

Using HTK Hidden Markov Model Toolkit

Chi-Yueh Lin
2006/07/17



HTK

- HTK (Hidden Markov Model Toolkit) 是由劍橋大學電機系所發展出來的隱藏式馬可夫模型(HMM)開發工具，藉由HTK可以方便且快速地發展出一套以HMM為架構的語音辨識系統。
- 命令列

HTK

- 在架構一個語音辨識系統之前，事先要準備的資料不外乎以下幾種：
 - 想要辨識的單字跟句型
 - 備妥訓練語料之語音資料庫(Speech Corpus)以及標註檔(Transcription/Label files)
 - 使用字層次的模型，還是音素層次的模型

HTK

- 訓練階段
 - HCopy – 萃取語音參數
 - HInit & HRest – 訓練模型(Label)
 - HCompV & HERest -訓練模型(Transcription)
 - HHEd – 分裂高斯混合模型
- 辨認階段
 - HParse – 產生欲辨識句子的文法結構
 - HVite – 辨識語音

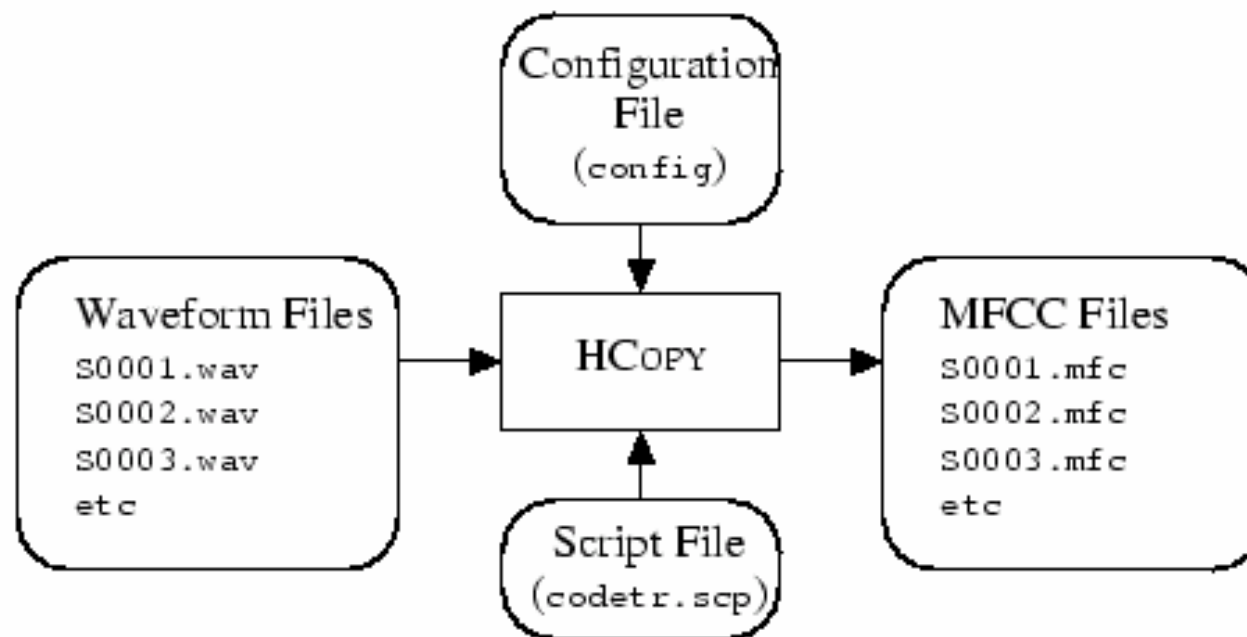


訓練階段

HCopy - abstract


- HCopy 是 HTK 轉換資料型態的指令，它可以
 - 將波形資料轉換成特徵參數
 - 參數間的互相轉換
 - 萃取某一段波形或參數
- 最常用的功能是用來將波形資料轉換成特徵參數，如把**PCM**編碼的波形資料轉成**MFCC**

HCOPY – block diagram



備妥語音資料，以及撰寫 config 及 script 檔

HCopy – config file

- NATURALREADORDER=TRUE
 - NATURALWRITEORDER=TRUE
 - SOURCEFORMAT=NOHEAD
 - SOURCEKIND=WAVEFORM
 - TARGETKIND=MFCC_E_D_A
 - SOURCERATE=1250
 - TARGETRATE=100000
 - WINDOWSIZE=250000
 - ZMEANSOURCE=T
 - USEHAMMING=T
 - PREEMCOEF=0.97
 - NUMCHANS=31
 - USEPOWER=F
 - NUMCEPS=13
 - ENORMALISE=T
 - LOFREQ=200
 - HIFREQ=3500
 - DELTAWINDOW=2
 - ACCWINDOW=2
 - SAVECOMPRESSED=F
 - SAVEWITHCRC=F
- 
- 1
- 2
- 3
- 4
- 5
- 6
- 7

