國 立 交 通 大 學

電信工程研究所

碩士論文

基於隱藏式馬可夫模型之英文語音合成系統實作
An Implementation of HMM-based English Speech Synthesis

研 究 生：劉冠驛
指導教授：陳信宏 博士

中 華 民 國 一〇〇 年 八 月

基於隱藏式馬可夫模型之英文語音合成系統實作

An Implementation of HMM-based English Speech Synthesis

研 究 生 ：劉冠驛　　Student：Kuan-Yi Liu

指導教授 ：陳信宏 博士　　Advisor：Dr. Sin-Horng Chen

國 立 交 通 大 學

電 信 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Communication Engineering

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in Communication Engineering

August 2011

Hsinchu, Taiwan, Republic of China

中 華 民 國 一 〇 〇 年 八 月

基於隱藏式馬可夫模型之英文語音合成系統實作

研 究 生：劉冠驛　　指導教授：陳信宏 博士

國立交通大學電信工程研究所碩士班

中文摘要

本論文使用一個以中文為母語的女性語者，以托福考試文章為內容的語料庫，實作一個線上英文語音合成系統。先透過一個不錯的三連音模型為語料庫做切割，再使用 cmu 字典與 Stanford-Postagger 在標記中加上音素與音節、詞、片語、句子五層結構的相關位置的韻律資訊，加以建立口腔、基頻與狀態持續時間模型，以期增加合成語音的韻律、節奏的自然度。

由實驗結果顯示，產生的韻律仍不夠自然，雖和國外其它網站合成的語音比較起來，整體韻律起伏較為明顯一點，但聲音則明顯模糊不清與細部奇怪的音調起伏，推測是因為目前只使用規則法去估計各韻律標記，所預估的韻律資訊仍不夠準確，以致合成的音檔大體的韻律正確，但較細部的音調有忽高忽低的問題。

An Implementation of HMM-based English Speech Synthesis

Student：Kuan-Yi Liu Advisor：Dr. Sin-Horng Chen

Institute of Communication Engineering
National Chiao Tung University

Abstract

The thesis establishes an online English text to speech system. Using the data base based on a woman whose mother language is China read TOEFL article. First through a good tri-phone model to segment data base, then using CMU dictionary and Stanford-Postagger software labeled phone, syllable, word, phrase and sentence five level structure relative position and prosodic information, to establish vocal cave, fundamental frequency, and duration model, expected to product more prosody and rhythm.

According to experiment result, the synthesized prosody still not natural enough. Although compare with speech synthesized from foreign web site, our prosody is more ripple but more blurred and weird rise and fall. Suppose to use rule based method to estimate variety prosodic labels still not accurate enough. So synthesized speech prosody right in general, but having strange ripple in detail.

誌謝

這篇感謝文章，是我口試前一晚打的。回想二年前的這個時後，我還在為在讀了四年畢不了業，且在電信所多媒體組找不到新老師肯收我，而痛苦不已。當時，我信心幾乎完全崩潰，要放棄這個碩士學位。就在我寄出最後一封信給陳信宏老師時（我的第一封信也是寄給陳老師），陳老師隔了一星期後回我信（一星期，我都已經放棄再找新老師了，不要這個學位了）信上，陳老師說，肯在我當完兵之後，當我的指導教授。我好開心，終於有老師肯收我了。我找新老師找了約三個多星期！期間一次又一次的不斷的被約見，也不斷的被拒絕。心裡實在一次比一次難過，信心打擊也很大。

去年 9/27 我一退伍，10/1 人就歸心似箭的搬到新竹，效率之高，從來沒有。可惜實驗室沒有我的位子，也沒有我的電腦。辛好！有一個我從來沒見過而且很混的碩二學生，聽說他是日夜不停的玩星海 2，於是在我每天的蠶食鯨吞下，漸漸的佔領了他的位子與電腦。由於我什麼也不會，於是上學期就跟著碩一學弟們一起做無聊、煩瑣的語音切割+建立 mono phone model。做了一學期，學到的東西實在有限，想說，再這樣下去不是辦法，得趕快開始做屬於我自己的研究，才能在明年暑假和碩二生一起畢業，於是從寒假開始，從開始學灌 HTS，到自己寫程式做 lab 檔，直到 6/4 我才成功合出第一版的聲音，前期幫助我最多的，當屬合哥和銘傑。不過因為合哥與銘傑都不是做合成的，所以也只能幫我一些 c 程式上的問題，對於合成的原理，合哥也不是很了。後期，自從我 5/9 修練完成 Python 神功第一層心法，接下來的日子，我的程式能力，有如增加一甲子功力。我用 c 寫一個月的程式，我用 Python 重寫，只花了一星期不到，而且我還可以在 Linux 系統中，任意呼叫各類程式，並把輸出在各程式間傳來傳去，自從有了 Python，我在寫程式上，不再需要求助任何人（話說回來，實驗室也沒人會 Python）。幫助我第二多的是和我一樣做合成的文良與性獸，文良因為位子和我在同一間 lab，找他方便，又有耐心。性獸是浸垠在合成領域多時，總能一下子就發現我的問題，有時才會立刻給我解答，大部份不會，不過可能是因為我做英文的關係，性獸對於我的研究，感覺比較沒什麼興趣（和文良的中文合成比起來）。

最後要感謝的還是陳老師，給了我一個其它老師不肯給我的機會，讓我能在明天口試。陳老師幾乎每天都會來實驗室，問學生進度。每天的壓力大是大，不過見我在一個地方卡太久，還會不斷的叫性獸幫我，或叫我轉移目標，繼續往下做。這是我前一位老師不願做的。我如能在明天順利畢業，最感謝的，還是陳信宏老師。

目錄

表目錄

圖目錄

第一章 緒論

1.1 研究動機

隨著科技的蓬勃發展，電腦運算能力的不斷提升，使得電腦有能力處理以溝通和訊息交換為主的研究，在這過程中，早期的研究主要是致力於如何提供最有用，最有價值的資訊，然而，資訊最終的目的是要提供給使用者，所以人與電腦間的溝通就顯得格外的重要。

語音是人類自然的溝通方式，它也可以成為一種人機溝通方式，一個 TTS (text-to-speech)系統就是其中一種。在近幾年 TTS 系統已經發展成一種人機介面的輸出裝置，而且被使用在許多應用領域，例如：汽車導行系統、語音郵件、語言轉換系統等等。

然而傳統上使用不同語音單元，以連續串接的方式，合成各式各樣的語音特性，例如：不同語者、不同情緒，需要大量的語音資料庫，可是大量的語料不容易去收集、切割並儲存它們。從這角度看來，為了建立一個可以產生各種語音特性的語音合成系統。一種基於 HMM(Hidden Markov Model)的 TTS 系統被提出。本論文主要著重在設計一套以語料庫為基礎的英文文句翻語音系統，期能使聲音音質的自然流暢度更為提升。

1.2 文獻回顧

傳統上，研究語言文字轉語音系統多半以實現整個系統為主，多以 Corpus-based 為基礎的語音合成系統。Corpus-based 共有兩種主要技術， Sample-based 合成法與統計(statistical)合成法。Sample-based 合成法像是單元選取(unit selection)合成法[1][2]，一種直接從語料庫中選取聲音單元，再串接起來成為語音波形。unit selection 合成法的一個最主要好處是，藉著串接自然的聲音單元，可以得到保留原本語者特性的高品質聲音。

然而因為擁有目標單元特性(attribute)的聲音單元，不總是在語料庫中，因此，類似目標單元特性的其它單元將被代替使用，當串接這樣不同的單元在一起時，常會造成聽覺上不連續的現象。為減少此現象以達到高品質的聲音，一個廣泛包含各種特性的語料庫是必須的。因為有許許多多會影響韻律特性的上下文(contextual)因子，使語料庫變得非常巨大，要建立如此巨大的語料庫將會非常吃力，而且此種方法本質上很難具有彈性去合成出不同聲音特性的各種語音。

另一方面，統計合成法使用抽取語料庫中各音檔的各種統計參數來合成語音，根據這些統計參數合成的語音，具有較連續且較一致的品質。像是[3][4][5]在所有統計方法中，我們將把焦點放在 HMM-based 語音合成法[6][7]。HMM-based 語音合成法具有下列優點，(1)HMM 已被廣為人知，適合模擬語音參數的時間序列。(2)可以將許多原本運用在語音辨識上的計術，運用到語音合成上。(3)因為 HMM 在數學上較易處理，可藉著修改聲音統計參數，達到改變合成後的語音特性。

1.3 研究方向

這篇論文中，我們應用 HTS(HMM-based speech synthesis system)[8]的方法及另一個語音合成語音軟體 Festival[9]的資料 ，去合成英文語音。類似其它 data-driven 語音合成方法，HTS 有一個精簡的語言相依模組：一串文本因子(contextual factors)，透過自己用 Python 程式語言寫的程式抽取特徵(feature)。再使用 HTS 運算核心引擎合成語音，最後再實作一個可以線上 demo 的伺服器端程式與用戶端圖形使用者介面。

1.4 語料庫簡介

本論文所使用的語料是由一位以中文為母語的女性所錄製而成，以托福考試的英文

文章為內容，總長度約為 39 分，總 word 數為 12,535，總音節數 21,321，音檔取樣頻率 16kHz，16 位元數的 PCM 格式，以下是資料庫的統計資料

表 1.1：訓練語料中各母音出現次數

AA	852	IH	3584
IY	2167	OW	548
AE	1437	EY	1003
EH	1339	UH	110
AH	6237	AW	209
UW	616	AY	784
OY	131	ER	1491
AO	825		

表 1.2：訓練語料中各子音出現次數

DH	1521	M	1621	ZH	49
HH	423	L	2440	G	379
CH	401	N	3959	F	972
JH	325	P	1324	K	1985
D	2167	S	2711	V	1324
NG	501	R	2674	Y	356
TH	317	T	3588	Z	2015
B	858	W	719	SH	522

表 1.3：每個音節中，各種母、子音組合的出現次數

CCV	995	VCCC	11	CV	7452	CCCV	25
VCC	576	CCVCC	242	VC	2810	CVC	5415
CCCVCC	12	CVCC	1405	V	1321	CCVC	728
CCCVC	83	CVCCC	240	CCVCCC	18		

表 1.4：word 中音節個數的統計

1	7306	4	661	7	2
2	2847	5	196		
3	1473	6	44		

表 1.5：每種詞性 word 個數統計

PRP$	96	VBZ	336	NNP	305
VBG	232	DT	1393	VB	368
VBD	280	RP	32	WRB	73
VBN	537	NN	2133	CC	504
VBP	247	TO	314	LS	1
WDT	97	PRP	188	PDT	8
JJ	1207	RB	572	RBS	13
WP	15	NNS	1292	RBR	24
CD	208	WP$	4	JJS	20
EX	13	MD	153	JJR	47
IN	1806	NNPS	11		

詞性使用 The Penn Treebank POS tag set，如表 1.6

表 1. 6：The Penn Treebank POS tag set[14]

Table 2
The Penn Treebank POS tagset.

1. CC	Coordinating conjunction	25. TO	*to*
2. CD	Cardinal number	26. UH	Interjection
3. DT	Determiner	27. VB	Verb, base form
4. EX	Existential *there*	28. VBD	Verb, past tense
5. FW	Foreign word	29. VBG	Verb, gerund/present participle
6. IN	Preposition/subordinating conjunction	30. VBN	Verb, past participle
7. JJ	Adjective	31. VBP	Verb, non-3rd ps. sing. present
8. JJR	Adjective, comparative	32. VBZ	Verb, 3rd ps. sing. present
9. JJS	Adjective, superlative	33. WDT	*wh*-determiner
10. LS	List item marker	34. WP	*wh*-pronoun
11. MD	Modal	35. WP$	Possessive *wh*-pronoun
12. NN	Noun, singular or mass	36. WRB	*wh*-adverb
13. NNS	Noun, plural	37. #	Pound sign
14. NNP	Proper noun, singular	38. $	Dollar sign
15. NNPS	Proper noun, plural	39. .	Sentence-final punctuation
16. PDT	Predeterminer	40. ,	Comma
17. POS	Possessive ending	41. :	Colon, semi-colon
18. PRP	Personal pronoun	42. (	Left bracket character
19. PP$	Possessive pronoun	43.)	Right bracket character
20. RB	Adverb	44. "	Straight double quote
21. RBR	Adverb, comparative	45. `	Left open single quote
22. RBS	Adverb, superlative	46. ``	Left open double quote
23. RP	Particle	47. '	Right close single quote
24. SYM	Symbol (mathematical or scientific)	48. ''	Right close double quote

1.5 章節概要說明

本論文一共分為五章，其各章節內容分配如下：

第一章：緒論。

第二章：HMM-based 語音合成系統 HTS。

第三章：英文語音合成系統實作

第四章：實驗結果及分析

第五章：結論與未來展望。

第二章　HMM-based 語音合成系統

本章描述本論文的語音合成系統架構並簡介所使用的 HMM-based 語音合成系統(HTS)。各節內容如下：2.1 節：介紹整個 HTS 系統架構; 2.2 節：基本的英文語音特性; 2.3 節：來源濾波器模型; 2.4 節：廣義梅爾倒頻譜係數; 2.5 節：基頻參數模型; 2.6 節：狀態持續時間機率模型; 2.7 節：文本相關模型; 2.8 節：全域變異數。

2.1 HTS 基本架構簡介

隱藏式馬可夫模型早期大量應用在語音辨識系統中，它成功地以機率模型描述發音的現象。近年來則被應用到語音合成上，可說是目前語音合成系統中，合成品質相當好的系統。此系統是以統計參數式的方式來合成語音，更佳彈性且不需耗費大量時間錄製語料與大量空間儲存語料。並透過參數的轉換與調適[10][11]，可輕易產生出不同語者特性的語音。

本研究使用的 HTS 為日本名古屋大學資工研究所開發出來的 HTS 2.1(HMM-based Speech Synthesis System, version 2.1)[8]，些系統為基於 HTK(Hidden Markov Model Toolkit 3.4)技術，所發展出針對使用隱藏式馬可夫模型建構的語音合成系統。基於隱藏式馬可夫模型的語音合成系統如圖 2.1 所示：

