

國立交通大學

電信工程學系

碩士論文

中文連續語音辨認之進一步研究

A Further Study on Continuous Mandarin Speech
Recognition

研究生：張獻文

指導教授：陳信宏 博士

中華民國九十六年七月

中文連續語音辨認之進一步研究

A Further Study on Continuous Mandarin Speech
Recognition

研究生：張獻文

Student : Xian-Wen Zhang

指導教授：陳信宏

Advisor : Dr. Sin-Horng Chen

國立交通大學



A Thesis

Submitted to Department of Communication Engineering
College of Electrical Engineering and Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in Electrical Engineering

August 2007

Hsinchu, Taiwan, Republic of China

中華民國九十六年七月

中文連續語音辨認之進一步研究

研究生：張獻文

指導教授：陳信宏 博士

國立交通大學電信工程學系碩士班



在本論文中，可以分成兩個部份，第一個是探討詞典所收錄的內容，以人工的方式，觀察所收錄的詞是否合適，若有不適合的詞則刪除。另一方面，當我們做詞典精簡內容之後，延伸而來的問題是 Out of Vocabulary(OOV)。我們從 OOV rate 的大小，觀察對辨識結果的影響。要解決 OOV 的問題，我們想出將詞做重新斷詞，長詞變短詞，使 OOV rate 下降，進而提高辨識率。第二部份則是 POS based language model 的建立，試著加入詞長的資訊，以及混和 POS 和 word 的語言模型來達到辨識率改善。最後則是使用 linear interpolation 結合兩種不同型態的語言模型，使得我們能夠再進一步的改善辨識率。將最終的音節辨認率和基本系統相比，約提昇 2.5% 左右。

A Further Study on Continuous Mandarin Speech Recognition

Student: Xian-Wen Zhang

Advisor : Dr. Sin-Horng Chen

Institute of Communication Engineering National

Chiao Tung University



In this thesis, can be divided into two parts, one is to explore dictionary contains Observation contains words it appropriate, if not for the words were deleted. The other hand, when we do dictionary streamline content, A problem is Out of Vocabulary . From OOV rate, observation of the recognition rate . To solve the problem of OOV, we do find words to constantly re-term, long term change short term, OOV rate drop further enhance recognition rate. The second part is POS based language model build, try to join the long-term information and POS and word to the language model to improve the recognition rate. Another is the use of linear interpolation combining two different styles of language model, enables us to further improve the accuracy rate. To the final ,syllable recognition rate improve about 2.5% .

誌謝

首先誠摯的感謝指導教授陳信宏教授、王逸如教授，兩位老師悉心的教導使我得以一窺語音信號處理領域的深奧，不時的討論並指點我正確的方向，使我在這些年中獲益匪淺。老師對學問的嚴謹更是我輩學習的典範。

本論文的完成另外亦得感謝阿合學長的大力協助。因為有你的體諒及幫忙，使得本論文能夠更完整而嚴謹。

兩年裡的日子，實驗室裡共同的生活點滴，學術上的討論、言不及義的閒扯、讓人又愛又怕的宵夜、趕作業的革命情感、因為睡太晚而遮遮掩掩閃進實驗室.....，感謝眾位學長、同學、學弟的共同砥礪(墮落?)，你們的陪伴讓兩年的研究生活變得絢麗多彩。

感謝性獸、阿合、阿德、barking、希群，輝哥學長們不厭其煩的指出我研究中的缺失，且總能在我迷惘時為我解惑，也感謝宏宇、小傅、啟風、友駿、小迷彩、胤賢、小肚臍、小鄧同學，因為你們的陪伴，我才能順利的度過碩士生涯。

最後，謹以此文獻給我摯愛的雙親，因為有你們的支持，我才能努力到現在，謝謝你們！！

目錄

中文摘要.....	I
英文摘要.....	II
誌謝.....	III
目錄.....	IV
圖目錄.....	VI
表目錄.....	VIII
第一章 緒論.....	1
1.1 研究動機.....	1
1.2 研究方向.....	1
1.3 章節概要.....	2
第二章 TREEBANK 基本辨認系統.....	3
2.1 ACOUSTIC MODEL 建立.....	3
2.1.1 特徵參數：38 維語音特徵參數.....	3
2.1.2 聲學模型訓練.....	3
2.2 WORD-BASED BI-GRAM LANGUAGE MODEL 建立.....	5
2.2.1 n-grams 語言模型.....	5
2.2.2 機率的smoothing.....	6
2.2.3 語言模型建立流程.....	7
2.2.4 Perplexity 計算.....	8
2.2.5 實驗：加入語言模型後辨識效能.....	9
2.3 實驗分析.....	10
第三章 詞典的精簡及OUT OF VOCABULARY (OOV) 探討.....	12

3.1 精簡詞典內容.....	12
3.2 詞典的再增減.....	13
3.3 OOV探討.....	14
3.4 解決OOV.....	16
第四章 POS BASED LANGUAGE MODEL 建立.....	21
4.1 POS DATABASE 產生.....	21
4.2 POS BASED LANGUAGE MODEL.....	22
4.2.1 Nonoverlap:.....	22
4.2.2 Overlap.....	22
4.2.3 建立POS-Based Language Model.....	23
4.3 詞長資訊.....	26
4.4 POS AND WORDS MIXTURE LM.....	28
4.4.1 Database 建立.....	28
4.4.2 訓練POS And Words Mixture LM.....	29
第五章 LINEAR INTERPOLATION WORD BASED AND POS BASED LM	31
5.1 LINEAR INTERPOLATION.....	31
5.2 產生LINEAR INTERPOLATION LM.....	31
5.3 實驗分析.....	34
第六章 結論與展望.....	37
6.1 結論.....	37
6.2 展望.....	37