HCopy – config file (1)(2)

- NATURALREADORDER=TRUE
- NATURALWRITEORDER=TRUE
 - 這兩項設定跟機器、作業系統有關，一般都設定為 TRUE
- SOURCEFORMAT=NOHEAD
 - 來源檔是沒有檔頭的
- SOURCEKIND=WAVEFORM
 - 來源檔是波形
- TARGETKIND=MFCC_E_D_A
 - 目的檔是MFCC，以及 energy (E), delta (D), delta-delta (A)
 - SOURCEFORMAT 會自動設定為HTK

HCoppy – config file (3)

- SOURCERATE=1250
 - 來源檔取樣間隔時間為 0.125 ms
- TARGETRATE=100000
 - 音窗的間隔為 10 ms
- WINDOWSIZE=250000
 - 音窗的長度為 25 ms
- **注意：**在HTK當中，時間單位為 100ns

HCOPY – config file (4)

- ZMEANSOURCE=T
 - 將來源資料取 zero mean，即去掉DC值
- USEHAMMING=T
 - 使用 Hamming Window
- PREEMCOEF=0.97
 - 預強調係數為 0.97

HCopv – config file (5)

- NUMCHANS=31
 - 在 Mel 刻度下等分成31個頻帶
- USEPOWER=F
 - 不使用c(0)參數
- NUMCEPS=13
 - 最後使用13階MFCC參數

HCOPY – config file (6)

- ENORMALISE=T
 - 音框的能量值做正規化
- LOFREQ=200
 - 設定頻帶的下截止頻率
- HIFREQ=3500
 - 設定頻帶的上截止頻率
- DELTAWINDOW=2
- ACCWINDOW=2
 - 設定 delta 和 delta-delta 的計算參數

電話線品質，如果為麥克風錄音，LOFREQ設為0；HIFREQ設為8000

HCopy – config file (7)

- **SAVECOMPRESSED=F**
 - HTK可以壓縮產生出來的參數檔，一般設為False
- **SAVEWITHCRC=F**
 - HTK可以在產生的參數檔後頭加上CRC檢查碼，一般設為False