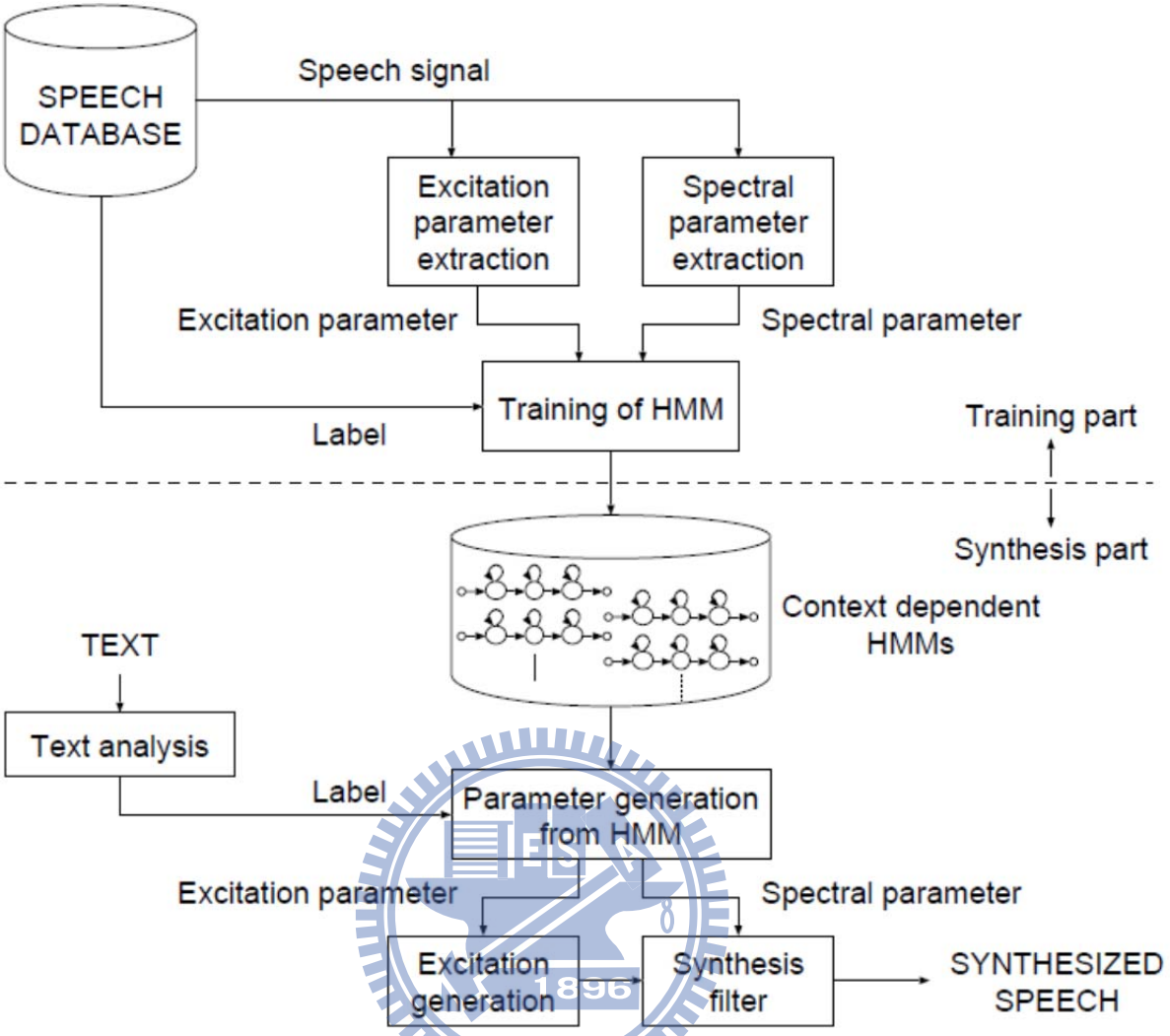


圖 2. 1 HMM-based 語音合成系統架構圖[12]

如圖 2.1 所示，HTS 分為訓練部分與合成部分，在訓練部分，由語料中抽取其頻譜成分，即廣義梅爾倒頻譜參數(Mel-Generalized Cepstral coefficients, MGC)及其動態特徵向量，與激發訊號部分 log F0 及其動態特徵向量。狀態持續時間機率密度函數去模擬語音在時間上的構造，搭配相對應的文字分析器產生文字標記，再配合適當的文脈相關問題集，訓練狀態合併分裂樹，產生與文脈相對應的 HMM 模型。合成部分則是輸入文字，透過文字分析器產生與前後文相關的文字標記檔，透過分類與回歸樹(CART)演算法，挑選對應的 HMM 模型，經由生成參數演算法，產生頻譜參數與激發訊號參數，再透過梅爾對數頻譜近似濾波器 (Mel Log Spectrum Approximation filter, MLFA filter)[13]產生語音信號。

2.2 英文語音特性

英文的音素可大致分為母音(15 個)和子音(24 個)。首先介紹母音，母音可根據舌頭在口腔上不同的位置做分類例：1. 前後；2. 上下。如圖 2.2。

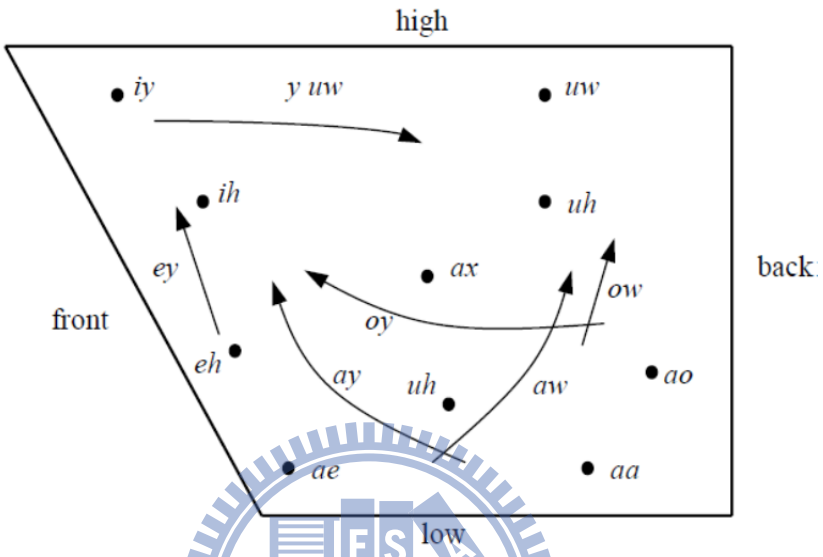


圖 2.2 發英文母音時舌頭的相對位置[14]

3. 嘴脣形狀是否是圓形，4. 是否為 tense：一種相對量，與其它發音相近的母音比較起來。例如/i/被分為 tense 母音，而/ɪ/被分為 lux 母音，/u/被分為 tense 母音，而/ʊ/被分為 lux 母音。依照上述母音發音特性，我們可將母音分類為如表 2.1。

表 2.1：英文母音在音系學上的特徵分類[14]

Vowel	high	low	front	back	round	tense
iy	+	-	+	-	-	+
ih	+	-	+	-	-	-
ae	-	+	+	-	-	+
aa	-	+	-	-	-	+
ah	-	-	-	-	-	+
ao	-	+	-	+	+	+
ax	-	-	-	-	-	-
eh	-	-	+	-	-	-
ow	-	-	-	+	+	+
uh	+	-	-	+	-	-
uw	+	-	-	+	-	+

子音相對於母音，氣流在咽喉或口腔內明顯受到壓縮或阻礙，子音可被分為有聲及無聲子音，例如無聲子音/s/、有聲子音/z/。另外因其發音方式的不同可分為如表 2.2

表 2.2：子音的發音方式分類

方式	音素例子	Word例子	發音機制
爆破音(Plosive)	/p/	tat, tap	口腔閉合
鼻音(Nasal)	/m/	team, meet	鼻腔閉合
摩擦音(Fricative)	/s/	sick, kiss	雜訊般的混亂氣流
捲舌流音(Retroflex liquid)	/r/	rat, tar	Vowel-like 舌頭高且向後捲
旁流音(Lateral liquid)	/l/	lean, kneel	Vowel-like 舌頭中間且氣流從旁過
滑音(Glide)	/y/,/w/	yes, well	Vowel-like

在子音發音時，如果聲道沒有完全被阻塞，這類子音被歸為半母音(semivowel)子音，包含流音與滑音。母音、流音與滑音都是響音(sonorant)，代表聲帶有在振動的發音，其中流音/l/及/r/是非常像母音的，有時會在某些位置上扮演為音節主音(syllabic)的角色。例如當/l/出現在末瑞時如單字 edible。

在發滑音/y/及/w/時口腔的形狀，就是在發母音/i/及/ u/時的初始位置，不過在同一音節裡，它們被要求發音時間更短，使得它們缺乏被加重音的能力，因此不夠使它們成為真正的母音，所以在子音裡它們也被歸為特殊的一群。

有時後子音是由其它複合的子音發聲方式組合而成。這種子音被定義為破擦音(affricate)，例如一個塞音(e.g. /t/)後面接著一個摩擦音(e.g. /sh/)並迅速的完成發音，使之成為一種新的單元(e.g., {t+sh} = ch)。破擦音在英文中總是有聲音，後面接無聲音的組合：像是/j/(d+ʒ)和/*ch*/ (*t* + *sh*)。表 2.3 是完整的英文子音發聲方式分類表

表 2. 3：英文子音發聲方式分類表

Consonant Labels	Consonant Examples	Voiced?	Manner
b	big, able, tab	+	plosive
p	put, open, tap	-	plosive
dh	dig, idea, wad	+	plosive
t	talk, sat	-	plosive
g	gut, angle, tag	+	plosive
k	cut, oaken, take	-	plosive
v	vat, over, have	+	fricative
f	fork, after, if	-	fricative
z	zap, lazy, haze	+	fricative
s	sit, cast, toss	-	fricative
dh	then, father, scythe	+	fricative
th	thin, nothing, truth	-	fricative
zh	genre, azure, beige	+	fricative
sh	she, cushion, wash	-	fricative
jh	joy, agile, edge	+	affricate
ch	chin, archer, march	-	affricate
l	lid, elbow, sail	+	lateral
r	red, part, far	+	retroflex
y	yacht, onion, yard	+	glide
w	with, away	+	glide
hh	help, ahead, hotel	+	fricative
m	mat, amid, aim	+	nasal
n	no, end, pan	+	nasal
ng	sing, anger, drink	+	nasal

在英文裡，詞(Word)是最小有意義單位，每個詞有自己所屬的詞性(Part-of-Speech, POS)，每個詞由一至數個音節(syllable)所組成，而音節是由一個母音與數個子音合成的次單位，以上資訊都可藉由查詢字典所獲得。

2.3 來源濾波器模型

提出的TTS系統是基於來源濾波器模型(source-filter model)。為了建構這樣的系統，首先必須從資料庫中抽取特徵參數來訓練，抽取特徵參數的目的就是為了模擬說話時的口腔特性，將聲音分解為激發訊號與口腔模型兩部分，後面將針對這兩種特徵參數進行詳細討論。首先介紹的是來源濾波器模型，

為了將語音波形以數學化的形態表示，來源濾波器模型是一種時常被使用的表示法。如圖 2.3，轉移函數 H(z)模擬口腔結構。激發訊號的選擇，由語音是有聲音或無聲音決定。當激發訊號是有聲音時被模擬成一個周期性脈衝串，無聲音時被模擬成一個白雜訊。激發訊號 e(n)通過一個隨時間改變參數的線性系統 H(z)產生語音訊號 x(n)。

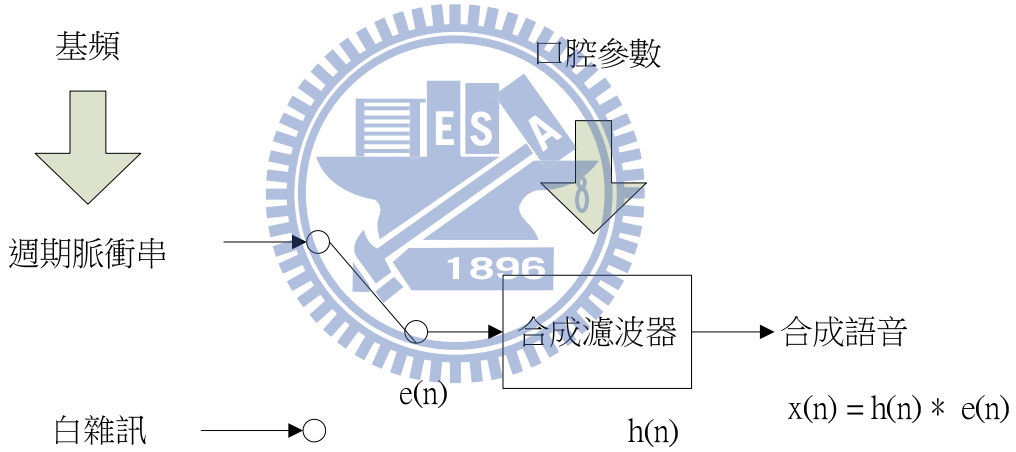


圖 2. 3 來源濾波器模型

藉由激發訊號 e(n)和口腔模型的頻率響應 h(n)做 convolution 可產生語音訊號 x(n)

$$x(n) = h(n) * e(n) \tag{2.1}$$

符號*代表離散的 convolution。

2.4 廣義梅爾倒頻譜係數

為求得口腔模型的頻譜參數的方法通常有兩種，分別是線性預估法(linear prediction)與倒頻譜分析法(cepstrum analysis)。(1)線性預估法是一種早已被廣泛使用的方法，可用來得到口腔的全極點模型，線性預估分析常用來找出語音頻譜的包絡線，描述語音頻譜。(2)倒頻譜分析是利用人耳在不同頻帶對音強的敏感度不同所設計，另外此種分析方法的好處是可以有效分離發音腔調模型與激發訊號，使得口腔模型的參數預估更為準確。在語音辨識系統已被證實其效果較線性預估法好。但使用較低的倒頻譜係數維度時，會有共振峰頻寬過度估計的問題[15]

本論文使用廣義梅爾倒頻譜係數[16]來模擬口腔模型。廣義梅爾倒頻譜係數的求取步驟為 1. 對語音訊號做傅利葉轉換。2. 將轉換後的係數取廣義對數函數。3.在頻率校正(warped frequency scale)下作反傅利葉轉換。可寫成(2-1)式：

$$S_\gamma(X(e^{j\omega})) = \sum_{m=-\infty}^{\infty} c_{\alpha,\gamma}(m) e^{-j\beta_\alpha(\omega)m} \tag{2.2}$$

