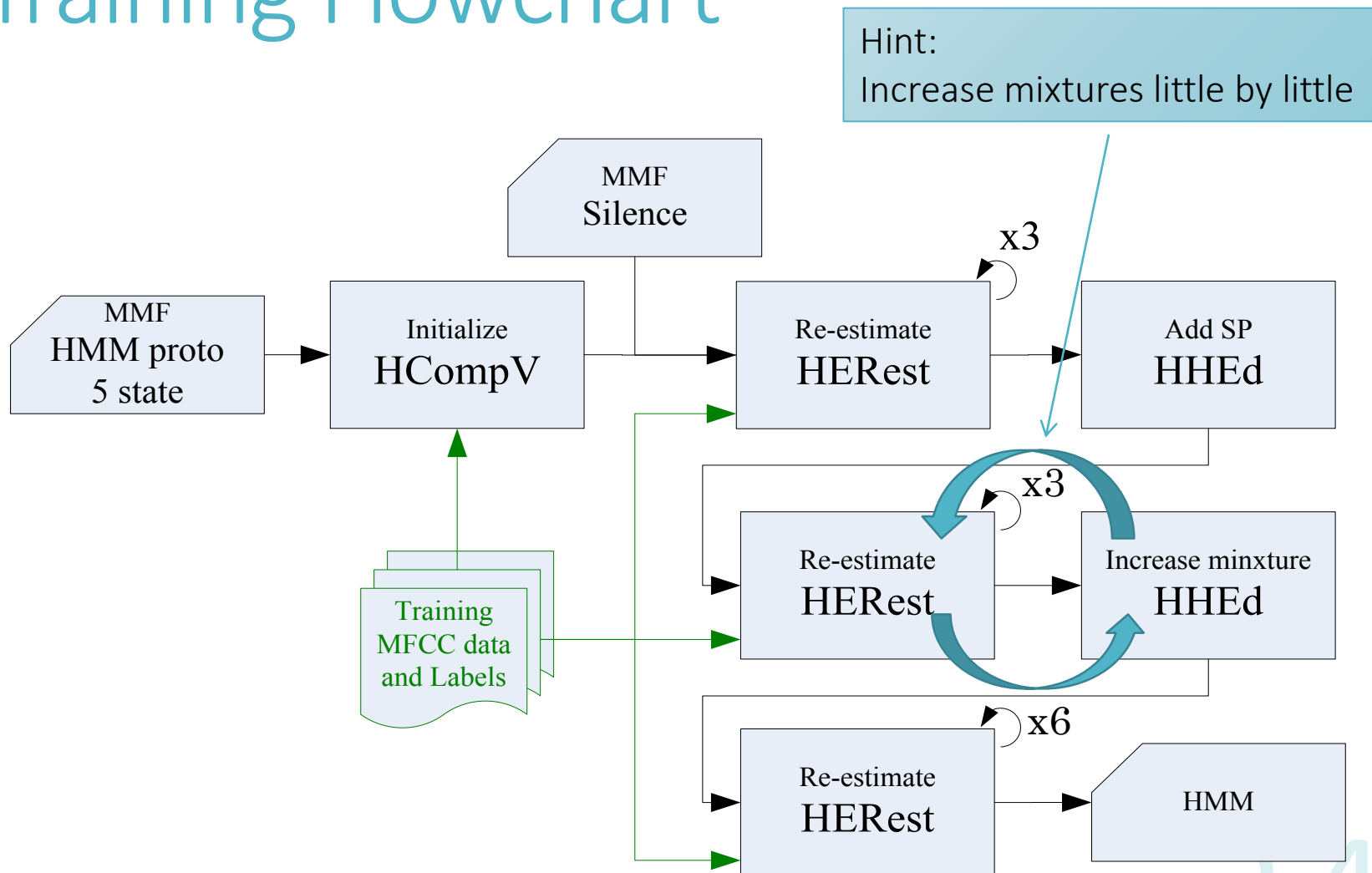
You can increase # Gaussian mixture here.