HCopy – script file

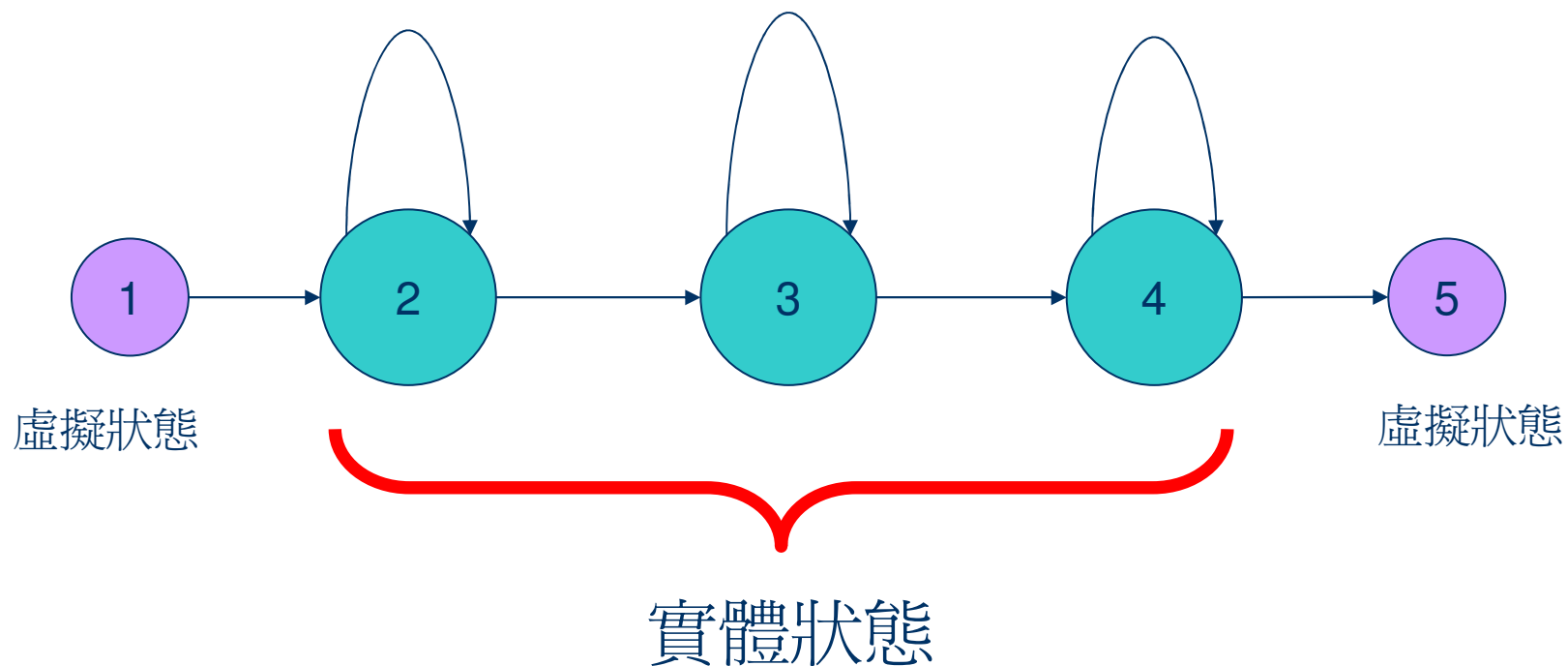
- Script 檔是告訴 HCopy 來源檔在哪裡，以及產生出來的目的檔要放在哪裡。
- 範例
/source/dir/001.wav /target/dir/001.mfc
/source/dir/002.wav /target/dir/002.mfc
/source/dir/003.wav /target/dir/003.mfc
- 左右兩段文字，左邊放source，右邊放target

HCopy - Usage

- `$ Hcopy -T 1 -C XXX.config -S XXX.script`
 - `-T 1` 是設 Trace Level 為1，這樣可以在螢幕上看到hcopy的過程
 - `-C` (大寫) 讀入 config 檔案，檔名及副檔名可以自訂，一般副檔名都設為 config 或 cfg
 - `-S` (大寫) 讀入 script 檔案

HMM - definition

3-state Left-to-Right HMM



HMM - definition

- 設定初始HMM模型檔
- ~o 設定 observation
- <VecSize> 39
 - 向量維度為 39
- <MFCC_0_D_A>
 - 參數類型，要跟 hcopy 的 config 檔案裡設定的相同
- ~h “proto” 這是 HMM 的 prototype，之後的HMM都由這個proto所產生出來

```
~o <VecSize> 39 <MFCC_0_D_A>
~h "proto"
<BeginHMM>
  <NumStates> 5
  <State> 2
    <Mean> 39
      0.0 0.0 0.0 ...
    <Variance> 39
      1.0 1.0 1.0 ...
  <State> 3
    <Mean> 39
      0.0 0.0 0.0 ...
    <Variance> 39
      1.0 1.0 1.0 ...
  <State> 4
    <Mean> 39
      0.0 0.0 0.0 ...
    <Variance> 39
      1.0 1.0 1.0 ...
  <TransP> 5
    0.0 1.0 0.0 0.0 0.0
    0.0 0.6 0.4 0.0 0.0
    0.0 0.0 0.6 0.4 0.0
    0.0 0.0 0.0 0.7 0.3
    0.0 0.0 0.0 0.0 0.0
<EndHMM>
```

HMM - definition

- HMM的設定由 <BeginHMM> 和 <EndHMM> 所包圍
- <NumStates> 5
 - 共有5個狀態，3個實體+2個虛擬
- 設定每一個實體狀態的 <Mean> 和 <Variance>
 - <Mean> 39
 - 39 個 0.0
 - <Variance> 39
 - 39 個 1.0