其中 $X(e^{j\omega})$ 為語音訊號 x(n)的傅利葉轉換，S_γ 為廣義對數函數，定義如下：

$$S_\gamma = \begin{cases} (\omega^\gamma - 1)/\gamma & 0 < |\gamma| \le 1 \\ \log\omega & \gamma = 0 \end{cases} \tag{2.3}$$

參數 γ 用來調整將頻譜平滑化的程度。頻率校正 $\beta_\alpha(\omega)$ 定義為一個全通系統 $\Psi_\alpha(z)$ 的相位響應(phase response)定義如下：

$$\Psi_\alpha(z) = \left.\frac{z^{-1}-\alpha}{1-\alpha z^{-1}}\right|_{z=e^{jw}} = e^{-j\beta_\alpha(\omega)}, |\alpha| < 1 \tag{2.4}$$

$$\beta_\alpha(\omega) = \tan^{-1}\frac{(1-\alpha^2)\sin\omega}{(1+\alpha^2)\cos\omega - 2\omega} \tag{2.5}$$

因此可將語音頻譜 H(Z)以 M+1 階梅爾倒頻譜係數表示為：

$$H(z)=S_{\gamma}^{-1}\left(\sum_{m=0}^{\infty}c_{\alpha,\gamma}(m)\Psi_{\alpha}^{m}(z)\right)$$

$$=\begin{cases}\left(1+\lambda\sum_{m=0}^{\infty}c_{\alpha,\gamma}(m)\Psi_{\alpha}^{m}(z)\right)^{1/\gamma} & ,0<|\gamma|\leq 1\\ \exp\sum_{m=0}^{\infty}c_{\alpha,\gamma}(m)\Psi_{\alpha}^{m}(z) & ,\gamma=0\end{cases} \tag{2.6}$$

由(2-3)、(2-5)可以看出當(α,γ)=(0,-1)時廣義梅爾倒頻譜分析等於線性分析，而(α,γ)=(0,0)時等於倒頻譜分析。圖 2.4 及表 2.4[15]表示參數(α,γ)對應的各種分析法。

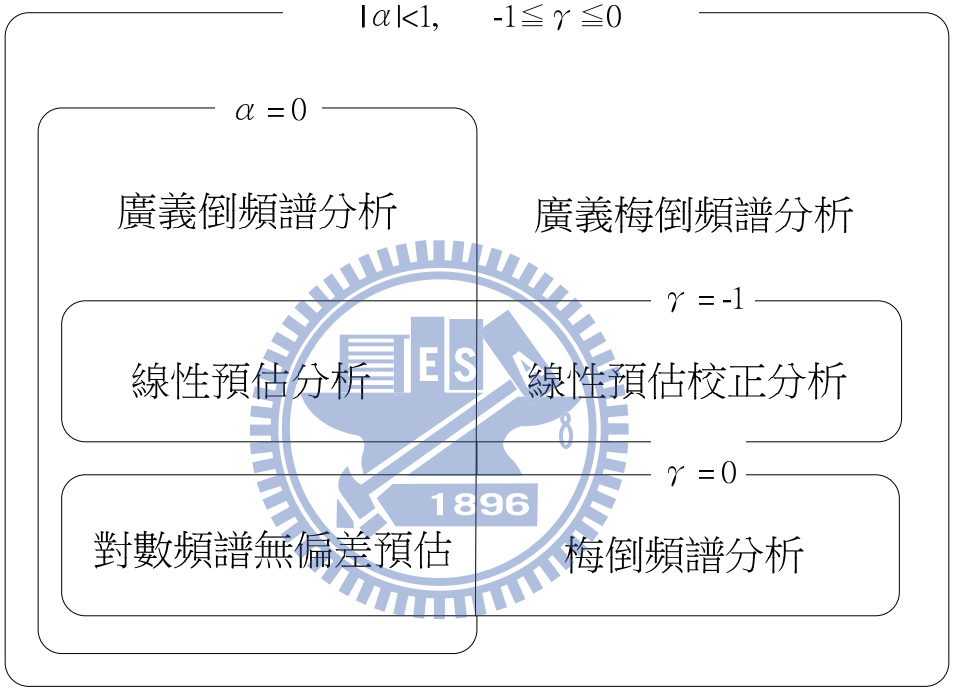


圖 2. 4 廣義梅爾倒頻譜分析和其它分析法關係圖

表 2.4：各頻譜模型表示與參數之間關係

	$\alpha=0$	$\lvert\alpha\rvert<1\neq 0$
$\gamma=-1$	全極點模型	全極點校正模型 (warped all-pole)
$\gamma=0$	倒頻譜	梅爾倒頻譜
$\gamma=1$	全零點模型	全零點校正模型
$-1\leqq\gamma\leqq 1$	廣義倒頻譜	廣義梅爾倒頻譜

廣義梅爾倒頻譜藉由調整參數（α,γ），可以藉由找出最適當的參數(α,γ)，在相同階數清況下，更有彈性的表示頻譜特性，獲得更精準的頻譜分析，因此在語音辨識、語音編碼和語音合成上很廣泛的應用。

2.5 基頻參數模型

基頻(F0)在發有聲(voiced)音時是一個連續的變數，而在無聲音時為 0，因此並不能使用單純的離散或連續 HMM 建立 F0 模型。此論文 F0 輸出狀態的機率是用多維空間機率分佈 MSD(multi-space probability distribution)[17]，做法是將 F0 由一維的連續變數表示有聲區間，0 維的單一值 0 表示無聲音。

2.5.1 多空間機率分佈

首先將先介紹多維度空間機率分佈(Multi-Space Probability Distribution)。考慮一個樣本空間Ω如圖 2.4

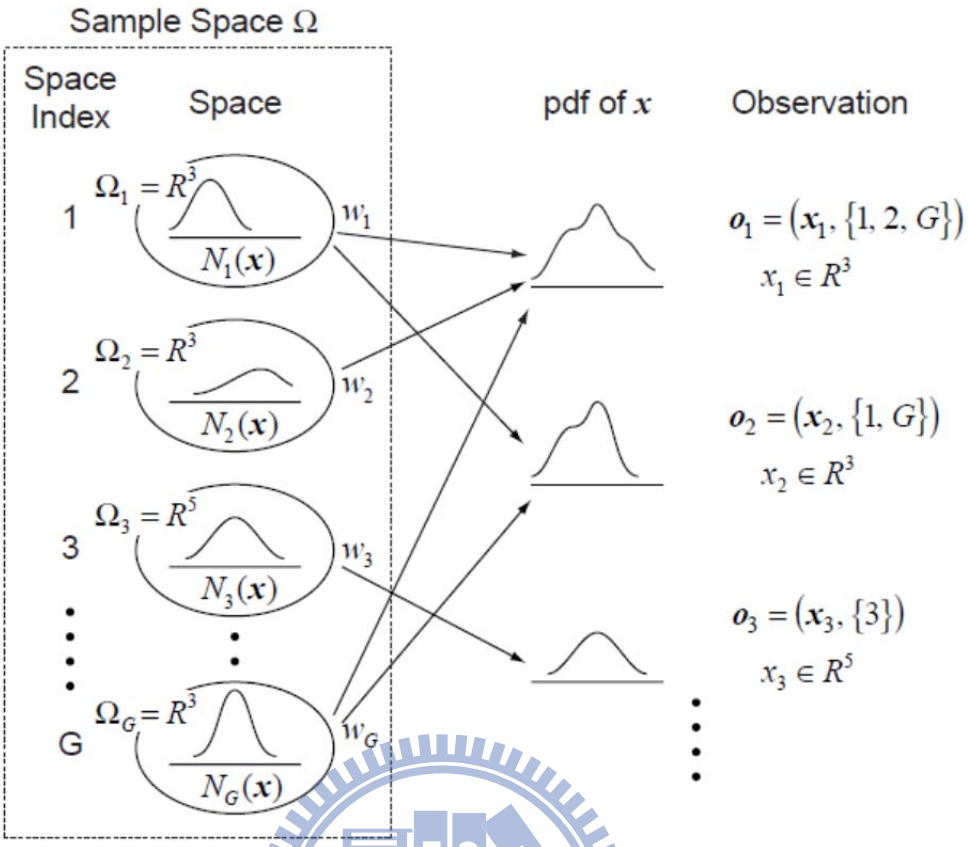


圖 2.5 多維空間機率分佈與觀測資料[12]

這裡Ω_g是一個n_g維的實數空間R^{n_g}，空間索引是g。每個空間Ω_g有自己的出現機率ω_g, i.e. $P(\Omega_g) = \omega_g$。如果$n_g > 0$每個空間有一個機率密度函數$N_g(\mathbf{x})$, $\mathbf{x} \in R^{n_g}$，這裡$\int_{R^{n_g}} N_g(\mathbf{x})d\mathbf{x} = 1$。我們假設當$n_g = 0$時，$\Omega_g$只包含一個樣本點。根據上述，$P(E)$是機率分佈我們有

$$P(\Omega) = \sum_{g=1}^{G} P(\Omega_g) = \sum_{g=1}^{G} \omega_g \int_{R^{n_g}} N_g(\mathbf{x})d\mathbf{x} = 1 \tag{2.7}$$

值得注意的是，雖然當$n_g = 0$，因為Ω_g只包含一個樣本點，$N_g(\mathbf{x})$不存在。為簡化表示，我們定義當$n_g = 0$時$N_g(\mathbf{x}) \equiv 1$。

本論文中所考慮的每個事件 E，被表示成一個隨機變數$\boldsymbol{O}$，由一組 n 維連續隨機變數$\boldsymbol{x} \in R^{n_g}$和一組空間索引 X 所組成

$$o = (\mathbf{x}, X) \tag{2.8}$$

所有的空間索引 X 都是 n 維，O 的觀測機率被定義為

$$b(o) = \sum_{g \in S(o)} w_g N_g(V(o)) \tag{2.9}$$

這裡

$$V(o) = x, \qquad S(o) = X \tag{2.10}$$

在圖 2.5 中有一些觀測點的例子。一個觀測點 O_1 是由一個三維向量 $\boldsymbol{x}_1 \in R^3$ 和一組空間索引 X_1={1, 2, G}所組成。因此隨機變數 $\boldsymbol{x}$ 是從三個空間索引 Ω_1, Ω_2, $\Omega_G \in R^3$ 中的其中一個被選出。而 $\boldsymbol{x}$ 機率密度函數是 $w_1 N_1(\mathbf{x}) + w_2 N_2(\mathbf{x}) + w_G N_G(\mathbf{x})$ 。

上述定義的機率分佈，就是本論文中的多空間機率分佈。當 $n_g \equiv 0$ 和 $n_g \equiv m > 0$ 分別等於離散和連續分佈。更進一步，如果 $S(\boldsymbol{o}) \equiv$ {1, 2, 3…, G},此連續分佈代表一個 G-mixture 機率密度函數。因此多空間機率分佈是一個比離散和隨機分佈更一般的表示法。

2.5.2 HMM 的多空間分佈

在 MSD-HMM 中每個狀態的輸出機率，是由在前一小節提到的多空間機率分佈所定義的。一個 N 個狀態的 MSD-HMM λ 有一個初始狀態機率 $\pi = \{\pi_j\}_{j=1}^{N}$、轉移狀態機率分佈 $\mathrm{A} = \{a_{ij}\}_{i,j=1}^{N}$ 和輸出機率密度 $\mathrm{B} = \{b_i(\cdot)\}_{i=1}^{N}$ ，這裡

$$b_i(o) = \sum_{g \in S(o)} w_{ig} N_{ig}(V(o)), \quad i = 1, 2, ..., N. \tag{2.11}$$

就像圖 2.6，每個狀態 i 有 G 個機率密度函數 $N_{i1}(\cdot)$, $N_{i2}(\cdot)$, … $N_{iG}(\cdot)$, 和它們的權重 ω_{i1}, ω_{i2}, …, ω_{iG}。

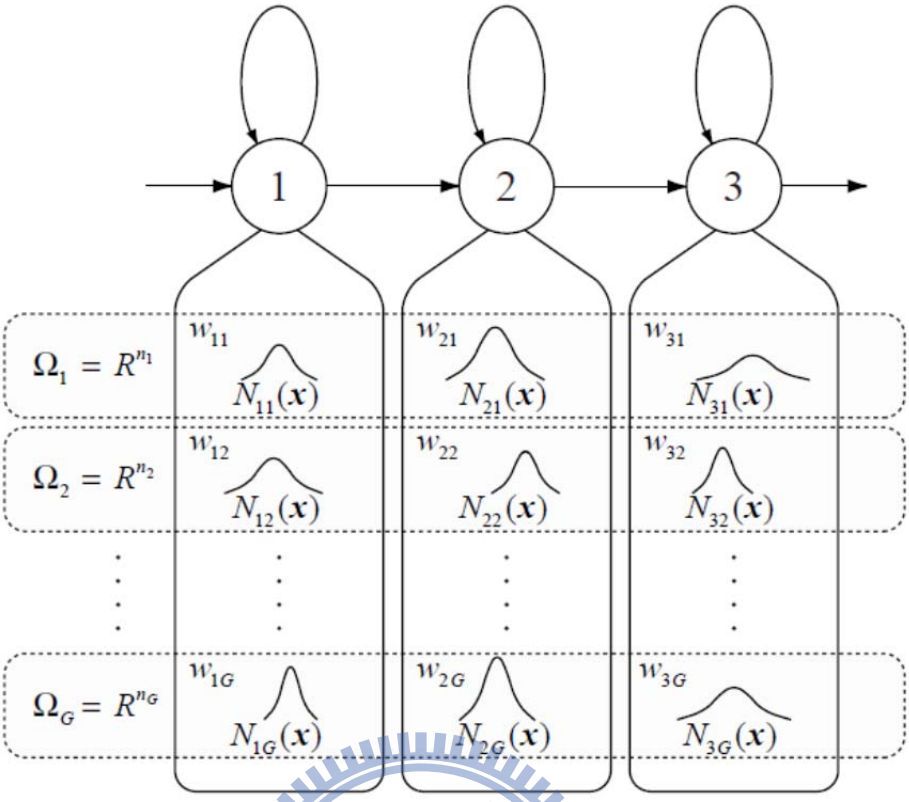


圖 2. 6 多空間機率分佈的 HMM[17]

觀測機率 $\boldsymbol{O}=\{\boldsymbol{o}_1, \boldsymbol{o}_2, \cdots \boldsymbol{o}_T\}$可被寫成

$$P(\mathbf{O}|\lambda) = \sum_{all\ \mathbf{q}} \prod_{t=1}^{T} a_{q_{t-1}q_t} b_{q_t}(\mathbf{o}_t) = \sum_{all\ \mathbf{q},\mathbf{l}} \prod_{t=1}^{T} a_{q_{t-1}q_t} \omega_{q_t l_t} N_{q_t l_t}(V(o_t)) \tag{2.12}$$

