行政院國家科學委員會專題研究計畫成果報告

新世代自動語音辨識技術之研究－

子計畫二：語音、韻律之屬性與事件偵測之研究

執行期限：94 年 8 月 1 日至 97 年 7 月 31 日

計畫編號： NSC94－2213－E－009－134

NSC95－2221－E－009－255

NSC96－2221－E－009－041

主　持　人 ： 王逸如　國立交通大學電信工程系

共同主持人 ： 廖元甫　國立台北大學電子工程系

執行單位： 國立交通大學電信工程系

中　華　民　國　97　年　10　月　30　日

摘要

在傳統語音辨認方法中，通常只使用語音的頻譜參數作辨認。但是在新世代自動語音辨識技術中，將結合語音與語言學知識，以多種語音屬性（attribution）與語音事件（event）偵測器群，盡可能從語音信號中擷取各種聲學、韻律及語言相關的訊息，在交與後級『語音事件及相關知識整合』及『語音證據確認』單元，做語音辨認甚至於語意瞭解，以期突破傳統隱藏式馬可夫模型（hidden Markov model, HMM）方式的困境。本計畫中即擬進行國語語音之各種語音屬性、音節邊界、基頻軌跡、韻律資訊之偵測研究，以做為新世代自動語音辨識系統之前端處理器。研究重點如下：

1. 中文語音屬性（attribution）與各種語音事件（event），包括偵測發音方法(articulation manner) ，發音部位(articulation position)與其他語音特徵參數（distinctive feature）。
2. 中文音節界標（boundary landmark）偵測器，提供後級正確時序訊號。
3. 中文基頻軌跡偵測器，包括新的求取方法與軌跡特徵。
4. 中文音調與韻律訊息偵測器。

關鍵詞：新世代自動語音辨識系統，語音屬性偵測，語音事件偵測，基頻特徵偵測。

Abstract

In this project, various speech attribution-, speech event-, syllable boundary-, pitch contour- and prosodic information -detectors will be studied and act as the front-end of the next-generation automatic speech recognizer. The focuses of the research include:

(1) Mandarin speech attribution, event and other distinctive feature detectors including articulation manner and articulation position
(2) Mandarin syllable boundary detector to provide syllable timing information
(3) Mandarin pitch contour extraction and feature
(4) prosodic information detection and tone recognizer

Keywords: next generation ASR, speech attribution detection, speech event detection, pitch contour feature

目錄

一、 緣由與目的

回顧現今自動語音辨識技術，大詞彙的連續語音辨識(large vocabulary continuous speech recognition, LVCSR)技術被開發出來，所依賴的就是大量的語音資料與語言資料。各個國家都針對其所用的語言進行大量語音與語言資料的收集，就特定的一些應用領域發展語音辨識系統，例如聽寫機(voice dictation machine)、交談系統(conversational system)、口語文件擷取(spoken document retrieval)、口語翻譯(spoken language translation)等。但大家發現現有的這些技術還是不夠好，仍無法與人類辨識語音的能力相比，而現有技術的進步空間有限，為了將來語音辨識技術的發展，近年國際上已不斷有學者主張，應該回頭將語音與語言的知識帶進來，建立一個以知識為基礎(knowledge-based)加上資料驅動的(data-driven)模式，開放測試平台，共享一個合作的設計與評量機制，將自動語音辨認推向新一代的技術[Lee, 2004]。在[Lee, 2004]中，為新一代自動語音辨識技術建立之平台及架構圖如圖 1-1 所示。

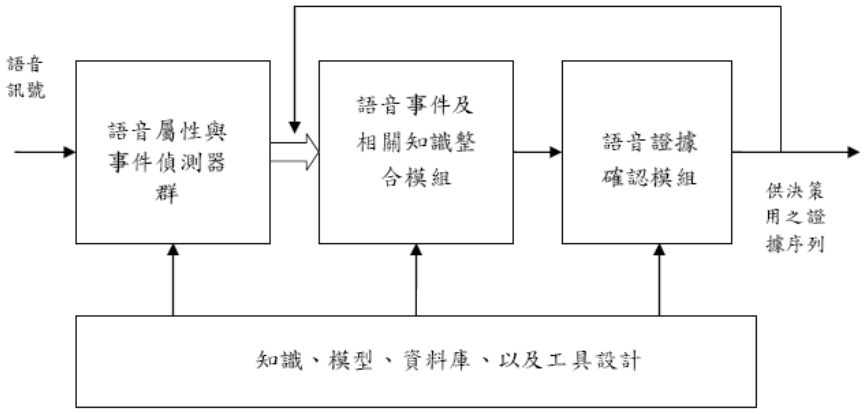


圖 1-1:新世代自動語音辨識技術架構圖。

在本子計畫中，我們將致力研究新世代自動語音辨識技術架構(圖)中之語音屬性與事件之偵測器。在語音屬性與事件之偵測器中對於語音訊號，不只是抽取語音特徵，而且要偵測某一時段中語音的屬性。因此除了傳統的聲學特徵參數，還要依據語音學或語言學的知識，抽取韻律相關的訊息與發音方式的訊息，協助區辨容易混淆的聲音。利用語音的特徵與屬性的發生，描述一段語音事件(event)的發生，而從事件序列來做語音辨識的決策。也就是以事件序列，判斷是否有某一段語音的發生。抽取不同的語音特徵與屬性，可以偵測出若干個事件序列，將

更有助於作正確的判斷，提高語音辨識的正確率。

接著，將計畫中之各行就結果分類於下面章節。

二、 以高斯混合模型為架構的語音屬性偵測器

因為製作 NG-ASR 的語音屬性偵測器必須要有一個語料庫具有 phone-level transcription 資料。在國語語料中缺乏這樣的語料庫，所以計畫中先從英文語料庫進行研究。本節內容為介紹使用英文語料庫 TIMIT Corpus，以及使用 HTK(Hidden Markov Model Toolkit)，建立 GMM-Based 的英文語音屬性之偵測器，其中語音屬性包含發音方法（Articulation manner）與發音位置(Articulation position)。且這兩類偵測器輸出將會有每個音框(frame)是屬於何種類別(class)的機率值，以提供 NG-ASR 的第二級作事件的整合。

1. 語音資料庫

首先由於我們要製作英文語音屬性的偵測器，因此我們採用的語料庫為 TIMIT Corpus，語料內容為 2342 句平衡語料，由分佈在美國八個不同方言的地區共 630 位語者，每人錄製 10 句，共有 6300 句語音，其中 438 位男性、192 位女性。並以其中 4620 句、語料長度總和約為 3 小時 49 分 10 秒的語音訊號作為訓練語料，另外 1680 句、語料長度總和約為 1 小時 23 分 51 秒的語音，作為測試語料。語料的音訊格式為 PCM，取樣頻率為 16 kHz，位元解析度為 16 bits，檔頭為 1024 bytes (original : 12 bytes)。

TIMIT Corpus [Garofolo, 1993]已經有 manual phonetic transcription，所以可以由 transcription 取得 phone 及語音屬性的參考答案。而其 transcription 的起始以及終止時間的單位為取樣的點數，因此在下面的實驗中會將 manual transcription 轉換成以 100ns 為單位的時間資訊的 labeling 資訊。

在英文發音方法部份，TIMIT 附有英文發音方法的分類表，因此我們將 TIMIT 原有的 61 個 phonemes 依照其分類表分成七類發音方法。

表 2-1: TIMIT Corpus 發音方法及部位分類表。

	Bilabial	Lab-dent	Dental	Alveolar	Velar	Glottal	Rhotic	Front	Central	back
Stop	b, p			d, t, dx	g, k	q				
Nasal	m, em			n, en, nx	ng, eng					
Fricative		f, v	th, dh	s, z	sh, zh					

Glide						hh	r	y	l, hv, el	w
affricate				jh, ch						
vowel							er, axr	iy, ih, eh, ey, ae, ay, ix	aa, aw, ax, ax-h	ah, ao, oy, ow, uh, uw, ux

2. 特徵參數

在語音特徵參數抽取部分， 我們依然採用傳統的 MFCC 參數，以 32ms 為單位取一音框，每隔 10ms 重取音框。每一音框，包含 12 維的倒頻譜參數，12 維的一階差量倒頻譜參數，12 維的二階差量倒頻譜參數，1 維的一階差量對數能量以及 1 維的二階差量對數能量，共計 38 維。

3. 高斯混合模型

高斯混合模型是以高斯機率分佈為主體，包含多個高斯機率分佈，下面式子(2.1)為 N 個基本高斯機率分佈加權和(weighted summation)之高斯混合模型:

$$p(x|\theta) = \sum_{i=1}^{N} C_i \cdot N(\mu_i, \Sigma_i) \qquad (2.1)$$

$$N(\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} \exp[-\frac{1}{2}(x-\mu_i)^T(\Sigma_i)^{-1}(x-\mu_i)] \qquad (2.2)$$

$$\theta = \{(C_i, \mu_i, \Sigma_i), 1 \le i \le N\} \qquad (2.3)$$

其中 x 為一 D 維度大小之特徵參數向量， θ為高斯混合模型，$N(\mu_i, \Sigma_i)$ 為 GMM 中各高斯分佈之機率密度函數，μ_i 為平均向量(Mean Vector)，Σ_i 為共變異矩陣(Covariance Matrix)，C_i 為混合加權值(Weight)，且須滿足 $\sum_{i=1}^{N} C_i = 1$。而在接

下來的實驗中，我們假設共變異矩陣為一對角矩陣(Diagonal Matrix)。

4. 高斯混合模型的發音方法、發音位置偵測器之效能

由於我們已把每一個發音方法、發音位置偵測器的兩種 model 的 mixture 數均升至 256，與 128，因此我們將看看對於測試語料，各種偵測器的偵測效果。

在此每一個發音方法、發音位置偵測器皆將利用最大事後機率法則(Maximum a Posteriori, Criterion)，去對測試語料每一個 frame 偵測是否為所要偵測的種類。下式為 MAP Criterion：

$$\frac{\mathrm{p}(x|\theta)}{\mathrm{p}(x|\hat{\theta})} \geq \frac{\mathrm{p}(\hat{\theta})}{\mathrm{p}(\theta)} = \zeta \quad , \quad x \to \theta \tag{2.7}$$

And, where $\mathrm{p}(\theta) + \mathrm{p}(\hat{\theta}) = 1$。

其中 θ 為 target model，而 $\hat{\theta}$ 為 anti-model，x 為每一個 frame 的特徵參數向量，ζ 為 threshold。若(2.7)式成立，則將此 frame 判定為 target，反之則否。

首先 $p(\theta)$ 與 $p(\hat{\theta})$ 為各個 class 的 targe 與 anti-target 的事先機率，而在 threshold 這個值的選定，我們利用 TIMIT 訓練語料中，每個 class 的 target 與 anti-target 的出現音框總數當作其事先機率，兩者相除便得到一個 threshold 的值，再利用其值，對 TIMIT 的測試語料作偵測，可以得到 False Alarm 以及 False Reject 的值，而後再行調整 Theshold，使其可以得到 False Alarm 以及 False Reject 的其它值，最後可將所有值畫出一個 FA-FR 的曲線圖。並且可以得到 FA Rate 等於 FR Rate 時的 EER(Equal Error Rate)。

以下為 FA、FR、Frame Error Rate 的定義 ：

FA Rate = # of FAs / total # of non-target (2.8)
FR Rate = # of FRs / total # of targets (2.9)
Frame Error Rate = (# of FAs + # of FRs)/ total # of labels (2.10)

其中 FA=False Alarm，FR=False Reject

下兩表為 TIMIT Corpus 相關統計資料：

表 2-2: TIMIT Corpus 發音方法出現次數，與平均 frame 數等的統計資料。

	TIMIT Traning Data					TIMIT Testing Data				
	total files : 4620 total frames : 1416713					total files : 1680 total frames : 513526				
Manner	Times	Frame amount	Min Frame	Average frame	Max Frame	Times	Frame amount	Min Frame	Average frame	Max Frame
Vowel	57463	549896	<1	9.57	43	20911	202289	1	9.67	48
Fricative	21424	195416	<1	9.12	38	7724	71036	<1	9.20	33
Stop	25871	106575	<1	4.12	28	9176	37755	<1	4.11	30
Nasal	14157	80454	<1	5.68	26	5104	29043	<1	5.69	22
Glide	20257	129666	<1	6.40	25	7822	51199	1	6.55	24
Silence	35877	340525	<1	9.48	300	12777	117734	<1	9.20	464
Affricate	2031	14181	2	6.98	34	631	4470	2	7.08	23

表 2-3: TIMIT Corpus 發音位置出現次數，與平均 frame 數等的統計資料。

	TIMIT Traning Data					TIMIT Testing Data				
	total files : 4620 total frames : 1416713					total files : 1680 total frames : 513526				
Position	Times	Frame amount	Min Frame	Average frame	Max Frame	Times	Frame amount	Min Frame	Average frame	Max Frame
bilabial	8796	40182	<1	4.57	26	3416	15486	<1	4.53	20
labdent	4210	34866	1	8.28	31	1622	13638	1	8.41	30
dental	3577	17536	<1	4.90	31	1320	6373	<1	4.83	22
alveolar	32662	214114	<1	6.56	38	11375	75028	<1	6.60	33
velar	10648	66628	<1	6.26	30	3658	23504	1	6.43	27
glottal	4547	29533	1	6.50	28	1600	10671	<1	6.67	30
rhotic	11992	91398	<1	7.62	34	4708	36827	1	7.82	37
front	34883	316266	1	9.07	43	12503	114284	1	9.14	39
central	15684	119361	<1	7.61	42	5881	45035	1	7.66	48
back	14204	146304	1	10.30	43	5285	54946	<1	10.40	39
silence	35877	340525	<1	9.49	300	12777	117575	<1	9.20	464

5. 高斯混合模型的發音方法之效能

下表為 GMM-Based 發音方法偵測器國外學者用不同偵測器架構所做出來的性能[Lee, 2005]作比較。

表 2-4: GMM-Based mixture256 與其它偵測架構的發音方法偵測效能比較。

Equal ErrorRate(%)	Baseline(GMM)	ANN*	HMM	SEG_MCE
Vowel	12.3	9.0	1.7	1.8
Fricative	10.0	11.3	6.4	3.6
Stop	16.7	14.5	9.9	5.4
Nasal	8.7	12.2	11.2	5.4
Glide (Approximant)	16.3	15.9	8.0	6.1
Sil	9.7	3.7	2.1	0.8
Affricate	7.2			

在表 2-4 的 ANN 部份的作法，各個偵測器其網路輸入部分有 9 個 frame，每個 frame 有 13 維的特徵向量(12MFCCs+energy)，因此共有 117 個輸入節點(input nodes)，而 frame rate 為 10ms。且有一個隱藏層其中有 100 個節點。輸出部份僅有一個節點。偵測器輸出的 threshold 值為 0.5。

由表 2-4 可以看出，以 GMM-Based 的 Fricative 與 Nasal 偵測器，其效能較佳於 ANN，尤其是 Nasal 偵測器改善了約 3%，其他的偵測器均較 ANN 差，尤其是 silence 偵測器差了 6%。另外由表 2-4 可以看出，以 HMM Segment-Based 做發音方法偵測器普遍比 GMM 以及 ANN 架構好，這提供了我們未來在做其他偵測器一個參考的依據。

6. 高斯混合模型的發音位置偵測器之效能

在發音位置的效能部分，由於我們將 target model 與 anti-model 的 mixture 數目皆訓練至 128，在此我們隨便挑一個發音位置偵測器來觀察其在 mixture 64 與 mixture128 的 FA-FR chart 的差異。

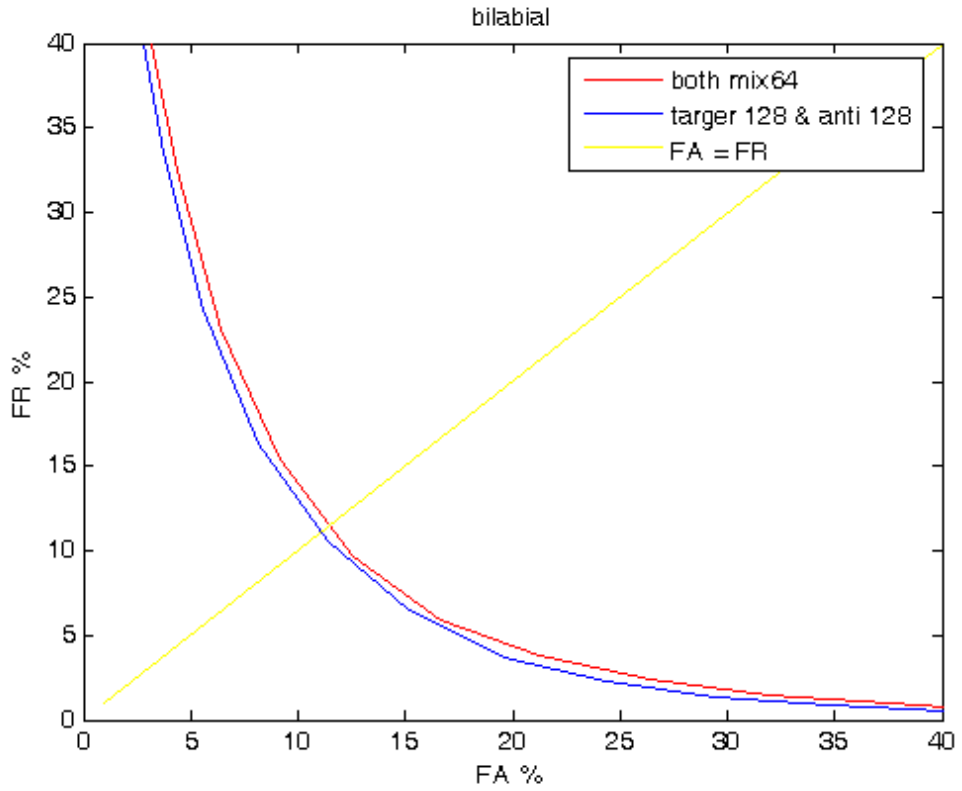


圖 2.1: 發音位置偵測器 “bilabial” 之 FA-FR 圖。

由圖 2.1 可看出在 mixture 64 與 mixture 128 時，bilabial 偵測器的 FA-FR 曲線圖相當接近，且其兩者的 EER 差距不到 0.5%，因此在發音位置偵測器的部份我們取 mixture 數為 64。下表為 GMM-Based 發音位置偵測器的 EER。

表 2-5: GMM-Based mixture64 的發音位置偵測效能比較。

Equal ErrorRate(%)	Baseline(GMM)
Bilabial	12.2
Lab-dent	11.0
Dental	12.7
Alveolar	12.0
Velar	12.4
Glottal	18.3
Rhotic	9.4
Front	13.5
Central	17.7
Back	17.8

由表 2-5 可以看出幾乎全部的發音位置偵測器的 EER 均大於 10%以上、除了 Rhotic 偵測器、但也很接近 10%。其中以 Glottal、Central、Back 偵測器錯誤率皆大於 17%以上為最差。

7. 高斯混合模型的發音方法與發音位置偵測器混合之效能

在 NG-ASR 第一級的各個偵測器，其輸出為語音屬性或者事件的機率值，因此，由於我們現在已訓練出各個發音方法與發音位置偵測器的 target model 以及 anti-model，我們將可以求出每一個 frame 是否為 target 的機率。

$$\begin{aligned} p(\theta \mid x) &= \frac{p(x \mid \theta)\, p(\theta)}{p(x)} \\ &= \frac{p(x \mid \theta)\, p(\theta)}{p(\theta)\, p(x \mid \theta) + p(\hat{\theta})\, p(x \mid \hat{\theta})} \\ &= \frac{p(x \mid \theta)}{p(x \mid \theta) + \zeta\, p(x \mid \hat{\theta})} \end{aligned} \tag{2.11}$$

其中 $p(x \mid \theta)$ 、 $p(x \mid \hat{\theta})$ 皆為已知，如此便能得到每一個 frame 為 target 的機率。

由於我們欲將發音方法偵測器與發音位置偵測器的結果結合起來，因此我們將利用發音方法與發音位置獨立的特性，藉由機率得到合在一起的結果。

$$p(manner \cap position \mid x) = p(manner \mid x) \times p(position \mid x) \tag{2.12}$$

其中 $p(class \mid x)$ 為這個 frame 屬於這一個 class 的機率值

而由 TIMIT 語料發音方法與發音位置分類，可以得知六種發音方法與 10 種 positions 共有 21 種組合。而發音方法偵測器其 target model 與 anti target model 皆為 256 而發音位置偵測器其 target model 與 anti target model 皆為 64。

而觀察所有的發音方法結合發音位置的 FA-FR 曲線圖，可以得出 Stop + Velar、Nasal+ Bilabial、Fricative+ Alveolar、Fricative + Velar、Glide + Central、Glide + Back 共六種的發音方法與發音位置的組合，其 EER 比其原先結合的兩種較佳，而 Stop + Glottal、Fricative + Lab-dental、Fricative + Dental、Glide + Glottal、Vowel + Rhotic、Vowel + Front、Vowel +Central、Vowel +Back 共八種的發音方法與發音位置的組合其 EER 比其原先結合的兩種較差，而其餘的七種組合則其結合的 EER 則介於其原先結合的兩種之間。

8. 高斯混合模型的發音方法與發音位置偵測器之辨認

由於發音方法各偵測器的輸出有每個 frame 的機率值，因此我們藉由所有發音方法偵測器(包含 silence 偵測器)的每個 frame 機率值，依照其機率值大小對 TIMIT 測試語料作辨識，也就是說，每一個 frame 在哪一種發音方法偵測器的機率值最高，就決定該 frame 屬於該種發音方法。而發音位置偵測器之辨認其作法亦同於上述步驟。

下表為發音方法偵測器對 TIMIT 測試語料所做的發音方法辨識,，其各別發音方法的統計資料 ：

表 2-6: 發音方法偵測器所做的發音方法辨識，其各別發音方法的統計資料。

TIMIT Testing Data			
Total frame error rate = 24.533%			
manners	FA rate	FR rate	error_rate
Vowel	10.21 %	24.33 %	12.97 %
Fricative	24.01 %	20.35 %	6.30 %
Stop	50.43 %	42.79 %	7.43 %
Nasal	40.94 %	18.20 %	4.24 %
Glide	46.97 %	32.10 %	9.20 %
Silence	14.18 %	18.94 %	7.41 %
Affricate	71.04 %	47.99 %	1.53 %