圖目錄

圖 1-1 辨識流程	1
圖 2-1 已知切割模型之訓練圖	4
圖 2-2 Bigram語言模型建立流程圖	8
圖 2-3 syllable比較圖	10
圖 3-1 詞長分佈圖	13
圖 3-2 詞長分佈圖	14
圖 3-3 辨識率比較圖	16
圖 3-4 character辨識率比較圖	19
圖 3-5 character和syllable辨識率比較圖	20
圖 4-1 POS bigram model建立流程	24
圖 4-2 POS-based LM建立	24
圖 4-3 syllable辨識率比較圖	25
圖 4-4 character辨識率比較圖	27
圖 4-5 syllable辨識率比較圖	27
圖 4-6 character辨識率比較圖	29
圖 4-7 syllable辨識率比較圖	30
圖 5-1 建立linear interpolation LM流程圖	32
圖 5-2 Test-set word perplexity as a function of the interpolation coefficient λ_w, λ_c	32
圖 5-3 character辨識率比較圖	34

圖 5-4 syllable辨識率比較圖.....35

圖 5-5 character辨識率比較圖.....35



表目錄

表 2-1 參數抽取設定檔	3
表 2-2 Inside test	4
表 2-3 outside test	5
表 2-4 Database統計	7
表 2-5 測試語料Database統計	9
表 2-6 Perplexity	9
表 2-7 word辨識率	9
表 2-8 character辨識率	10
表 2-9 syllable辨識率	10
表 2-10 syllable 辨識率	11
表 2-11 syllable 辨識率	12
表 2-12 syllable error rate and error reduction	12
表 3-1 詞長比例表	13
表 3-2 詞長比例表	14
表 3-3 OOV統計表	15
表 3-4 word辨識率	15
表 3-5 character辨識率	15
表 3-6 syllable辨識率	16
表 3-7 OOV統計表	17

表 3-8 perplexity(52k)	17
表 3-9 perplexity(54k)	17
表 3-10 word辨識率	17
表 3-11 word辨識率	18
表 3-12 characters辨識率	18
表 3-13 syllable辨識率	18
表 3-14 OOV統計表	19
表 3-15 characters辨識率	20
表 3-16 syllable辨識率	20
表 4-1 POS統計表	21
表 4-2 詞典一字詞POS overlap統計表	23
表 4-3 詞典二字詞POS overlap統計表	23
表 4-4 word辨識率	25
表 4-5 character辨識率	25
表 4-6 syllable辨識率	25
表 4-7 character辨識率	26
表 4-8 syllable辨識率	26
表 4-9 character辨識率	29
表 4-10 syllable辨識率	29
表 5-1 perplexity比較表	32
表 5-2 perplexity比較表	33



表 5-3 perplexity比較表	33
表 5-4 character辨識率	33
表 5-5 syllable辨識率	34



第一章 緒論

1.1 研究動機

現代科技中的一項重要發展便是用電腦來處理語言問題，最終的目標，就是利用語言辨識技術來建立人與機器之間溝通的橋樑【1】，由於信號處理、演算法和電腦設備的進步，語音辨識技術在過去的十到二十年間有了長足的發展，例如資料驅使(Data-driven)方法，聲學模型和語言模型建立方式，以及基於動態編輯程序 (Dynamic Programming-based) 之搜尋方法等【2】。

近年來，大家都從語音訊號的方面去著手，來研究聲音的特性，以便提昇辨識率。若是回歸基本面，語音和語言是密不可分的，因此對語言這方面做研究，或許能夠提高辨識的效果。基於以上理由，本論文將以本實驗室所錄製的單一語者的 treebank 語料庫，來對語言這方面做深入的研究和探討



1.2 研究方向

一個基本的語音辨識系統方塊圖如下：

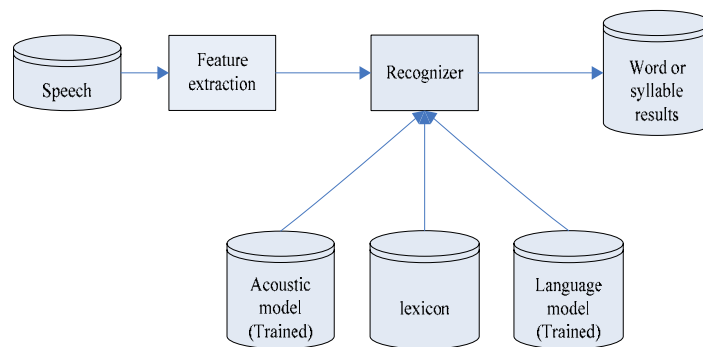


圖 1-1 辨識流程

如上圖所示，一個基本系辨識系統包含：求取語音參數，聲學模型(Acoustic Model, AM)的訓練，語言模型(Language Model, LM)的訓練還有辨識的方法，根據不同的領域，我們必須對辨識系統做調整。

近年來，我們所用的詞典大小雖然是根據統計所得來的，但是我們從未去思考我們所用的詞典是否適合於辨識系統，因此在本論文中，將會探討詞典的內容。

通常我們加入辨識器的語言模型，是屬於 word-based language model，而 word-based 的語言模型通常需要大量的訓練語料，才可以訓練出好的模型，因此在本論文中，我們將探討如何訓練出 part of speech (POS)-based 的語言模型，而它只需少量的語料即可訓練而成。

1.3 章節概要

本篇論文章節內容區分如下：

第一章 **緒論**：介紹研究動機、研究方向及章節概要。

第二章 **Treebank 基本系統建立**：訓練聲學模型和語言模型，並檢視其辨識效能。

第三章 **詞典的精簡及 Out of Vocabulary (OOV) 探討**：說明精簡詞典的內容，以及精簡後的詞典所產生的 OOV 問題，及解決方法。

第四章 **POS-based Language Model 建立**：利用 POS 的資訊做分類，使用 class-based language model 的訓練方式，將 POS-based language model 建立起來，並嘗試加入詞長資訊和建立混和 POS 及 word 的語言模型，來達到辨識結果的改善。