這裡 $\boldsymbol{q} = \{q_1, q_2, \cdots, q_T\}$代表可能的狀態序列，$\boldsymbol{l} = \{l_1, l_2, \cdots, l_T\} \in \{S(\boldsymbol{o}_1) \times S(\boldsymbol{o}_2) \times \cdots \times S(\boldsymbol{o}_T)\}$是可能得到觀測序列 $\boldsymbol{O}$ 的空間索引序列，$a_{q_0,j}$ 定義為 π_j。

向前、向後變數為：

$$\alpha_t(i) = P(\mathbf{o}_1, \mathbf{o}_2, \cdots \mathbf{o}_t, q_t = i \mid \lambda) \tag{2.13}$$

$$\beta_t(i) = P(\mathbf{o}_{t+1}, \mathbf{o}_{t+2}, \cdots \mathbf{o}_T \mid q_t = i, \lambda) \tag{2.14}$$

使用類似一般 HMM 的向前向後演算法，即可求得上式。根據公式(2.12)可被如下式計算

$$P(\mathbf{O}\mid\lambda)=\sum_{i=1}^{N}\alpha_T(i)=\sum_{i=1}^{N}\beta_1(i) \tag{2.15}$$

向前、向後變數將在下一小節中用來計算重新估計公式。

2.5.3 MSD-HMM 訓練階段的重新估計演算法

對於一個給定的觀測序列 $\boldsymbol{O}$ 和一個特別選定的 MSD-HMM，使用最大相似度估計演算法最大化 $P(\boldsymbol{O}|\lambda)$，下面是 MSD-HMM 的最大相似度重新估計公式。

一個輔助函式 $Q(\lambda',\lambda)$的定義如下，其中 λ' 是目前的參數，λ 是新參數。

$$Q(\lambda',\lambda)=\sum_{all\ \mathbf{q},\mathbf{l}}P(\mathbf{O},\mathbf{q},\mathbf{l}\mid\lambda')\log P(\mathbf{O},\mathbf{q},\mathbf{l}\mid\lambda) \tag{2.16}$$

我們假設 $N_{ig}(\cdot)$是一個高斯分佈，平均值向量是 μ_{ig}、共變異矩陣是 Σ_{ig}。給定一個觀測序列 $\boldsymbol{O}$ 和一個模型 λ'，導出一個新的模型參數 λ，使得 Q 函式為最大值。從(2.12) $\log P(\mathbf{O},\mathbf{q},\mathbf{l}\mid\lambda)$ 可以表示為

$$\log P(\mathbf{O},\mathbf{q},\mathbf{l}\mid\lambda)=\sum_{t=1}^{T}(\log a_{q_{t-1}q_t}+\log\omega_{q_t l_t}+\log N_{q_t l_t}(V(o_t))) \tag{2.17}$$

因此 Q 函式(2.16)可以被表示為

$$\begin{aligned}Q(\lambda',\lambda)=&\sum_{i=1}^{N}P(\mathbf{O},q_1=i\mid\lambda')\log\pi_i\\&+\sum_{i,j=1}^{N}\sum_{t=1}^{T-1}P(\mathbf{O},q_t=i,q_{t+1}=j\mid\lambda')\log a_{ij}\\&+\sum_{i=1}^{N}\sum_{g=1}^{G}\sum_{t\in T(\mathbf{O},g)}P(\mathbf{O},q_t=i,l_t=g\mid\lambda')\log w_{ig}\\&+\sum_{i=1}^{N}\sum_{g=1}^{G}\sum_{t\in T(\mathbf{O},g)}P(\mathbf{O},q_t=i,l_t=g\mid\lambda')\log N_{ig}(V(o_t))\end{aligned} \tag{2.18}$$

其中

$$T(\mathbf{O},g)=\{t\mid g\in S(\mathbf{o}_t)\} \tag{2.19}$$

參數集 $\lambda=(\pi, A, B)$ 將在 $\sum_{i=1}^{N}\pi_i=1$，$\sum_{j=1}^{N}a_{ij}=1$和 $\sum_{g=1}^{G}w_g=1$的限制下使得(2.18)式為最大化，可以導出

$$\pi_i=\sum_{g\in S(\mathbf{o}_1)}\gamma_1'(i,g) \tag{2.20}$$

$$a_{ij}=\frac{\sum_{t=1}^{T-1}\xi_t'(i,j)}{\sum_{t=1}^{T-1}\sum_{t\in S(o_t)}\gamma_t'(i,g)} \tag{2.21}$$

$$w_{ig}=\frac{\sum_{t\in T(\mathbf{O},g)}\gamma_t'(i,g)}{\sum_{h=1}^{G}\sum_{t\in T(O,h)}\gamma_t'(i,h)} \tag{2.22}$$

$$\mu_{ig}=\frac{\sum_{t\in T(\mathbf{O},g)}\gamma_t'(i,g)V(\mathbf{o}_t)}{\sum_{t\in T(O,g)}\gamma_t'(i,g)},\quad n_g>0 \tag{2.23}$$

$$\mathbf{\Sigma}_{ig}=\frac{\sum_{t\in T(\mathbf{O},g)}\gamma_t'(i,g)(V(\mathbf{o}_t)-\boldsymbol{\mu}_{ig})(V(\mathbf{o}_t)-\boldsymbol{\mu}_{ig})^T}{\sum_{t\in T(O,g)}\gamma_t'(i,g)},\quad n_g>0 \tag{2.24}$$

這裡 $\gamma_t(i,h)$ 是在時間 t 時，狀態在 i 且空間索引是 h 的機率，而 $\xi_t(i,j)$ 是在時間 t 時狀態在 i，而在下一個時間狀態在 j 的機率。都可以使用向前、向後變數 $\alpha_t(i)$、$\beta_t(i)$ 計算，如下式

$$\begin{aligned}\gamma_t(i,h)&=P(q_t=i,l_t=h\mid\mathbf{O},\lambda)\\&=\frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N}\alpha_t(i)\beta_t(i)}\cdot\frac{w_{ih}N_{ih}(V(\mathbf{o}_t))}{\sum_{g\in S(\mathbf{o}_t)}w_{ig}N_{ig}(V(\mathbf{o}_t))}\end{aligned} \tag{2.25}$$

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j \mid \mathbf{O}, \lambda)$$
$$= \frac{\alpha_t(i) a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{\sum_{h=1}^{N}\sum_{k=1}^{N} \alpha_t(h) a_{hk} b_k(\mathbf{o}_{t+1}) \beta_{t+1}(k)} \tag{2.26}$$

當 F0 的觀測值在發有聲區音時是一個連續值，而發無聲音時為 0。我們可以模擬這種觀測序列，在發有聲音時觀測到的 F0 值，出現在一維空間，被假設成一個連續 (G-1)-mixture 機率密度函數，即設定 $n_g = 1 (g = 1, 2, \cdots, G-1)$, ，而無聲音時，出現在零維空間，$n_G = 0$ 如下式

$$S(\mathbf{o}_t) = \begin{cases} \{1, 2, \cdots, G-1\} & (voiced) \\ \{G\} & (unvoiced) \end{cases} \tag{2.27}$$

2.6 狀態持續時間機率模型

對於在合成時，假設語音長度固定為 T，要找尋最佳狀態持續時間序列，目標是獲得一狀態序列 $q=\{q_1, q_2, q_3 \cdots, q_T\}$，使得(2.29)式在滿足(2.28)式時為最大：

$$T = \sum_{k=1}^{K} d_k \tag{2.28}$$

$$\log P(q \mid \lambda, T) = \sum_{k=1}^{K} \log P_k(d_k) \tag{2.29}$$

$P_k(d_k)$表示在某狀態 k 持續時間為 d_k 的機率，K 是整句隱藏式馬可夫模型中的狀態個數。因為在某一狀態 k 中，$P_k(d_k)$為單一高斯機率分布，能使得(2.29)最大化的狀態持續時間序列$\{d_k\}_{k=1}^{K}$為：

$$d_k = \xi(k) + \rho \cdot \sigma^2(k) \tag{2.30}$$

其中

$$\rho = \frac{T - \sum_{k=1}^{K} \xi(k)}{\sum_{k=1}^{K} \sigma^2(k)} \tag{2.31}$$

ξ(k)和σ^2(k)分別是在狀態 k 中，持續時間分布的平均值和變異數。由(2.31)式，可知ρ和 T 相關，所以可藉由控制ρ來控制合成語音的說話速率，由(2.30)可看出，若希望合成出來的語音說話速率為語料的平均值，則ρ需設為 0，ρ的值是正或負分別影響合成語音說話速率的快或慢，σ^2(k)代表第 k 個狀態可變化的彈性。

而每個音素的狀態持續時間模型，被模擬成一個多維的連續高斯分佈，其中第 n 維的狀態持續時間變數，代表音素中的第 n 個狀態，且假設隱藏式馬可夫模型的狀態為由左到右，不可跳躍。隱藏式馬可夫模型與狀態持續時間模型關係如圖 2.5。

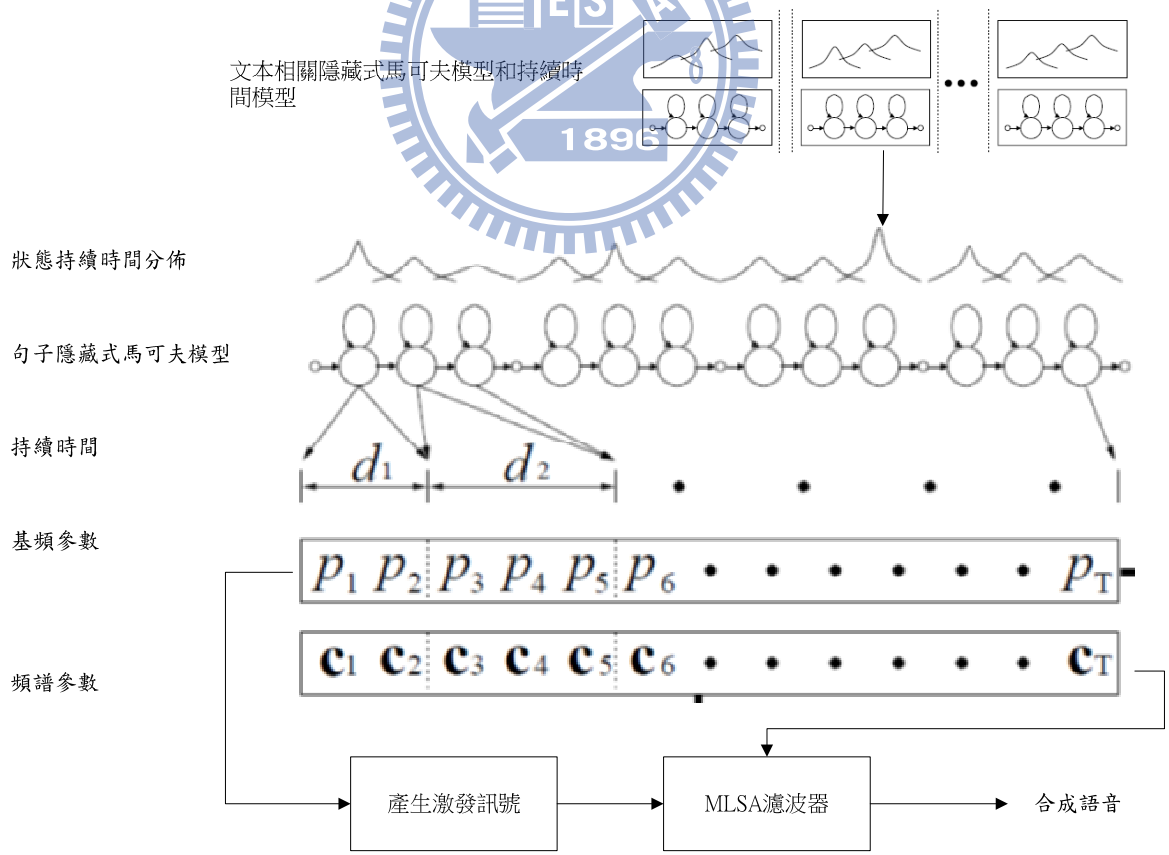


圖 2. 7 狀態持續時間機率模型在語音合成系統架構中關係圖[17]

其中每一維度的高斯分佈中的平均值ξ(i)與變異數σ^2(i)，可用下列兩式計算

$$\xi(i)=\frac{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)(t_1-t_0+1)}{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)} \tag{2.32}$$

$$\sigma^2(i)=\frac{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)(t_1-t_0+1)^2}{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)}-\xi^2(i) \tag{2.33}$$

這裡$\chi_{t_0,t_1}(i)$定義為某音素在狀態 i 中，持續時間在 t_0 至 t_1 的機率，可表示為

$$\chi_{t_0,t_1}(i)=(1-\gamma_{t_0-1}(i))\cdot\prod_{t=t_0}^{t_1}\gamma_t(i)\cdot(1-\gamma_{t_1+1}(i)) \tag{2.34}$$

其中γ_t(i)定義為在時間 t 時，在狀態 i 中的機率，另外定義邊界機率

$$\gamma_{-1}(i)=\gamma_{T+1}(i)=0 \tag{2.35}$$

2.7 文本相關模型

有許多的文本相關因素（例：音素、重音、位置因素等）會影響頻譜、基頻、持續時間。當建立一個文本相關模型，並考慮許多的前後文(context)因素，我們希望能得到一個適合的模型。然而，當前後文因素增加時，它們的組合也會呈現指數增加。因此在有限的訓練語料下，無法建立一個足夠正確的模型參數，更不可能準備一個包含所有因素組合的語料庫。

2.7.1 基於文本相關決策樹分類法

為克服上述問題，我們應用了一個基於前後文群集技術的決策樹[18]去分類頻譜、F0 和狀態持續時間。因為頻譜、F0 和狀態持續時間，有它們各自的影響因素，像是音

素本身特性、重音所在、或是前後文相關位置等變因，將上述的各項變因列入考慮而嘗試建構各特徵參數模型，目的就是希望在合成語音時，能藉由已經訓練好的模型，準確的預估特徵參數，使得合成語音聽起來自然流暢。因此，考慮到現實上訓練語料數量上的限制，當變因的組合太多時，符合其中某種組合條件的樣本數會太少，甚至沒有符合的樣本，導致模型的參數無預估的準確或根本無法預估。