下表為發音方法偵測器所作的發音方法辨識，其 confusion matrix。

表 2-7: 發音方法偵測器所作的發音方法辨識，其 confusion matrix。

recognize / ref	vowel	fricative	stop	nasal	glide	silence	affricate
vowel	75.67 %	2.20 %	3.70 %	4.12 %	12.03 %	1.93 %	0.35 %
fricative	1.37 %	79.65 %	5.67 %	1.43 %	1.04 %	7.20 %	3.64 %
stop	7.61 %	10.94 %	57.21 %	2.81 %	7.30 %	11.06 %	3.06 %
nasal	6.03 %	1.85 %	2.09 %	81.80 %	3.46 %	4.56 %	0.22 %
glide	19.29 %	1.72 %	5.34 %	3.35 %	67.90 %	2.09 %	0.30 %
silence	1.61 %	5.46 %	5.67 %	3.68 %	1.62 %	81.06 %	0.89 %

affricate	0.87 %	32.15 %	9.96 %	0.22 %	0.89 %	3.89 %	52.01 %

由表 2-6 可知以發音方法偵測器對 TIMIT 測試語料所作的辨識，其 frame error rate 為 24.533%，另外可以看出除了 Affricate 辨識錯誤率最低以外，nasal 的辨識錯誤率是次低，因為可由先前的偵測器的效能看出，而辨識錯誤率最高的是 vowel 與 glide，vowel 錯誤率高的其中一個原因可能是由於其資料量龐大所造成。另外由 confusion matrix 可以看出 vowel 與 glide 互為容易辨識錯誤的一對。

9. 高斯混合模型的發音位置偵測器之辨認

下表為發音方法偵測器對 TIMIT 測試語料所做的發音方法辨識，其各別發音方法的統計資料。

表 2-8: 發音位置偵測器所做的發音方法辨識，其各別發音方法的統計資料。

TIMIT Testing Data			
Total frame error rate = 35.779%			
positions	FA rate	FR rate	error_rate
bilabial	65.01 %	40.29 %	4.56 %
labdent	64.60 %	44.02 %	3.88 %
dental	83.12 %	54.37 %	3.46 %
alveolar	22.81 %	37.73 %	8.20 %
velar	52.69 %	40.55 %	4.89 %
glottal	76.36 %	53.10 %	4.25 %
rhotic	31.83 %	27.19 %	4.39 %
front	21.99 %	30.73 %	11.18 %
central	51.88 %	51.47 %	9.10 %
back	42.58 %	49.02 %	9.29 %
Silence	13.32 %	24.86 %	8.35 %

由表 2-8 可以看出 front、central、back 的辨識錯誤率相當的高，而這三個發音位置的 phonemes 大部分亦是屬於發音方法 vowel，且 vowel 偵測器的錯誤率與辨識錯誤率亦是相當的高。

下表列出 front，central，back 的 confusion matrix。

表 2-9: 發音位置偵測器所作的發音方法辨識，發音位置 front、central、back 的 confusion matrix。

Recognize / ref	front	central	back
Front	69.27 %	6.45 %	6.28 %
Central	11.00 %	48.53 %	20.32 %
Back	13.65 %	19.88 %	50.98 %

由表 2-9 可以看出 front、central、back 這三種彼此相互辨認錯誤率相當高，因此這些與 front、central、back 和 vowel 相交集的 phonemes 之辨識較不易，需要有更好的作法解決。

三、 國語語音 Phone-Based HMM 以及 TIMIT GMM 語音屬性偵測器進行音素切割之效能比較

因為現有之國語語料庫都沒有人工音素切割資訊，而在 NG-ASR 架構中之語音屬性偵測性又是 frame-based 的偵測性，與傳統的 HMM 語音辨認器以音素單元作辨認結果有很大的差異。所以傳統的 HMM force-alignment 所獲得的切割位置的精確性將首先在此被檢驗。在此我們將探討使用人工切割位置所訓練之英文語音屬性偵測器架構的辨識效能與傳統以 HMM-Based 的辨識效能作比較，觀察其差異。

而本節主要重點放在發音方法上的比較，因此僅考慮發音方法。

一、 國語語音資料庫之預處理

我們對國語語料庫 TCC-300 [TCC, 2008]作以下之預處理：

(1). 使用中文語料庫 TCC300，訓練以 Phone-Based 為架構的 HMMs，而其每個 phone model 皆設為 3 state 的 HMM，且取 38 維 MFCC 參數，window size 為 32ms ， frame shift 為 10ms，以 flat start 開始訓練 HMMs， HMMs 的 mixture 數時，會依照該 model 在語料庫的資料量作調整，而在此我們利用此一功能藉由 iteration 12 次將每一個 model 的每一個 state 平均升至 mixture 128 個。接著我們將訓練好的 HMMs 拿來對 TCC300 的訓練語料作 Forced Alignment，因此我們可以得到一個有粗略位置資訊的 phone-level 的 labeling file，最後我們可以利用中文發音方法的分類表，將 phone-level 的 labeling file 轉為 manner-level 的 TCC300 的訓練語料的 labeling file。

(2). 在這個實驗中，我們將利用前一節所作的以英文(TIMIT)語料庫所作的英文發音方法偵測器，在此我們將其已訓練到 mixture256 的七種 Target models 拿來對中文語料庫 TCC300 的訓練語料作 Forced Alignment ，因此可得到另一種切割方式的 manner-level 的 TCC300 訓練語料的 labeling file。

(3). 用 TIMIT 的每一個發音方法偵測器去對 TCC300 的訓練語料作偵測，將每個偵測器所得到的結果依照其每個 frame 的機率值去作辨識，也就是說在每一個 frame 其若在某一個偵測器的機率值為最高，便決定該 frame 屬於該偵測器的屬性。因此最後會得到一個 TCC300 的訓練語料的 manner-level 的 labeling file。

二、 切割位置的差距

首先我們先比較前一小節的 1,2 項，而這兩種皆是對 TCC300 的訓練語料作 forced alignment，因此已知正確答案的切割，所以我們可以觀察這兩種切割方式每種 segment 的切割位置，並觀察其差異。在此我們將以第二種 labeling file 也就是用 TIMIT 的發音方法的 GMM model 對 TCC300 作切割的 labeling file 當作參考，去觀察傳統的 HMM-Based 的切割效能。

下表為以 TIMIT 的 manner models 對 TCC300 測試語料作 forced alignment 的統計資料

表 3.1: 以 TIMIT 的 manner models 對 TCC300 測試語料作 forced alignment 的統計資料。

manners	times	Frame amount	min_frame	Average_frame	max_frame
Vowel	418337	3661466	1	8.75	127
Fricative	74276	604075	1	8.13	99
Stop	76291	316615	1	4.15	68
Nasal	119535	827703	1	6.92	73
Liquid	14653	109749	1	7.49	117
Silence	350316	2770498	1	7.91	2314
Affricate	75889	291470	1	3.84	32

表 3-2: 由 Phone-Based HMM 對 TCC300 測試語料作 forced alignment 後轉為 manner-level 的統計資料。

manners	times	Frame amount	min_frame	Average_frame	max_frame
Vowel	418337	4088079	3	9.77	60
Fricative	74276	829482	3	11.17	45
Stop	76291	632948	3	8.30	31
Nasal	119535	692825	3	5.80	55
Liquid	14653	100047	3	6.83	35

Silence	350316	1456902	0	4.16	2313
Affricate	75889	781293	3	10.30	47

下表為上述所講的第二項當參考的 labeling 檔，其相對的 HMM 切割位置差異量的統計，其中負值表示使用英文偵測器之切割位置相對較為前面。(單位為 frame)

表 3-3: 以 HMM 作切割相對於以 TIMIT Manner GMMs 作切割的前後切割位置統計資料。

manners	front_min	front_avg	front_max	back_mix	back_avg	back_max
Vowel	-190	-3.52	241	-206	-2.50	249
Fricative	-208	-3.72	219	-190	-0.68	224
Stop	-183	-5.35	220	-179	-1.20	225
Nasal	-174	0.54	249	-173	-0.59	249
Liquid	-169	-2.08	246	-170	-2.74	228
Silence	-2312	-1.67	249	-2312	-5.42	246
Affricate	-195	-6.74	227	-183	-0.28	241

下表為這兩種切割的 Confusion Matrix

表 3-4: 以 TIMIT Manner GMMs 作 forced alignment 當參考答案與以 HMM 作切割的 Confusion Matrix。

recognize / ref	vowel	fricative	stop	nasal	liquid	silence	affricate
vowel	89.10 %	3.69 %	1.19 %	2.62 %	1.31 %	0.22 %	1.87 %
fricative	14.27 %	78.90 %	0.58 %	0.28 %	0.07 %	0.87 %	5.03 %
stop	32.57 %	2.22 %	61.24 %	0.27 %	0.36 %	0.58 %	2.76 %
nasal	29.94 %	2.18 %	1.75 %	61.71 %	1.28 %	0.42 %	2.71 %
liquid	72.74 %	3.82 %	1.57 %	1.67 %	19.29 %	0.33 %	0.58 %
silence	8.92 %	6.58 %	13.54 %	2.93 %	0.68 %	51.88 %	15.47 %
affricate	21.04 %	2.10 %	0.26 %	0.19 %	0.05 %	0.22 %	76.14 %

三、 英文發音方法偵測器辨識效能與英文發音方法高斯混合模型對 TCC300 作切割的效能比較

在此我們一樣取以 TIMIT Manner GMMs 作 forced alignment 的 labeling file 當參考答案，來看看與英文發音方法偵測器對 TCC300 所作的辨識兩者間的誤差，也就是看第 2 項與第 3 項的誤差。

下表為以 TIMIT Manner GMMs 作 forced alignment 當參考答案與以英文發音方法偵測器所作的辨識的統計資料。

表 3-5:以 TIMIT Manner GMMs 作 forced alignment 之結果當參考答案與以英文發音方法偵測器所作的辨識的統計資料。

TCC300 Training Data			
Total frame error rate = 32.062%			
manners	FA rate %	FR rate %	error_rate %
Vowel	8.96	28.89	15.31
Fricative	57.70	31.06	8.81
Stop	74.48	44.49	7.62
Nasal	37.08	26.95	6.75
Liquid	93.02	48.72	9.36
Silence	9.56	32.27	12.73
Affricate	53.12	66.71	3.55

下表為這兩種切割的 Confusion Matrix。

表 3-5: 以 TIMIT Manner GMMs 作 forced alignment 當參考答案與以英文發音方法偵測器所作的辨識的 Confusion Matrix。

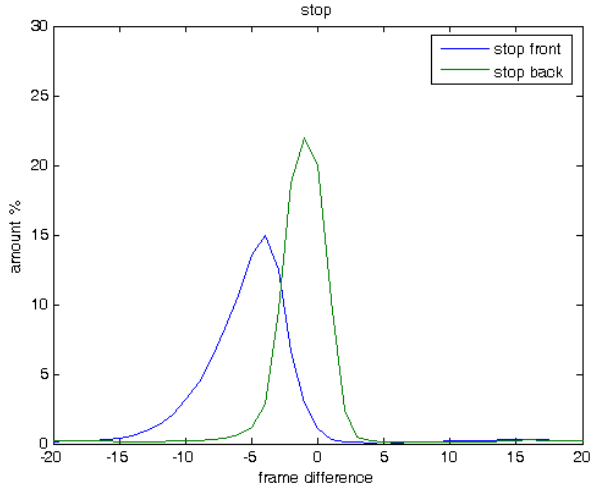
ref \ recognize	vowel	fricative	stop	nasal	liquid	silence	affricate
vowel	71.11 %	2.04 %	4.86 %	4.34 %	15.33 %	1.63 %	0.69 %
fricative	2.64 %	68.94 %	9.07 %	2.11 %	2.15 %	6.66 %	8.43 %
stop	7.42 %	7.97 %	55.51 %	3.01 %	12.81 %	12.25 %	1.03 %
nasal	12.59 %	1.92 %	1.44 %	73.05 %	5.12 %	5.67 %	0.22 %
liquid	25.75 %	1.30 %	9.36 %	9.52 %	51.28 %	2.43 %	0.36 %
silence	2.85 %	12.13 %	7.27 %	5.84 %	3.17 %	67.73 %	1.02 %

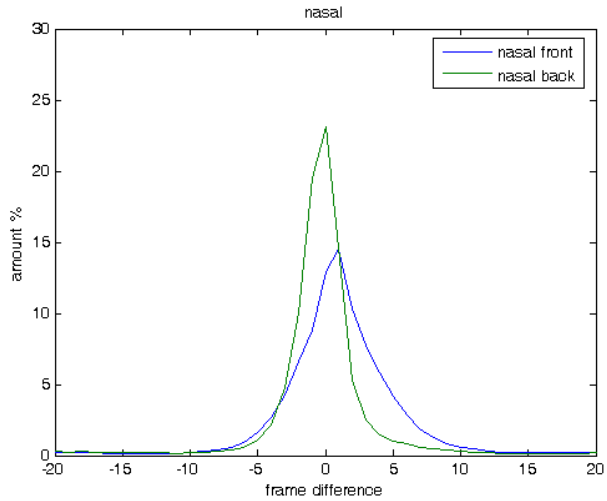
affricate	1.83 %	39.33 %	19.48 %	0.99 %	1.61 %	3.46 %	33.29 %

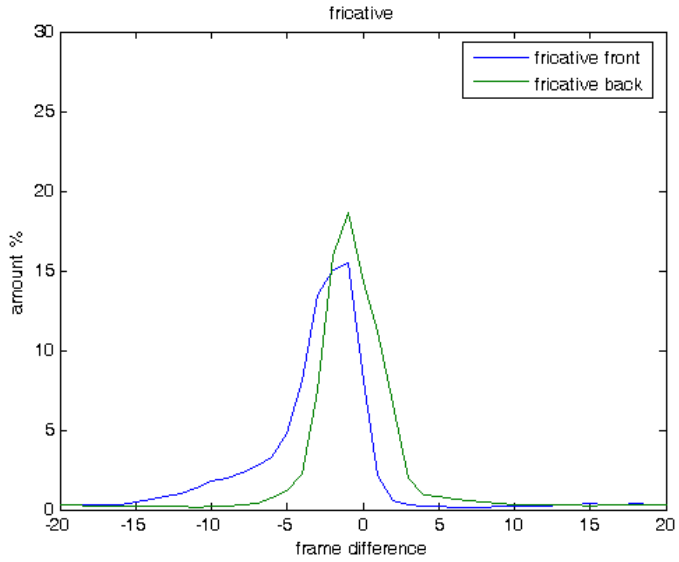
尤以上結果可以知道傳統 HMM 所獲得之切割位置的精確度對用來標示 frame-based 語音屬性偵測器的訓練語料而言仍然不足。

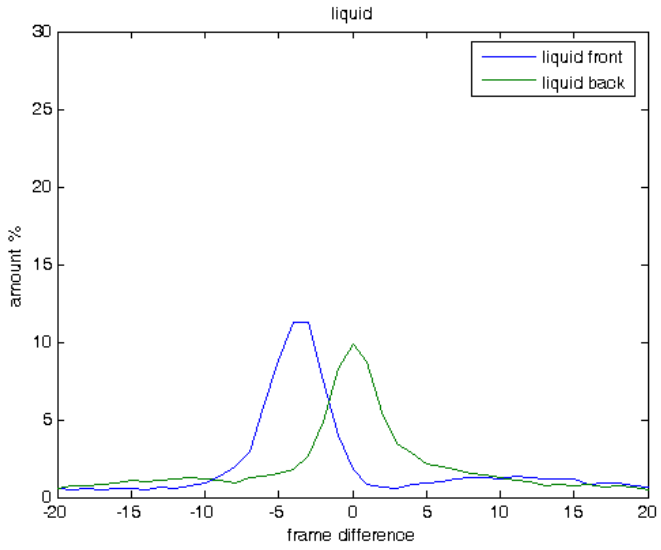
附錄 1、Phone-Based HMM 以及 TIMIT GMM 語音屬性偵測器進行音素切割之效能比較範例

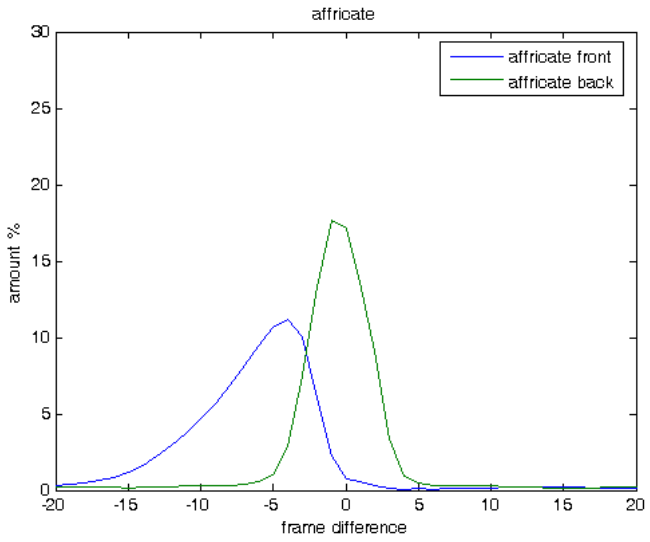
表 3-3 資訊之分佈圖，也就是使用語音屬性偵測器進行音素切割之位置當參考的 labeling 檔，其相對的 HMM 切割位置差異量的統計圖。

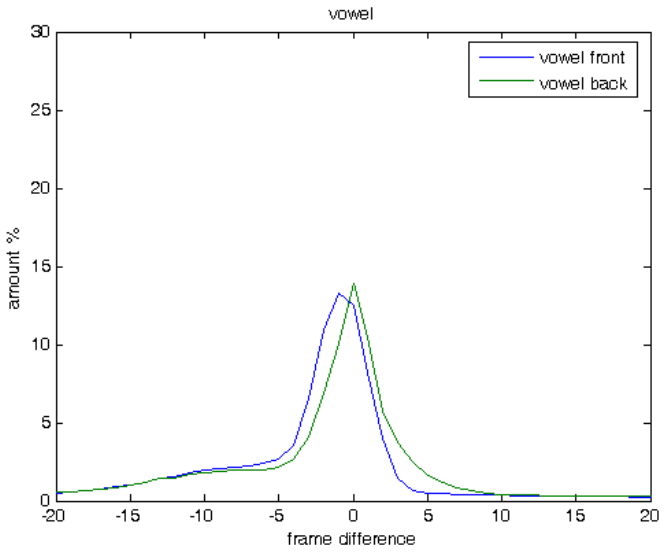





fricative
fricative front
fricative back
amount %
frame difference
30
25
20
15
10
5
0
-20
-15
-10
-5
0
5
10
15
20


liquid
liquid front
liquid back
amount %
frame difference
30
25
20
15
10
5
0
-20
-15
-10
-5
0
5
10
15
20


affricate
affricate front
affricate back
amount %
frame difference
30
25
20
15
10
5
0
-20
-15
-10
-5
0
5
10
15
20


vowel
vowel front
vowel back
amount %
frame difference
30
25
20
15
10
5
0
-20
-15
-10
-5
0
5
10
15
20

四、 中文音節標記檔的訂正、自動切割與語音屬性偵測器之製作

TCC300 語料庫原始的音節 transcription 檔可能存在某部份人為標記的錯誤，這個現象直接會污染我們訓練的音節模型，使得 HMM 強迫切割的結果變差，進而造成偵測器模型參雜了不正確的訓練資料影響到偵測器的效能，因此我們構想一個機制如下圖所示，首先能夠自動的找出大部分可能發生錯誤的標記錯誤，接著再用人工去聽音檔確認並且改正錯誤的標記，首先我們先訓練 411 音節的 HMM，接著對於語料做辨認，如果標示答案未出現於音節辨認結果的 Top-N 當中，我們將用人工檢查是否為標示錯誤。在我們所使用的辨認器之辨識率已經很高的情況下，這樣將可以有效的找出語料當中的標示錯誤。

我們將 TCC-300 語料作 force-alignment 後對每一音節做 top-N 辨認，在 top 20 若沒有包含 transcription 中所標示之音節則自動篩選出來，這些就是可能有問題的音節。最後在使用人工去聽音檔對照檢查，確認該 syllable 是否有音節標記錯誤或是其他類型的錯誤，下表是錯誤類別的統計：

將以上統計結果中實際上音節標記錯誤的資料量佔所有語料的比例統計:

表 4.1：音節標記錯誤佔所有語料庫的資料量比例。

語料	總音節數	實際上標記錯誤的音節數
TCC300_train	約 300000	287(約 0.1%)
TCC300_test	約 33000	29(約 0.09%)

由上述統計結果可以看出，音節標記錯誤佔所有語料庫資料量中約 0.1%，這些標記的錯誤不但會使訓練語料拿來訓練偵測器模型時由於錯誤的標記位置學習到不正確的資料影響偵測器的效能，同時被拿當來當作偵測實驗當中的參考答案的測試語料中的標記錯誤，更是可能直接使得錯誤率升高的原因之一。

1. 中文音素切割位置的取得

接下來我們首先訓練音節的 HMM 模型進行強迫切割取得音節的切割位置，然後以音節切割位置開始訓練狀態數為 3 的音素的 HMM 模型，同時訓練 short pause (sp) 與 silence 模型並且將 non-speech signal(如 breath , noise…等)模型隱藏在 short pause 以及 silence 的 HMM 狀態當中，允許狀態跳躍來切出更合理的非語音段，下圖 4.1 為非語音模型當中的狀態轉移示意圖:

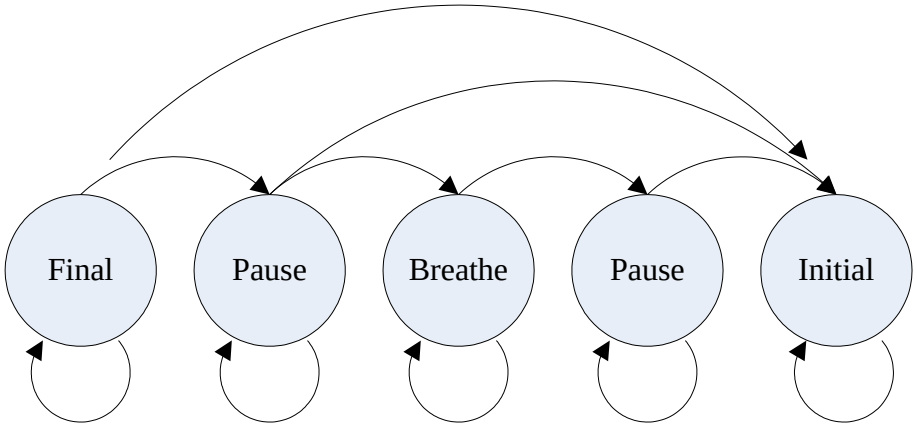


圖 4.1：非語音段模型狀態轉移圖。

有了音素的 HMM 模型之後緊接著便對語料庫進行強迫切割，雖然把呼吸聲切割出來能夠切出比較乾淨的 short pause 以及 silence，然而以切割的角度來看 HMM 的切割位置仍舊不夠精準，底下是一個簡單的例子:

語者呼吸聲

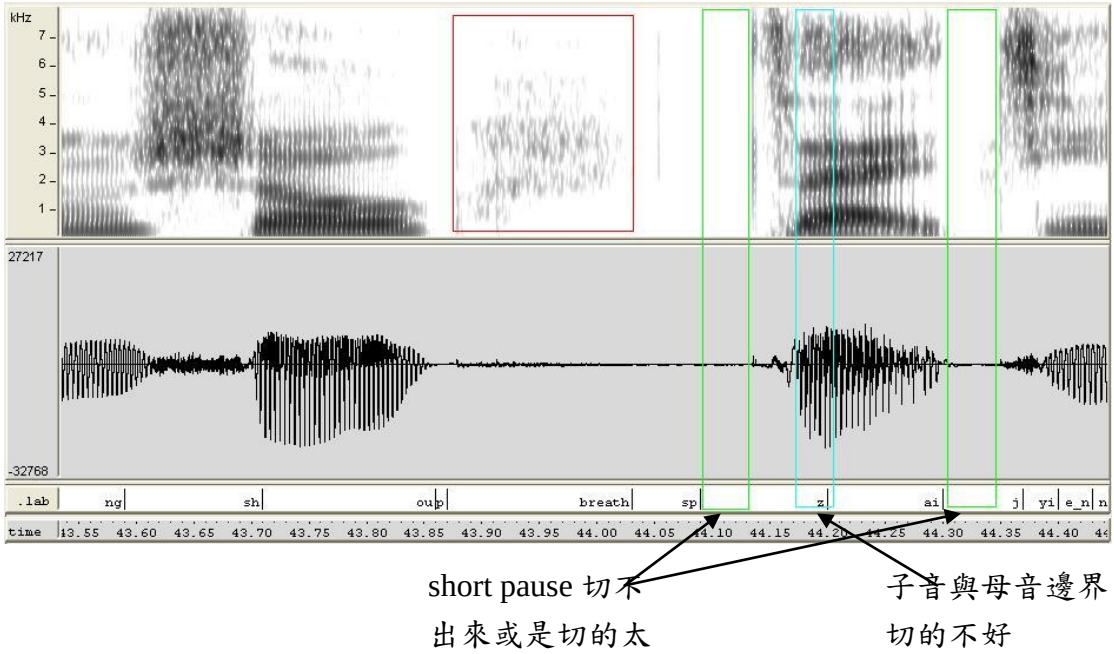


圖 4.2：音素 HMM 強迫切割不準確的例子。

我們可以看到 HMM 自動強迫切割的結果對於音素的邊界常常有不小的誤差，尤其是子音之前的短暫 short pause 往往都切不出來或者是切的太短，造成子音的長度普遍過長，同時子音與其後母音的交界切割也不甚理想，以下是 HMM 強迫切割的各發音方法平均音長的統計:

表 4.3：強迫切割各發音方法平均音長。

單位:音框 發音方法	HMM 強迫切割平均音長
Vowel	9.62
affricate	9.62
Liquid	6.69
fricative	10.99
Nasal	6.59
Stop	7.78
Silence	15.74

我們拿音長切的過長最明顯的 Stop 音出來更細部的觀察音素的平均音長狀況，其中音長特別短的ㄅ、ㄉ、ㄍ三個音的音長做統計:

表 4.4：ㄅㄉㄍ三種音素的平均音長。(單位：frame)

	平均長度
ㄅ(b)	6.36
ㄉ(d)	5.86
ㄍ(g)	6.85

這三個音的音長實際上大約在 3 個音框長以下，但 HMM 自動切割的平均長度竟然整整多了一倍，代表說以 HMM 強迫切割取得的切割位置確實將子音的長度切的過長，其中又以音長較短的子音特別明顯。

因此底下我們提出以使用局部樣本(local sample)之 Segmental K-means segmentation algorithm 的方法來調整音素的切割位置，它是一種廣為人知拿來對於資料分群的 K-means iterative procedure，它能夠藉著 Viterbi algorithm 找到最佳

的分段序列，重新將樣本點的資料分類，因此我們將這方法應用來對我們音節之間的 sp 以及子音母音的邊界進行切割位置調整。首先我們固定呼吸聲的切割位置，假設 Observation sequence $O = (o_1 o_2 ... o_N)$ 用來代表由一音節之 final 起始點至下一音節 initial 之終止點間語音信號參數，並且使用 HMM 之音節切割位置並且將之分成 I = 3 段落 (final， short pause， initial)， observation vector o_j 由 13 維 MFCC 參數組成，而 i th (1 ≦ i ≦ I) segment；S_i ；的音框訓練一個高斯模型 $\Phi_i = N(\mu_i, \sum_i)$，其中 μ_i 為 mean-vector，$\sum_i$ 為 covariance matrix，這些參數都可由第 i th 段落當中 n_i 個音框求得：

$$\hat{\mu} = \frac{1}{n_i} \sum_{k=1}^{n_i} o_k \tag{4.1}$$

$$\hat{\Sigma}_i = \frac{1}{n_i} \sum_{k=1}^{n_i} (o_k - \hat{\mu}_i)\ (o_k - \hat{\mu}_i)^t \tag{4.2}$$

而在調整音節間 short pause 切割位置的步驟當中，likelihood 方程式可以寫成：

$$\prod_S p(o_j \mid \Phi_{S_i}) = \prod \frac{1}{2\pi \mid \hat{\Sigma}_i \mid^{1/2}} \exp\left[-\frac{1}{2} (o_j - \hat{\mu}_i)^t \times \hat{\Sigma}_i^{-1} (o_j - \hat{\mu}_i) \right] \tag{4.3}$$

使用 maximum likelihood 的要求利用 Viterbi search 來找到最佳之切割位置 S。

我們針對每個句子，收集該句當中較可靠的 short pause，silence，breath 音框(該非語音段落長度至少長於 5 ms)，用 VQ 將這些資料依據 energy 大小分為兩群，能量較大的一群定為"non-speech signal"，能量較低的一群定為"silence"，將能量較低的一群抽取 13 維的 MFCC 參數拿來訓練 Gaussian 模型當做該句中 short pause 模型，之後從句子的開頭開始循序往後處理每個音節之間的 short pause，處理的方式如下：拿 sp 之前的 final 音段當中所有的音框拿來訓練一個 13 維 MFCC 的 final 高斯模型，同時拿 short pause 之後的 initial 音段當中較可靠的音框(一般來說是所有音框，但是針對某些音的特性而有些限制，比如說爆破音當中ㄅ，ㄉ，ㄍ這三個音特別的短，因此僅取該音結束點往前的 3 個音框)拿來訓練 13 維 MFCC 的 initial 模型，然後從 short pause 前的 final 起始點開始往後逐個音框對於 final， short pause，non-speech signal， initial 這幾個模型如下圖 4.3 進行 Viterbi Search 的比對，不過比對結束之後決定每個 state 的區段位置時必須保留呼吸聲保留原始的切割位置，因此實際上調整的有兩部分 ：final 與 short pause 之間的切割位置以及 initial 與 short pause 之間的切割位置。

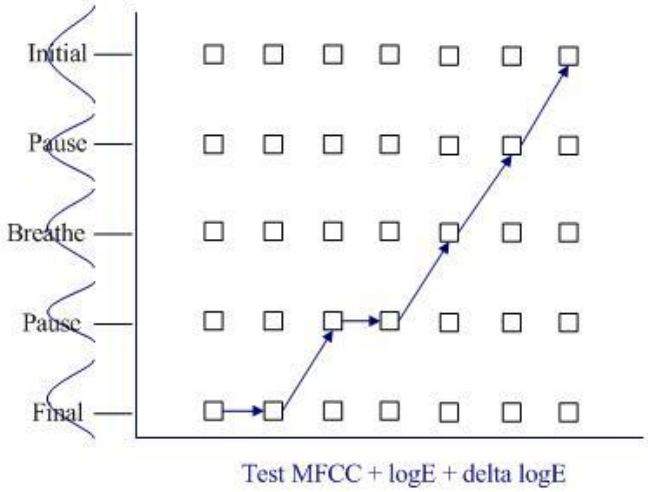


圖 4.3：音節間進行 Viterbi search 比對。

每執行完一次 Viterbi search，規定結束點狀態必須是 initial，起始點狀態必須是 final 之後用 back trace 決定各個狀態的音框數，獲得新的切割位置之後，取新的 initial 與 final 音段音框分別更新 initial 與 final 的高斯模型，再次執行上述的流程，直到各個狀態音框段落都收斂之後，就算處理完一個音節之間的 short pause，之後以此類推處理完整個音檔，至於調整 initial 與 final 之間切割位置，所不同的是僅有兩個狀態轉換之間去做 Viterbi search，圖 4.4 為取得音素切割位置的流程:

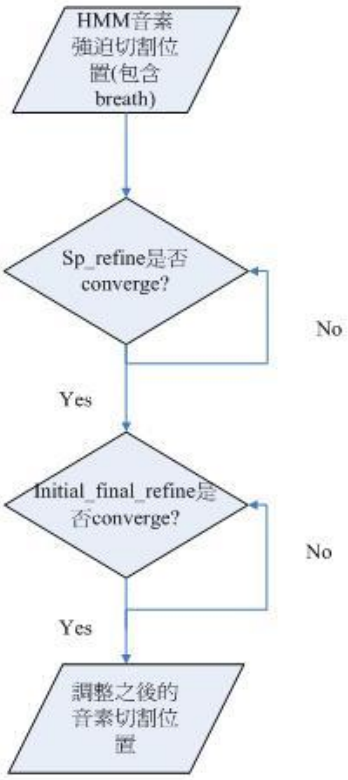
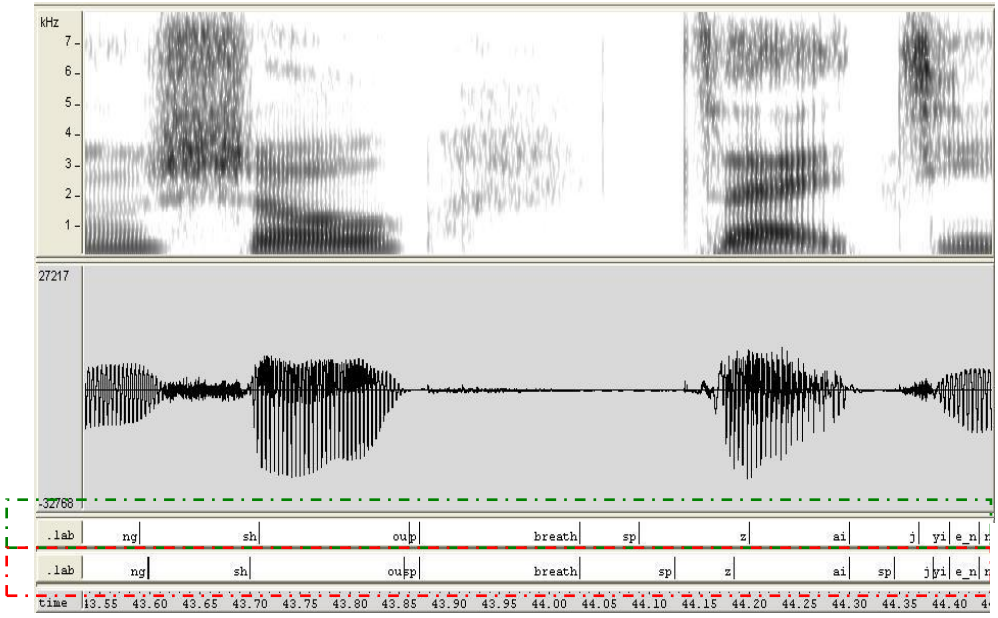


圖 4.4：取得音素切割位置的流程

舉個例子來觀察經過調整之後的切割位置更趨近於人工標記的切割位置:



未經調整的切割位置

經過調整的切割位置

圖 4.6：音素切割位置調整前後之比較。

由上圖我們可以觀察到不管是音節與音節之間的 sp 或者是子音與母音之間

的邊界切割位置都切的更準確，底下我們更細部的去觀察統計切割位置改進的情形，首先統計各發音方法調整之後的平均音長:

表 4.5：調整前後的發音方法平均音長統計。

發音方法	調整前平均音長	調整後平均音長
Vowel	9.62	9.49
affricate	9.62	7.95
Liquid	6.69	6.13
fricative	10.99	10.88
Nasal	6.59	6.90
Stop	7.78	4.89
Silence	15.74	4.83
Breath	16.67	16.67

子音的音長除了 nasal 稍微變長以外都有或多或少的下降，這是由於子音前的 sp 能夠被有效的切出來因此使得子音的平均音長下降，特別是在發音的時候必須先緊閉聲道的 stop 以及 affricate 特別明顯，也因為音節間短暫的 sp 被有效的還原出來因此造成 silence 的平均長度被拉低了，這時我們在將音長特別短的ㄅ、ㄉ、ㄍ這三個音的音長做比較:

表 4.6：ㄅㄉㄍ三種音素調整前後平均音長比較。

	原始平均長度	調整之後平均長度
ㄅ(b)	6.36	3.41
ㄉ(d)	5.86	3.46
ㄍ(g)	6.85	3.58

很明顯的看到，ㄅ、ㄉ、ㄍ經過調整之後的平均長度都大幅的降低約 3 個音框長，明顯的比調整之前的音長要更趨近於合理的長度，接著我們進一步定量的分析實際上經過調整之後的 initial 之前的 sp 能夠大量的被還原出來使得 initial 的長度變的比較合理，我們針對此一現象特別明顯的 stop 音來抽樣，我們抽樣取 100 個 stop 音事件當中的前三個音框，而這些 stop 音事件分別平均分布於 10 個不同的音檔，我們用人工去實際比對音檔與切割位置，觀察比較未經過調整之前 stop 音的前三個音框有很大的成分實際上是沒被切出來的 sp，底下用直方圖來表示會很清楚：

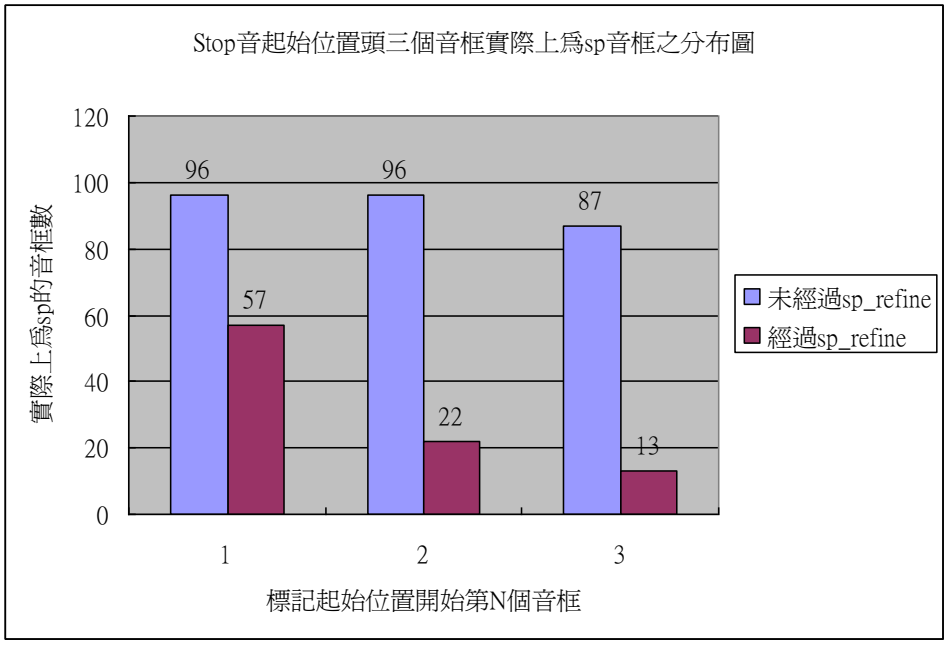


圖 4.7：Stop 音改善切割位置的比較統計。

我們可以看到藉由抽樣觀察 100 個 stop 音事件當中，未經過調整的 stop 音切割位置幾乎前三個音框實際上都是 sp，而經過自動調整之後的切割位置大幅度的將 stop 音起始的頭三個音框(實際上是 sp)還原為 sp 音框。接下來為了分析調整子音與母音之後切割位置的改進，同樣取來自不同語者的 10 個音檔，其中每一句平均取若干個 initial 的事件來做統計，用直條圖來表示經過調整前後切割位置與實際人工標記比較的改進情況:

Total 樣本: 100 samples: fricative，affricate

50 samples: stop，nasal

40 samples: liquid

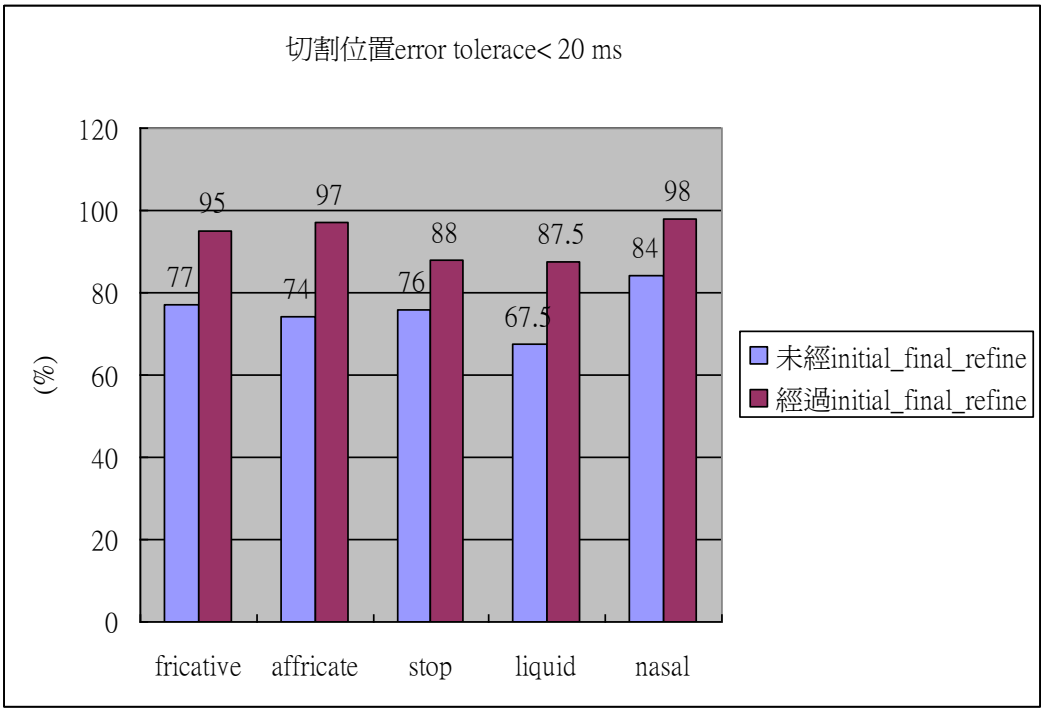


圖 2.8：子音與母音邊界切割位置改善的統計。

由上圖統計資料可以看出，調整之前的子音母音交界切割位置誤差在 20 ms 以內的比例大約是還算不錯的 75%左右，但是經過自動調整之後的子音母音交界切割位置誤差在 20 ms 以內幾乎可以達到約 90%以上，由以上的切割位置改善統計結果可以看出，雖然 TCC300 語料庫沒有人工標記的音素切割位置，但是經由 Segmental K-means segmentation algorithm 自動的調整切割位置之後，已經能夠得到趨近於人工標記的音素切割位置。

2. 中文語音屬性偵測器之初步建立

在前一節當中我們已經自動的調整 HMM 強迫切割的結果得到可靠的音素切割位置，接著將利用中文音素的發音方法分類表(表 4.7)，將訓練語料以及測試語料的音素切割位置轉為發音方法的切割位置，在發音方法分類當中直得注意的是，原本在參考資料[5]當中ㄖ這個音的分類是屬於摩擦音，但是參考資料當中同樣有統計ㄖ這個音被 Liquid 偵測器偵測為 Liquid 的比例高達 76%，這是因為如果單獨念ㄖ這個音聲學特徵確實是屬於摩擦音，但是在中文連續語音當中語者往往因為連音的現象因此只有唸出ㄖ這個音的前半捲舌音(類似於 r 系音)因此也符合於參考文獻[4]當中對於 Liquid 這類音素的定義，因此在本計畫中我們將ㄖ這個音素由摩擦音移至 Liquid 音的分類當中。

表 4.7：中文發音方法分類表。

1	爆破音（Stop）	ㄅ (p)	ㄆ (b)	ㄉ (t)	ㄍ (g)	ㄍ(k)	ㄎ (k)
2	鼻音　（Nasal）	ㄇ (m)	ㄋ (n)	n_n , ng			
3	摩擦音（Fricative）	ㄈ (f)	ㄙ (s)	ㄒ (x)	ㄏ (h)	ㄕ (sh)	
4	塞擦音（Affricate）	ㄓ (zh)	ㄔ (ch)	ㄑ (q)	ㄐ (t)	ㄘ（z）	ㄗ（s）
5	流音　（Liquid）	ㄌ (l)	ㄖ (r)				
6	母音　（Vowel）	others					

n_n , ng 為ㄤㄣㄢㄥ的鼻音韻尾

而此訓練語料的發音方法切割位置便作為我們在製作中文發音方法高斯混合模型貝氏偵測器的切割位置，最後再將製作出來的中文發音方法偵測器對測試語料作偵測求取偵測效能。

我們從音節切割位置起始著手訓練音素的馬可夫模型後，對語料庫作切割的方法，同時切出非語音的呼吸聲，接著再半自動的調整音素切割位置，而我們也將以此較可靠的切割位置當作是中文 TCC300 訓練語料的切割位置，訓練各個發音方法的高斯混合模型偵測器，以下將用此結果與使用音素 HMM 對語料進行強迫切割的切割位置所訓練的高斯模型製作的偵測器偵測效能做比較。