第五章 **Linear Interpolation word-based LM and POS-based LM**：結合兩者不同型態的語言模型，來擷取兩個語言模型的優點，來達到辨識結果的改善。

第六章 **結論與展望**

第二章 Treebank 基本辨認系統

近年來相關的語音研究，最常採用的聲學模型是利用隱藏式馬可夫模型 (Hidden Markov Model, HMM)，藉由這種機率模型，來描述發音過程的狀態(State) 轉移現象和輸出結果，這種方法可以有不錯的辨識效能，所以本系統也採用這模型。本章節的第二部份則是加入語言模型，藉由語言模型的幫助來提高辨識率。

2.1 Acoustic model 建立

2.1.1 特徵參數：38 維語音特徵參數

進行語音辨認系統之訓練、測試，首先的前處理工作就是將語音參數從輸入的語音中抽取出來。因為語音訊號之短時間穩定特性(Short Term Stationary)，加上考慮到人耳聽覺效應的補償作用，使用的參數為 MFCC (Mel-Frequency Cepstral Coefficients，梅爾倒頻譜參數)，它的成份包括 12 維 MFCC 加上能量共 13 維，取 Delta 和 Delta-Delta，加起來總共 39 維，而能量的大小對於語音辨認並不重要，所以省略了能量參數，最後得到的參數便是 38 維的語音參數。

表 2-1：參數抽取設定檔

取樣頻率	16 kHz
音框長度	30 ms
音框平移	10 ms
Filter bank 個數	24 個梅爾刻度三角濾波器

2.1.2 聲學模型訓練

我們採用的是 left-to-right HMM，雖然口腔聲道會隨時間而變，但語音訊號

具備短時間的穩定特性，因此假設在同一音框 (Frame) 中，口腔狀態是相同的。此外，代表音框與各狀態的相似程度的狀態觀測機率 (State Observation Probability)，使用混和高斯模型 (Mixture Gaussian Model) 來表示。

訓練模型、估計參數時採用的方法則利用 Baum-Welch 參數估計法，從已知狀態序列，根據轉移規則，推出每個音框所屬的最佳口腔狀態，並重複估測直到穩態為止。所使用的訓練語料共有 47244 個字，時間長度約為 3.5 個小時。下圖為 HMM model 的訓練流程。採用的訓練軟體為英國劍橋大學開發的 HMM Tool Kit (HTK) 【3】。

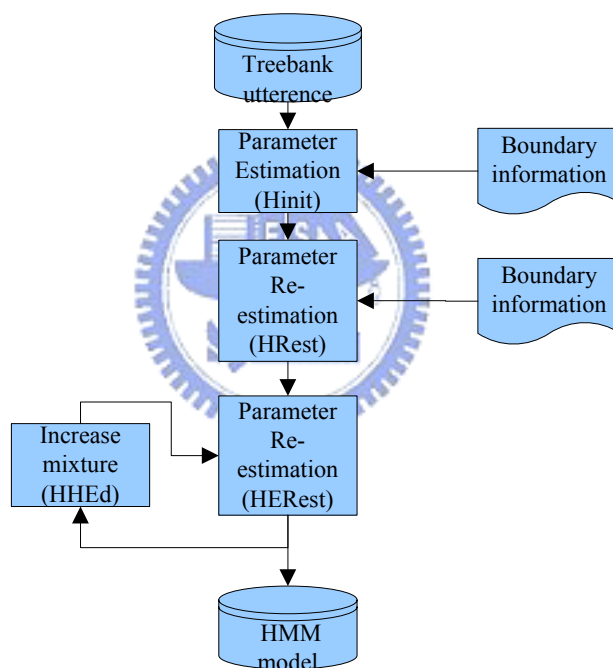


圖 2-1 已知切割模型之訓練圖

實驗結果

表 2-2 Inside test

Inside test				
Total syllable	Deletion	Substitution	Insertion	Accuracy
47244	0.04%	0.78%	0.03%	99.05%

表 2-3 outside test

Outside test				
Total syllable	Deletion	Substitution	Insertion	Accuracy
4948	0.32%	9.08%	0.42%	90.18%

實驗分析

由於是單一語者的辨認，在 inside test 的結果，syllable 的辨識率很高，超過了 99%，而 outside test 表現也不差，辨識率也超過了 90%。

2.2 Word-based bi-gram language model 建立

語言模型，可區分為兩種，一種是根據語言的文法、詞性，訂定文章出現一定符合規則之語言模型 (Rule-based LM)；另一種則是藉由處理大量的文字資料，利用統計的方式，計算詞和詞之間的聯結規則而建立的語言模型 (Statistic-Based LM)



2.2.1 n-grams 語言模型

假設有一個句子，其構成單元為詞，在句子中有 m 個詞「 w_1, w_2, \dots, w_m 」，其中「 w_i 」表示句子中的第 i 個詞。這句子的發生的機率，如下式，

$$\begin{aligned}
 P(w_1, w_2, \dots, w_m) &= \prod_{i=1}^m P(w_i | w_1, w_2, \dots, w_{i-1}) \\
 &\approx \prod_{i=1}^m P(w_i | w_{i-n+1}, \dots, w_{i-1}) \quad (2.1) \\
 &\approx \prod_{i=1}^m P(w_i | w_{i-1}) \quad (\text{bi-gram}) \\
 P(w_i | w_{i-1}) &= \frac{\text{Count}(w_{i-1}, w_i)}{\text{Count}(w_{i-1})}
 \end{aligned}$$