決策樹是一種二元樹，每一節點都是一個決策點，給定一個條件，把資料群根據符合條件與否分為兩類，從決策樹根部開始，每個節點都有一個相關於文本內容的問題，每一個樣本根據標記文本上符合問題條件與否，持續分類直到子葉。

最後被分到同一子葉的樣本群，經由上述的篩選，在文本內容上必定十分相似，因此，同一子葉的樣本群訓練成一個模型，好處是在合成語音時遇到訓練語料中沒有出現過的文本組合，也可以藉由決策樹分類找出最相近的模型套用。由於影響各特徵參數的文本相關因子不同，所以分別會為它們各自建決策樹，如圖 2.8

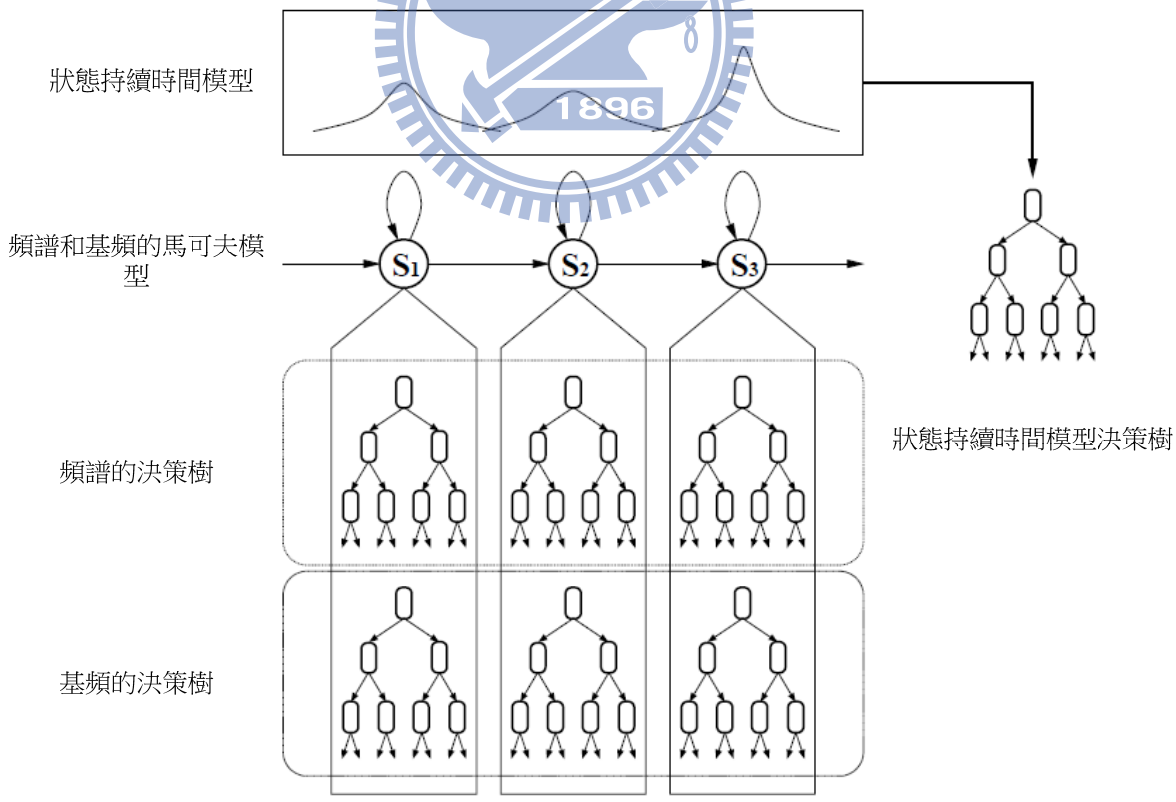


圖 2.8 決策樹[17]

傳統決策樹停止分裂的規則為，為防止葉節點內的資料太少，預先設定「最小音框佔有期」臨界點(minimum frame occupancy threshold)和為防止過度訓練而設的「最小相似度增加值」臨界點，因為過大的樹將過份專化(overspecialized)訓練語料，使得一般化的表現不好。另一方面過小的樹將使的最後建立的模型因參數過少不夠精準。而適當的臨界點需要用嘗試錯誤法去慢慢找出來。因此改用下小節介紹的自動停止規則。

2.7.2 最小描述原理

最小描述原理(MDL Minimum Description Length)原理[19]在這被使用當作決策樹停止分裂的原則，已被證明是一個有效的選擇最佳機率模型方法。根據 MDL 原理，對於資料 $x^N = x_1, \cdots, x_N$，是從 I 個模型中 $i = 1, \cdots, I$ 的其中一個建立的，要找出最佳模型，則擁有最小描述長度 l 的模型被選為最佳模型，對於模型 i 描述長度 $l(i)$ 的定義為

$$l(i) = -\log P_{\hat{\theta}(i)}(x^N) + \frac{\alpha_i}{2}\log N + \log I \qquad (2.36)$$

其中 α_i 是模型 i 中，自由參數的個數，$\hat{\theta}(i)$ 是模型 i 中使用最大相似度演算法估計的參數，第一項是相似度的負值、第二項表示模型的複雜度、第三項只是模型總個數。當一個模型變得更複雜，即可用的自由參數變更多時，相似度上升，第一項的值會減少、第二項的值會增加、第三項只是個常數。模型在一個適當的複雜度時，會出現最小描述長度如圖 2.7。而且，在(2.36)中會發現，MDL 原理不需要任何外部設定的參數，屬於某資料庫的最佳模型會自動從一組模型中被選出。

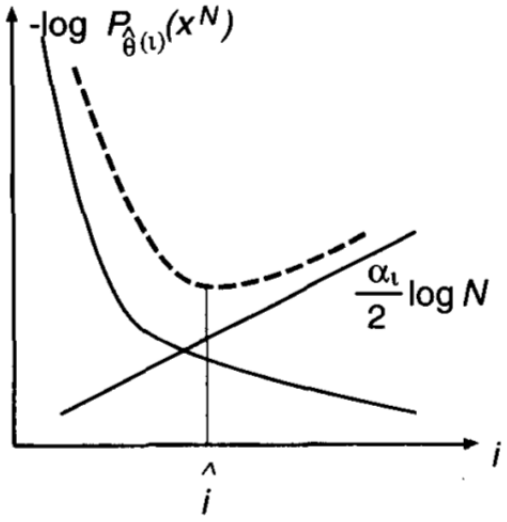


圖 2. 9 The MDL criterion[19]

MDL 原理被應用在 MSD-HMM[20]決策樹分群裡，被用來建立 f0 模型。假設一個分群(cluster)集 S 是分群後的結果，定義為 $S = \{S_1, S_2, \cdots, S_i, \cdots, S_M\}$。對數相似度 L 被計算如下式：

$$L = -\sum_{s\in S}\sum_{g=1}^{G}\frac{1}{2}(n_g(\log(2\pi)+1)+\log|\Sigma_{sg}|-2\log\omega_{sg})\sum_{t\in T(O,g)}\gamma_t(s,g) \qquad (2.37)$$

這裡 g 是空間索引，ω_{sg} 是空間索引 g 在分群集 S 中的權重。$T(O, g)$是一組時間，滿足觀察向量所使用的空間集中，有包含 g 空間的時間。$\gamma_t(s, g)$是在時間 t 時，觀察值是由分群子集 s 中的空間 g 產生的機率。當在 0 維空間時，在(2.37)式中的 $\log|\Sigma_{sg}| = 0$。使用相似度 L，描述長度 l 被表示成

$$\begin{aligned} l = &\sum_{s\in S}\sum_{g=1}^{G}\frac{1}{2}(n_g(\log(2\pi)+1)+\log|\Sigma_{sg}|-2\log\omega_{sg})\sum_{t\in T(O,g)}\gamma_t(s,g) \\ &+\left(\sum_{s\in S}\sum_{g=1}^{G}\frac{1}{2}(2n_g+1)\right)\cdot\left(\log\sum_{s\in S}\sum_{g=1}^{G}\sum_{t\in T(O,g)}\gamma_t(s,g)\right) \end{aligned} \qquad (2.38)$$

當一個節點被分裂為兩個子節點，兩個子節點的描述長度是 l'，描述長度的變化量 δl 被下式計算

$$
\begin{aligned}
\delta l &= l' - l \\
&= \sum_{s\in\{S_{i+},S_{i-}\}} \sum_{g=1}^{G} \frac{1}{2}(\log|\Sigma_{sg}| - 2\log\omega_{sg}) \cdot \sum_{t\in T(O,g)} \gamma_t(s,g) \\
&- \sum_{s\in\{S_i\}} \sum_{g=1}^{G} \frac{1}{2}(\log|\Sigma_{sg}| - 2\log\omega_{sg}) \cdot \sum_{t\in T(O,g)} \gamma_t(s,g) \\
&+ \left(\sum_{g=1}^{G} \frac{1}{2}(2n_g + 1)\right) \cdot \left(\log \sum_{s\in S} \sum_{g=1}^{G} \sum_{t\in T(O,g)} \gamma_t(s,g)\right)
\end{aligned}
\tag{2.39}
$$

如果 $\delta l < 0$ 節點被分裂，反之則否。

第三章 英文語音合成系統實作

3.1 系統環境、語言及程序工具簡介

本英文語音合成系統建構在Fedora 12作業系統下，整個語音合成系統的主要部分，是使用日本名古屋大學資工研究所開發的HTS 2.1。HTS 2.1[8]是基於HTK 3.4[22]的修改版本。HTK是由英國劍橋大學電機系開發出來的隱藏式馬可夫模型開發工具，提供了非常多的指令，方便使用者實作隱藏式馬可夫模型的建立，主要是作為語音辨認之用；HTS保留了大部分HTK的指令，只針對語音合成上的一些需要做更動。整個合成系統的架構以及使用的工具語言如圖3.1：

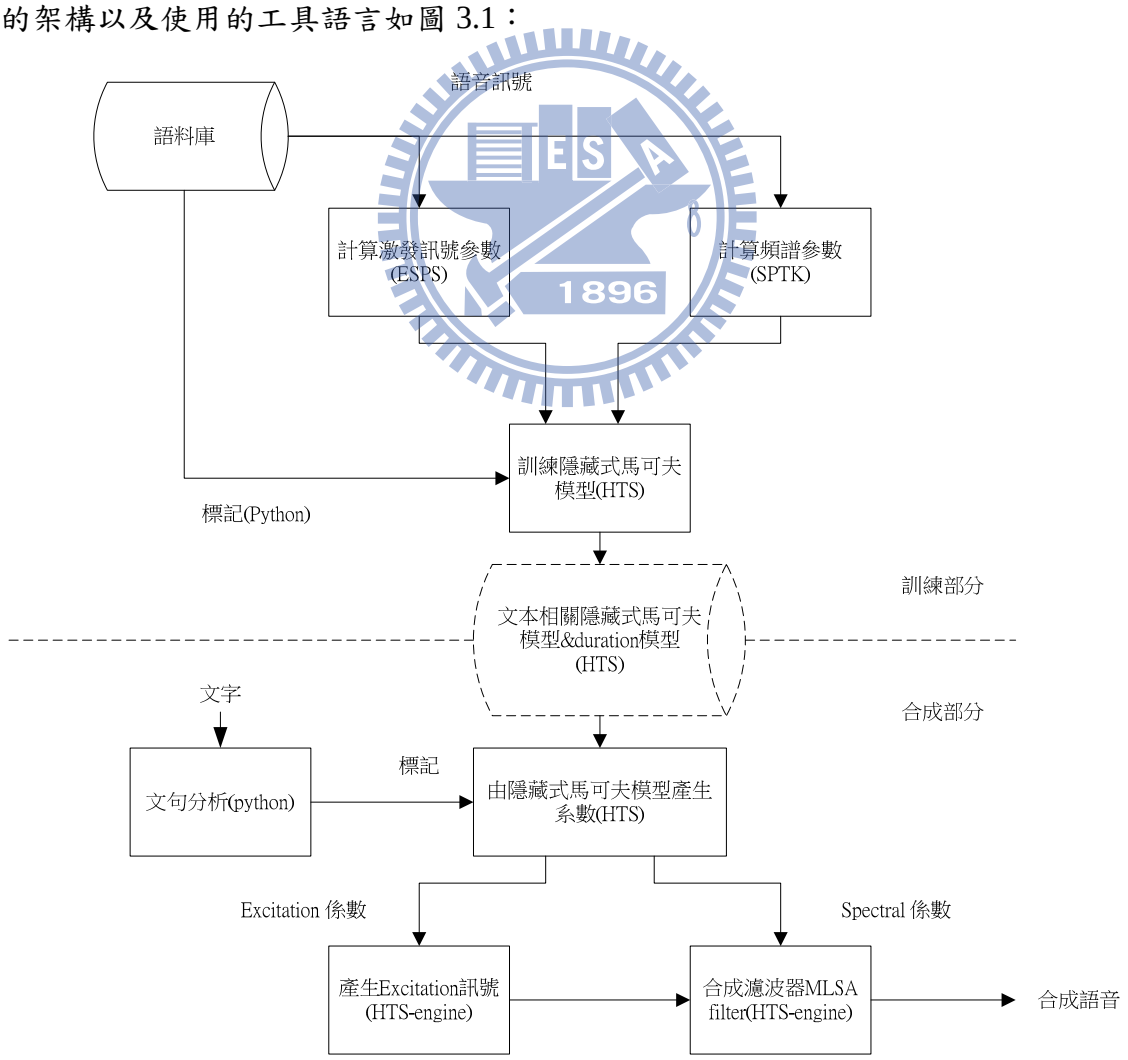


圖 3. 1 基於隱藏式馬可夫模型英文語音合成系統

目前的 HTS 版本中都不包含文句分析功能，因此自己用 Python 語言寫了一個簡單的文句分析器；另外語料檔所唸的內文，也經由 Python 語言撰寫程式，轉寫成 HTS 可接受的格式；抽取基頻參數部分，使用 SPTK(Speech signal process toolkit)[23]工具；抽取基頻參數部分，使用 Tcl 語言引入 Snack Library 來計算，而最後合成部分，使用合成軟體 HTS_engine[24]合成輸出語音。

3.2 語料的前處理

第一步：將訓練語料各音檔標記音素的切割位置

從[25]下載國外研究者已建好的三連音模型。模型的訓練語料如下

Train data : WSJ all + TIMIT

Tied states (approx) : 10000

Gaussians : 32

Silence gaussians : 64

使用下載後的 acoustic model 對自己的語料庫做 forced-alignment 找初始切割位置，此步驟完成後，即可得到 phone level 的 feature。

第二步：將訓練語料標記音節邊界

由於之前做 force-alignment 時所用的字典是 cmu 0.7a，以 import 這個 word 為例

IMPORT　IH0 M P AO1 R T

IMPORT　IH1 M P AO0 R T

只有兩欄，第一欄是 word，第二欄是此 word 的發音，每個母音後面都有一個數字，代表此母音是否是重音，1 表重音，2 表次重音，0 表沒有。此字典的優點是詞彙量多(約 13 萬)，各種字的發音齊全，很適合用在語音辨識，一個具有 multi-pronunciation 的詞大

部分發音都有收錄。缺點是它雖然也有一詞多音的支援，可是沒有提供詞性，無法得知在什麼詞性下該發什麼音，也不提供音節邊界的標記。使得它不適合作為語音合成之用，但它卻很適合拿來作為 force-alignment 的字典。

有了 force alignment 的結果，就有各個 phone 的時間切割資訊，接著為了得到 cmu 0.7a 所不提供的音節邊界標記，於是使用 festival[9]這套軟體裡所用的 cmu 0.4 字典(檔名為 cmu.out)格式說明如下，可得 word 的 syllable boundary 資訊。字典的內容如下：

```
          1       2          3
          ↓       ↓          ↓
("import"       n        (((ih m) 1) ((p ao r t) 0)))
("import"       v        (((ih m) 0) ((p ao r t) 1)))
("animal"       nil      (((ae) 1) ((n ax) 0) ((m ax l) 0)))
```

第一行代表詞，第二行代表這個詞的詞性，如果這個詞在此字典中只有一種發音，則它的詞性即用 nil 作代表。第三行代表這個詞的音素，同在一個音節中的音素會用小括號括起來，後面的數字 1 代表，這個音節是有重音的，0 則沒有。

利用 Stanford POS Tagger, v. 3.0[26]可將文本標記所屬的 POS。輸出格式如下：

They_PRP breathe_VBP through_IN lungs_NNS ,_, not_RB through_IN gills_NNS ,_, and_CC give_VB birth_NN to_TO live_VB young_JJ ._.