1. 以 HMM 強迫切割的音素切割位置訓練的高斯模型製作偵測器。
2. 以 HMM 強迫切割之後的音素切割位置再經過自動調整的切割位置訓練的高斯模型偵測器。

表 4.8：以調整前後的切割位置訓練高斯混合模型偵測器偵測實驗。

訓練語料切割位置 / 發音方法偵測	HMM 強迫切割位置 (EER%)	HMM 強迫切割經過調整後的切割位置(EER%)
Stop	12.17	11.12
Nasal	11.90	11.57
Vowel	12.33	11.05
Affricate	11.91	10.98
Fricative	12.47	11.38
Liquid	9.73	9.16
Silence	11.98	7.25

上表顯示出經過調整切割位置之後的切割位置訓練偵測器，各種發音方法的

偵測錯誤率都有明顯的下降，除了 Nasal 與 Liquid 之外，其餘發音方法偵測器的等錯誤率都有約 1%以上的下降，特別是 Silence 偵測器等錯誤率大幅的降低了 4%以上，這主要也是因為經過自動調整之後還原了許多音節間的 sp 的緣故，所以對於發音方法屬性偵測而言，經過調整之後的切割位置確實是比 HMM 強迫切割的切割位置要好，因此往後的章節當中將以不同方法訓練發音方法偵側器以及偵測實驗都將以此調整後的音素切割位置來當作訓練各個發音方法模型後製作偵測器以及測試語料的依據。

五、 使用類神經網路的國語語音屬性偵測器

我們取得了相當可靠的 TCC300 國語語料庫的音素切割位置之後，接著建立最基礎的 frame-based 高斯模型中文發音方法偵測器。由於非線性的類神經網路架構已經證明在資料類別分類上有優於線性高斯混合模型的效能，因此在本節當中首先建立屬於類神經網路的多層感知機 (Multi-layer perceptrons, MLP)模型為基礎的中文發音方法偵測器。然而在連續語音的語音屬性偵測當中單純的只考慮每個音框本身的資訊其實是不大合理的，因為即使是以音框為偵測的基本單元，每個音框仍舊會受到前後音框以及一些語言特性的影響，因此本章接著會加入類似以音段為基礎(segment-based)的概念，在原本的 MLP 模型為基礎的偵測器上加入 target 與 anti-model 這兩個狀態轉換的機率(transition probability)分數改善偵測器的效能，最後我們將由 MLP 發音方法偵測器為基礎建立階層式的語音屬性信任度量測(Confidence Measure)，如此一來便能夠評量偵測器偵測結果的可靠性，提供給自動語音辨識架構後級辨識器更可靠的語音資訊。

1. 以 MLP 模型為基礎發音方法貝氏偵測器之製作

我們採用的發音方法 MLP 模型分為三層如圖 5.1，包含一個輸入層、一個隱藏層、一個輸出層，根據我們輸入每個音框的 38 維 MFCC 參數因此輸入層點數設定為 38，而隱藏層點數設定為 50，而輸出層由於我們同樣要訓練 target model 以及 anti model 因此設定點數為 2。

MLP 網路是一種正向饋入(feed-forward)網路，每一個第 i 層神經元的輸出 $O_k^{(i)}$ 都是第 i-1 層輸出加權總合的非線性函數，其數學式如下:

$$O_k^{(l)} = f\left(\sum_{i=1}^{N_\Delta} w_{ki}^{(l)} O_i^{(\Delta)} + \theta_k^{(l)}\right) \tag{5.1}$$

其中

$$f(x) = \frac{1.0}{1+e^{-x}} \tag{5.2}$$

為 sigmoid function，N_Δ 為第 Δ 層的神經元數目，$\theta_k^{(l)}$ 為 bias，而符號 $w_{ki}^{(l)}$ 則表示由 Δ 層的第 i 個神經元到 l 層的第 k 個神經元的加權值。

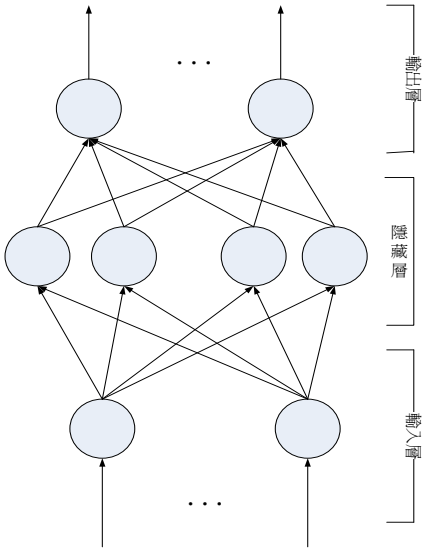


圖 5.1： MLP 網路架構

先前得到相當可靠的音素切割位置，有了切割位置之後我們便直接拿來訓練各個中文發音方法的偵測器，然後與高斯混合模型為基礎的偵測實驗結果相比較:

1. 以 TCC300 中文語料庫經過調整後的音素切割位置訓練 38 維 MFCC 參數的高斯混合模型發音方法偵測器。
2. 以 TCC300 中文語料庫經過調整後的音素切割位置訓練 38 維 MFCC 參數的 MLP 模型發音方法偵測器。

表 5.1：GMM 及 MLP 為基礎的發音方法偵測效能比較。

EER(%) / manner	GMM	MLP
Vowel	11.05	8.29
Stop	11.12	9.98
Fricative	11.38	10.06
Affricate	10.98	9.17
Nasal	11.57	9.25
Liquid	9.15	9.16
Silence	7.25	5.72

由結果可以得知，除了 Liquid 的等錯誤率幾無變化之外，其餘的發音方法偵測等錯誤率都有約 1~2.5%的下降，特別是 Vowel 以及 Nasal 還有 Silence 這三類偵測器，error reduction 都有超過 20%的下降，而這三類的發音方法資料量約佔語料庫總資料量的 70%，因此這三類偵測器錯誤率明顯的下降對於整體偵測器

的效能有顯著的提升。

在得到了 frame-based MLP 中文發音方法偵測器的結果之後，我們將以 segment 的角度來觀察 frame-based 的 MLP 偵測器的偵測錯誤情形類別做分析，首先是錯誤拒絕的部份:

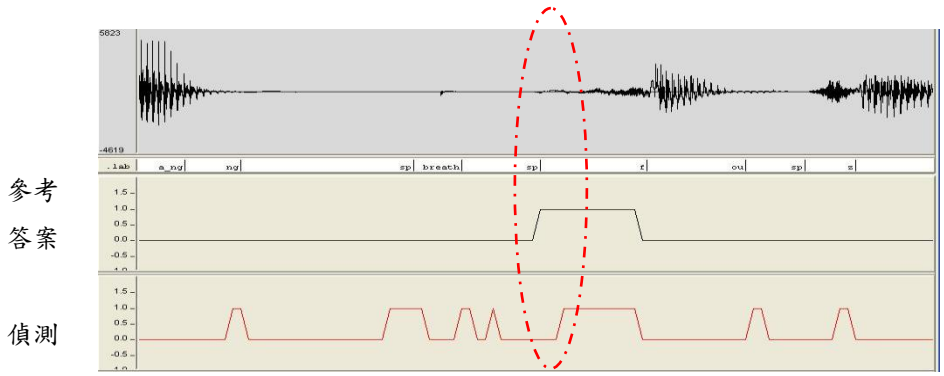


圖 5.2：偵測的結果稍微向內縮。

這類型的錯誤多半是因為邊界附近的聲學特徵還不是很穩定因此發生錯誤拒絕的偵測錯誤，但是以 segment 的角度來看這類型偵測錯誤的情形並不算嚴重，而這類型的錯誤對於 frame-based 偵測器的偵測錯誤影響我們將在後面的章節再作較深入的分析。

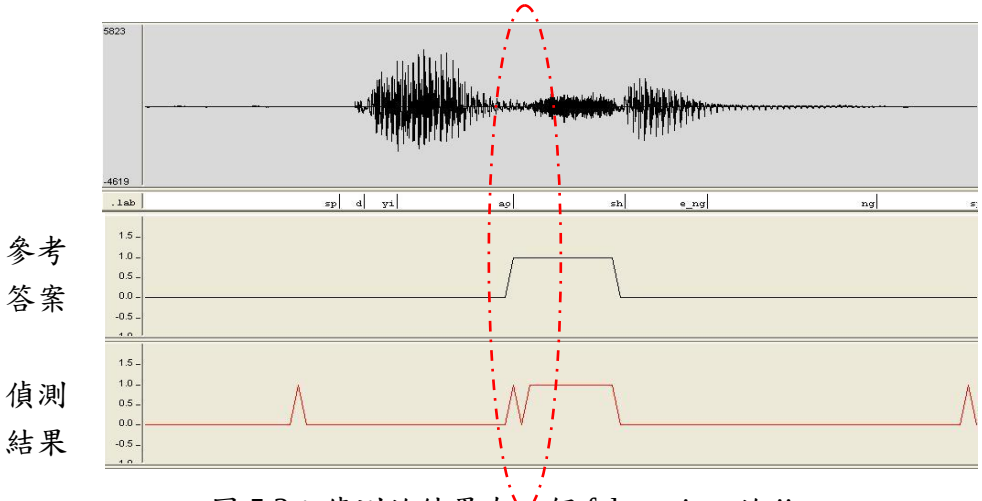


圖 5.3：偵測的結果有一個 false-reject 的 jitter。

這類型的錯誤非常常見但是實際上這些短暫的 jitter 是造成偵測錯誤的主要來源之一，並且也不能提供後級的辨識器可靠的資訊，如果能夠加入一些 segment 概念的資訊應該就能夠有效抑制這類型的錯誤，因此在下一節當中我們將會加入 target 與 anti-model 這兩種狀態轉移的機率分數來試圖克服此類型的偵測錯誤。

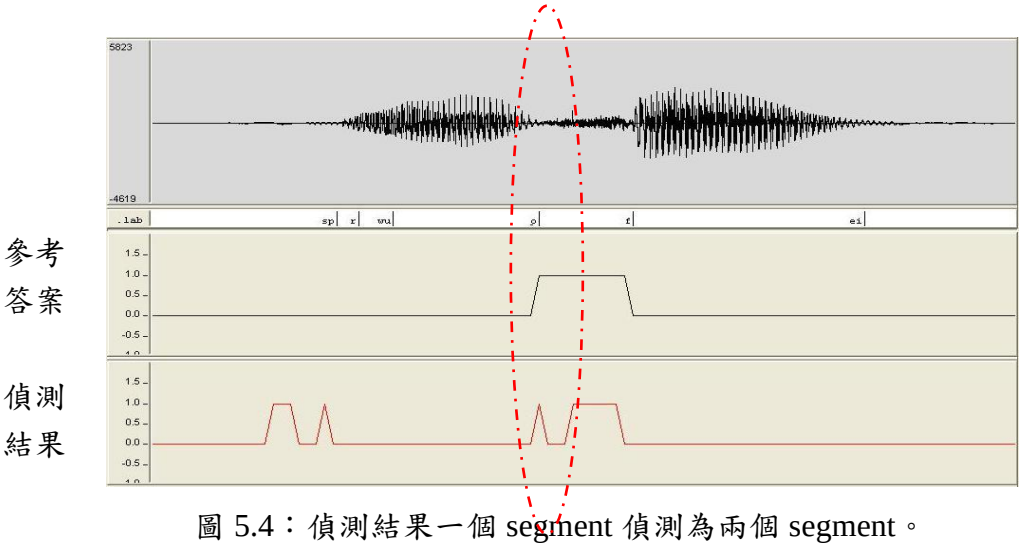


圖 5.4：偵測結果一個 segment 偵測為兩個 segment。

這類型的錯誤時常出現於各種發音方法中能量相對比較小的摩擦音段當中音頭以及音尾部分以及 Silence 段中可能夾雜些微背景雜訊時。

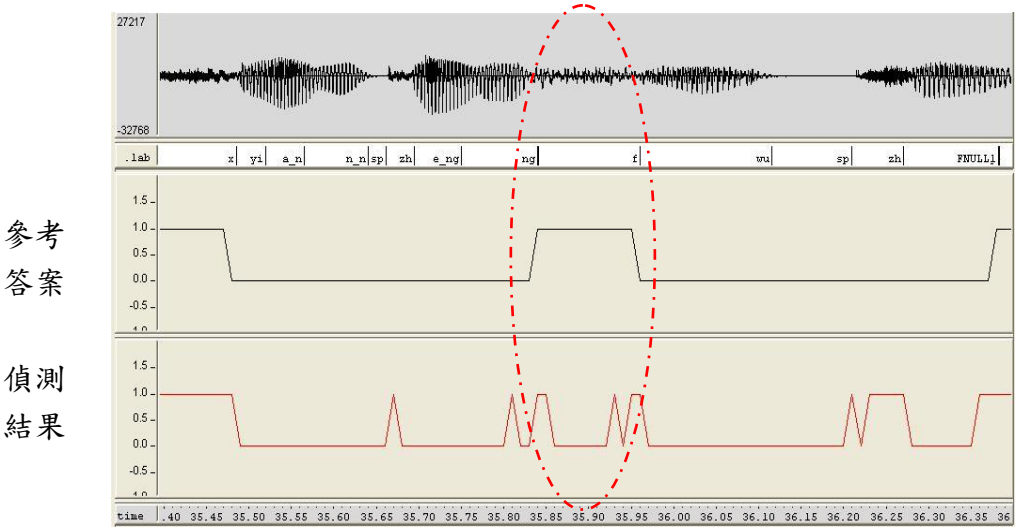


圖 5.5：偵測結果一個 segment 偵測為數個 segment。

這類型的偵測錯誤通常發生在聲學特徵較不穩定的音素當中，比如說摩擦音當中的ㄈ這個音，在隨機抽樣觀察的幾個句子當中發現到ㄈ這個音常常有錯誤拒絕很嚴重的情形，因此我們特別針對這個音去對摩擦音偵測器作偵測，同樣發現到ㄈ這個音雖然屬於摩擦音但是對於摩擦音偵測器的錯誤拒絕率卻高達 32%，並且對於 Silence 偵測器的錯誤警戒率(也就是被偵測為 Silence)將近有 50%，這是因為ㄈ這個音的聲學特徵其實有些類似於語者呼吸聲，因此也許此音素在發音方法屬性偵測的分類上因其特殊的聲學特性而有需要獨立出來自成一類。在分析完了錯誤拒絕類型的錯誤之後我們接著對於錯誤警戒類型的偵測錯誤同樣用 segment 的角度來做各種偵測警戒錯誤的類型分析:

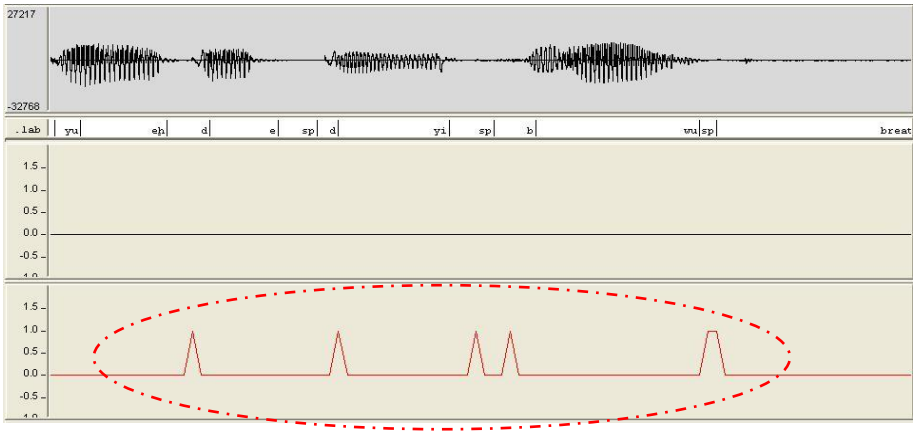


圖 3.6：false-alarm jitter。

同偵測錯誤拒絕當中的第 b 類型錯誤，事實上除了一部分 Stop 音以及 Silence 以外其他類發音方法幾乎不可能單獨出現這麼短的 target segment(1~2 個音框)，因此這類十分明類的錯誤便是我們下一節當中提出狀態轉移機率概念最主要要解決的偵測錯誤類型。

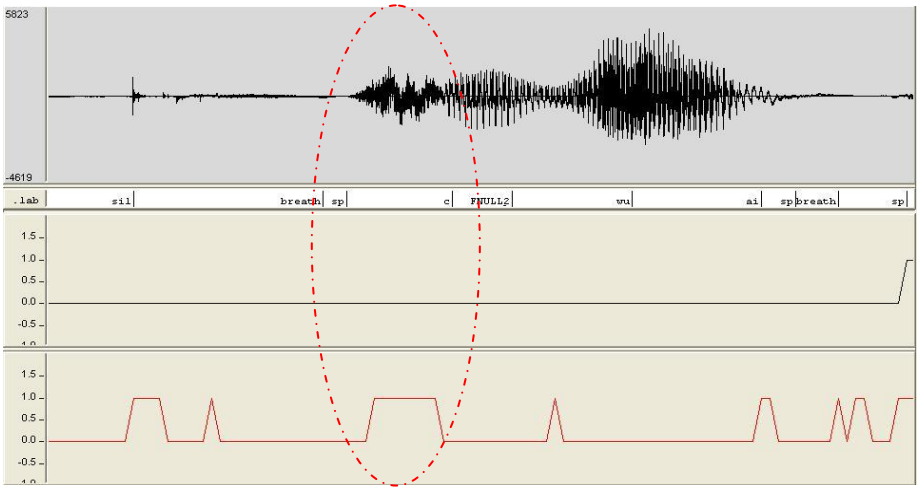


圖 5.7：一長段連續 false-alarm。

這類型的錯誤通常發生在該種發音方法與某種聲學特徵相近的發音方法之間互相混淆情形非常嚴重時，最明顯的情形就是如上圖當中的例子，該音素ㄘ(c)是屬於 Affricate，但是由於 Affricate 與 Fricative 混淆的情形十分嚴重因此整段被 Fricative 偵測為錯誤警戒的錯誤，同樣的錯誤類型也常見於聲學特徵類似的 Nasal 與 Liquid 之間以及 Nasal 與 Vowel 之間。

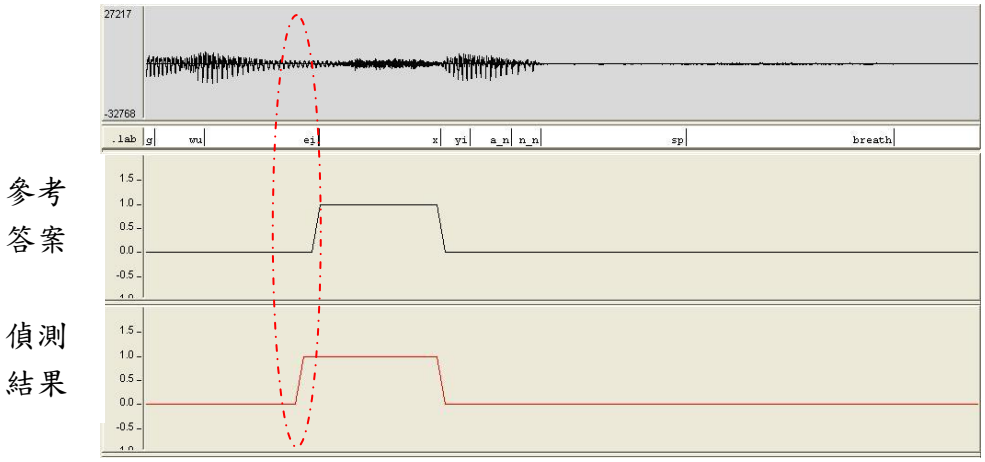


圖 5.8：segment 邊緣向外稍微延伸。

此類型的錯誤類似於錯誤拒絕當中的 a 類型錯誤，因此同樣以 segment 的角度來看這類型的錯誤時這類型並不算是嚴重的偵測錯誤。

以上是對於 frame-based MLP 屬性偵測的錯誤類型的分析，之後的章節我們將針對考量 frame-based MLP 屬性偵測器偵測錯誤的缺失為基礎，提出加入提供偵測器更多資訊的方式，期望能夠降低偵測器一部分的偵測錯誤。

2. MLP 模型為基礎加上狀態轉移機率的發音方法偵測

在前一節當中我們利用類神經網路當中 MLP 模型架構所訓練的偵測器雖然已經能夠得到偵測器效能明顯的提升，但是如果提供更多的聲學資訊加入偵測器的訓練當中應該能再進一步的提升偵測器效能，因此接下來我們將 MLP 的偵測器的輸出分數加入狀態轉移機率的分數作整合。

由於在之前 frame-based 的 MLP 發音方法偵測實驗中，我們已經得到每一個音框在偵測器的分數以及偵測結果，然而偵測的結果純粹是用等錯誤率的狀況下該音框在 target model 上的事後機率分數是否大於 anti model 上的事後機率分數來判定是 target 還是 anti-model，接下來我們取出偵測器的分數，加入 duration model 限制的概念，在每一句當中考慮 target，anti-model 這兩類狀態轉換的機率，讓句子當中的每一個音框進行 Viterbi Search，找到最佳的偵測結果。假定經過 Viterbi Search 後的最佳化 utterance score 為 Q^*，其數學表示式如下:

$$Q^* = \arg\max_{s} S(C$$

$$= \sum_{t=1}^{N} Q_{MLP}(S_t, O_t) + \sum_{t=1}^{N} Q_A(S_t, S_{t-1}) \qquad (5.3)$$

其中 S =1、2； O 為 observation；當 t=1 的時候，若偵測的 target 為 silence，則 S_1 為 2，若偵測的 target 為其餘發音方法，則 S_1 為 1，因此原式可寫成:

$$Q^* = \sum_{t=1}^{N} Q_{MLP}(S_t, O_t) + \sum_{t=2}^{N} Q_A(S_t, S_{t-1}) \tag{5.4}$$

而 $Q_{MLP}(S_t, O_t)$ 即為原本 frame-based 偵測器輸出每個音框在 target model 上的分數取對數，而狀態轉移分數 $Q_A(S_t, S_{t-1})$ 為狀態轉移機率取對數，也就是 $\log(P(S_t \mid S_{t-1}))$，狀態轉移機率用訓練語料當中 target-segment，anti-model segment 的平均長度求得，假設 target-segment 平均長度為 L2,anti_segment 平均長度為 L1 則狀態轉移機率為:

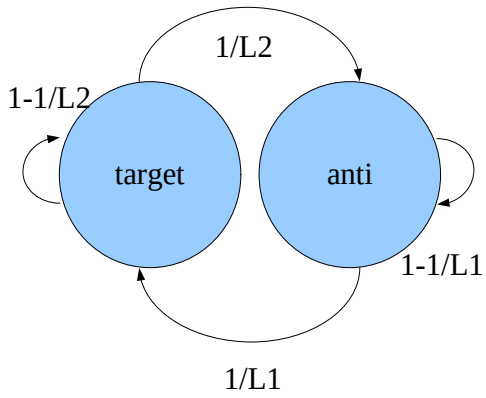


圖 5.9：發音方法音長模型狀態轉移圖。

以下我們將加入狀態轉移機率分數之後的偵測結果與之前 frame-based 的 MLP 偵測器之偵測結果比較:

表 5.2：加入音長模型之後的錯誤率統計。

	MLP	MLP + duration model		
manner	EER(%)	False alarm rate(%)	False reject rate(%)	Frame error rate(%)
Vowel	8.29	8.71	7.05	7.92
Stop	9.98	6.99	10.53	7.13
Fricative	10.06	7.08	9.68	7.31
Affricate	9.17	7.74	8.47	7.80

Nasal	9.25	7.03	9.67	7.30
Liquid	9.16	6.91	9.26	6.95
Silence	5.72	3.59	8.21	4.52

雖然這裡求得的錯誤率並不是等錯誤率，但是我們仍舊可以從統計的結果看出偵測錯誤率下降的現象，如上表當中用紅色數字標記錯誤率的 Fricative 以及 Affricate，其偵測的結果無論是錯誤警戒率(FA rate)或者是錯誤拒絕率(FR rate)都較原本的等錯誤率(錯誤警誡律=錯誤拒絕率)低的多，因此很明顯的這兩類的偵測器效能獲得很明顯的提升，至於 Vowel，Stop，Nasal，Liquid 這四類發音方法的偵測結果，其中一種錯誤率大幅的下降但是同一時間另外錯誤率卻小幅的上昇，雖然錯誤警戒以及錯誤拒絕率沒有同時下降，但是依照比例以及音框錯誤率(frame error rate)來看整體來說偵測器的效能依然是比原本沒有加上音長模型的效能要好，至於 Silence 偵測器由於錯誤警戒率以及錯誤拒絕率的變動以及差距較大，因此比較不能判斷偵測器的好壞。

下面我們觀察加上狀態轉移機率之後的偵測情形來分析，舉個 Nasal 的偵測情形觀察是否 jitter 類型的偵測錯誤能夠有效的被抑制:

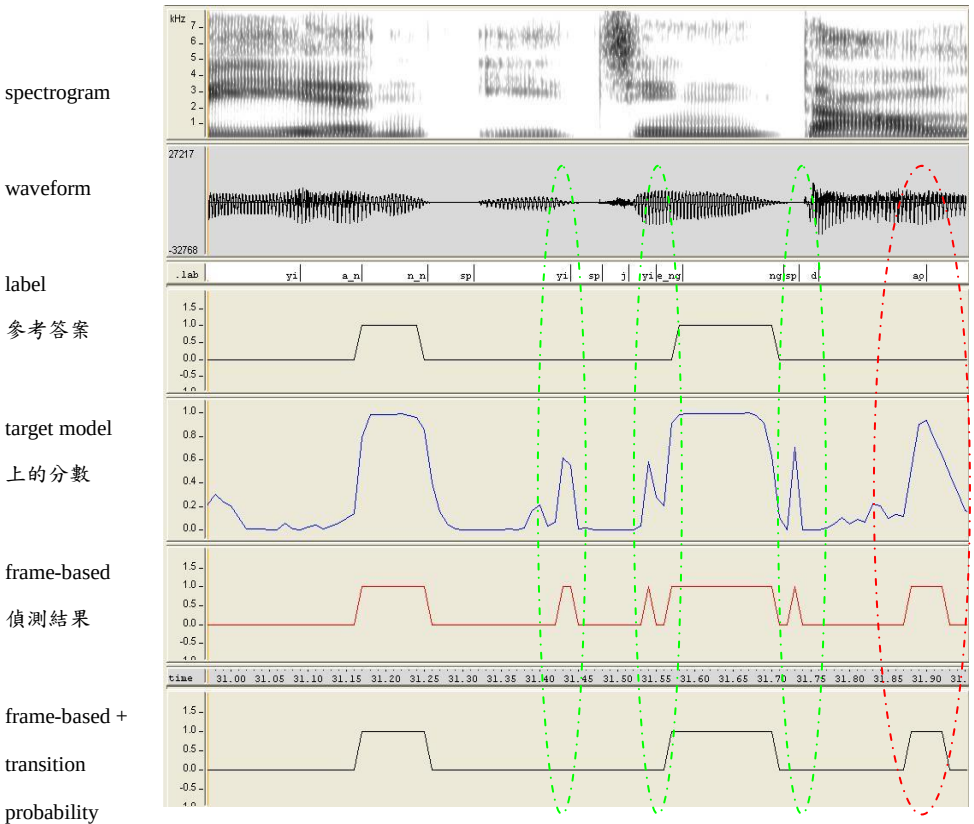


圖 5.10： Nasal 發音方法偵測實例。

從上面的例子可以看到，雖然一整個音段聲學特徵都非常類似 Nasal 而造成的整段偵測錯誤警戒(如紅色虛線所示)這類型的偵測錯誤沒有明顯改善，但是在 frame-based 偵測錯誤當中十分常見的 jitter 偵測錯誤類型(綠色虛線標示)的偵測錯誤幾乎都被排除了，代表說加入狀態轉移機率分數確實能夠有效的排除不合理的 jitter 類型偵測錯誤。

接著圖 5.11~5.17 我們統計出原本各發音方法音長段落的分布、frame-based MLP 偵測結果段落長度分布以及加入狀態轉移機率的偵測結果段落長度分布:

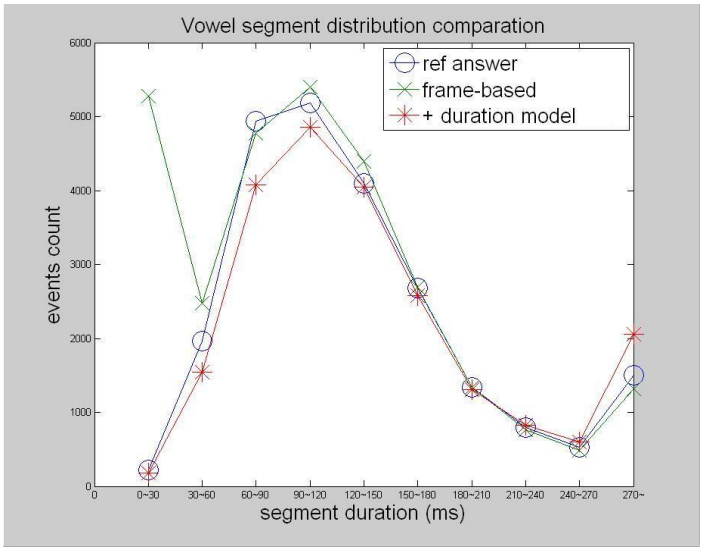


圖 5.11：Vowel 段落音長分佈比較(藍色為參考答案音長分布，綠色為 frame-based 偵測結果，紅色為加上轉移機率的偵測結果) 。

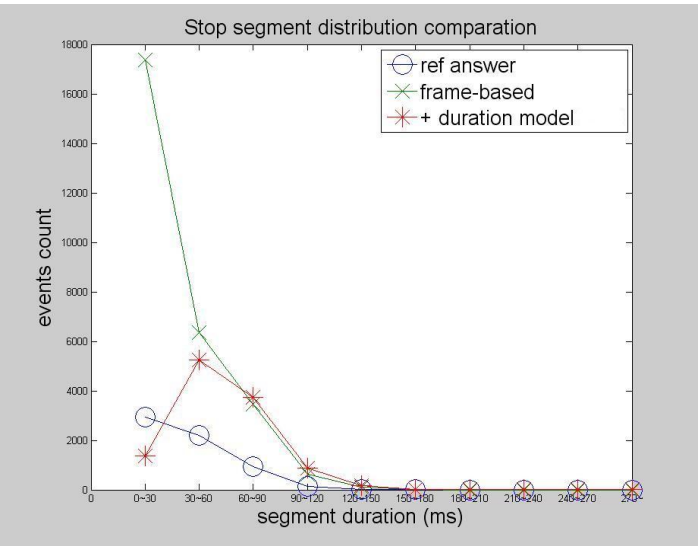


圖 5.12：Stop 段落音長分佈比較(藍色為參考答案音長分布，綠色為 frame-based 偵測結果，紅色為加上轉移機率的偵測結果) 。

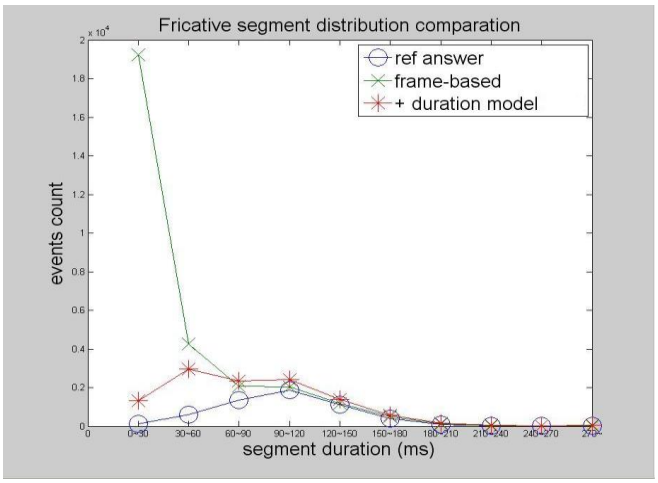


圖 5.13：Fricative 段落音長分佈比較(藍色為參考答案音長分布，綠色為 frame-based 偵測結果，紅色為加上轉移機率的偵測結果) 。

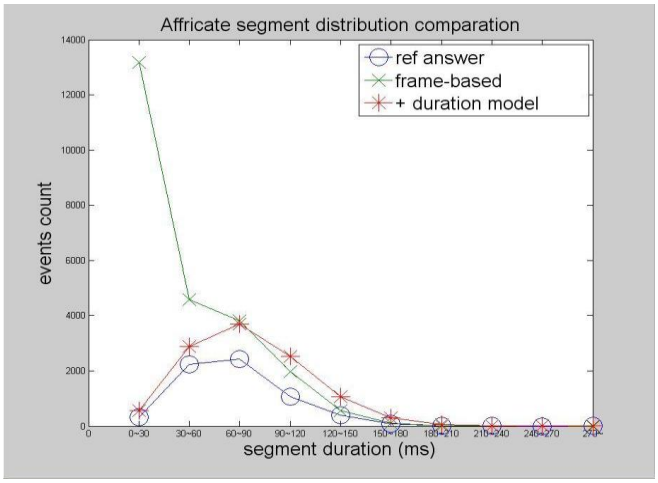


圖 5.14：Affricate 段落音長分佈比較(藍色為參考答案音長分布，綠色為 frame-based 偵測結果，紅色為加上轉移機率的偵測結果) 。

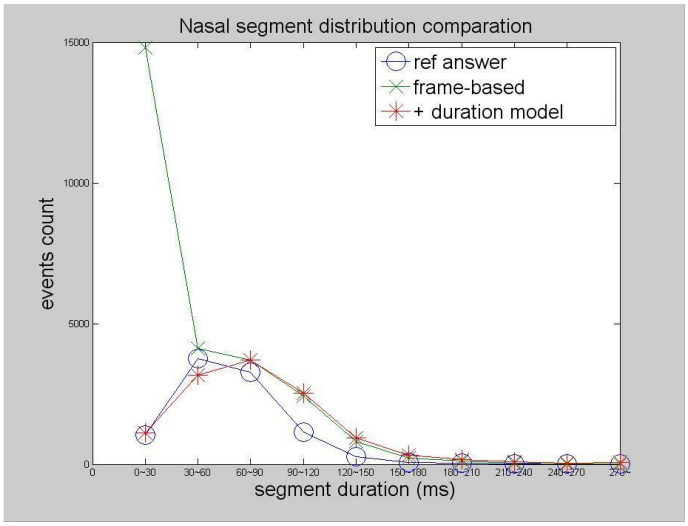


圖 5.15：Nasal 段落音長分佈比較(藍色為參考答案音長分布，綠色為 frame-based 偵測結果，紅色為加上轉移機率的偵測結果) 。

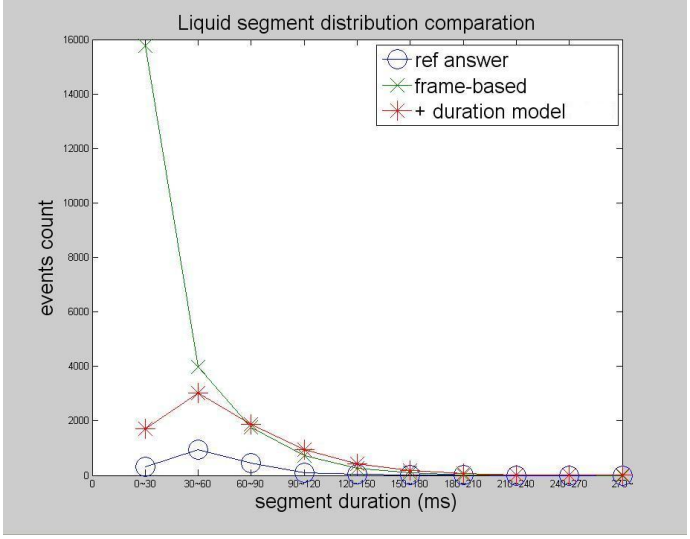


圖 5.16：Liquid 段落音長分佈比較(藍色為參考答案音長分布，綠色為 frame-based 偵測結果，紅色為加上轉移機率的偵測結果) 。

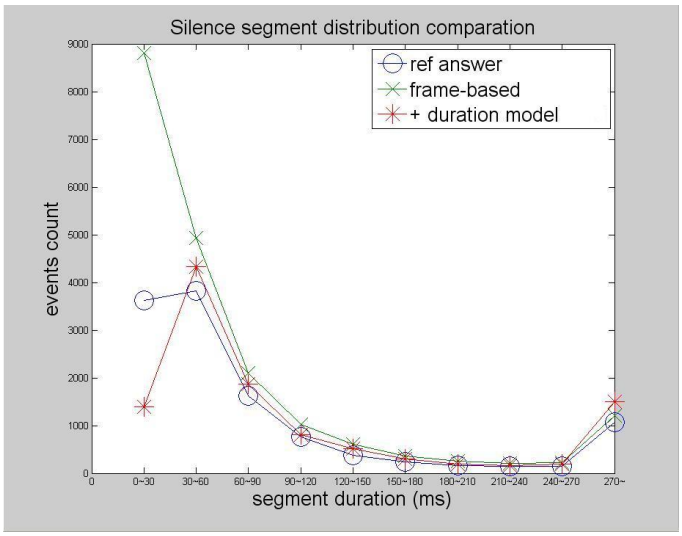


圖 5.17：Silence 段落音長分佈比較(藍色為參考答案音長分布，綠色為 frame-based 偵測結果，紅色為加上轉移機率的偵測結果) 。

由各種發音方法偵測結果音長分布的統計我們可以清楚的看到，原始的 MLP 偵測結果偵測出太多 jitter 型態的段落，造成了音長分布大多偏向音長很短的段落，而加入狀態轉移機率之後的偵測結果，等於說是加入各種發音方法在 segment 音段長度的資訊，使得偵測的結果 segment 長度分佈明顯較趨近於實際上的音長分佈。

不過一般來說偵測器的效能仍舊是以求得等錯誤率來評斷，因為等錯誤率考量到 target 資料量與 anti 資料量不一定相當的問題，因此接下來我們將導入以下的數學式加入一個可以調整的權重值，取得偵測器錯誤拒絕率與錯誤警戒率相等的等錯誤率。

$$AP'(S=1)=\frac{AP(S=1)\times C}{AP(S=1)\times C+AP(S=0)} \tag{5.5}$$

$$AP'(S=0)=\frac{AP(S=0)}{AP(S=1)\times C+AP(S=0)} \tag{5.6}$$

$AP(S=1)$，$AP(S=0)$ 分別為音框在 target model 以及 anti model 上的事後機率分數，C 是用來將錯誤率調適成等錯誤率的權重參數，$AP'(S=1)$ 為調適之後新的 target model 分數，$AP'(S=0)$ 為調適之後新的 anti model 分數。底下是調整權重使得錯誤率為等錯誤率的結果比較:

表 5.3：加入狀態轉移機率前後的等錯誤率偵測結果比較。

EER(%) / manner	frame-based MLP	frame-based MLP+ transition probability
Vowel	8.29	7.93
Stop	9.98	8.58
Fricative	10.06	8.32
Affricate	9.17	8.07
Nasal	9.25	8.30
Liquid	9.16	8.01
Silence	5.72	5.39

由上表的偵測結果可以看到 Vowel 與 Silence 的偵測器錯誤率僅降低了 0.3%，不過其餘各類的發音方法偵測結果等錯誤率均有 1%以上的下降，證明在等錯誤率的情形下加入 segment 資訊的偵測器效能比單純僅有音框資訊的偵測器要明顯提升許多。

統計完了等錯誤率的比較之後，我們將等錯誤率情形下的偵測結果與之前沒有調至等錯誤率的偵測結果(如圖 5.10)做比較觀察偵測結果段落的差異:

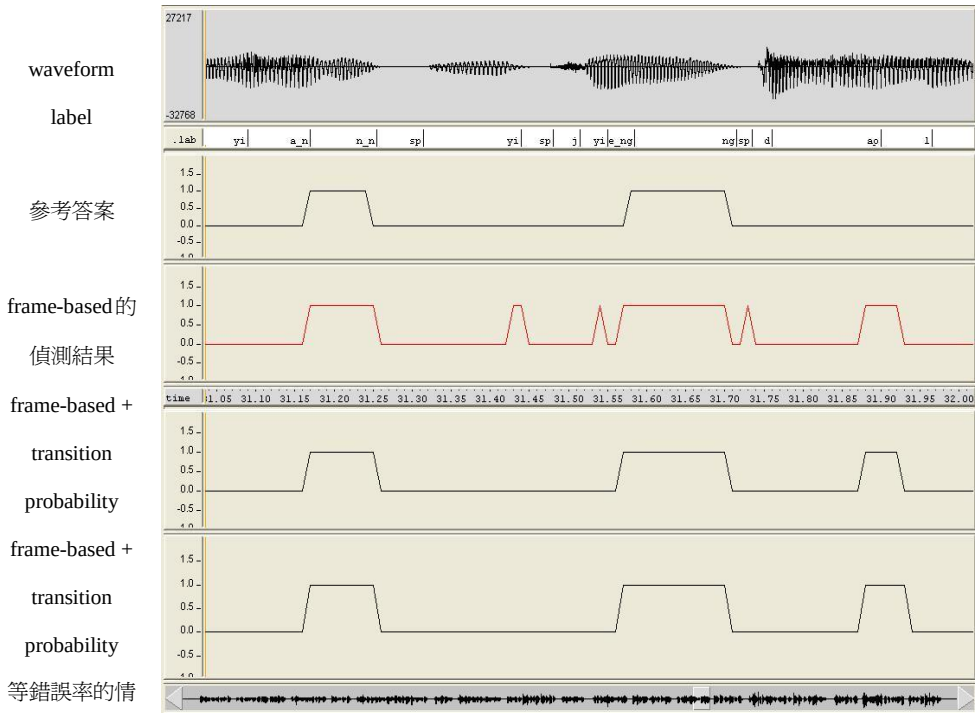


圖 5.18：調整成等錯誤率前後的偵測結果實例(I) 。

由這個例子可以看到偵測結果其實沒有什麼差異，同樣保留了去除 jitter 類型錯誤的優點，底下我們再找另外一個差異較明顯的例子:

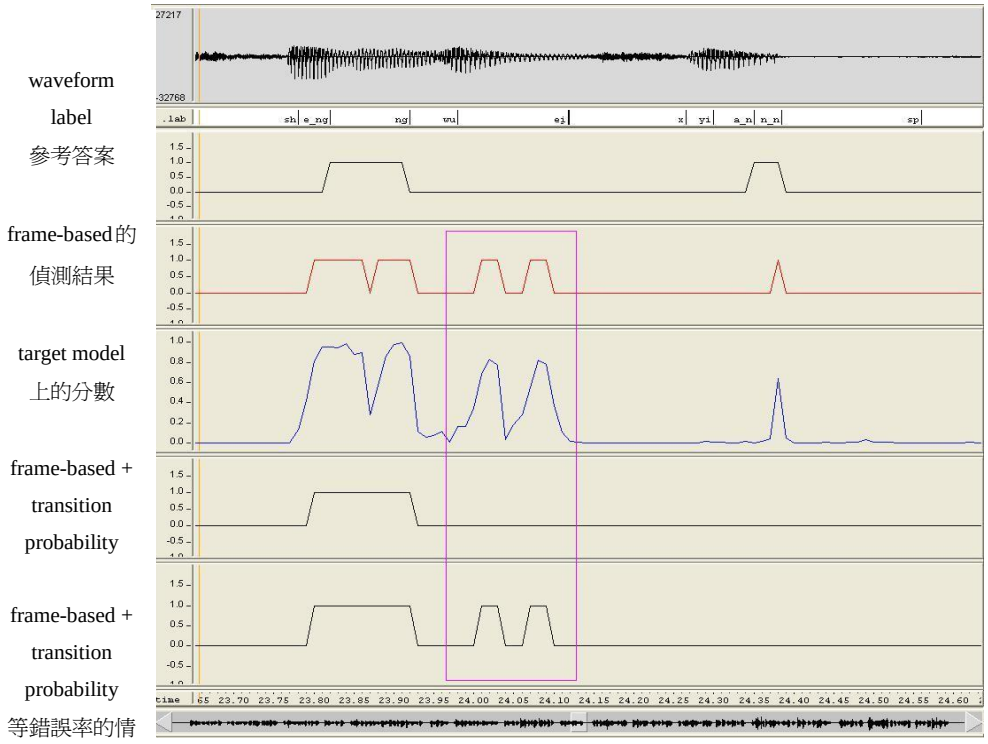


圖 5.19：調整成等錯誤率前後的偵測結果實例(II) 。

由圖 5.18、圖 5.19 觀察得知，調整成等錯誤率雖然會稍微增加一部分的錯

誤率，不過加入轉移機率最主要優點是抑制 jitter 這一點仍然保留了下來，在附錄當中有統計等錯誤率狀況下偵測結果的 segment 長度分布與原本未調至等錯誤率的分布差不多同樣證明了這一點，因此我們還是可以肯定偵測器加入 segment 的資訊能夠有效提升偵測器效能。