HMM - definition

- 設定 Transition Matrix <TransP>

<TransP> 5

0.0 1.0 0.0 0.0 0.0

0.0 0.6 0.4 0.0 0.0

0.0 0.0 0.6 0.4 0.0

0.0 0.0 0.0 0.7 0.3

0.0 0.0 0.0 0.0 0.0

← 虛擬狀態

← 虛擬狀態

1. 第一行一定是 0.0 1.0 開頭
2. 最後一行全部都為 0.0
3. 中間實體狀態的轉移機率初始值並不重要，不為0即可

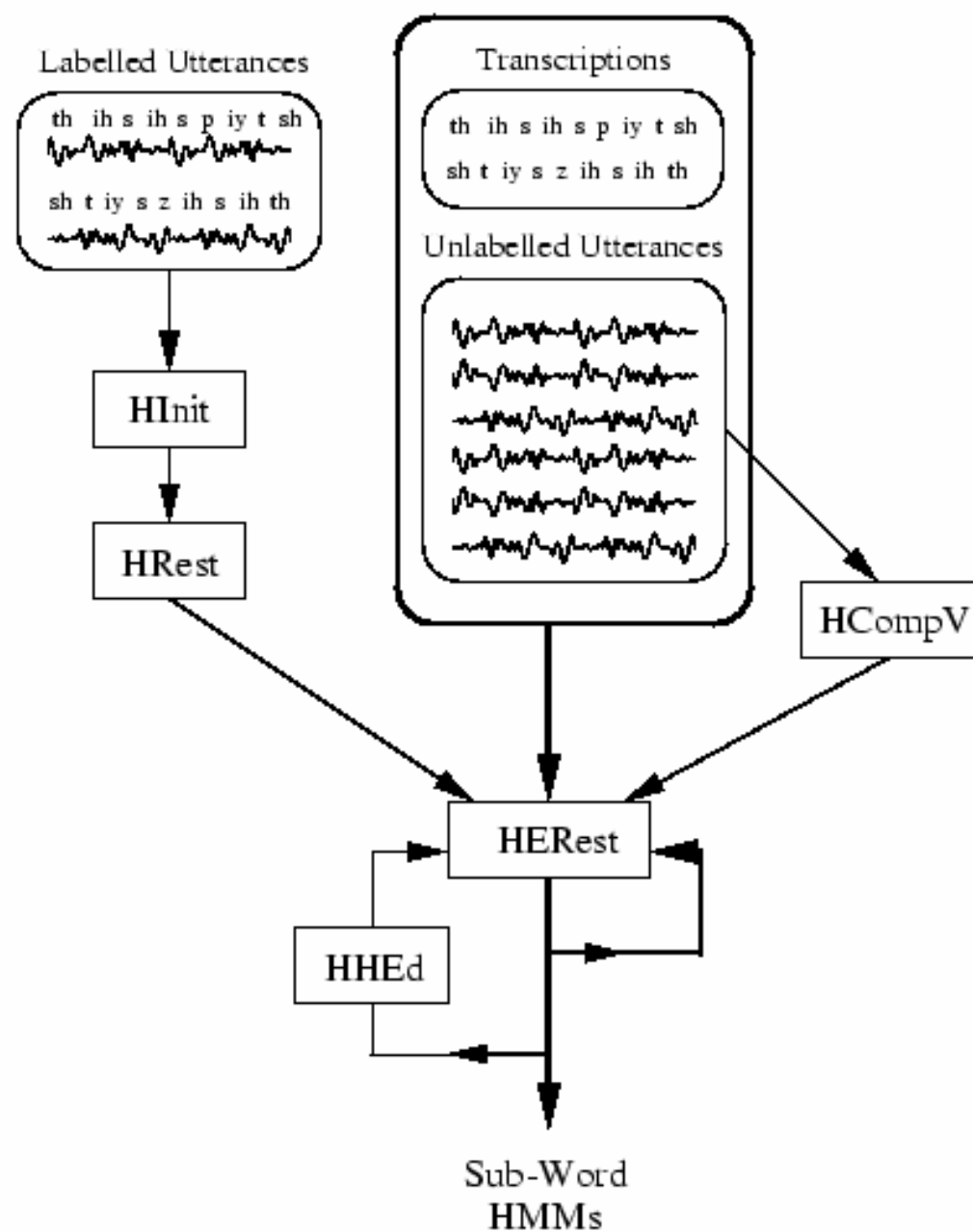
HMM - Training

- 在進行模型訓練之前，重要的是確定手邊的標註檔是屬於何種類型
 - **Label** : 標註檔含有時間資訊
 - 0 180000 sil
 - 180000 450000 voc
 - 450000 610000 voc
 - **Transcription** : 標註檔不含時間資訊
 - sil
 - voc
 - voc