# HERest - Adjust HMMs Again

```
HERest -C lib/config.cfg -S scripts/training.scp  
-I labels/Cleano8TR_sp.mlf -H hmm/macros  
-H hmm/models -M hmm lib/models_sp.lst
```

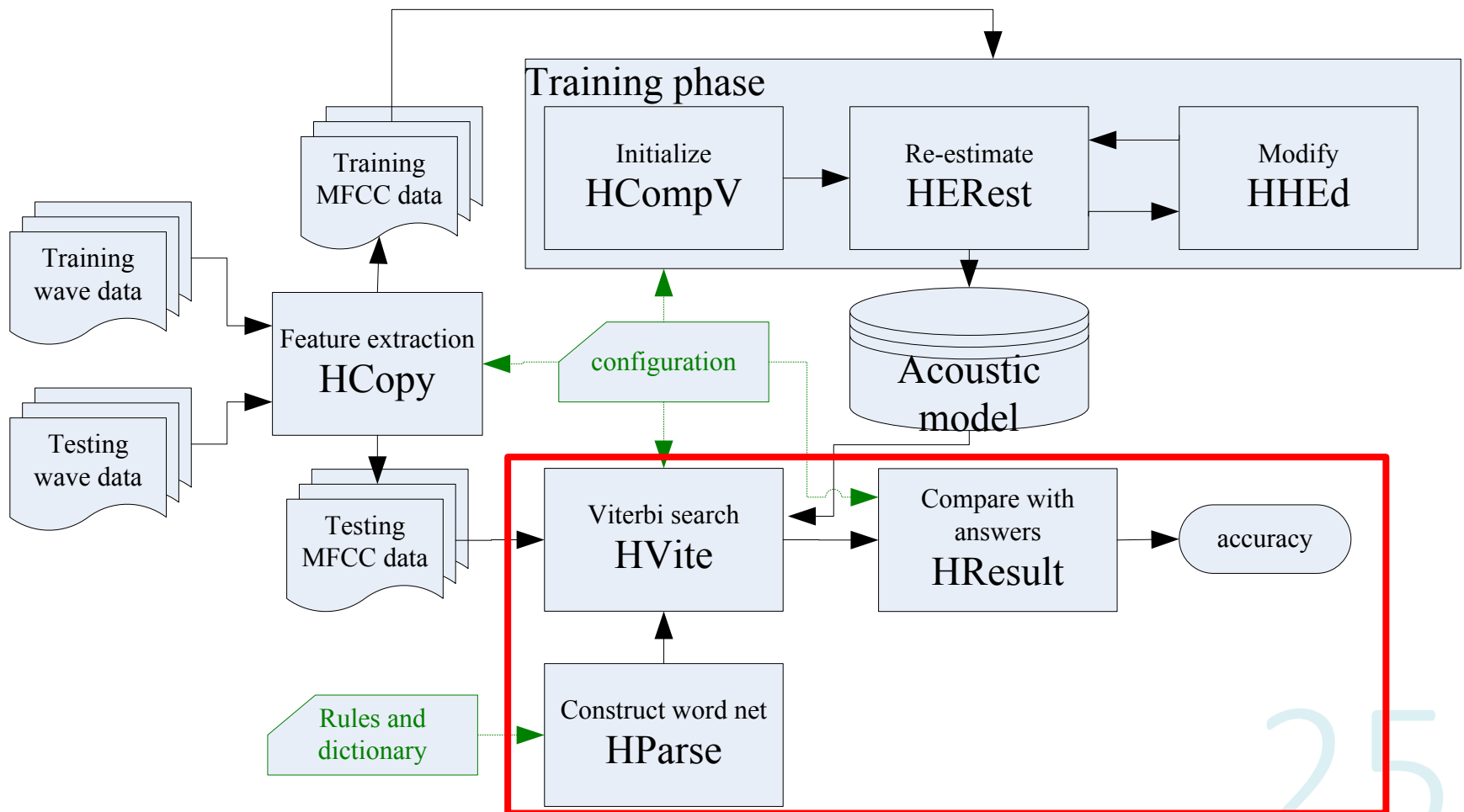


# Training Flowchart





# Testing Flowchart



# HParse - Construct Word Net

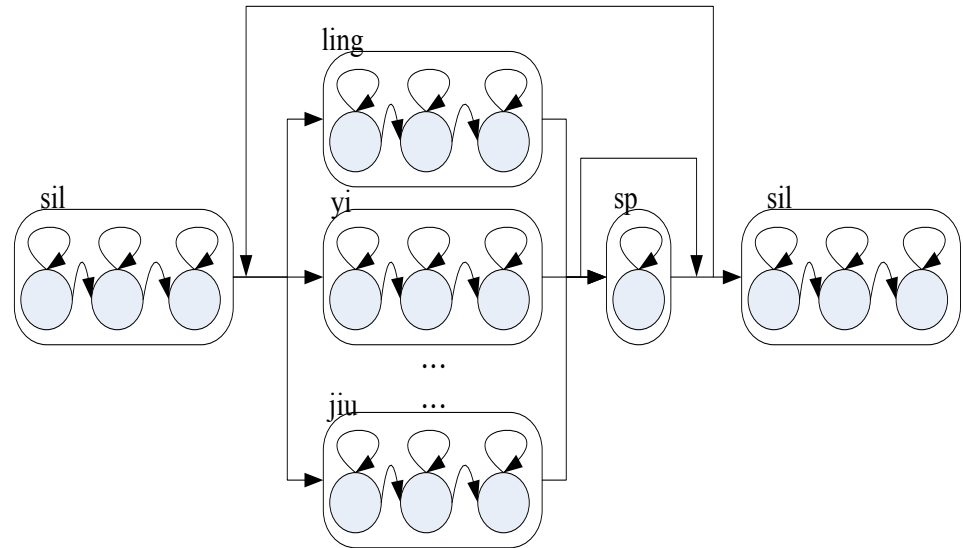
HParse `lib/grammar_sp` `lib/wdnet_sp`

`lib/grammar_sp`

- regular expression

`lib/wdnet_sp`

- output word net



# HVite - Viterbi Search

```
HVite -H hmm/macros -H hmm/models -S scripts/testing.scp  
-C lib/config.cfg -w lib/wdnet_sp -l '*' -i result/result.mlf  
-p 0.0 -s 0.0 lib/dict lib/models_sp.lst
```

**-w lib/wdnet\_sp**

- input word net

**-i result/result.mlf**

- output MLF file

**lib/dict**

- dictionary: a mapping from word to phone sequences

*ling -> liN, er -> #er, ... . —-> sic\_i i, ㄥ-> chi\_i i*

# HResult - Compared With Answer

```
HResults -e "???" sil -e "???" sp  
-I labels/answer.mlf lib/models_sp.lst result/result.mlf
```

## Longest Common Subsequence (LCS)

===== HTK Results Analysis =====

Date: Wed Apr 17 00:26:54 2013

Ref : labels/answer.mlf

Rec : result/result.mlf

----- Overall Results -----

SENT: %Correct=38.54 [H=185, S=295, N=480]

WORD: %Corr=96.61, Acc=74.34 [H=1679, D=13, S=46, I=387, N=1738]

=====

# Part 1 (40%) – Run Baseline

Download HTK tools and homework package

Set PATH for HTK tools

- `set_htk_path.sh`

Execute (bash shell script)

- `(00_clean_all.sh)`
- `01_run_HCopy.sh`
- `02_run_HCompV.sh`
- `03_training.sh`
- `04_testing.sh`

You can find accuracy in “result/accuracy”