3. 以 frame-based MLP 偵測器為基礎之階層式信任度量測

在第一小節當中我們已經得到 frame-based MLP 發音方法的等錯誤率偵測結果，但是屬性偵測的目的是當作自動語音辨識系統的前端，提供可靠的語音資訊提供給後端的辨識器使用，因此得到偵測器的結果之後我們必須對於偵測的結果進行信任度的量測(Confidence Measure)，信任度較高的偵測結果才能提供有效的語音資訊給後級辨識器。

我們提出階層式(hierarchical)的信任度量測架構，最底層為七種發音方法加上呼吸聲(Breath)共八類偵測器的偵測結果進行信任度量測，而第二層將聲學特徵極為類似的 Fricative 與 Affricate 以及 Vowel 與 Nasal 分別合併，再上一層便將非響音(non-sonorant)包含 Vowel、Nasal、Liquid 以及響音(sonorant)的 Fricative、Stop、Affricate 合併在一起，而最上層便是將語音(speech)的部份包括 Vowel、Nasal、Liquid、Fricative、Affricate、Stop 合併以及非語音(non-speech)的部份包括 Silence、Breath 合併。架構運作的方式為欲確認偵測結果可靠程度之音框假使在最底層的偵測信任度就超過門檻值，我們就確認該音框的屬性偵測結果，假如說該音框在最底層的信任度低於門檻值，我們便對該音框進行第二層語音屬性分類偵測的信任度量測，假如該音框在第二層的偵測信任度量測高於門檻值，我們便確認該音框的偵測結果為第二層當中對應的屬性分類，假使該音框在第二層的信任度仍舊低於門檻，便將該音框進行第三層的信任度量測，以此類推，而下圖 5.20 是架構圖:

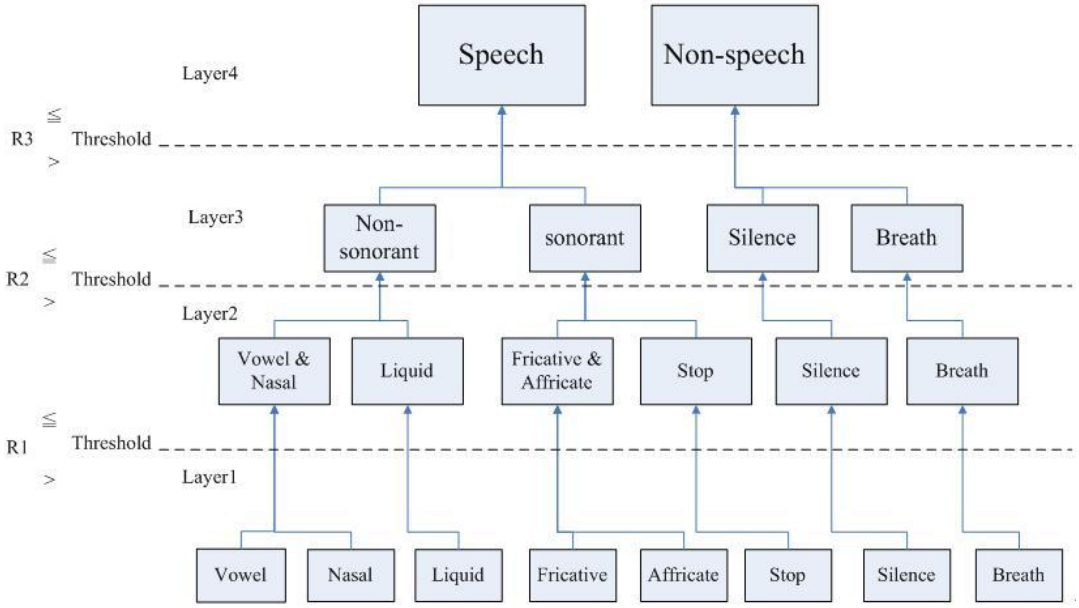


圖 5.20：階層式的屬性偵測器信任度量測架構圖。

為了計算屬性偵測結果的信任度，首先將八類偵測器的偵測結果 normalize 使其總和值為 1:

$$AP_{total} = \sum_{i=1}^{N} AP_i \qquad N=8 \qquad (5.7)$$

$$AP_i' = \frac{AP_i}{AP_{total}} \qquad 1 \leqq i \leqq 8 \qquad (5.8)$$

AP_i 為各種偵測器在 target model 上的分數，接著將上式 normalized 後的結果 AP_i' 運算每個音框偵測結果的亂度(entropy) [Mengusoglu, 2001]:

$$H = \sum_{i=1}^{N} AP_i' \log(\frac{1}{AP_i'}) \qquad N=8 \qquad (5.9)$$

計算出亂度 H 之後便將 H 代入 sigmoid function 得到信任度 R，亂度 H 若是越低會得到較高的 R 值，也就是說該音框的偵測結果較為可信:

$$R = \frac{1}{1+\exp(\lambda(H-\beta))} \qquad (5.10)$$

其中 λ、β 的值目前調整為適當的值使得 R 值的分布在 0~1 之間。得到信任度之後我們將信任度必須超過一個門檻值的音框偵測結果才認定為可靠的資訊。

以下是對於各個階層以不同的信任度門檻值分別求得的可靠資訊正確率:

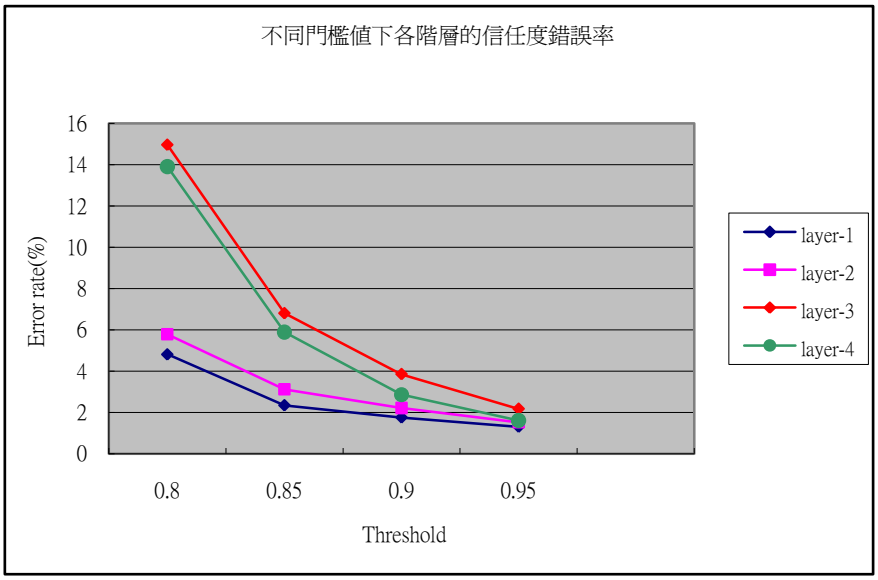


圖 5.21：不同門檻值下各階層的信任度錯誤率。

我們可以從統計的結果看到，如果將可靠度的門檻訂的越高，則提供給後級辨識器的資料的錯誤率就越低，底下我們統計各個門檻值之下，各階層信任度高於門檻值的資料涵蓋所有資料量的比例(inclusion rate)再與圖 5.21 一起做分析:

表 5.4：各門檻值下各階層信任度高於門檻值的資料比例。

門檻值 / 階層	0.8	0.85	0.9	0.95
1-layer	28.6%	21.8%	16.7%	10.3%
2-layer	20.9%	21.7%	20.9%	19.2%
3-layer	23.6%	21.0%	19.8%	18.5%
4-layer	26.9%	27.1%	27.0%	30.0%
Total	100%	91.6%	84.4%	78.0%

對於第一層而言分類的數目最多，因此若是信任度量測要高於門檻值，必須要僅有一類發音方法偵測器的分數比其餘發音方法偵測器分數高很多，因此隨著門檻值提高，涵蓋率有明顯的下降趨勢，同時由於分類較細，因此在高門檻值之下較不容易發生信任度量測錯誤的現象。而第二層將容易混淆的 Fricative 與 Affricate 合併，同時將容易因為鼻音化母音與鼻音韻尾相混淆的 Nasal 及 Vowel 合併，從圖 5.21 上可以得到此一合併的結果能夠得到不錯的錯誤率，不過由於合併之後的分類涵蓋了一種以上的發音方法，因此同一分類當中個別發音方法的偵測錯誤可能在分類合併時加成在一起，造成信任度錯誤率稍有提高。而第三層

將響音(sonorant)包含 Vowel、Nasal、Liquid 合併，並且將非響音包含 Fricative、Affricate、Stop 合併，由統計的結果看來此一合併的信任度量測結果由於多種發音方法的個別偵測錯誤加成在一起，因此在較低的門檻值下信任度量測的效能會大幅的降低。而最上層的語音(speech)與非語音(non-speech)的信任度量測同樣有與第三層類似的問題，在較低的門檻值下錯誤率升高的很快，不過整體來說錯誤率要比第三層為低。綜合圖 5.21 以及表 5.4 的結果，我們將在不同門檻值下整體信任度量測正確資料量的比例做統計:

表 5.5：各門檻值下信任度量測整體正確率。

門檻值	0.8	0.85	0.9	0.95
涵蓋率	100%	91.6%	84.4%	78.0%
信任度量測整體正確率	90.1%	95.4%	97.3%	98.3%

由上表的統計可以看到隨著門檻值越來越高，信任度量測的涵蓋率便隨之降低,但是即使將門檻值提高到 0.9 以上,信任度量測的涵蓋率依然有大約 85%，同時信任度的正確率達到 97%以上，不過值得注意的是，每個階層的分類不同，因此各個階層不一定要用相同的門檻值，由統計中可以看出，即使在門檻值為 0.8 時涵蓋率僅有 28.6%，代表說在最底層其實將很多的資料認定為”不可靠”而進行更上層的信任度量測。有了信任度量測結果的統計之後接著在下圖當中我們對於信任度量測的結果以最底層的信任度量測結果與各偵測器的偵測結果之間的關係做初步的分析:

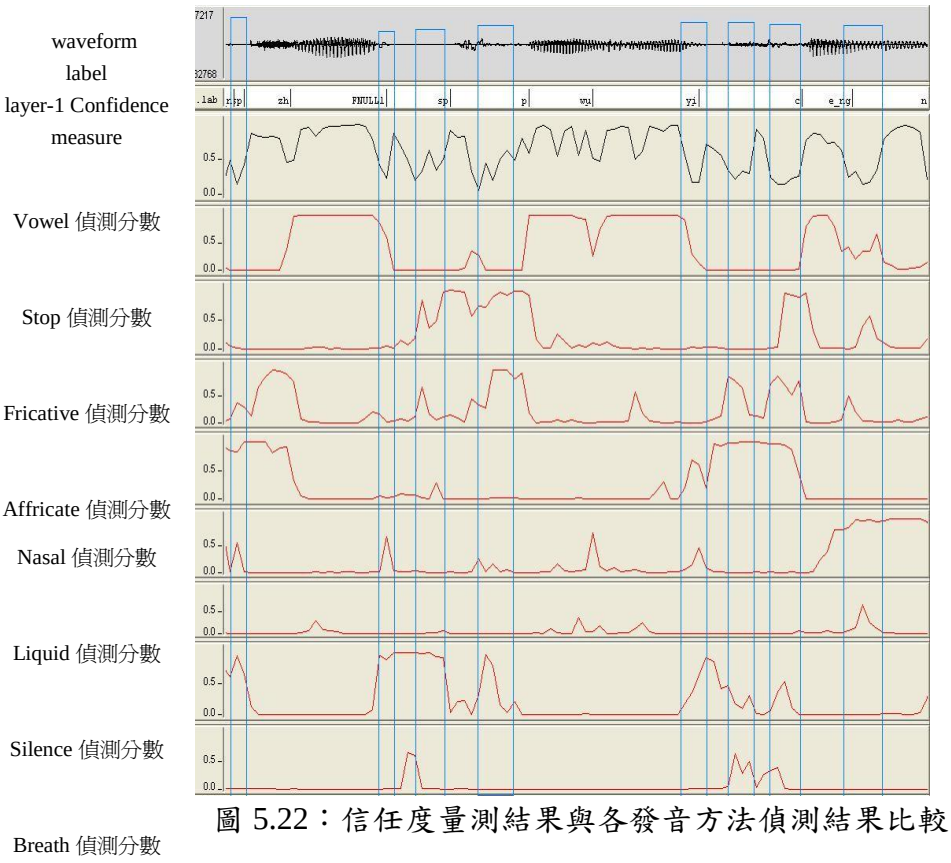


圖 5.22：信任度量測結果與各發音方法偵測結果比較。

從上圖中可以看到，大部分信任度不足的情形(如藍色框框標示)發生在音素間的交界附近以及發音方法容易混淆的地方，因為音素間交界的部份往往因為前後音素不同兩類發音方法偵測分數都很高，因此該音素交界區域偵測結果的亂度較高造成信任度的降低，同時容易混淆的音框往往有兩類以上的發音方法偵測分數均頗高，因此造成偵測結果同樣不可靠，而在較不容易混淆的音框由於大部分僅有一類發音方法偵測的分數很高，因此該區段偵測結果的亂度較低所以信任度也就相對的提高，代表說該區段的偵測結果是較為可信的。

六、 語者調適之 HMM 自動切割與使用 CRF 之語音屬性整合

1. 語者調適之 HMM 自動切割

傳統 HMM 的 training criterion 本來就不是 optimal segmentation position，我們檢視傳統的 HMM 語音辨認架構在做音節切割時的精確度，根據文獻記載，在 ±20 msec (也就是視傳統的 HMM 的 2 個 frame)的誤差下，精確度也僅能達到 90% 上下[Grande, 2003] [Kotropoulos, 2008]。。雖然，王新明博士提出了 minimum segmentation error 的 HMM 訓練方法[Kuo, 2006]，但是是一個 supervised training algorithm，也就是需要有人工正確切割語料庫來訓練模型。在計畫中我們使用了一套以語者調適訓練的 HMM 模型來對不特定語者作切割的機制，根據言厭結果發現可獲得精確度較高之切割資訊。如圖 6.1 所示，在不特定語者 HMM phone-like unit model training 後，我們再使用做 speaker adaptation training(SAT) [Makhoul, 1996]；SAT 就是使用 constraint MLLR(CMLLR)對不同語者做做語音參數的轉換；使用經語者轉換(CMLLR)後之語音參數再重新訓練新的 HMM 模型將可獲得較佳之 speaker-dependent HMM 模型。做完 SAT 後，我們再做 HMM 做 model adaptation，使用 MLLR 技術來調適 HMM 模型[Gales, 1996] [Gales, 1998]，它和 SAT 會又加成性的效果。如此就可以獲得較佳的 HMM 模型來做 force alignment，作為語料庫 syllable boundaries 的啟始切割位置。

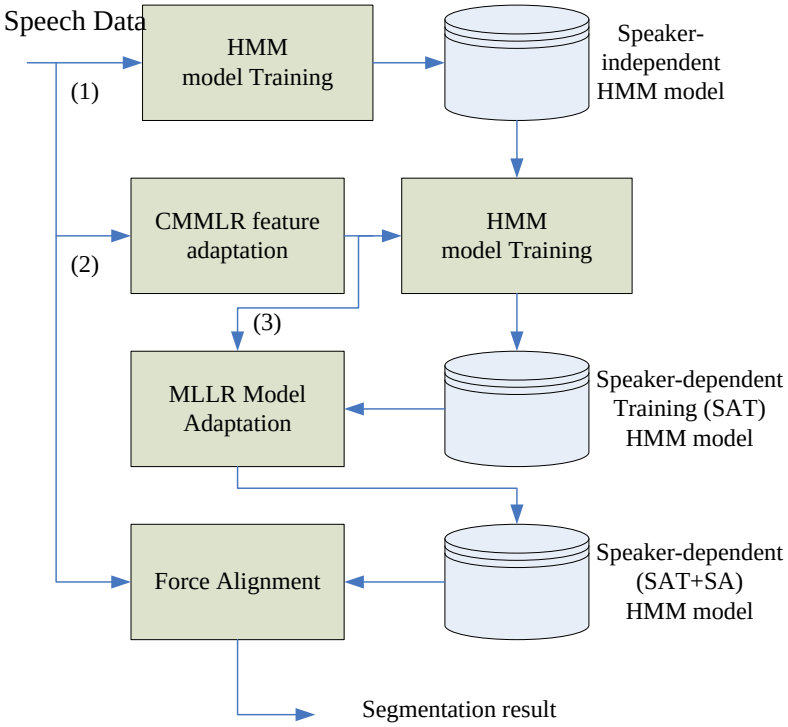


圖 6.1：使用語者調適之 HMM 自動切割流程圖。

使用語者調適 HMM 模型後，經觀察 phone-like unit 之切割位置在一些明顯錯誤處有所改善，而平均音長分佈如表 6.1 所示。

表 6.1：使用語者調適 HMM 模型後的發音方法平均音長統計。

發音方法	調整前平均音長	調整後平均音長
Vowel	9.49	9.92
affricate	7.95	9.28
Liquid	6.13	6.45
fricative	10.88	11.42
Nasal	6.90	6.39
Stop	4.89	5.28
Silence	4.83	-
Breath	16.67	-

2. MLP 為基礎的語音屬性偵測器

使用語者調適 HMM 模型後之自動標示資訊，重新訓練 MLP 為基礎的發音

方法偵測器，其效能如圖 6.2 所示。

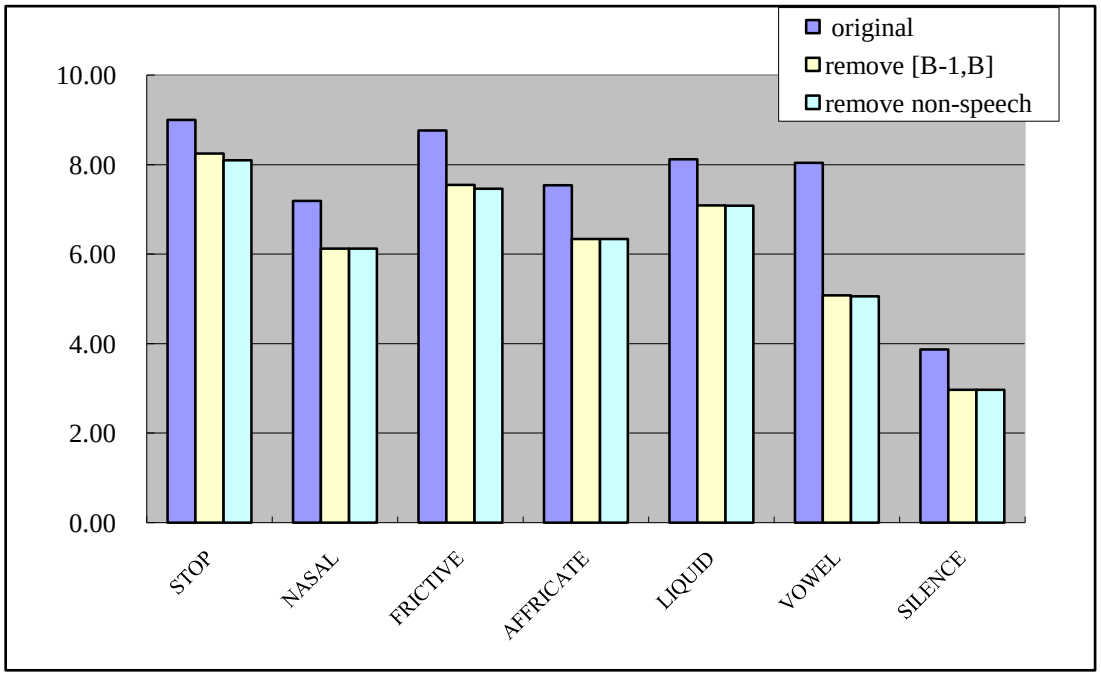


圖 6.2：使用語者調適 HMM 模型後之自動標示資訊，重新訓練之 MLP 為基礎的發音方法偵測器之效能(EER in %)。

偵測器之效能明顯較先前的結果好。因為使用語者調適 HMM 模型做自動標示及語音屬性偵測器是以音框為單位；所以切割位置誤差在一個音框以下可認為是容許誤差，所以在圖 6.2 也標示出考慮切割位置正負一個音框之容許誤差後之發音位置偵測器之效能。最後因 TCC 語料中有一些背景雜訊存在，若將自動切割時標示為非語音或靜音信號之音框其偵測結果後，偵測器之效能會進一步提升；由途中也可以看出非語音信號常會被偵測為 stop。

重新訓練之 MLP 為基礎的發音位置偵測器之效能則如圖 6.3 所示。由圖 6.2、6.3 所示，以自動標示之語料及音框為單位的語音屬性偵測器而言，這樣的效能可以說是相當的好了。