HMM - Training

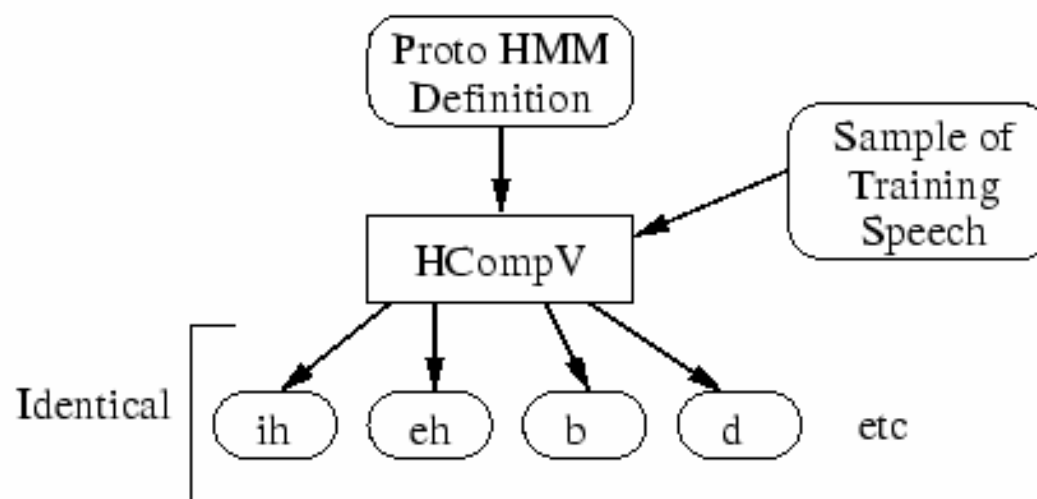
- 手邊的是 Label 檔
 - 用 **Hinit** + **HRest** 指令
- 手邊的是 Transcription 檔
 - 用 **HCompV** + **HERest** 指令
- 兩者都有？
 - 先用 Hinit 和 HRest 處理有 Label 的檔案，再用 HCompV 和 HERest 處理 Transcription 檔案

HMM -



HMM - Training

- HCompV
 - 在 HTK 中稱之為 flat start
 - 將語料等分成數個段落，再計算一個統一的 Mean 和 Variance



HMM - Training

- HCompV
 - **\$ HCompV -C config -S script -M dir1 -l aa -o aa -I label.mlf proto**
 - **config** 只需放以下兩行
 - SOURCEFORMAT=HTK
 - SOURCEKIND=MFCC_E_D_A
 - **script** 放要拿來訓練的檔案列表
 - **-l aa -o aa** 估計aa的HMM檔案，並存成aa
 - **-M dir1** 指定要存放訓練檔案的目錄
 - **-I** 指定master label file，亦可用 **-L** 指定目錄
 - **proto** 先前設定的 HMM prototype 檔

HMM - Training

- HERest

- **\$ HERest -C config -S script -I label.mlf -d dir1 -M dir2 hmmlist**
 - **-I** 指定master label file，亦可用 **-L** 指定目錄
 - **-d** 尋找HMM檔案的目錄(從HCompV產生出來的)
 - **-M** 指定儲存新HMM的目錄
 - **hmmlist** 裡面放所有HMM的名稱
- 執行完後，會在dir2產生一個新的HMM Macro File

HMM – mixture incrementing

- HHEd
 - 編寫 split.hed 檔
 - MU 16 {*.state[2-4].mix}
 - 每個HMM的實體狀態(2~4)分裂成16個mixture
 - MU 8 {aa.state[2-4].mix}
 - aa的HMM實體狀態(2~4)分裂成8個mixture
 - **\$ HHed -M mix2 -w newHMM -d mix1 split.hed**
 - **-M mix2** 分裂完的HMM檔會存在 mix2 目錄
 - **-w newHMM** 分裂完的HMM檔名
 - **-d mix1** 在mix1目錄尋找原始的HMM檔