底線後面的代號代表那個字所屬的 pos。Stanford POS Tagger 裡代號所代表的詞性如表 1.6，不含標點符號共 36 類。得到的這個字的詞性，即可藉由詞性及 cmu.out 判斷正確的發音。做到這裡會發生某些問題。例如：

Q1: 做 force-alignment 所使用的 triphone model，將/ʌ/, /ə/這兩個 phone 都標示成 AH，這點和我所使用的 cmu 0.7a 的標示法一樣。而 cmu.out 是將這兩個 phone 分別標示為 AH、AX，目前的解決之道是將 cmu.out 中所有出現 AX 的地方都取代成 AH。

Q2: 以 particually 這個單字為例，在 cmu 0.7a 中有兩種發音，在此代表兩種唸法都有人在唸，但不代表這個字具有兩種以上詞性，正統的英文發音是選後者

PARTICULARLY　P AA2 R T IH1 K Y AH0 L ER0 L IY0　← force-alignment

PARTICULARLY　P ER0|T IH1|K Y AH0|L ER0|L IY0　← cmu.out

而 force-alignment 的結果，假設選到前者(具有 12 個 phone，5 個母音)。可是 cmu.out 中 particularly 這個單字只有一種發音—後者(具有 11 個 phone，5 個母音)，音節切割結果如上圖紅線標示。因此，如果想用 cmu.out 字典所提供的音節資訊，替 force-alignment 的結果切音節，會發生 phone 數不合的問題，以及如下圖的不合理結果：第二個音節中沒有母音，而最後一個音節中卻有兩個母音的奇怪情形。

P AA2 | R T | IH1 K Y | AH0 L | ER0 L IY0

Q3:再看一個例子，在 cmu 0.7a 字典中 natural 這個字有兩種發音

NATURAL　N　AE1 | CH　ER0 | AH0　L　← cmu.out

NATURAL　N　AE1　CH　R　AH0　L　← force alignment

而 force-alignment 選到了後者的結果（6 個 phone，2 個母音），但 cmu.out 中只有前者的發音（6 個 phone，3 個母音），音節切割結果如上圖紅線標示，如果用 cmu.out 的音節資訊，替 force-alignment 的結果切音節，會發現如下圖的不合理結果：第二個音節中沒有母音

NATURAL　N　AE1 | CH　R | AH0　L

類似的單字還有許多，我一方面想要 cmu 0.7a 中 force-alignment 的 phone sequence 時間資訊，另一方面又想要 cmu.out 的音節資訊。當兩種結果互相衝突時，我選則了保留時間資訊。再使用其它方法切音節。方法如下。

由[27]得知

狀況 1：當兩個母音中間沒有子音 → 直接切在兩個母音中間。

狀況 2：當兩個母音中間有一子音，絕大部份的情形是將那子音，切給後面的母音，例外，當前面有音節有明顯的短發音，於是我將 1 連音的規則定為 0:1(表 0 個子音切給前面母音，1 個子音切給後面)。

狀況 3：當兩個母音中間有二子音，一邊分一個子音。

狀況 4：當兩個母音中間有三個以上子音 → 沒有規則

對於狀況 4 我試著統計文本中，三連子音、四連子音中的個數，分別是 180 與 20 個，英文裡沒有五連子音以上的情形。從三連子音中，隨機抽 20 個來看發現 1:2(表示一個子音切給前面母音，2 個子音切給後面)的有 14 例，2:1 的有 6 例，0:3 有 1 例。於是將三連音的規則定為 1:2。再看另外 20 個四連音中的音節切割情形，發現，1:3 的有 18 例，2:2 的有 2 例，於是將四連音的規則定為 1:3。如此，就定下了自己的音節切割規則。

第三步：標加強音（Accent）邊界

將所有詞分為內容詞(content word)與功能詞(function word)兩類，其中根據[14]功能詞的定義為「連結詞、限定詞、介系詞、代名詞」共四類，參考 Stanford POS Tagger 程式的結果，將'CC', 'DT', 'IN', 'PDT', 'PRP', 'PRP$', 'WDT', 'WP', 'WP$', 'MD', 'RP', 'TO'共 12 類歸為功能詞，其於為內容詞，對於加強音(accent)定義：參考 festival[28]文件中說明，對於英文加強音的位置，在一般資料庫中，有高達 80%的機率會出現在 content word 中重音出現的位置。

詞層次中關於詞性的定義，festival 中使用前後文獨立的詞性定義，將所有詞性共分為 9 種，各類詞性的定義如下：

(1)in：of for in on that with by at from as if that against about before because if under after over into while without

(2) to： to

(3) det： the a an no some this that each another those every all any these both neither no many through new between among until per up down

(4) md：will may would can could should must ought might

(5) cc：and but or plus yet nor

(6) wp：who what where how when

(7) pps：her his their its our their its mine

(8) aux：is am are was were has have had be

(9) content：剩下的 word

我自己的做法，則是參考 Stanford POS Tagger 程式的結果，將原本的 36 類再併為 14 類，分別為

(1) aux：is am are was were has have had be

(2) in：IN

(3) to：TO

(4) cc：CC

(5) uh：UH

(6) md：MD

(7) wp：WP、WRB

(8) det：DT、PDT、RP、WDT

(9) pps：POS、PRP$、WP$

(10) n：LS、NN、NNP、NNPS、NNS、CD、FW、SYM

(11) v：VB、VBD、VBG、VBN、VBP、VBZ

(12) adj：JJ、JJR、JJS

(13) pro：PRP、EX

(14) adv：RB、RBR、RBS

第四步：標定片語邊界

首先先定義片語，根據 festival[28]裡頭文件所定義的 phrase, 「對於片語邊界的位置，可以標在每句中任何 silence 音素之前。」，於是在訓練階段，片語邊界是切在任何標點符號的位置上，和 pause 長度≧ 200ms 的地方。

在測試階段 phrase boundary 是切在任何標點符號的位置上，除此之外，參考 festival[28]文件，符合下列條件也會出現 phrase boundary。

(1) 現在的 word 詞性是 content word，而下一個 word 是 function word，切在兩者之間。

(2) 從上一個標點符號，到將要切的位置之間的 word 個數，必須超過五個。

(3) 從將要切的位置，到下一個標點符號之間的 word 個數，必須超過五個。

這裡 function word 的定義是根據[14]中的定義即為「連結詞、限定詞、介系詞、代名詞」共四類，再參考 Stanford POS Tagger 程式的結果，將'CC', 'DT', 'IN', 'PDT', 'PRP', 'PRP$', 'WDT', 'WP', 'WP$', 'MD', 'RP', 'TO'共 12 類歸為 function word，其於為 content word Sentence boundary，即為出現驚嘆號(!)、問號(?)、句號(.)這三種符號所在的位置。有了以上資訊之後，即可將文本資訊，轉成 phone & prosody feature。

3.3 特徵參數抽取

SPTK(speech signal toolkit)[23]是一個功能強大的語音訊號處理工具，在進行模型訓練之前，除了準備語料的標記檔，還需要語料的特徵參數，包括頻譜參數、基頻參數。頻譜部分的計算是使用 SPTK 處理，音檔的取樣頻率為 16k，音框長度 25ms、音框位移 5ms、視窗為 Hamming window。頻譜參數部分使用 24 階的廣義梅爾倒頻譜係數，及其動態特徵向量(delta & delta-delta)，由左到右不可跳躍的三狀態 HMM。基頻參數部分使用 ESPS 套裝軟體中的 get_F0.tcl script，引入 snack library 計算，設定尋找基頻的上下限為 80Hz~350Hz，求得基頻及其動態特徵向量。取得上述兩部分特徵向量後，再將兩部分特徵向量正規化並結合成*.cmp 檔以符合 HTK 格式。下一小節是自己用 python 語言寫的程式，將文本，force alignment 過後的切割資訊，參考字典轉為文本標示資訊。

3.4 文本標示資訊與問題集設計

文本標示資訊為HTS相當重要的一環，採用那些語言參數會直接影響context dependent

model的狀態分裂合併結果。這裡所用的詞類分法是依據Penn Treebank project[29] 共36類。根據2.2節所定義的英文音素模型，加上利用前後文相關的語參數，輸出文本標示。本論文所採用的語參數，可粗分為五大類：音素層次(phone level)、音節層次(syllable level)、詞層次(word level)、片語層次(phrase level)、句子層次(sentence level)，詳細所使用的文脈相關語參數格式為下，符號代表的意義如表3.1所示：

p1^p2-p3+p4=p5@p6_p7

/A:a1_a2

/B:b1-b2@b3-b4&b5-b6#b7-b8!b9-b10|b11

/C:c1+c2

/D:d1_d2 /E:e1+e2@e3+e4 /F:f1_f2

/G:g1_g2 /H:h1=h2@h3=h4 /I:i1_i2

/J: j1+ j2- j3

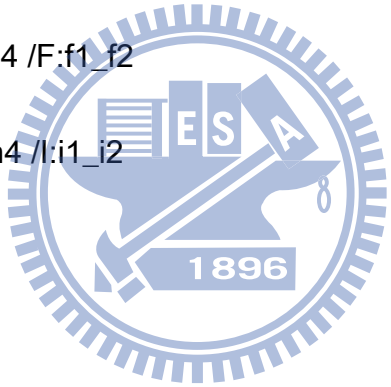


表 3. 1：文脈資訊

level	ID	Description
Phone level	p_1	the phoneme identity before the previous phoneme
	p_2	the previous phoneme identity
	p_3	the current phoneme identity
	p_4	the next phoneme identity
	p_5	the phoneme after the next phoneme identity
	p_6	position of the current phoneme identity in the current syllable (forward)
	p_7	position of the current phoneme identity in the current syllable (backward)
Syllable level	a_1	whether the previous syllable stressed or not (0: not stressed, 1: stressed)
	a_2	the number of phonemes in the previous syllable
	b_1	whether the current syllable stressed or not (0: not stressed, 1: stressed)
	b_2	the number of phonemes in the current syllable
	b_3	position of the current syllable in the current word (forward)
	b_4	position of the current syllable in the current word (backward)
	b_5	position of the current syllable in the current phrase (forward)
	b_6	position of the current syllable in the current phrase (backward)
	b_7	the number of stressed syllables before the current syllable in the current phrase
	b_8	the number of stressed syllables after the current syllable in the current phrase
	b_9	the number of syllables from the previous stressed syllable to the current syllable
	b_{10}	the number of syllables from the current syllable to the next stressed syllable
	b_{11}	name of the vowel of the current syllable
	c_1	whether the next syllable stressed or not (0: not stressed, 1: stressed)
	c_2	the number of phonemes in the next syllable
Word level	d_1	gpos (guess part-of-speech) of the previous word
	d_2	the number of syllables in the previous word
	e_1	gpos (guess part-of-speech) of the current word
	e_2	the number of syllables in the current word
	e_3	position of the current word in the current phrase (forward)
	e_4	position of the current word in the current phrase (backward)
	f_1	gpos (guess part-of-speech) of the next word
	f_2	the number of syllables in the next word
Phrase level	g_1	the number of syllables in the previous phrase
	g_2	the number of words in the previous phrase
	h_1	the number of syllables in the current phrase
	h_2	the number of words in the current phrase
	h_3	position of the current phrase in this utterance (forward)
	h_4	position of the current phrase in this utterance (backward)
	i_1	the number of syllables in the next phrase
	i_2	the number of words in the next phrase
Sentence level	j_1	the number of syllables in this sentence
	j_2	the number of words in this sentence
	j_3	the number of phrases in this utterance

建立好文脈標示後，接著根據表 3.1 的參數設計相關問題集，為達到最佳狀態分裂合併結果，考量五大類問題集，說明如下：

1. 音素層次(phone level):

- {之前、目前、之後}的音素
- 音素在目前的音節中的位置
- 母音發音類別：舌前後音、圓音、舌高低音、、、
- 子音發聲類別：爆破音、摩擦音、鼻音、、、

2. 音素層次(syllable level):

- {之前、目前、之後}的音節中音素個數
- {之前、目前、之後}的音節是否有強調(accent)
- {之前、目前、之後}的音節是否有加重音(stress)
- 目前的音節在目前詞中的位置
- 在目前的片語中，{往前、往後}數的重音(stress)個數
- 在目前的片語中，{往前、往後}數的強調音(accent)個數
- {往前、往後}數，到下一個重音之間共有幾個音節
- {往前、往後}數，到下一個強調音之間共有幾個音節
- 目前這音節中的母音