$\text{Count}(\cdot)$ 表示詞串出現次數

2.2.2 機率的 smoothing

如果分子在 $Count(\cdot)$ 值為 0 時，則 bi-gram 機率會等於零，表示在訓練語料中並未出現，但是這並不是代表測試語料不會出現，因此這種情況下機率的給定是不合理的，而在消息理論(Information Theorem)上來看機率 0 會使得資訊量無窮大，造成錯誤的估計。此外，當 $Count(\cdot)$ 的值很小時，所計算出的 n-gram 機率也是不準確。所以必須對計算出的機率做 smoothing 的動作【5】，使所有的 n-gram 機率均能被良好的估計。一般常見的 smoothing 方式如下：

$$P(w_i | w_{i-n+1}, \dots, w_{i-1}) = \begin{cases} a(w_{i-n+1}, \dots, w_{i-1})P(w_i | w_{i-n+1}, \dots, w_{i-1}) & Count(w_{i-n+1}, \dots, w_i) = 0 \\ d_a \frac{Count(w_{i-n+1}, \dots, w_i)}{Count(w_{i-n+1}, \dots, w_{i-1})} & 1 \leq Count(w_{i-n+1}, \dots, w_i) \leq k \\ \frac{Count(w_{i-n+1}, \dots, w_i)}{Count(w_{i-n+1}, \dots, w_{i-1})} & Count(w_{i-n+1}, \dots, w_i) > k \end{cases} \quad (2.2)$$

其中， $a(w_{i-n+1}, \dots, w_{i-1})$ 為 back-off 係數，當計算出次數為 0 時，則利用 (n-1)-gram，乘上 back-off 係數，來表示出現次數為 0 的機率。並分配給它一個適當的機率值。 $a(w_{i-n+1}, \dots, w_{i-1})$ 的選定，還會經過 normalization，令其滿足：

$$\sum_{w \in V} P(w_i = w | w_{i-n+1}, \dots, w_{i-1}) = 1 \quad (2.3)$$

若是 $Count(\cdot)$ 的值很小，造成機率預測不準確時，解決方式則是當詞串次數小於 k 次時，則會乘上一個根據 Good-Turning discounting 所計算出來的值 d_a (Discount Coefficient Factor)，減低其機率，並將扣除的機率分給未出現的 n-gram 機率使用。

2.2.3 語言模型建立流程

利用大量的文字資料訓練出一個涵蓋範圍廣泛，適用於各個領域的語言模型，基於此種模型的普遍性，稱為「General LM」。一個好的語言模型，它所需要的條件，是必須擁有大量的文字資料庫，在此建立的語言模型共有兩個，分別介紹如下：

第一個是光華雜誌，其內容是一般雜誌的文章，蒐集範圍為 1976 年到 2000 年之間，第二個是 NTCIR，是建立資訊檢索系統的標竿測試集，其內容是各個不同學科領域的文章構成。下表則是兩大資料庫的 word 及 character 統計表：

表 2-4 Database 統計

訓練語料	詞數 (Word)	字數 (Character)
光華雜誌	9,950,311	16,228,258
NTCIR	101,618,417	169,186,222
合計	111,568,728	185,414,480

藉由輸入大量的文字資料，統計出各種詞串在文章中累積出現的次數，接者利用(2.2)式和(2.3)式的 smoothing 方法，計算出建立語言模型所需的 n-gram 機率。在此我們建立出了 unigram 及 bigram 機率。但是要把語言模型及聲學模型在辨識系統中共同使用，我們需將語言模型轉換成 word-net 的形式。其描述著詞與詞之間的串接機率。訓練語言模型所使用的軟體為 SRI Language Modeling Toolkit 【4】。

整體語言模型建立流程可參考下方流程圖：

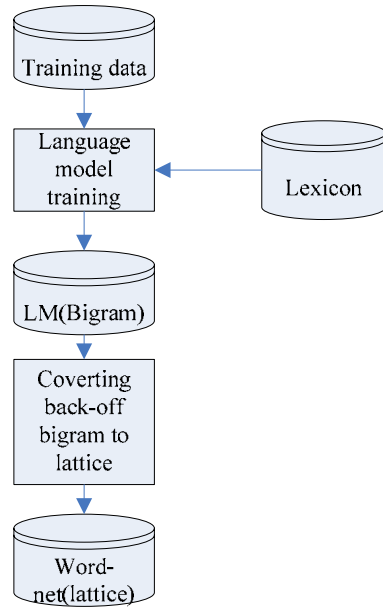


圖 2-2 Bigram 語言模型建立流程圖 【6】

2.2.4 Perplexity 計算

將光華雜誌和 NTCIR 兩個 database 當做訓練語料，平衡語料庫做測試語料。使用五萬兩千詞、五萬四千詞和六萬詞這三個詞典個別訓練出三種不同的語言模型。語言模型的好壞我們可以由 perplexity(PP)來測量，定義如下：

$$PP = 2^{\hat{H}}$$

$$\hat{H} = -\frac{1}{m} \log_2 P(w_1, w_2, \dots, w_m) \quad (2.4)$$

上式是一個句子之內容由 m 個詞所組成的例子，其中並對於每個新詞提供的平均資訊量，entropy (H)，經過了 ergodic 的假設與適當的化簡，最後以上式來對 H 做近似。計算 log probability 是以 10 為底，因此數學式修改如下：

$$PP = 10^{\hat{H}}$$

$$\hat{H} = -\frac{1}{m} \log_{10} P(w_1, w_2, \dots, w_m) \quad (2.5)$$

其中

$$P(w_1, w_2, \dots, w_m) \approx \prod_{i=1}^m P(w_i | w_{i-1})$$

$$\begin{aligned}
\log_{10} P(w_1, w_2, \dots, w_m) &= \log_{10} \left(\prod_{i=1}^m P(w_i | w_{i-1}) \right) \\
&= \log_{10} P(w_1) + \log_{10} P(w_2 | w_1) + \dots + \log_{10} P(w_m | w_{m-1}) \\
&= \sum_{i=1}^m \log_{10} P(w_i | w_{i-1})
\end{aligned}$$