HMM – mixture incrementing

- HHEd

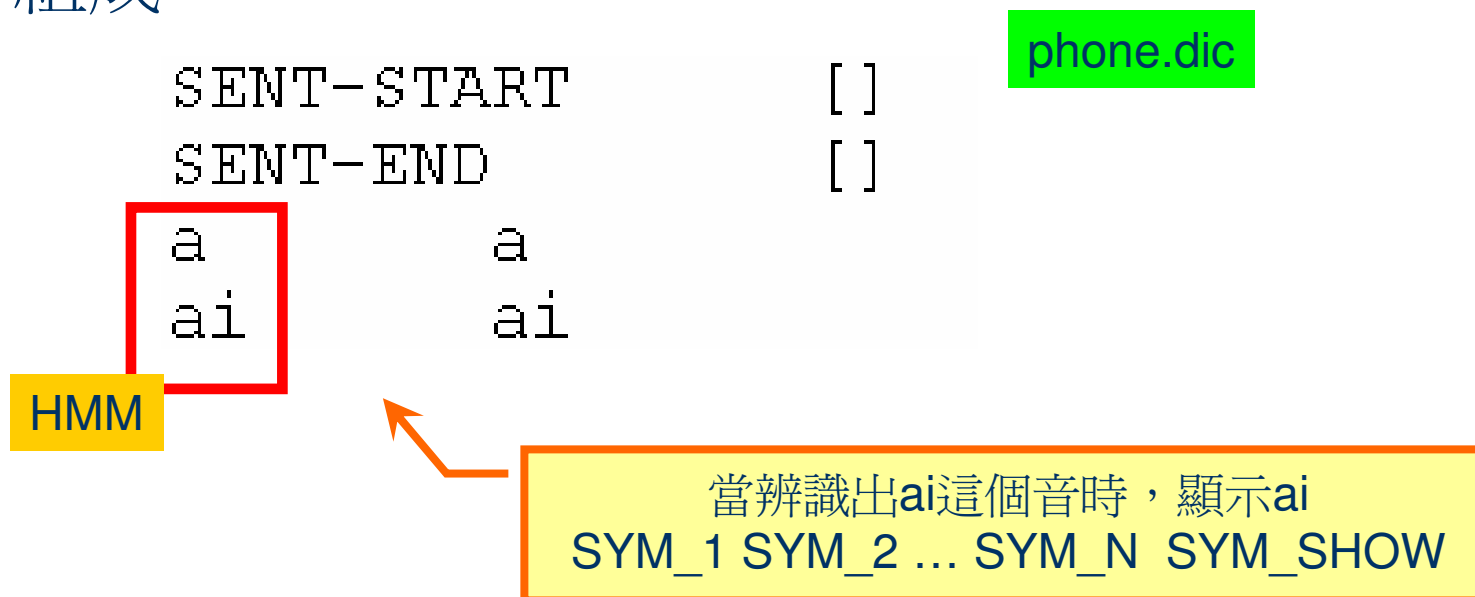
- 若用文字編輯器開啓產生出來的 `newHMM`，可以發現每個 `mixture` 的權重被設成相等，因此還要執行 `HERest` 數次，直到相似度值漸漸收斂為止。
- 收斂完的模型，即是最終的 `HMM` 模型，這時取名為 `HMM.model`