- the baseline accuracy is 74.34%

# Useful tips

## To unzip files

- `unzip XXXX.zip`
- `tar -zxvf XXXX.tar.gz`

## To set path in “set\_htk\_path.sh”

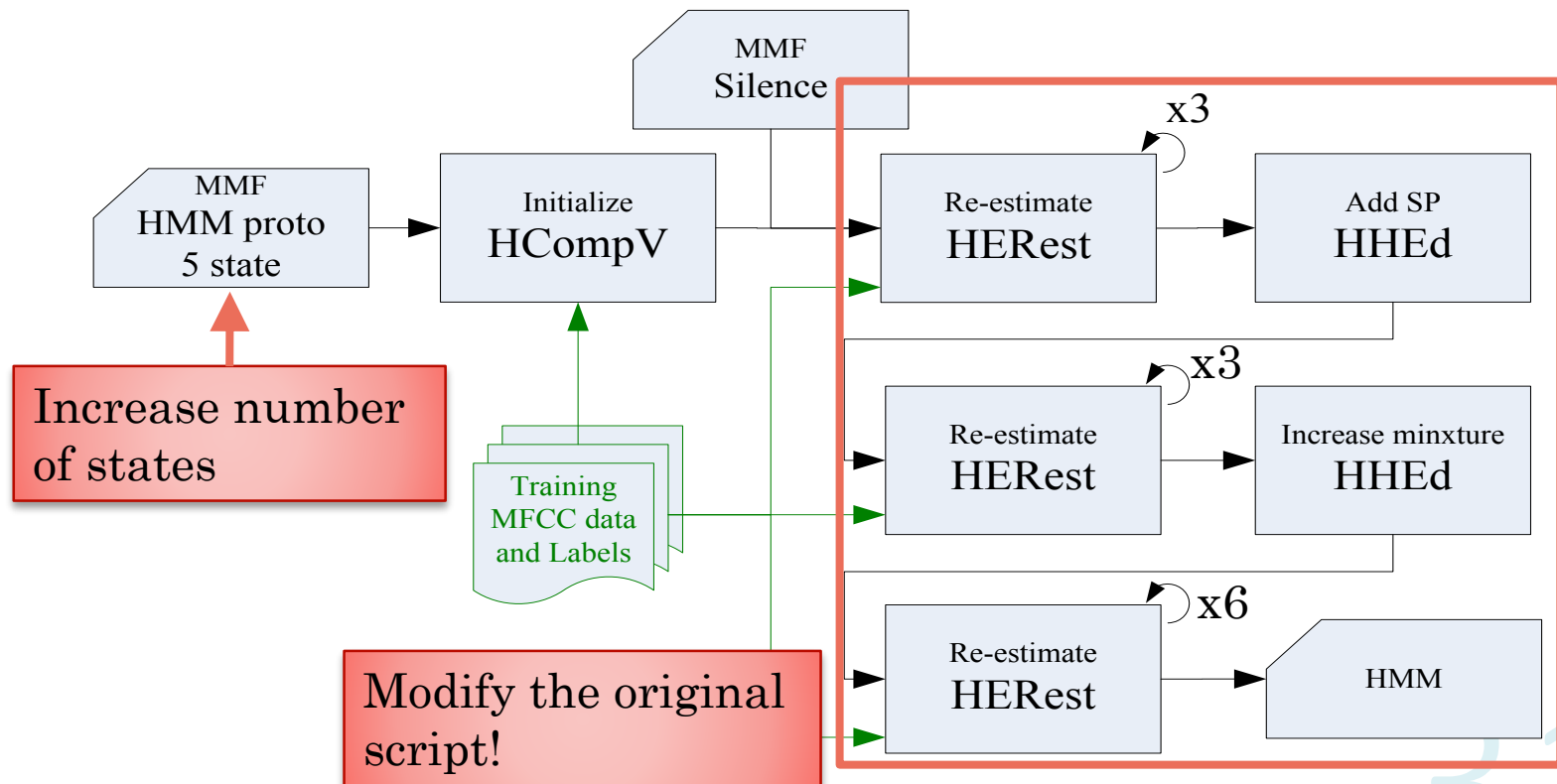
- `PATH=$PATH:“~/XXXX/XXXX”`

## In case shell script is not permitted to run...

- `chmod 744 XXXX.sh`

# Part 2 (40%) – Improve Recognition Accuracy

Acc > 95% for full credit ; 90~95% for partial credit



# Attention(1)

- Executing **03\_training.sh** twice is **different** from doubling the number of training iterations. To increase the number of training iterations, please modify the script, rather than run it many times.
- If you executed **03\_training.sh** more than once, you will get some penalty.

```
#####  
# re-adjust mean, var  
echo "step 00 [HErest]: adjust mean, var..."  
for i in 0 1 2 ;  
do  
    echo "iteration $i"  
    HErest -C $config -I $label \  
        -t 250.0 150.0 1000.0 -S $data_list \  
        -H $macro -H $model -M $mmf_dir $model_list  
done
```



# Attention(2)

- Every time you modified **any parameter or file**, you should run **oo\_clean\_all.sh** to remove all the files that were produced before, and restart all the procedures. If not, the new settings will be performed on the previous files, and hence you will be not able to analyze the new results.

(Of course, you should record your current results before starting the next experiment.)

```
#!/bin/bash

if [ -d MFCC/ ]; then
    echo "rm MFCC/ -r"
    rm MFCC/ -r
    echo "mkdir MFCC"
    mkdir MFCC
fi

if [ -d hmm/ ]; then
    echo "rm hmm/ -r"
    rm hmm/ -r
    echo "mkdir hmm"
    mkdir hmm
fi

if [ -d result/ ]; then
    echo "rm result/ -r"
    rm result/ -r
    echo "mkdir result"
    mkdir result
fi

cd bin; make clean; cd ..
```

## Part 3 (20%)

Write a report describing your training process and accuracy.

- Number of states, Gaussian mixtures, iterations, ...
- How some changes effect the performance
- Other interesting discoveries

Well-written report may get +10% bonus.

# Submission Requirements

## 4 shell scripts

- your modified 01~04\_XXXX.sh

## 1 accuracy file

- with only your best accuracy (The baseline result is not needed.)

## proto

- your modified hmm prototype

## mix2\_10.hed

- your modified file which specifies the number of GMMs of each state

## 1 report (in PDF format)

- the filename should be hw2-1\_bXXXXXXXXX.pdf (your student ID)

Put above 8 files in a folder (named after your student ID), and compress into 1 zip file and upload it to Ceiba.

- 10% of the final score will be taken off for each day of late submission

# If you have any problem...

Check for hints in the shell scripts.

Check the HTK book.

Ask friends who are familiar with Linux commands or Cygwin.

- This should solve all your technical problems.

Contact the TA by email.

But please allow a few days to respond.

盧宏宗 [r03922011@ntu.edu.tw](mailto:r03922011@ntu.edu.tw)