3. 詞層次(word level)

- {之前、目前、之後}的詞的詞性
- {之前、目前、之後}的詞所含的音節個數
- 目前的詞在目前片語中的位置
- 在目前的片語中，{往前、往後}數，內容詞(content word)的個數
- {往前、往後}數，到下一個內容詞(content word)之間的詞個數

4. 片語層次(phrase level)

- {之前、目前、之後}的片語中，音節的個數

- 目前的片語在這句中位置

5. 句子層次(sentence level)

- 這句中的音節、詞、片語個數

綜合以上類別的考量，本論文使用約 1400 個問題集，問題集內詳細的內容附在附錄一，並依照圖 2.1 的 HTS 方塊圖，實作出一套以 HMM 為基礎的英文語音合成系統。

3.5 全域變異數

傳統的參數產生演算法中，在合成階段，需要產生靜態特徵參數以供合成，而產生的靜態特徵向量的條件，是要能使得整段特徵向量(包含靜態和動態特徵向量)對所給定的隱藏式馬可夫模型的相似度（likelihood）為最大值。上述這種參數產生演算法，因為在模型參數估計時，為得到較小的誤差值而產生較接近平均值的結果，常導致參數過度平滑化，而造成合成語音含糊不清。為減輕過度平滑化的問題，HTS 加入了全或變異數(Global Variance)參數產生演算法[21]。這種考慮到全域變異的演算法，特色是在產生特徵參數向量時，不只考慮對特定隱藏式馬可夫模型的相似性，同時對考慮對全域變數的相似性，藉此對過度平滑化的參數作補償，能有效提升合成語音的自然度。

3.6 模型訓練

上述所需要準備的標記、特徵參數、問題集都準備好之後，即可開始隱藏式馬可夫模型的訓練。

訓練步驟如下：

- 模型參數初始化及重新預估(initialization & reestimation)

- 建立單音的 mmf 檔案(making a monophone master macro file)
- 內嵌式單音模型參數重新估算(embedded reestimation(monophone))
- 複製單音 mmf 檔案成為全文本相關的 mmf 檔案(copying monophone mmf to fullcontext one)
- 內嵌式全文模型參數重新估算(embedded reestimation (fullcontext))
- 決策樹文本分類(tree-based context clustering)
- 內嵌式分類後模型參數重新估算(embedded reestimation(clustered))
- 解開參數共享結構(untying the parameter sharing structure)
- 內嵌式解開後的模型參數預估(embedded reestimation(untied))
- 決策樹文本分類(tree-based context clustering)
- 重新分類過後的模型參數再作內嵌式重新預估(embedded reestimation(re-clustered))

合成步驟如下：

- 把 mmf 檔案轉換成 hts_engine 合成軟體所需格式。
- 使用 hts_engine 合成語音。

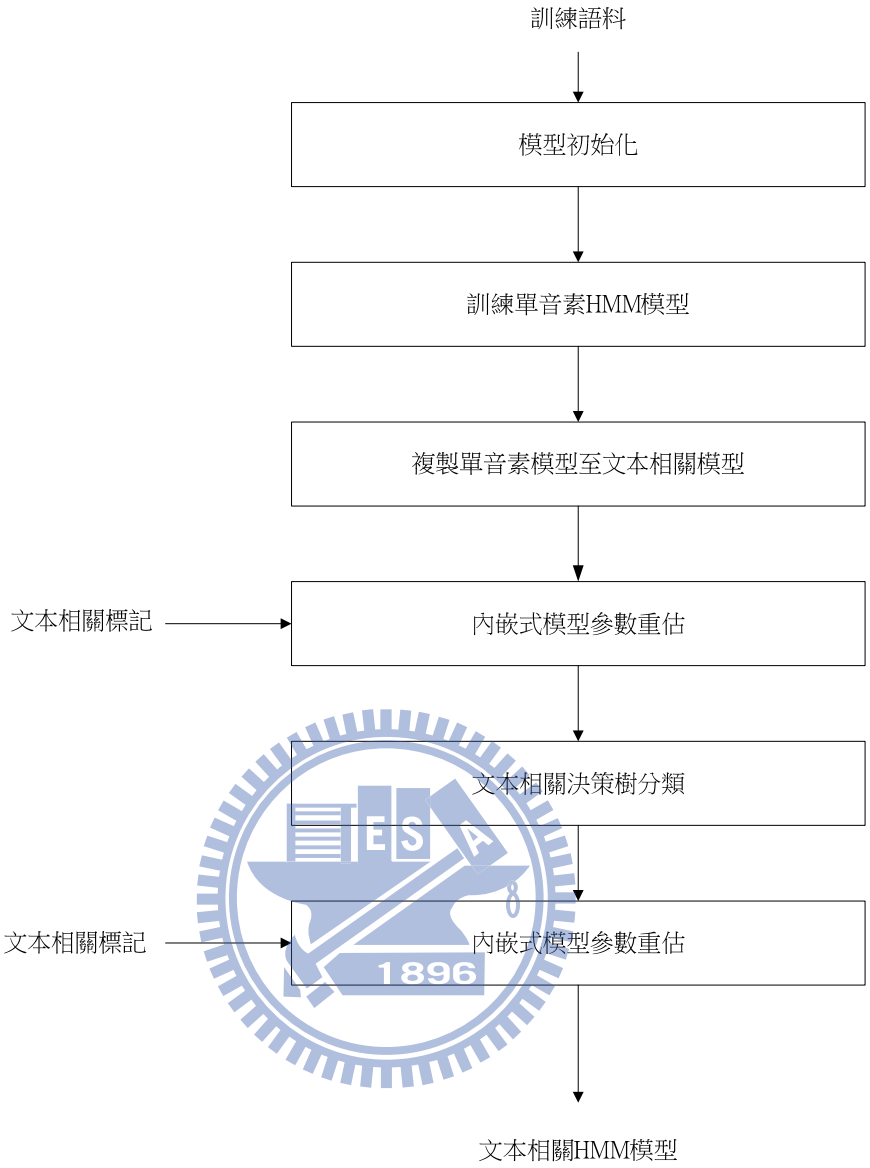


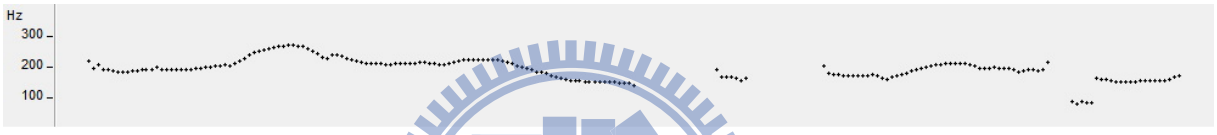
圖 3. 2 HTS 訓練階段流程圖

第四章 實驗結果與分析

本章共分三小節，分別是 4.1 基頻曲線圖比較、4.2 主觀式評估比較與 4.3 實驗結果分析。將利用 HTS 所合成的音檔，與其它實驗室使用相同引擎，線上 demo 版合成的音檔作比較。

4.1 基頻曲線圖比較

文句一：No matter where you live or travel on Earth



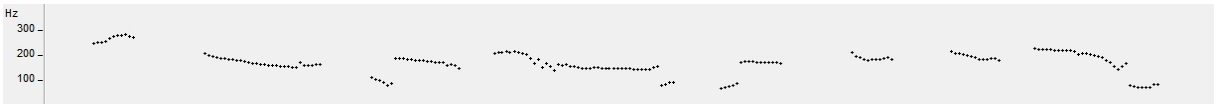
(a) 自然語音 Pitch contour

(b) 本論文合成的語音 Pitch contour

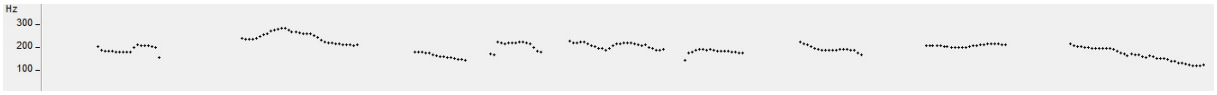
(c) 國外網站合成的語音 Pitch contour

圖 4. 1 內文編號 11-1 合成語音與自然語音基頻曲線圖對照

文句二：Some storms are deadly or cause great damage



(a) 自然語音 Pitch contour



(b) 本論文合成的語音 Pitch contour

(c) 國外網站合成的語音 Pitch contour

圖 4. 2 內文編號 11-2 合成語音與自然語音基頻曲線圖對照

由圖 4.2 可以發現，自然語音的的韻律起伏是最明顯，其次是自己合成的語音，國外網站合成的語音是最沒有高低起伏。可是由圖 4.1 可以很明顯的看出來，我合成出來的 pitch 有時會在不該有劇烈高低起伏的地方，產生奇怪的起落。

4.2 主觀式評估比較

本實驗的合成語音評量利用主觀式評估法，對合成系統做進一步的評估，採用平均鑑定分數(Mean Opinion Scores, MOS)作為評估的標準[30]，這種評估方式將合成語音輸出的自然度分為優良、良好、尚可、差、極差五個等級，分別給予 5 至 1 不等的分數。測試人員在聽過合成語音後，根據所感覺到的自然度評分。

測試是由[31]網站的線上即時 demo 系統，與自己做的系統，合成相同的中文句，做對照實驗。在此實驗中，合成二十個不含在訓練語料中的句子，由 5 位測試人員，聆聽並根據自己所感受的語音自然度打分數，最後取一個平均。

實驗中，比較兩套系統 (A)、(B)，在合成語音自然度上的差異。

(A) 系統是利用自己做的系統。

(B) 系統是網站上的系統。

表 4. 1：品質主觀評量

	平均 MOS
(A)	2.1
(B)	3.2

由上表結果可發現，目前單純利用 HTS 引擎所提供的音質，仍無法與其它使用相同引擎合出來的音質好聽，聲音不夠乾淨。

4.3 實驗結果分析

很明顯的目前合成出來的聲音仍不夠好聽，推測的原因如下：

1. /ʌ/, /ə/這兩個 phone，在訓練與合成時的處理都當成同一個音素表示，但聽覺上，這兩個音素並不是那麼難以分辨。
2. 當在訓練端，模型音節切割，若遇到 force-alignment 的結果與字典結果不合時，目前是採用，根據資料庫裡得到的統計資料所定的規則法，例如將三連子音的規則定為 1:2，但仍嫌不夠準確。
3. 標加強音時，雖然利用規則法將加強音標在 content word 中重音出現的位置。不過根據[28]文件，此種方法，仍有 20%錯誤的機會
4. 英文裡片語的定義是很不容易找出一個通用規則去描述，在這裡只用一個粗估的定義「現在的詞的詞性是 content word，而下一個詞是 function word，且前後詞數需大於 5，則切在兩詞之間」，仍不夠精確。
5. 合成階段，目前在給定任意文字下，韻律資訊中 pause 的位置，目前只標在各標點符號與各片語邊界的地方。並無法準確預估文句中所有 pause 的位置。

6. 切割位置不完整的影響，切割位置的不完整包括有前面不完整或後面不完整兩類，實驗結果發現，當合成單元受到連音效應的影響使切割位置不容易準確或切割位置錯誤，導致切割位置會在該音素之前或是前一音素尚未結束時，這屬於切割位置前面不完整，相對的，對於前一音素而言，則屬於切割位置及早結束，即後面不完整。

第五章 結論與未來展望

基於隱藏式馬可夫模型的英文語音合成，已經越來越廣泛的應泛在各種實際系統中。然而如何使得合成語音具有像自然語音那樣生動的韻律和節奏，一直是語音合成領域中的一大挑戰。本論文透過在標記中加上音素與音節、詞、片語、句子四層結構的相關位置的韻律資訊，增加合成語音的韻律、節奏的自然度。

本論文更進一步利用考慮前後文較精確的 POS 估計結果，為文句分析做標記。並將結果做更精簡的分群，能夠提升合成語音韻律的自然度。

本論文所使用的文句分析器較為陽春，目前若測試文字檔中如出現阿拉伯數字，縮寫(ex, Dr.)，首字母大寫縮寫字(IEEE, AIDS)，及字典中找不到的字，還沒有加入字母轉發音的方法。因此對於上述情形系統會出現跳過不唸結果，在未來日子這些仍是可以補強的地方。

參考文獻

[1] N. Iwahashi, N. Kaiki, and Y. Sagisaka, “speech segment selection for concatenative synthesis based on spectral distortion minimization,”IEICE Trans. Fundamentals, vol.E76-A, no.11, pp.1942-1948, Nov.1998.

[2] A.J. Hunt and A.W. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” Proc. ICASSP, pp.373-376, Atlanta, USA, May 1996.

[3] S. Nakajima and H. Hamada, “Automatic generation of synthesis units based on context oriented clustering,” Proc. ICASSP, pp.659-662, New York, USA, April 1988.

[4] M. Akamine and T. Kagoshima, “Analytic generation of synthesis units by closed loop training for totally speaker driven text to speech system (TOS Drive TTS),” Proc. ICSLP, pp.1927-1930, Sydney, Australia, Dec. 1998.

[5] T. Mizutani and T. Kagoshima, “Concatenative speech synthesis based on the plural unit selection and fusion method,” IEICE Trans. Inf. & Syst., vol.E88-D, no.11, pp.2565-2572, Nov. 2005.

[6] K. Tokuda, H. Zen, and A. W. Black, “An HMM-based speech synthesis system applied to English,” Proc. IEEE 2002 Workshop on speech Synthesis, Santa Monica, USA, Sept. 2002.

[7] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T.Kitamura, “Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis,” Proc. EUROSPEECH, pp.2347-2350, Budapest, Hungary, Sept. 1999.

[8] Zen, H., Nose, T., Yamagishi, J., Sako, S. and Tokuda, K., The HMM-based Speech System(HTS) Version 2.1 2007, http://hts.sp.nitech.ac.jp/

[9] Festival, http://www.cstr.ed.ac.uk/projects/festival/

[10] T. Yoshimura, T. Masuko, K. Tokuda, T. Kobayashi, T. Kitamura, “Speaker interpolation for HMM-based speech synthesis system,” J. Acoust. Soc. Jpn. (E), vol.21, no.4,

pp.199-206, 2000

[11] M. Tamura, T. Masuko, K. Tokuda, T. Kobayashi, “Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR,” Proc of ICASSP, pp.805-808, May 2001

[12] T. Yoshimura, “Simultaneous modeling of phonetic and prosodic parameters, and characteristic conversion for HMM-based text-to-speech systems,” Ph.D thesis, Nagoya Institute of Technology, Jan. 2002.