辨識階段

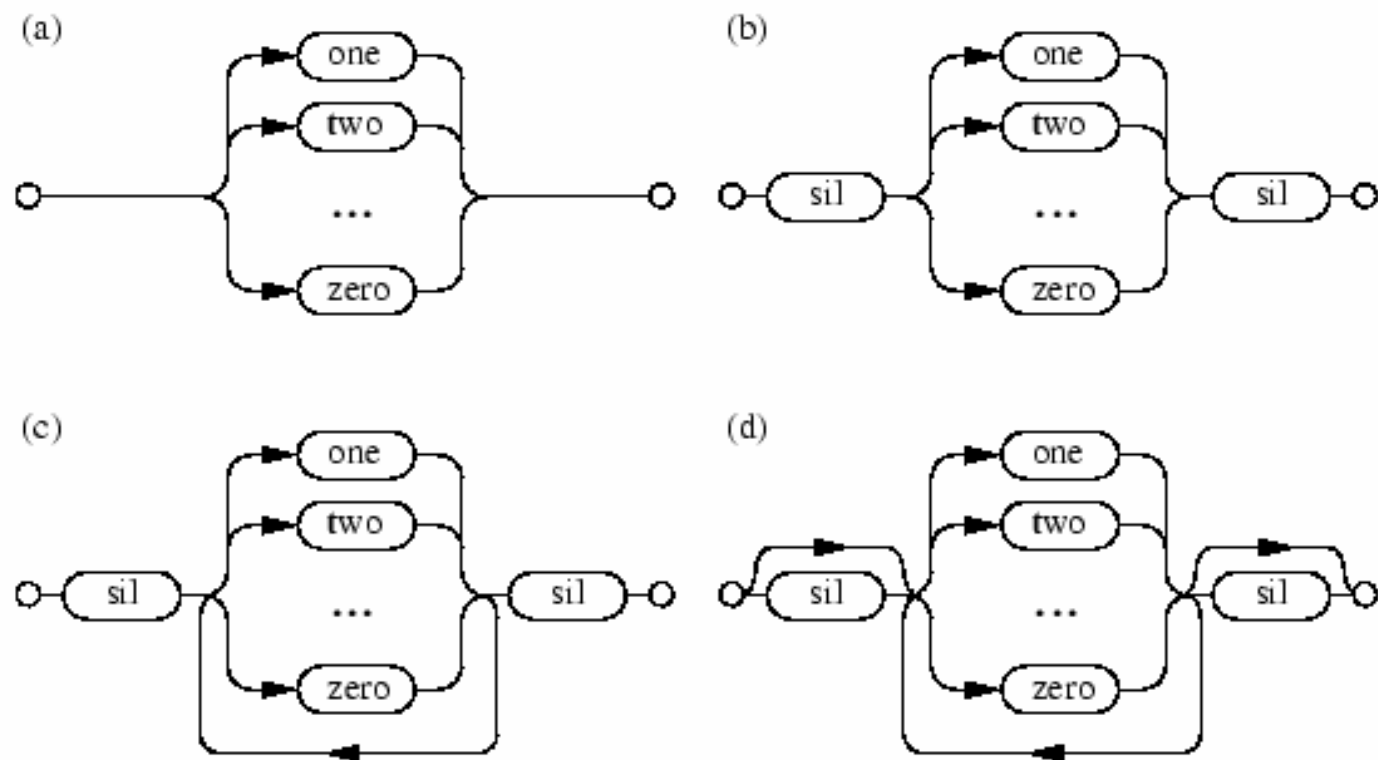
Recognition – Dictionary & WordNet

- Dictionary 的用意在於告訴HTK，你想要辨識出來的音是哪些？又這些音是由哪些HMM所組成。



Recognition – Dictionary & WordNet

- WordNet 告知辨識句子的文法結構



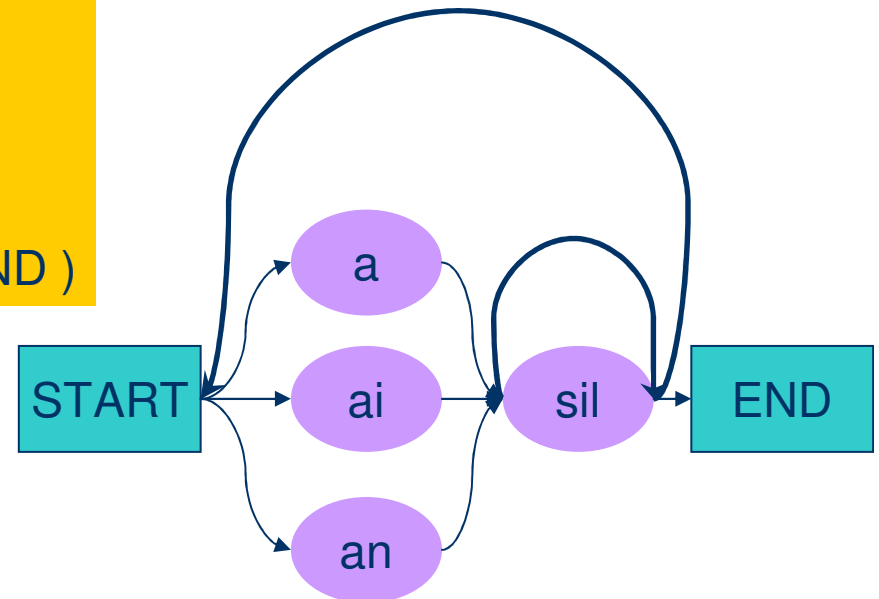
Recognition – Dictionary & WordNet

- WordNet

- 簡單的 WordNet, 可手動新增
- 複雜的 WordNet, 用 HParse 指令

```
$ci_phone =  
a      |  
ai     |  
an     ;  
( SENT-START < $ci_phone [sil] > SENT-END )
```