語言模型測試

因為 treebank 的語料是出自於平衡語料庫，所以我們將平衡語料庫視為測試語料，下表為平衡語料庫的統計表及所算出的 Perplexity。

表 2-5 測試語料 Database 統計

測試語料	詞數 (Word)	字數 (Character)
平衡語料庫	4,890,875	7,926,700

表 2-6 測試語料 Perplexity

LM	Perplexity
Word-based(60k)	626.643

2.2.5 實驗：加入語言模型後辨識效能

實驗中使用到的語言模型是使用兩大語料庫所建立的 bigram LM，配合聲學模型進行辨識，為了加快 Viterbi search 的速度，使用 beam search。因為語言模型的使用，辨識結果的基本單元將不再是音節，辨識器輸出單元將以詞為主。因此我們可以計算詞、字元 (character) 和 syllable 三種不同的辨識率。

實驗結果

加入 word-based language model，下表為 word 和 Character 及 syllable 三者的辨識率。

表 2-7 word 辨識率

Total word	Deletion	Substitution	Insertion	Accuracy
2975	3.22%	22.72%	3.49%	70.55%

表 2-8 character 辨識率

Total character	Deletion	Substitution	Insertion	Accuracy
4948	0.51%	18.09%	0.30%	81.10%

表 2-9 syllable 辨識率

Total syllable	Deletion	Substitution	Insertion	Accuracy
4948	0.44%	6.79%	0.30%	92.52%

2.3 實驗分析

在此將比較之前只用聲學模型及同時使用聲學模型及語言模型兩者的音節辨識率，比較圖如下圖：

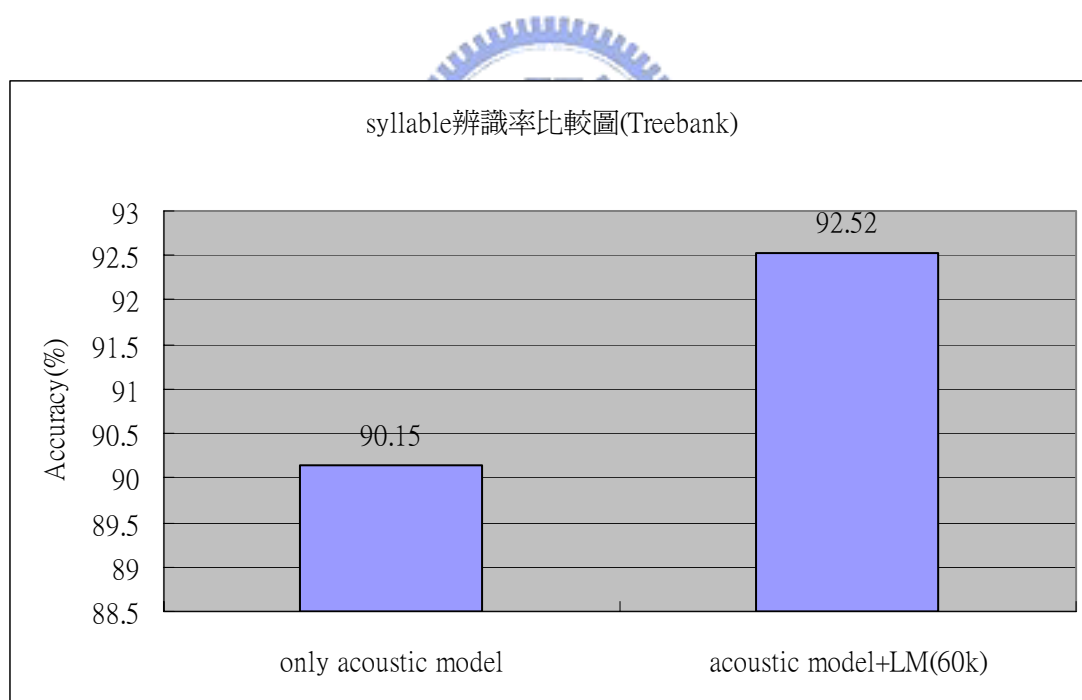


圖 2-3 syllable 比較圖

由上圖得知，當辨識器中加入語言模型之後，syllable 的辨識率能夠有顯著的提昇，這是因為人講話時大都會符合文法規則，因此若是在辨識系統中加入文法規則，則能夠限制、修正搜尋的範圍，則可以產生叫符合文法規則的辨識結果

而增加辨識的正確率。

若是和 multi-speaker 做比較，我們拿 TCC300 語料庫中由交通大學所錄製而成的音檔，計算出 syllable 的辨識率。如下表所：

表 2-10 syllable 辨識率

Total syllable	Deletion	Substitution	Insertion	Accuracy
13313	1.71%	26.70%	1.31%	70.14%

表 2-11 syllable 辨識率(加入 language model)

Total syllable	Deletion	Substitution	Insertion	Accuracy
13313	1.25%	15.32%	0.40%	83.03%

表 2-12 syllable error rate and error rate reduction

	only acoustic model	acoustic model and language model	error rate reduction
Treebank	9.85(%)	7.48(%)	24.6(%)
TCC300_NCTU	29.86(%)	16.97(%)	43.17(%)

單一語者在為加入語言模型之前，其 syllable error rate 為 9.85%，而 multi-speaker 則是 29.86%，在加入語言模型之後 syllable error rate，則分別是 7.48%和 16.97%，error rate reduction 則分別是 24.6 和 43.17%。我們發現到單一語者的 error rate reduction 小於 multi-speaker，其原因是，單一語者因為原本的 syllable 辨識率已經很高了，即使加入語言模型，也只有一定程度的改進辨識率。因此在之後的研究，syllable 的辨識率也許只能有少許的改進。

第三章 詞典的精簡及 Out of Vocabulary (OOV) 探討

目前所使用的六萬詞詞典，在訓練語言模型時，雖然能夠使包含率超過 99.6%，但是詞典中某些詞是不常出現，除非是在某些特定的情況，否則在一般情況下，是不會說的。另外，若是把詞做刪減，可想而知的是，OOV rate 一定會上升。因此解決 OOV 的問題，對於辨識結果或許會有幫助。