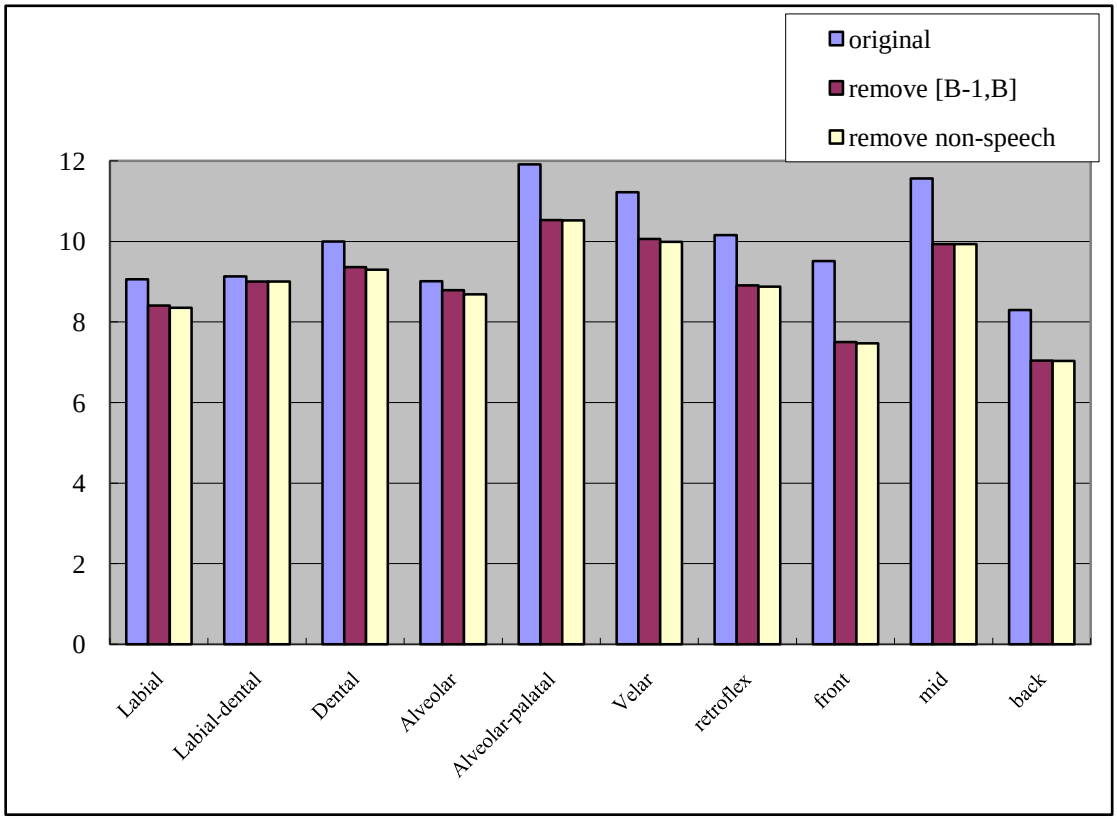
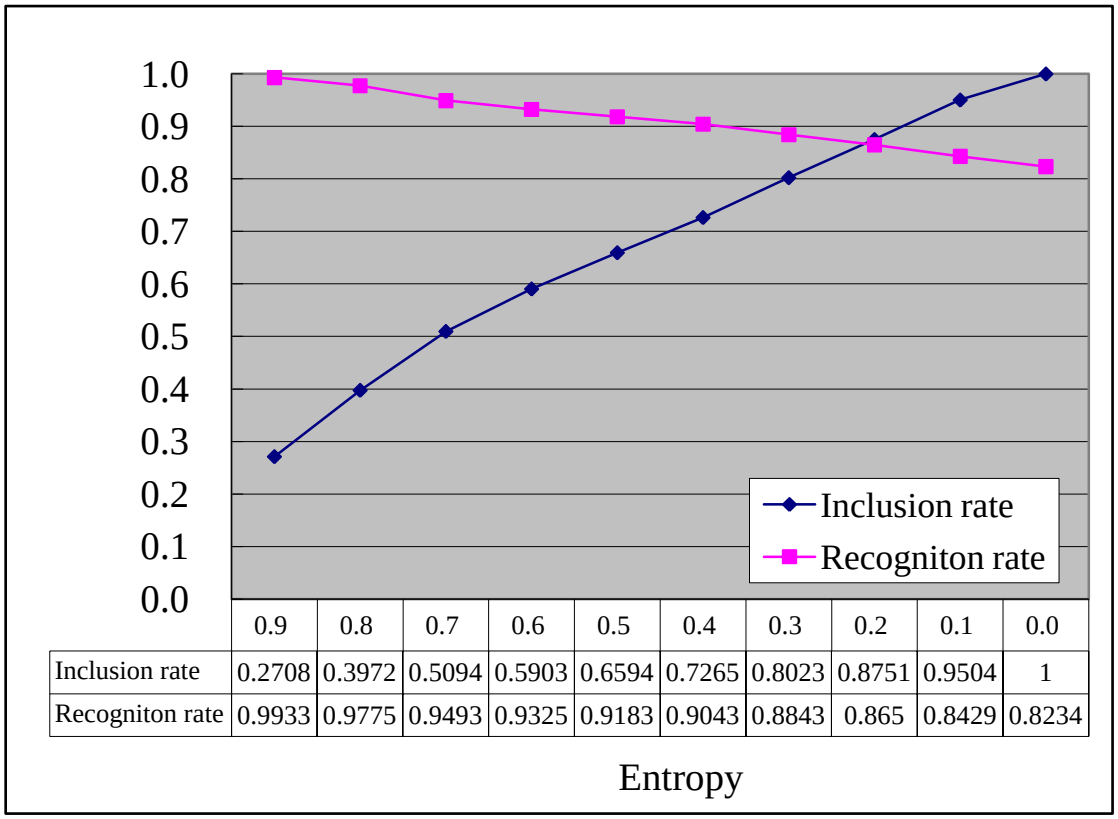


圖 6.3：使用語者調適 HMM 模型後之自動標示資訊，重新訓練之 MLP 為基礎的發音位置偵測器之效能(EER in %)。

3. 附加信任度量測的語音屬性辨認器

接著我們使用前面 5.3 節所提出之觀念為 frame-based MLP 偵測器之辨認結果加上信任度量測。我們對重新訓練的 MLP 發音方法偵測器之輸出做信任度量測，在圖 6.4 中可以看到對不同的 confidence threshold 時，以 frame-based MLP 偵測器輸出所製作之辨認器之含概率及其辨認率。由圖 6.4 可以看見若將 frame-based MLP 偵測器的信任率 threshold 設為 0.4，則有 72%的音框可辨認出發音方法而其辨認率可達 90%，可以看出結果較前面 5.3 節為佳。一般若在發音方法改變之邊界位置正負兩個音框範圍外的音框總數大概是佔總音框數的 70-80%。



	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0.0
Inclusion rate	0.2708	0.3972	0.5094	0.5903	0.6594	0.7265	0.8023	0.8751	0.9504	1
Recogniton rate	0.9933	0.9775	0.9493	0.9325	0.9183	0.9043	0.8843	0.865	0.8429	0.8234

圖 6.4：frame-based MLP 發音方法偵測器之辨認結果之信任度量測。

4. 語音屬性偵測器之錯誤分析

我們將本章所製作之利用 MLP-based 發音方法偵測器之錯誤列於表 6.2。例如第一行第二列是說 54%的/*h*/音素會使得 stop 發音方法偵測器有輸出，第二行第四列是說 17%的/*e-ng*/音素會使得 nasal 發音方法偵測器 miss detected。

我們將表 6.2 所得到的結果與語言學知識做比對可以發現一些在語言學家看到的現象(linguistics knowledge) [謝國平, 1998]，如：

- Backward nasal assimilation – 國語語音中鼻音韻尾的同化現象；
- Vowel unvoicing –韻母的非韻母化現象；
- denasalization – 去鼻音化現象，通常是因為進入鼻腔的空氣量減少所造成；

所以由圖 6.5 可以發現鼻音發生整個 phone missing detection 的比例事實上還算較高的。

表 6.2：最常見的發音方法偵測錯誤。

STOP	/h/ 0.54	/f/ 0.41	/ch/ 0.22	/e/ 0.22	/er/ 0.20	/c/ 0.19
NASAL	/l/ 0.27	/FNULL2/ 0.17	/e_ng/ 0.17	/r/ 0.15	/e_n/ 0.14	/wu/ 0.14
FRICTIVE	/q/ 0.54	/c/ 0.42	/z/ 0.42	/k/ 0.41	/ch/ 0.38	/p/ 0.37
AFFRICATE	/s/ 0.55	/x/ 0.54	/sh/ 0.46	/t/ 0.32	/f/ 0.17	/d/ 0.10
LIQUID	/n/ 0.52	/m/ 0.41	/er/ 0.27	/yu/ 0.25	/d/ 0.21	/FNULL1/ 0.20
VOWEL	/r/ 0.35	/n/ 0.25	/d/ 0.19	/l/ 0.16	/g/ 0.14	/h/ 0.13
SIL	/f/ 0.40	/p/ 0.18	/t/ 0.16	/c/ 0.14	/b/ 0.12	/k/ 0.12

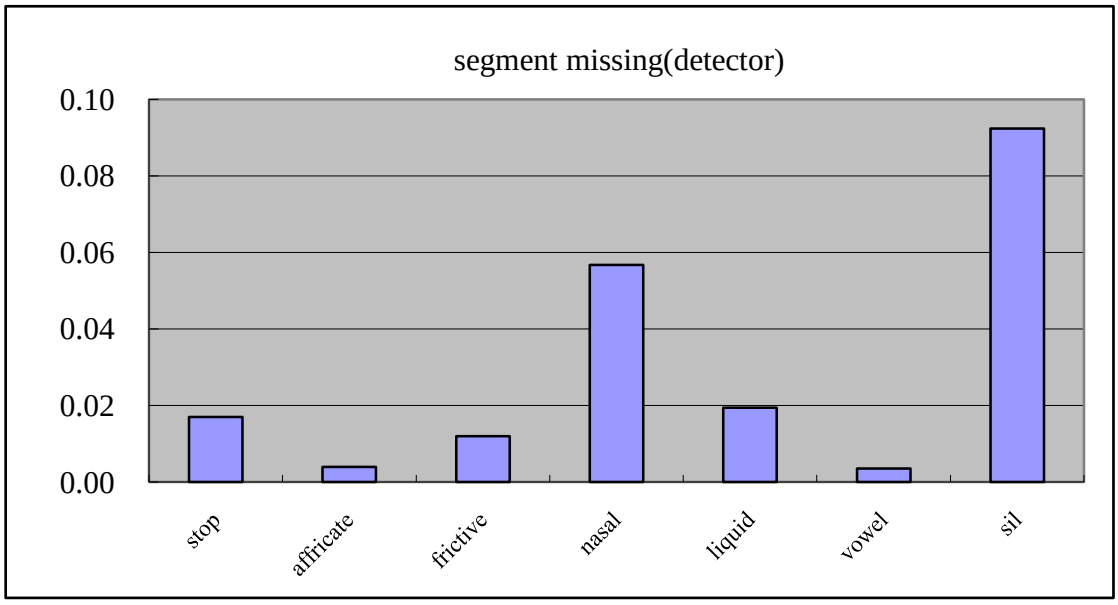


圖 6.5：frame-based MLP 發音方法偵測器中整個 phone missing detection 之統計。

接著，我們將利用 MLP-based 發音位置偵測器之錯誤列於表 6.3。

表 6.3：最常見的發音位置偵測錯誤。

Labial	/t/ 0.45	/n/ 0.44	/f/ 0.43	/h/ 0.34	/l/ 0.33	/k/ 0.32
Labial-dental	/b/ 0.60	/p/ 0.34	/g/ 0.33	/k/ 0.32	/h/ 0.27	/sil/ 0.23
Dental	/ch/ 0.76	/zh/ 0.70	/sh/ 0.63	/t/ 0.36	/f/ 0.31	/j/ 0.30
Alveolar	/p/ 0.73	/b/ 0.56	/k/ 0.49	/e/ 0.35	/h/ 0.33	/er/ 0.31
Alveolar-palatal	/ng/ 0.65	/m/ 0.59	/r/ 0.37	/sh/ 0.35	/zh/ 0.27	/c/ 0.24
Velar	/n_n/ 0.78	/m/ 0.39	/p/ 0.35	/n/ 0.32	/t/ 0.31	/wu/ 0.16
retroflex	/s/ 0.85	/z/ 0.81	/c/ 0.78	/x/ 0.58	/q/ 0.41	/j/ 0.36
front	/e_n/ 0.31	/r/ 0.23	/e_ng/ 0.23	/er/ 0.21	/FNULL1/ 0.18	/n/ 0.17
mid	/ou/ 0.36	/an/ 0.33	/ei/ 0.29	/d/ 0.26	/n/ 0.24	/eh/ 0.23
back	/ao/ 0.31	/e_ng/ 0.27	/g/ 0.27	/er/ 0.24	/h/ 0.23	/e/ 0.23

我們將表 6.3 所得到的結果與語言學知識做比對可以發現下列現象(linguistics knowledge)，如：

- Confusion set in Mandarin：如國語語音中捲舌與不捲舌聲母之混淆現象；
- Confusion set in Mandarin：ㄥ、ㄣ韻尾鼻音的混淆現象；
- Labial Assimilation：唇音同化現象 n_n -> m /_{labial, labial-dental}；

由上述錯誤觀察可以發現一些 linguistics knowledge 可以解釋語音與性在連續語音中有一些現象是過去的語音辨認器中尚未或因系統架構複雜而難以考慮的。而在 NG-ASR 的架構中則因將這些 linguistics knowledge 加入。

5. 使用 CRF 之語音屬性整合

我們將 frame-based 發音方法偵測器之輸出當作一個 CRF (Conditional Random Field，其示意圖如圖 6.5)的輸入 X_i，而 CRF detector 之輸出為某個發音方法之 detection 輸出，在 CRF 中我們可以將 conditional probability 表示為

$$P(Y_v \mid X, Y_W, w \neq v) = P(Y_v \mid X, Y_W, w \sim v) \tag{6.1}$$

其中，v 是 CRF graph 中與 **w** 相鄰的 vertices。

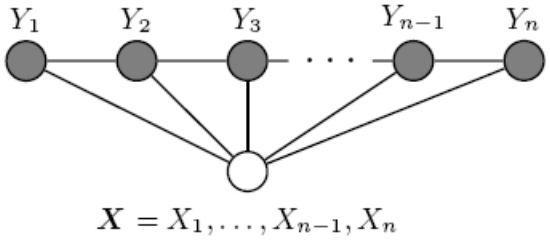


圖 6.5：CRF 示意圖。

我們可將 conditional probability 寫成

$$P(Y \mid X) = \exp\left(\sum_j \lambda_j t_j(y_{i-1}, y_i, x, i) + \sum_k \mu_k s_k(y_i, x, i) \right) \tag{6.2}$$

一般 $t_j(y_{i-1}, y_i, x, i)$ 稱為 transition function， $s_k(y_i, x, i)$ 稱為 state feature function。

在常用的 CRF 架構中 x_i 常以一個 discrete variable 表示，則

$$s_k(y_i, x, i) = \begin{cases} c; & (x_{i-n}, \cdots x_{i+m}) \text{ match to a specified pattern} \\ 0; & otherwise \end{cases} \quad (6.3)$$

所以我們將 MLP-based 發音方法偵測器之輸出 uniform quantized 成十等分來獲得 CRF 的 discrete 輸入資料。

在實驗中，我們使用[n-2, …, n+2]時間的 MLP-based 發音方法偵測器輸出來當 CRF classifier 的輸入，所獲得的發音方法 classifier 效能見表 6.4；表 6.4 為 CRF classifier 的 confusion table，由表中可以看到其中以 liquid 辨認率最低，最容易變成 vowel。

表 6.4：以 CRF 做語音屬性整合之發音方法辨認結果。

	p	a	f	n	l	v	s
p	0.86	0.03	0.04	0.01	0.01	0.02	0.03
a	0.01	0.92	0.05	0.00	0.00	0.00	0.02
f	0.02	0.05	0.90	0.01	0.00	0.01	0.02
n	0.00	0.00	0.00	0.91	0.02	0.06	0.01
l	0.03	0.01	0.00	0.05	0.74	0.17	0.00
v	0.00	0.00	0.00	0.01	0.01	0.97	0.00
s	0.00	0.00	0.01	0.00	0.00	0.00	0.98

其中 p-plosive, a-affricate, f-fricative, n-nasal, l-liquid, v-vowel, s-silence

七、 Tone Nucleus Model 及其在聲調辨認的應用

在計畫中，我們也對中文音調與韻律訊息偵測器進行研究，在這部分我們是與日本東京大學 Keikichi Hirose 教授進行合作，進行國語語音 tone neuclues 之搜取及 tone neuclues 在國語語音之聲調辨認上之應用研究。

Tone neuclus 是由東京大學的 Keikichi Hirose 教授及現在在北京语言大学信息科学学院任教的張勁松教授所提出；tone neuclus 是指國語音基週軌跡中之穩定部分，如圖 7-1 所示[Zhang, 2004]。如此將可將國語聲調以基週軌跡之平均值及斜率來描述。在圖 7-2 中則是 tone neuclues 抽取之演算法則，我們假設 pitch contour 可分為三段－onset(C1), nuclei(C2)及 offset(C3)，並分別使用直線來模擬機週軌跡，所以我們將 onset, nuclei 及 offset 的基週軌跡之平均值及斜率用高司分佈來描述，再使用 Viterbi search 來尋找最佳之分界點並取出 pitch contour 之 tone nuclei 部分。tone neuclues 能將 pitch 參數抽取不穩定部分去除，例如：voiced 及 unvoiced boundaries 附近。在有環境雜訊時，使用 tone neuclues 做國語語音之聲調辨認也可獲得較佳之效能。

其實 tone neuclus 很像語言學家對非聲調語言的基週軌跡的 stylization - 使用一些直線來描述基週軌跡，也就是 piecewise linear model。但是 tone nuclus model 則是 syllable-based，因為 syllable boundaries 對聲調是一個重要的信息。

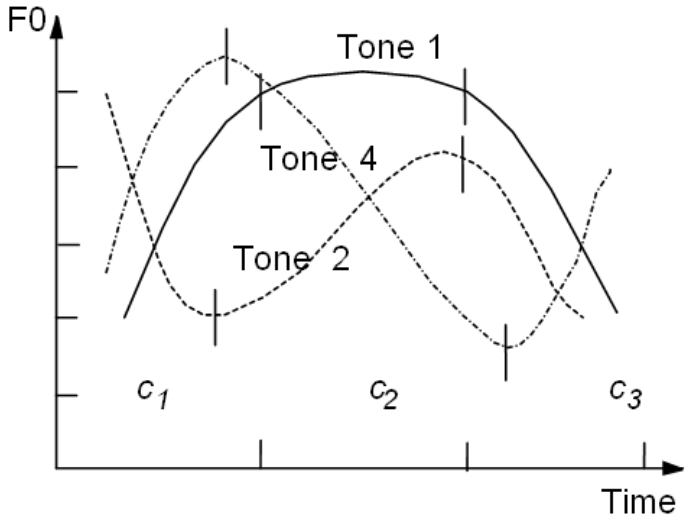


圖 7-1、國語 tone neucleus 之示意圖。

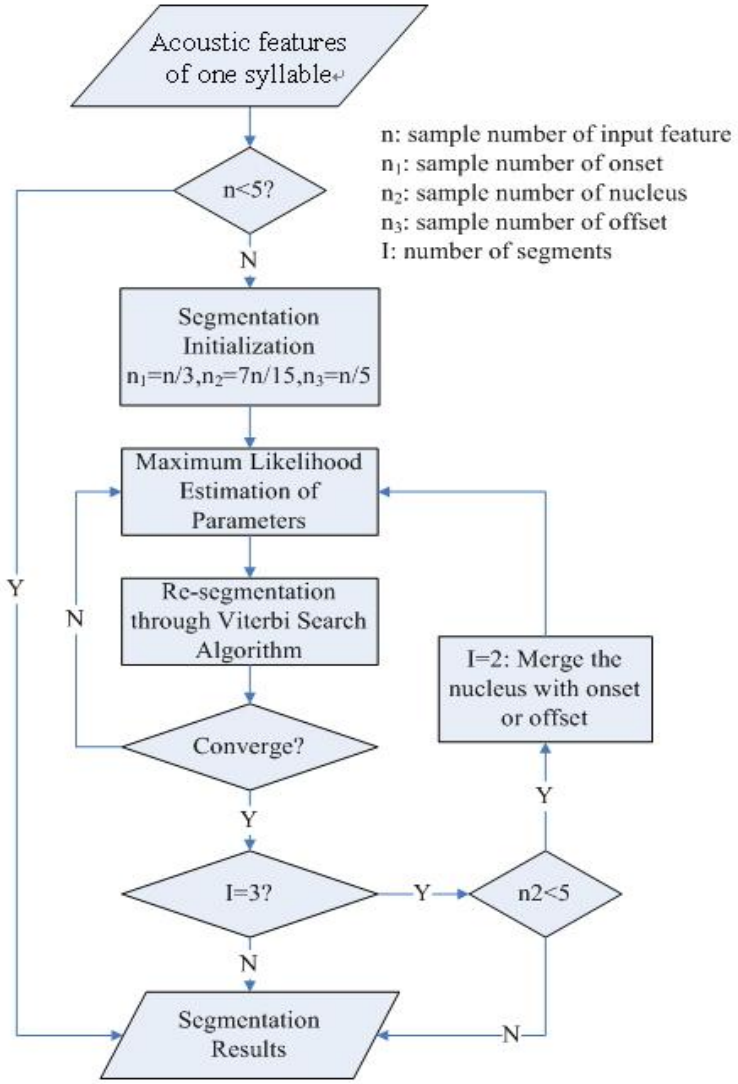


圖 7-2、國語 tone neucleus 之抽取演算法。

在國語連續語音之基週軌跡變化中一個音節的基週軌跡又會受前後文影響；語言學家將這些基週軌跡又會受前後文影響整理出一些規則，通稱為 tone sandhi。在國語連續語音之聲調辨認中，常會使用前後文相關(context-dependent)的模型，將一個 syllable 前後 syllable pitch contour 的資訊加入聲調辨認器之辨認參數。在張勁松教授的論文中則提出 tone anchor [Zhang, 2005]的觀念，其實就是 tone neuclues 間的 pitch jump/difference 會有特定的 pattern，或是語言學家所說的 tone sandhi 而已。在張勁松教授的論文中則提出 tone anchor 觀念常使用的參數則如圖 7-2 所示。

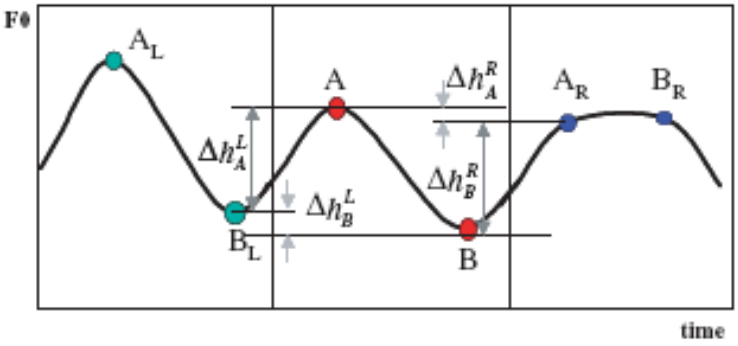


圖 7-2、tone anchor 常用參數示意圖。

在本計畫中，所使用的基週軌跡參數分別為：

1) tone neuclues 中 onset (C1), nuclei (C2)及 offset (C3)的能量、log-F0 平均值及 log-F0 斜率，各段起點之 log-F0 平均值;
2) 基週軌跡長度;
3) 前一音節 offset(C3)段的能量、log-F0 平均值及 log-F0 斜率, 後一音節 onset (C1)段的能量、log-F0 平均值及 log-F0 斜率;
4) 四個 tone anchor 參數，如圖 7-2 所示;
5) Syllable 前後 unvoiced 長度;
6) 兩個 indicator 來標示有無前一 syllable 及後一 syllable;