[13] S. Imai, “Cepstral analysis synthesis on the mel frequency scale,” Proc. Of ICASSP, pp.93–96, Feb. 1983.

[14]X. Huang, A. Acero, H. Hon, “Spoken Language Processing; A Guide to Theory, Algorithm and System Development” Prentice Hall; May 2001

[15] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, “Mel-generalized cepstral analysis – A unified approach to speech spectral estimation”, Proc. ICASSP, pp.1043-1046, 1994.

[16] T. Kobayashi and S. Imai, “Spectral analysis using generalized cepstrum,” IEEE Trans. Acoust., Speech, Signal processing, vol. ASSP-32, pp.1087-1089, Oct. 1984.

[17] K. Tokuda, T. Mausko, N. Miyazaki, T. Kobayashi, Multi-space probability distribution HMM, IEICE Trans. Inf. & Syst., vol.E85-D, no.3, pp.455-464, March 2002.

[18] J. J. Odell, “The Use of Context in Large Vocabulary Speech Recognition,”PhD dissertation, Cambridge University, 1995.

[19] K. Shinoda and T. Watanabe “MDL-based context-dependent subword modeling for speech recognition” Acoustical Science and Technology Vol. 21 (2000) , No. 2 pp.79-86

[20] K. Shinoda and T. Watanabe, “Speaker Adaptation with Autonomous Model Complexity Control by MDL Principle,” Proc. of ICASSP, pp.717–720, May 1996.

[21] T. Toda and K. Tokuda "A speech parameter generation algorithm considering global variance for HMM-based speech synthesis", IEICE Trans., vol. E90-D, p.816 , 2007.

[22] Hidden Markov Model Toolkit(HTK), http://htk.eng.cam.ac.uk/

[23] Speech Signal Processing Toolkit(SPTK), http://sp-tk.sourceforge.net/

[24] HTS engine, http://hts-engine.sourceforge.net/

[25] http://www.keithv.com/software/htk/us/

[26] Stanford POS Tagger, v. 3.0(http://nlp.stanford.edu/software/tagger.shtml)

[27] http://english.glendale.cc.ca.us/phonics.rules.html

[28] http://www.festvox.org/docs/manual-1.4.3/festival_toc.html

[29] http://www.cis.upenn.edu/ldc

[30] Min Chu and Hu Peng, “An Objective Measure for Estimating MOS of Synthesized Speech” in EuroSpeech 2001.

[31] http://www.sp.nitech.ac.jp/demo/flite+hts_engine/

附錄一 決策樹問題

1. Phone level

QS 前二個、前一個、目前、後一個、後二個的音素名稱

QS 前二個、前一個、目前、後一個、後二個的音素類型，以[目前]音素為例，共 60 類

QS Is left second phone is Vowel

QS Is left second phone is Consonant

QS Is left second phone is Stop {B,D,DX,G,K,P,T}

QS Is left second phone is Nasal {M,N,EN,NG}

QS Is left second phone is Fricative {CH,DH,F,HH,HV,S,SH,TH,V,Z,ZH}

QS Is left second phone is Liquid {EL,HH,L,R,W,Y}

QS Is left second phone is Front {AE,B,EH,EM,F,IH,IX,IY,M,P,V,W}

QS Is left second phone is Central {AH,AO,AXR,D,DH,DX,EL,EN,ER,L,N,R,S,T,TH,Z,ZH}

QS Is left second phone is Back {AA,AX,CH,G,HH,JH,K,NG,OW,SH,UH,UW,Y}

QS Is left second phone is Front_Vowel {AE,EH,EY,IH,IY}

QS Is left second phone is Central_Vowel {AA,AH,AO,AXR,ER}

QS Is left second phone is Vowel {AX,OW,UH,UW}

QS Is left second phone is Long_Vowel {AO,AW,EL,EM,EN,EN,IY,OW,UW}

QS Is left second phone is Short_Vowel {AA,AH,AX,AY,EH,EY,IH,IX,OY,UH}

QS Is left second phone is Dipthong_Vowel {AW,AXR,AY,EL,EM,EN,ER,EY,OY}

QS Is left second phone is Front_Start_Vowel {AW,AXR,ER,EY}

QS Is left second phone is Fronting_Vowel {AY,EY,OY}

QS Is left second phone is High_Vowel {IH,IX,IY,UH,UW}

QS Is left second phone is Medium_Vowel {AE,AH,AX,AXR,EH,EL,EM,EN,ER,EY,OW}

QS Is left second phone is Low_Vowel {AA,AE,AH,AO,AW,AY,OY}

QS Is left second phone is Rounded_Vowel {AO,OW,OY,UH,UW,W}

QS Is left second phone is Unrounded_Vowel
{AA,AE,AH,AW,AX,AXR,AY,EH,EL,EM,EN,ER,EY,HH,IH,IX,IY,L,R,Y}

QS Is left second phone is Reduced_Vowel {AX,AXR,IX}

QS Is left second phone is IVowel {IH,IX,IY}

QS Is left second phone is EVowel {EH,EY}

QS Is left second phone is AVowel {AA,AE,AW,AXR,AY,ER}

QS Is left second phone is OVowel {AO,OW,OY}

QS Is left second phone is UVowel {AH,AX,EL,EM,EN,UH,UW}

QS Is left second phone is Unvoiced_Consonant {CH,F,HH,K,P,S,SH,T,TH}

QS I Is left second phone is Voiced_Consonant
{B,D,DH,DX,EL,EM,EN,G,JH,L,M,N,NG,R,V,W,Y}

QS Is left second phone is Front_Consonant {B,EM,F,M,P,V,W}
QS Is left second phone is Central_Consonant {D,DH,DX,EL,EN,L,N,R,S,T,TH,Z,ZH}
QS Is left second phone is Back_Consonant {CH,G,HH,JH,K,NG,SH,Y}
QS Is left second phone is Fortis_Consonant {CH,F,K,P,S,SH,T,TH}
QS Is left second phone is Lenis_Consonant {B,D,DH,G,JH,V,Z,ZH}
QS Is left second phone is Neigther_F_or_L {EL,EM,EN,HH,L,M,N,NG,R,W,Y}
QS Is left second phone is Coronal_Consonant
{CH,D,DH,DX,EL,EN,JH,L,N,R,S,SH,T,TH,Z,ZH}
QS Is left second phone is Non_Coronal {B,EM,F,G,HH,K,M,NG,P,V,W,Y}
QS Is left second phone is Anterior_Consonant
{B,D,DH,DX,EL,EM,EN,F,L,M,N,P,S,T,TH,V, W,Z}
QS Is left second phone is Non_Anterior {CH,G,HH,JH,K,NG,R,SH,Y,ZH}
QS Is left second phone is Continuent
{DH,EL,EM,EN,F,HH,L,M,N,NG,R,S,SH,TH,V,W,Y,Z,ZH}
QS Is left second phone is No_Continuent {B,CH,D,G,JH,K,P,T}
QS Is left second phone is Positive_Strident {CH,JH,S,SH,Z,ZH}
QS Is left second phone is Negative_Strident {DH,F,HH,TH,V}
QS Is left second phone is Neutral_Strident {B,D,EL,EM,EN,G,K,L,M,N,NG,P,R,T,W,Y}
QS Is left second phone is Glide {HH,L,EL,R,Y,W}
QS Is left second phone is Syllabic_Consonant {AXR,EL,EM,EN,ER}
QS Is left second phone is Voiced_Stop {B,D,G}
QS Is left second phone is Unvoiced_Stop {P,T,K}
QS Is left second phone is Front_Stop {B,P}
QS Is left second phone is Central_Stop {D,T}
QS Is left second phone is Back_Stop {G,K}
QS Is left second phone is Voiced_Fricative {JH,DH,V,Z,ZH}
QS Is left second phone is Unvoiced_Fricative {CH,F,S,SH,TH}
QS Is left second phone is Front_Fricative {F,V}
QS Is left second phone is Central_Fricative {DH,S,TH,Z}
QS Is left second phone is Back_Fricative {CH,JH,SH,ZH}
QS Is left second phone is Affricate_Consonant {CH,JH}
QS Is left second phone is Not_Affricate {DH,F,S,SH,TH,V,Z,ZH}
QS Is left second phone is silences {sp,sil}

2. Syllable level

QS{之前、目前、之後}的音節中音素個數是否為{1}{2}{3}{4}{5}{6}{7}

QS{之前、目前、之後}的音節中音素個數是否{≦1}{≦2}{≦3}{≦4}{≦5}{≦6}{≦7}

QS{之前、目前、之後}的音節是否有強調(accent)

QS{之前、目前、之後}的音節是否有加重音(stress)

QS 目前的音節在目前詞中往前數的位置是否為{1}{2}{3}{4}{5}{6}{7}

QS 目前的音節在目前詞中往前數的位置是否{≦1}{≦2}{≦3}{≦4}{≦5}{≦6}{≦7}

QS 目前的音節在目前詞中往後數的位置是否為{1}{2}{3}{4}{5}{6}{7}

QS 目前的音節在目前詞中往後數的位置是否{≦1}{≦2}{≦3}{≦4}{≦5}{≦6}{≦7}

QS 目前的音節在目前片語中往前數的位置是否為{1}{2}{3}、、{19}{20}

QS 目前的音節在目前片語中往前數的位置是否{≦1}{≦2}、、{≦19}{≦20}

QS 目前的音節在目前片語中往後數的位置是否為{1}{2}{3}、、{19}{20}

QS 目前的音節在目前片語中往後數的位置是否{≦1}{≦2}、、{≦19}{≦20}

QS 在目前的片語中，往前數的重音(stress)個數是否為{1}{2}{3}、、{11}{12}

QS 在目前的片語中，往後數的重音(stress)個數{≦1}{≦2}、、{≦11}{≦12}

QS 在目前的片語中，往前數的強調音(accent)個數是否為{1}{2}{3}、、{5}{6}

QS 在目前的片語中，往前數的強調音(accent)個數是否{≦1}{≦2}、、{≦5}{≦6}

QS 在目前的片語中，往後數的強調音(accent)個數是否為{1}{2}{3}、、{6}{7}

QS 在目前的片語中，往後數的強調音(accent)個數是否{≦1}{≦2}、、{≦6}{≦7}

QS 往前數，到下一個重音之間的音節數是否為{1}{2}{3}、、{5}

QS 往前數，到下一個重音之間的音節數是否為{≦1}{≦2}、、{≦5}

QS 往後數，到下一個重音之間的音節數是否為{1}{2}{3}、、{5}

QS 往後數，到下一個重音之間的音節數是否為{≦1}{≦2}、、{≦5}

QS 往前數，到下一個強調音之間的音節數是否為{1}{2}{3}、、{16}

QS 往前數，到下一個強調音之間的音節數是否為{≦1}{≦2}、、{≦16}

QS 往後數，到下一個強調音之間的音節數是否為{1}{2}{3}、、{16}

QS 往後數，到下一個強調音之間的音節數是否為{≦1}{≦2}、、{≦16}

QS 目前這音節中的母音名稱

QS 前一個，目前、後一個音節是是否為重音

QS 前一個，目前、後一個音節是是否為強調音

word level

QS {之前、目前、之後}的詞的詞性

QS {之前、目前、之後}的詞所含的音節個數是否為{1}{2}{3}、、{7}

QS {之前、目前、之後}的詞所含的音節個數是否為{≦1}{≦2}、、{≦7}

QS 往前數過來，目前的詞在目前片語中的位置是否為{1}{2}{3}、、{13}

QS 往前數過來，目前的詞在目前片語中的位置是否為{≦1}{≦2}、、{≦13}

QS 往後數過來，目前的詞在目前片語中的位置是否為{1}{2}{3}、、{13}

QS 往後數過來，目前的詞在目前片語中的位置是否為{≦1}{≦2}、、{≦13}

QS 在目前的片語中，往前數過來，內容詞(content word)的個數是否為{1}{2}{3}、、{9}

QS 在目前的片語中，往前數過來，內容詞的個數是否為{≦1}{≦2}、、{≦9}

QS 在目前的片語中，往後數過來，內容詞(content word)的個數是否為{1}{2}{3}、、{8}

QS 在目前的片語中，往後數過來，內容詞的個數是否為{≦1}{≦2}、、{≦8}

QS 往前數，到下一個內容詞之間的詞個數是否為{1}{2}{3}、、{5}

QS 往前數，到下一個內容詞之間的詞個數是否為{≦1}{≦2}、、{≦5}

QS 往後數，到下一個內容詞之間的詞個數是否為{1}{2}{3}、、{5}

QS 往後數，到下一個內容詞之間的詞個數是否為{≦1}{≦2}、、{≦5}

Phrase level

QS 前一個片語中，音節的個數是否為{1}{2}{3}、、{20}

QS 前一個片語中，音節的個數是否為{≦1}{≦2}、、{≦20}

QS 前一個片語中，詞的個數是否為{1}{2}{3}、、{13}

QS 前一個片語中，詞的個數是否為{≦1}{≦2}、、{≦13}

QS 現在的片語中，音節的個數是否為{1}{2}{3}、、{20}

QS 現在的片語中，音節的個數是否為{≦1}{≦2}、、{≦20}

QS 現在的片語中，詞的個數是否為{1}{2}{3}、、{13}

QS 現在的片語中，詞的個數是否為{≦1}{≦2}、、{≦13}

QS 目前的片語在這句中位置往前數過來是否為{1}{2}、、{4}

QS 目前的片語在這句中位置往前數過來是否為{≦2}{≦3}{≦4}

QS 目前的片語在這句中位置往後數過來是否為{1}{2}、、{4}

QS 目前的片語在這句中位置往後數過來是否為{≦2}{≦3}{≦4}

QS 後一個片語中，音節的個數是否為{1}{2}{3}、、{20}

QS 後一個片語中，音節的個數是否為{≦1}{≦2}、、{≦20}

QS 後一個片語中，詞的個數是否為{1}{2}{3}、、{15}

QS 後一個片語中，詞的個數是否為{≦1}{≦2}、、{≦15}

Sentence level

QS 這句中的音節個數是否為{1}{2}{3}、、{28}

QS 這句中的音節個數是否為{≦1}{≦2}、、{≦28}

QS 這句中的詞個數是否為{1}{2}{3}、、{13}

QS 這句中的詞個數是否為{≦2}、、{≦13}

QS 這句中的片語個數是否為{1}{2}{3}{4}

QS 這句中的片語個數是否為{≦2}、、{≦4}