3.1 精簡詞典內容

主要是用人工的方式，將六萬詞的詞典刪減，刪除的項目如下所示：

1. 刪除人名：包含外國名，中文名
2. 刪除地名：包含國名，台灣地名
3. 刪除路名
4. 刪除單位名
5. 刪除斷詞錯誤的詞
6. 刪除不常見的一字詞



經以上刪除的步驟，原本六萬詞詞典，刪減到剩下五萬四千詞。刪除人名、地名、路名及單位名，主要是因為其組合的可能性太多了，只就人名而言，像中文的百家姓，可以組合成上萬種的組合。我們辨識器主要的目的並不是要辨識出這些專有名詞，因此為了訓練一個 General 語言模型，將人名、地名及單位名刪除。而刪除不常見的一字詞，是因為這些一字詞在平常的情形下，要唸到這些一字詞的機率是很小的，所以新的詞典將不會包含它們。下圖為新詞典及舊詞典各個詞長的分佈圖：

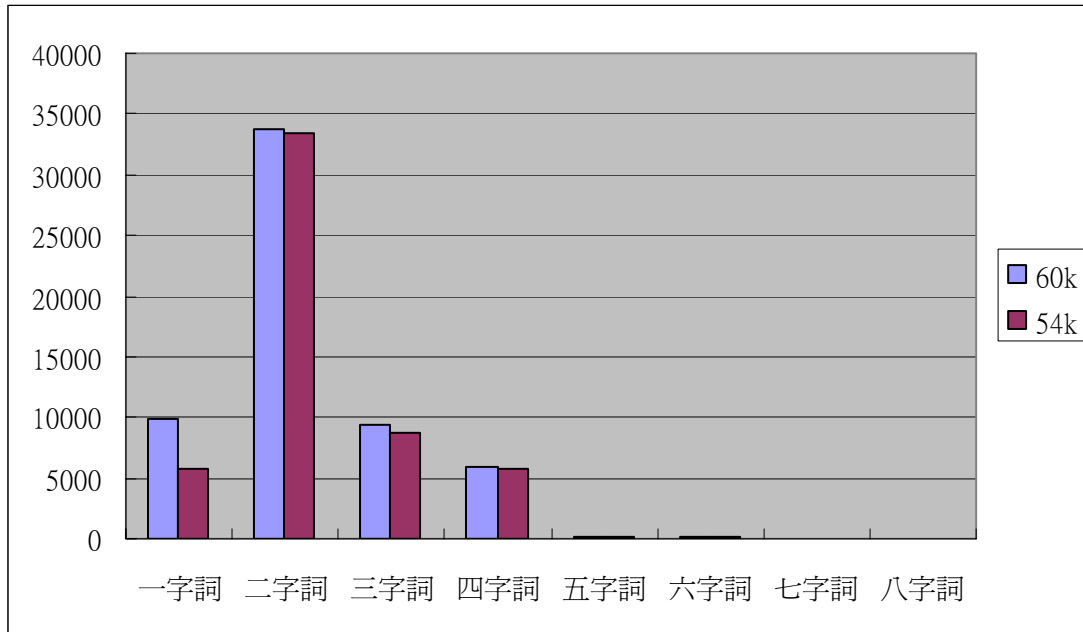


圖 3-1 詞長分佈圖

表 3-1 詞長比例表

詞長	1	2	3	4	5	6	7	8
百分比 (60k)	16.60%	56.97%	15.88%	9.93%	0.39%	0.21%	0.06%	0.04%
百分比 (54k)	10.75%	61.85%	16.09%	10.54%	0.43%	0.23%	0.06%	0.04%