共 27 為的特徵向量。

在實驗中，我們使用香港大學之 HKU96 國語語料庫進行 MLP-base 的中文語音聲調辨認研究[7]。我們所使用的訓練語料有 500 句，6,419 個音節；測試語料有 200 句，2,567 個音節。在實驗中，我們做三種聲調辨認器並比較其結果：(1) Baseline 1 - 使用 MLP 聲調辨認器，其輸入特徵參數為前述 1), 2), 3), 5), 6)，也就是不使用四個 tone anchor 參數；(2) Baseline 2 - 張勁松教授先前所提出之 HMM 聲調辨認器，詳見[Zhang, 2005] ；(3) 所提出之系統 - 使用 MLP 聲調辨認器，其輸入特徵參數除 baseline 1 系統所用，還加入前述 4)之 tone anchor 參數。

上述三系統所獲得之結果如所示圖 7-3。而所提出之系統之 confusion table 則如表 7-1 所示。

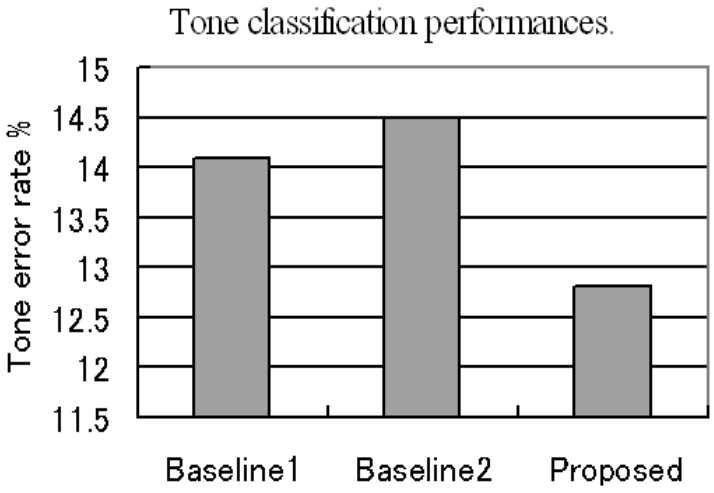


圖 7-3、使用 tone anchor 參數及基本聲調辨認系統效能比較圖。

表 7-1、 Confusion matrix for the proposed system (in %)。

Tone	T1	T2	T3	T4	T5
T1	**75.7**	6.34	0.6	15.6	1.7
T2	4.4	**88.9**	2.1	3.6	1.1
T3	0.6	3.5	**84.8**	9.3	1.2
T4	1.4	0.9	2.1	**94.8**	0.8
T5	3.2	2.5	10.8	15.8	**67.7**

由以上實驗，我們可以發現跟基週軌跡有關的語音屬性中 tone anchor 參數是一個對國語聲調描述十分有用的語音屬性。

八、 語音屬性偵測器之應用 - 利用屬性偵測概念做環境匹配調適

我們將語音屬性偵測器的概念運用於雜訊環境下之語音辨認效能之改進，只不過我們將原來偵測器所使用的輸入參數 frame-based 的頻譜相關參數的觀念使用到語音的 eigen-voice [Kuhn, 1998]上。在雜訊環境下之語音辨認時，我們可以將語音信號由原頻譜相關參數空間轉換至 eigen-voice 空間，然後我們在 eigen-voice 空間做一個語音特性偵測器，使用 eigen-voice 空間上之語音特性偵測器之資訊隊員語音辨認分數作重新計算分數(rescore)的工作；其系統架構圖如圖 8-1 所示。

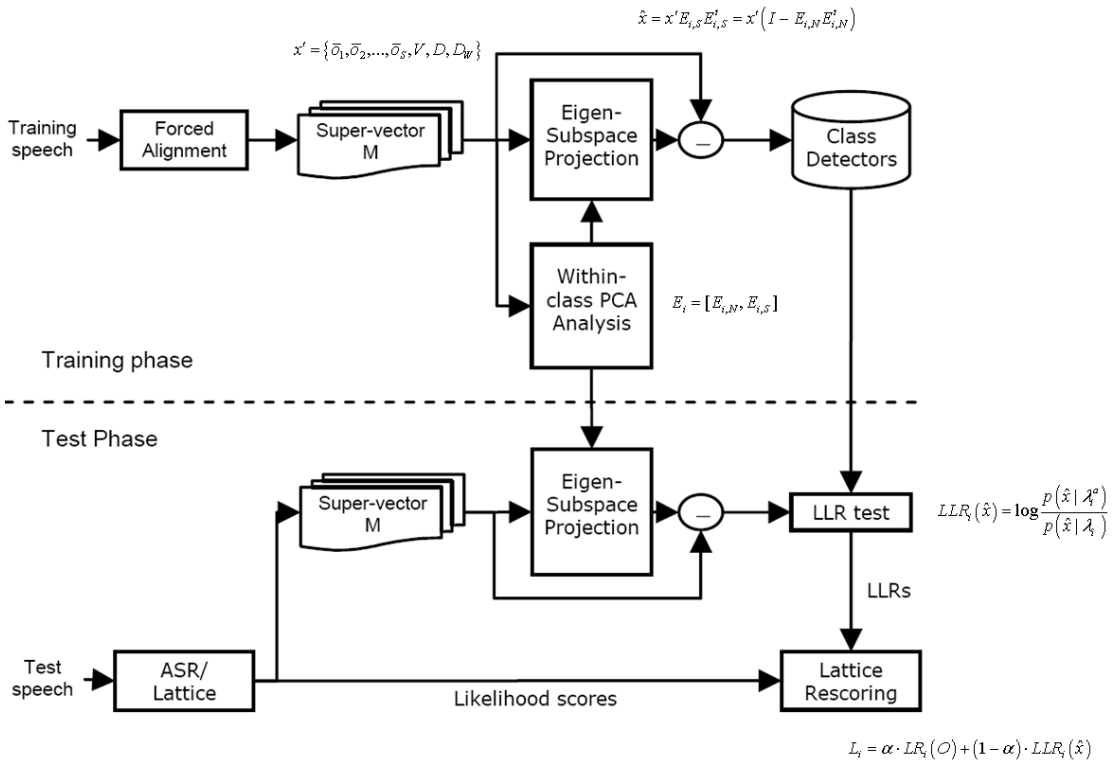


圖 8-1、利用屬性偵測概念做環境匹配調適辨認。

在圖 8-1 中，我們在訓練時首先將訓練語料用 HMM 做 force alignment，然後將求每一個 state 的 MFCC 參數之平均向量後，將每個字個個 state 的 MFCC

參數平均向量組成個 super-vector。

求得 super-vector 後，我們可以將帶辨認的字分為若干群， W_i $i=1, ..., N,$ 各自建立其 covariance matrix A_i。 接著我們對 covariance matrix, A_i, 用 PCA 做分解，建立 eigen-space, E_i。對每一個 eigen-space, E_i,可以分解為兩部分子空間 – 前幾維事實上會對應到語者及環境雜訊種類，我們將之表示為 $E_{i,N}$ 。其他的維度則用用來表是不同的 W_i ，我們將之表示為 $E_{i,S}$。也就是我們可以將 eigen-space, E_i, 表示為

$$E_i = [E_{i,N}, E_{i,S}] \tag{8.1}$$

如果一個字的 super-vector 為 x' ，屬於第 i 群，我們可以將它在與語者及環境雜訊種類有關的子空間 $E_{i,N}$ 的分量(投影)移除，以其提高語音辨識效果。將輸入語料在與語者及環境雜訊種類有關的子空間 $E_{i,N}$ 的分量移除的動作可以下式表示

$$\hat{x} = x' E_{i,S} E_{i,S}^t = x' \left(I - E_{i,N} E_{i,N}^t \right) \tag{8.2}$$

我們將上述的環境匹配調適稱為 within-class feature normalization(WCFN)。

接著我們對使用一組偵測器來偵測經 WCFN 後之參數 $\hat{x}$，是否屬於第 i 群。我們所使用的偵測器是 MLP 類神經網路， MLP 類神經網路偵測器的輸出是 $p\left(\hat{x} \mid \lambda_i\right)$ 。同時我們也可以訓練一組 anti-model 的 MLP 類神經網路偵測器其輸出為 $p\left(\hat{x} \mid \lambda_i^a\right)$。由上述兩組 MLP 類神經網路偵測器，我們可以找到 WCFN 後之參數 $\hat{x}$ 屬於第 i 群及不屬於第 i 群的 log-likelihood ratio (LLR) 如下式所示：

$$LLR_i\left(\hat{x}\right) = \log \frac{p\left(\hat{x} \mid \lambda_i^a\right)}{p\left(\hat{x} \mid \lambda_i\right)} \tag{8.3}$$

在辨認時，我們就可以利用上面屬於第 i 群及不屬於第 i 群的 MLP 類神經網路偵測器的輸出所獲得的 LLR 分數來對原 HMM 語音辨認器之分數作重新計分的工作，新的辨認分數即是原辨認分數與 MLP 類神經網路語音群組偵測器的輸出所獲得的 LLR 分數的加權和，如下式所示

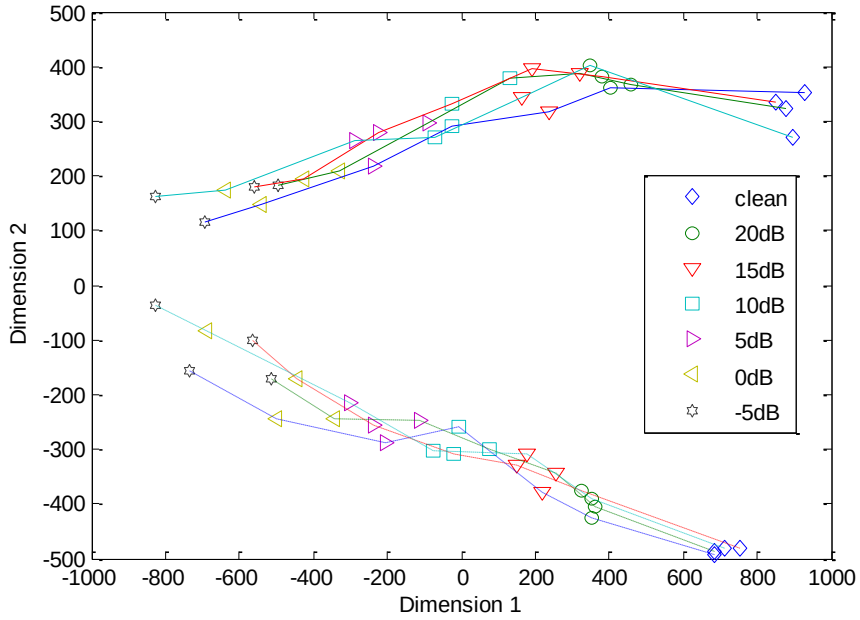
$$L_i = \alpha \cdot LR_i(O) + (1-\alpha) \cdot LLR_i(\hat{x}) \qquad (8.4)$$

其中α是 MLP 類神經網路語音群組偵測器的輸出所獲得的 LLR 分數的加權值。

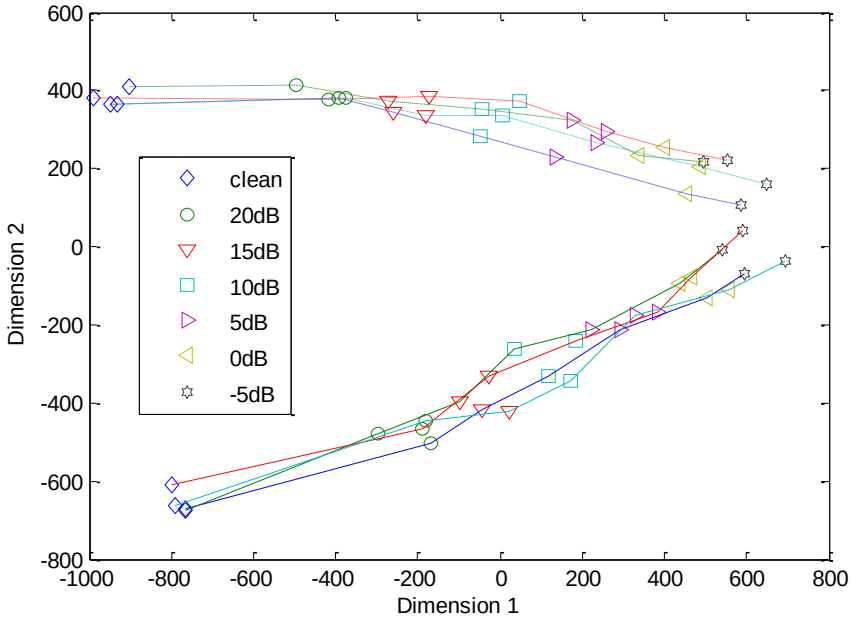
接著我們對 Aurora 2 語料庫做利用屬性偵測概念做環境匹配調適之辨認系統之實驗，其中之參數設定依據 Aurora 2 語料庫之標準設定。在實驗中我們將英文數字串中會出現的 12 個字各自視為一群。

首先，我們可以觀察我們求得 eigen-space 中的前兩維，如圖 8-2 所示，在圖中我們可以看到它們是與語者及環境雜訊種類有關的。

圖 8-2、在 Aurora 2 multi-condition 語料中所獲得 eigenvoice 中前兩維參數對不同訊雜訊比時之改變。



(a) Word – “one”



(b) Word – “two”

最後我們來看我們所提出之利用屬性偵測概念做環境匹配調適後之語音辨認器的效能比較。首先，我們先檢查 MLP 類神經網路語音群組偵測器效能如表 8-1 所示，前中所使用的參數為 16-state MFCC 參數平均值、MFCC variance 及 word 長度。

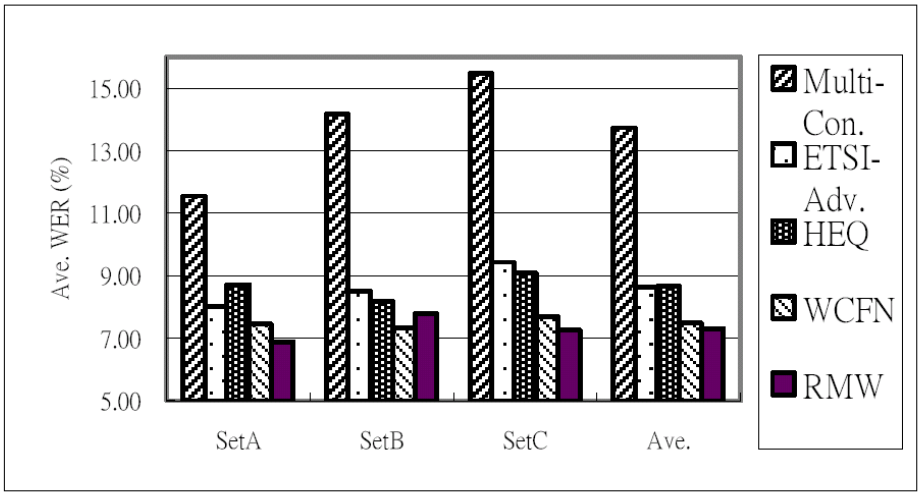
表 8-1、 MLP 類神經網路語音群組偵測器效能 (EER in %)。

Digit	MFCCs	+ Var.	+ Dur.
One	2.80	1.97	1.97
Two	3.41	3.13	2.56
Three	3.71	3.14	2.57
Four	2.86	2.57	1.73
Five	1.46	0.87	0.58
Six	1.98	1.98	1.42
Seven	1.72	1.72	1.43
Eight	1.72	1.72	1.72
Nine	2.90	2.32	2.03
Zero	2.56	2.56	1.70
Oh	3.41	3.13	2.84
Silence	0.15	0.15	0.15

Average	2.39	2.11	1.73

接著我們來看利用屬性偵測概念做環境匹配調適後之語音辨認器的辨認效能與其他已知方法的比較，如圖 8-3 所示。其中我們用來比較的方法有(1) 標準 multi-condition 訓練之辨認器 (2)使用 ETSI 前級之辨認器, (3) 使用 HEQ (histogram equalization) 之辨認器, (4) 使用這裡所提出 WCFN 參數之辨認器及 (5) 使用我們先前所提出 RMW (Reference Model Weighting) [Liao, 2007]參數之辨認器；其結果如圖 8-3 所示。由圖圖 8-3(a)中顯示使用 WCFN 參數之辨認器之 WER 為 7.45% 較方法(1)-(3)為佳。雖然低於 RMW，但 RMW 事實上是使用對環境雜訊的先驗知識；而 WCFN 則不需要。

圖 8-3、利用屬性偵測概念做環境匹配調適之辨認系統與其他系統之效能比較。



(a)

SNR(dB)	Multi-Con.	ETSI-Adv.	HEQ	WCFN	RMW
clean	1.35	0.98	1.21	1.10	1.27
20	2.47	1.55	1.42	1.19	1.23
15	4.05	2.25	2.15	1.80	1.79
10	6.38	4.17	3.76	3.34	3.10
5	14.53	9.44	9.00	7.89	7.57
0	41.73	25.82	26.95	23.22	22.86

(b)

九、 結論及計畫成果自評

本計畫在三年期間，建立了基本的國語語音屬性偵測器，可以提供國內相關研究的基本系統效能評比之用。在計畫進行中，我們發現對國內甚至國際上現有之國語語音資料中，缺乏一套像英語 TIMIT 語料庫一樣，有語音學家做到 phone-level 標示(transcript)且標示精確度到 sample 的語料庫。這對做 NG-ASR 架構語音辨認系統之研究事實分重要的資料，尤其是第一級的語音屬性偵測器。所以現在看到的一些 NG-ASR 研究多是以 TIMIT 語料庫作研究對象。所以本計畫中花了許多資源在使用工程方法整理 TCC-300 國語語料庫，希望成為一套國內相關研究人員在做相關研究時可以使用的語音資料庫。

在本計畫中完成下列工作並可提供國內相關研究人員在做相關研究時之基本資料庫

（1）TCC-300 國語語料庫之整理；
（2）TCC-300 國語語料庫 phone-like unit 自動切割位置之建立；
（3）國語發音位置及發音方法偵測器之建立；
（4）語言學知識(linguistics knowledge)之驗證；
- 如一些 phone assimilation 現象，在第二期的計畫中自動切割時考慮這些現象會是一個研究重點。

（5）利用 tone neuclus 這項基週相關語音屬性做國語語音聲調辨認；
（6）將語音屬性偵測器的觀念應用於傳統語音辨認器以改善其效能。

並發表下列與語音屬性偵測性相關之論文

1. Xiao-Dong Wang, Jin-Song Zhang, Keikichi Hirose, Nobuaki Minematsu, Chen-Yu Chiang, Yih-Ru Wang, Yuan-Fu Liao, “ Tone Recognition of Continuous Mandarin Speech Based on Tone Nucleus Model and MLP Nucleus Network, ”, IEICE technical report. Speech, Nagoya, Japan, Vol. 106, No. 443, pp. 107-112, SP2006-103, Dec., 2006.
2. Yuan-Fu Liao, Chi-Hui Hsu, Chi-Min Yang, Jeng-Shien Lin and Sen-Chia Chang, “ Within-Class Feature Normalization For Robust Speech Recognition “, Interspeech 2008.

參考資料

[Gales, 1996] M. J. F. Gales, “The generation and use of regression class trees for MLLR adaptation,” Cambridge Univ., Cambridge, U.K., Tech. Rep. CUED/F-INFENG/TR263, 1996; available via anonymous ftp from svr-ftp.eng.cam.ac.uk.

[Gales, 1998] M. J. F. Gales, “ Maximum likelihood linear transformations for HMM-based speech recognition, “, Computer Speech & Language, Volume 12, Issue 2, April 1998, pp. 75-98.

[Garofolo, 1993] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J.G. Fiscus, D. S. Pallett, and N. L. Dahlgren, “ DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus, “, U. S. Dept. of Commerce, NIST. Gaithersburg, MD, Feb. 1993.

[Grande, 2003] Toledano, D.T.; Gomez, L.A.H.; Grande, L.V., "Automatic phonetic segmentation," *Speech and Audio Processing, IEEE Transactions on* , vol.11, no.6, pp. 617-625, Nov. 2003.

[Kotropoulos, 2008] Georage Almpanidis, Comstrantine Kotropoulos, “ Phonemic segmentation using generalized Gamma distribution and small sample Bayesian information criterion, “, Speech Comm. 50, pp.38-50, 2008.

[Kuhn, 1998] R. Kuhn *et al.*, “Eigenfaces and eigenvoices: Dimensionality reduction for specialized pattern recognition,” in *Proc. IEEE Workshop Multimedia Signal Processing*, Dec. 1998.

[Kuo, 2006] Jen-Wei Kuo and Hsin-min Wang, "A Minimum Boundary Error Framework for Automatic Phonetic Segmentation," *International Symposium on Chinese Spoken Language Processing (ISCSLP2006)*, Lecture Notes in Artificial Intelligence, 4274, December 2006.

[Liao, 2007] Yuan-Fu Liao, Jyh-Her Yang, Chi-Hui Hsu, Cheng-Chang Lee and Jing-Teng Zeng, “A Reference Model Weighting-based Method for Robust Speech Recognition”, InterSpeech’2007.

[Lee, 2004] C.-H. Lee, “From knowledge-ignorant to knowledge-rich modeling: A new speech research paradigm for next generation automatic speech recognition,” Proc. ICSLP2004, Keynote speech, 2004.

[Lee, 2005] Y. Tsao, J. Li, and C. –H. Lee, “ A Study on Separation between Acoustic Models and its Applications, “, *Proc. InterSpeech2005*, pp. 1109-1112, Sep. 2005.

[Makhoul, 1996] T. Anastasakos, J. McDonough, R. Schwartz, J. Makhoul, “ A

compact model for speaker-adaptive training, “, ICSLP-96, pp. 1137-1140, 1996.

[Mengusoglu, 2001] Erhan Mengusoglu, Christophe Ris,” Use of Acoustic Prior Information for Confidence Measure in ASR application ”,TCT Lab , Mons , Belgium , Eurospeech 2001-Scandinavia.

[TCC, 2008] Speech Database in The Association for Computational Linguistics and Chinese Language Processing, http://www.aclclp.org.tw/corp_c.php.

[Zhang, 2004] J.-S. Zhang and K. Hirose, “Tone Nucleus Modeling for Chinese Lexical Tone Recognition”, Speech Communication, vol. 42, no. 4, pp. 447-466, 2004.

[Zhang, 2005] J.-S. Zhang, S. Nakamura, and K. Hirose, “Tone Nucleus-based Multi-level Robust Acoustic Tonal Modeling of Sentential F0 Variations for Chinese Continuous Speech Tone Recognition”, Speech Communication, vol. 46, no. 4, pp. 440-454, 2005.

[謝國平, 1998] 謝國平，《語言學概論》（增訂新版）。臺北：三民書局。