\$ HParse phone.syn phone.net
產生 phone.net 這個 WordNet 檔

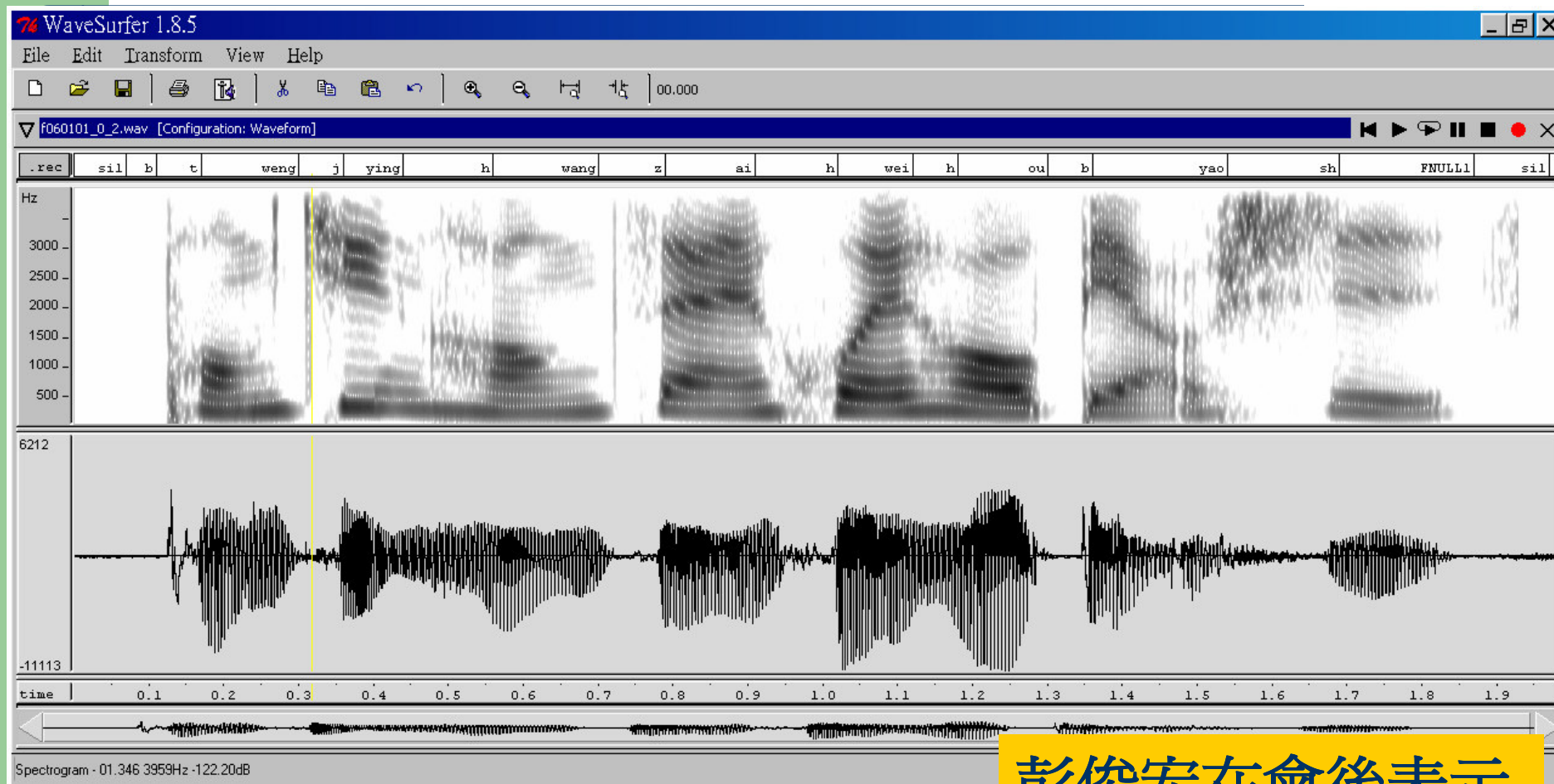


Recognition - HVite

- 當 WordNet、Dictionary 準備妥當，可以使用 HVite 來辨識
 - **\$ HVite -C config -w phone.net -H HMM.model
phone.dic hmmlist XXX.wav**
- 執行完後會產生 XXX.rec 檔，即為辨識結果

```
0 700000 sil -428.991882
700000 1100000 b -276.103149
1100000 1700000 t -493.964966
1700000 3000000 weng -1099.999023
```

Recognition - WaveSurfer



彭俊宏在會後表示

Recognition - WaveSurfer

- 彭 b t weng %前面爆破音”ㄅ”辨成”ㄅ”+”ㄨ”兩個音
- 俊 j ying %辨成比較像“靜”的音
- 宏 h wang %辨成比較像“黃”的音
- 在 z ai
- 會 h wei
- 後 h ou
- 表 b yao
- 示 sh FNULL1

Application