3.2 詞典的再增減

刪減內容

再次刪減不常見的一字詞。以中研院平衡語料庫為參考資料，將平衡語料庫中未出現的一字詞，從詞典中刪除。

增加內容

之前刪減的地名，其中的某些地名，出現的機率實在太高了，所以新的詞典將會重新納入常出現的地名，例如：歐洲，台灣縣市名……等。下圖為三個詞典的詞長分佈圖：

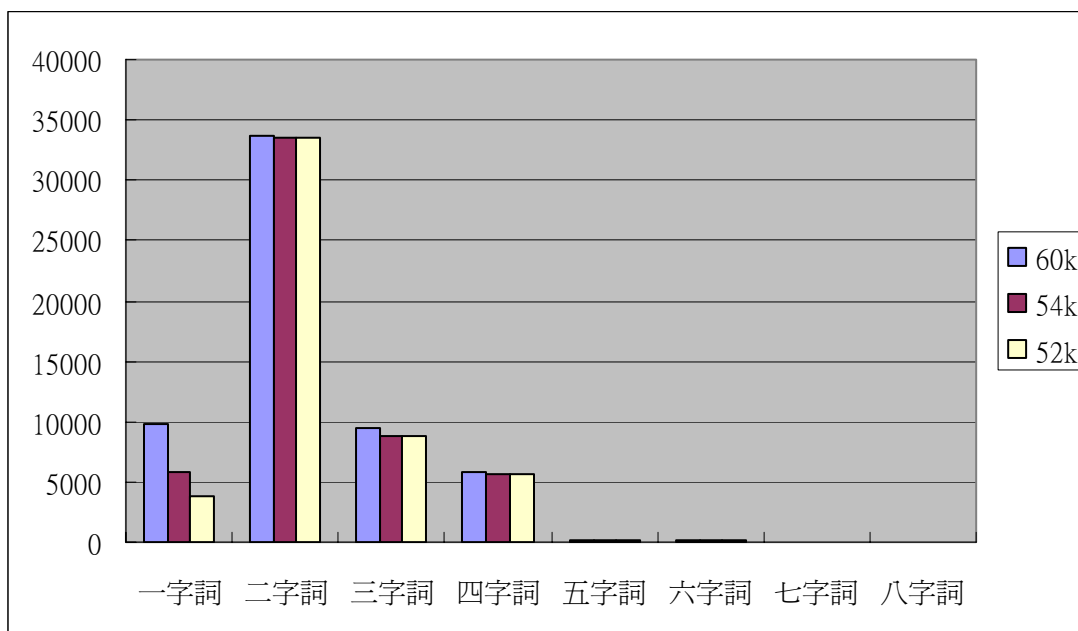


圖 3-2 詞長分佈圖

表 3-2 詞長比例表

詞長	1	2	3	4	5	6	7	8
百分比 (54k)	10.75%	61.85%	16.09%	10.54%	0.43%	0.23%	0.06%	0.04%
百分比 (52k)	7.18%	64.30%	16.78%	10.95%	0.44%	0.24%	0.07%	0.04%

3.3 OOV 探討

六萬詞詞典，由下表中可以看出，使用六萬詞詞典來建立語言模型而產生的 OOV 的詞數有 68 個，OOV rate 佔整個語言模型的訓練語料只有 0.398%，表示六萬詞詞典的 cover rate 超過 99.6%。但刪減過後的五萬四千詞詞典，OOV 詞數上升到 4024 個，OOV rate 從原本的不到 0.4% 上升到超過 1.5%，使得刪減過後的詞典的覆蓋率從超過 99.6% 降到 98.2% 左右。如果是用五萬兩千詞詞典來計算 OOV-rate，則是高達 2.875%，接近 3%。計算 OOV rate 參考式(3.1)。

表 3-3 OOV 統計表

詞典總數	OOV 詞數	OOV rate
60k	68	0.398%
54k	4024	1.800%
52k	5990	2.875%

OOV rate 計算：

$$\text{OOV rate} = (\text{Total OOV words}) / (\text{Total words in database}) \quad (3.1)$$

$$60k \text{ OOV rate} = 432060 / 111568728 = 3.87e-3 \quad \rightarrow 0.387\%$$

$$54k \text{ OOV rate} = 2024173 / 111568728 = 0.018 \quad \rightarrow 1.800\%$$

$$52k \text{ OOV rate} = 3208196 / 111568728 = 0.02875 \quad \rightarrow 2.875\%$$

接著利用六萬詞、五萬四千詞及五萬兩千詞這三種詞典，建立三種 Language model，並藉由辨識的結果，來觀察 OOV 對辨識會產生多大的影響。下表格為 word、character、syllable 三種辨識的結果。

表 3-4 word 辨識率

Outside test : Total 2975 words				
	Deletion	Substitution	Insertion	Accuracy
60k	3.22%	22.72%	3.49%	70.55%
54k	2.68	24.40%	5.95%	66.96%
52k	2.89%	24.87%	5.82%	66.42%

表 3-5 character 辨識率

Outside test : Total 4948 characters				
	Deletion	Substitution	Insertion	Accuracy
60k	0.51%	18.09%	0.30%	81.10%
54k	0.36%	19.52%	0.38%	79.73%
52k	0.36%	19.80%	0.38%	79.45%

表 3-6 syllable 辨識率

Outside test : Total 4948 syllable				
	Deletion	Substitution	Insertion	Accuracy
60k	0.44%	6.79%	0.30%	92.78%
54k	0.30%	6.70%	0.32%	92.66%
52k	0.30	6.79%	0.32	92.58%

實驗分析

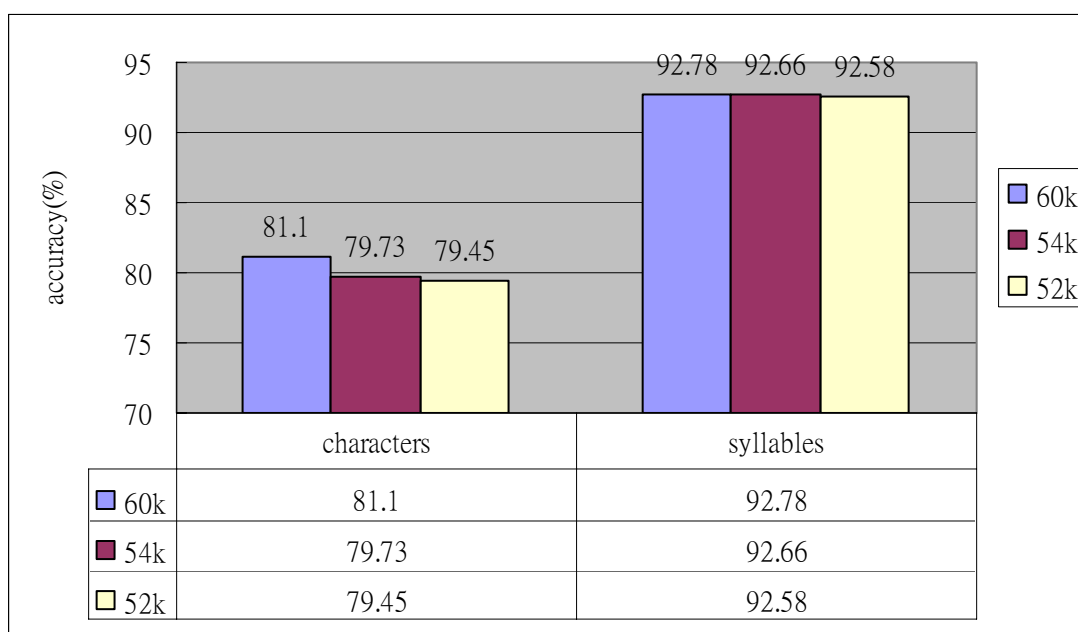


圖 3-3 辨識率比較圖

從以上圖表中，發現到當 OOV rate 從原本的 0.398% 上升到 1.80%，character 辨識率往下降了將近 2%，syllable 的辨識率也是下降了少許，由此看出 OOV rate 的多寡，會對辨識率造成影響。

3.4 解決 OOV

觀察那些 OOV 的詞，它們可以看成是由一些短詞組合而成，一個三字詞有可能是由一個二字詞和一個一字詞組合而成，一個四字詞有可能是由兩個二字詞組合而成【7】。

因此若是將 OOV 的詞做拆解，例：環保局，則拆解成（環保 局），三字詞被拆解成二字詞加一字詞的組合。經由上述分解的步驟，OOV rate 如下表，與之前 OOV 詞數相比，從 4024 個減少到 2838 個，OOV rate 也從之前的 1.8% 下降到了 0.43%，五萬兩千詞的 OOV rate，也從 2.875% 下降到了 2.04%，OOV 的詞只剩下一字詞。如下表所示。

表 3-7 OOV 統計表

	OOV 詞數	OOV rate
60k	68	0.398%
54k	2838	0.430%
52k	4807	2.04%

使用上述的斷詞法，產生新的訓練語料，接著重新訓練 Language model，並加入辨識器中。Perplexity 和新的 words 辨識率如下表：

表 3-8 perplexity(52k)

LM	Perplexity
Word-based(52k)	599.556

表 3-9 perplexity(54k)

LM	Perplexity
Word-based(54k)	617.512

表 3-10 word 辨識率

Outside test : Total 2975 words				
	Deletion	Substitution	Insertion	Accuracy
60k	3.22%	22.72%	3.49%	70.55%
54k	3.19%	23.96%	5.48%	67.36%
52k	3.22%	24.63%	5.34%	66.79%

若是將測試語料根據上述方法重新斷詞，其 word 辨識率如下表。

表 3-11 word 辨識率

Outside test : Total 3034 words				
	Deletion	Substitution	Insertion	Accuracy
54k	3.30%	22.67%	3.59%	70.44%
52k	3.36%	23.50%	3.49%	69.64%

上表中 total word 從原本的 2975 個詞增加到 3034 個詞，主要是因為，例：環保局，拆解成（環保 | 局），三字詞被拆解成二字詞加一字詞的組合。因此 word 的總數會增加。接著繼續觀察 characters 及 syllables 的辨識率，如下表所示：

表 3-12 characters 辨識率

Outside test : Total 4948 characters				
	Deletion	Substitution	Insertion	Accuracy
60k	0.51%	18.09%	0.30%	81.10%
54k	0.32%	18.53%	0.36%	80.78%
52k	0.32%	18.88%	0.38%	80.42%

表 3-13 syllable 辨識率

Outside test : Total 4948 syllable				
	Deletion	Substitution	Insertion	Accuracy
60k	0.44%	6.79%	0.30%	92.78%
54k	0.26%	6.75%	0.31%	92.68%
52k	0.24%	6.81%	0.40%	92.60%

比較 word 的辨認率，辨識率已從原本的 66.96% 上升到 67.13%，辨識率有了些許的改進，但是 insertion error 和 deletion error 差了 2% 以上，若是對測試語料重新斷詞，將長詞變成短詞的組合，insertion error 和 deletion error 則只差 0.2% 以下。

在此，以 characters 為觀察的準則，如下圖所示：

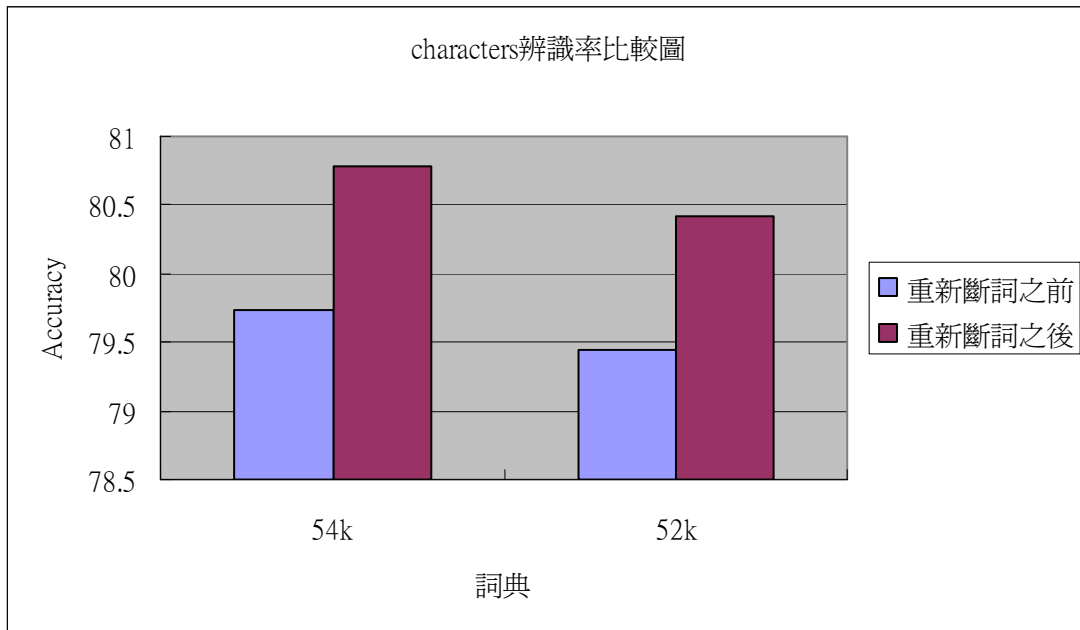


圖 3-4 character 辨識率比較圖

利用五萬四千詞及五萬兩千詞這兩個詞典所建立出的語言模型，在重新斷詞之前 characters 辨識率皆不到 80%，重新斷詞之後，辨識率都能夠有所改進，大約都提昇了 1% 左右。

之前詞的拆解動作，如果是人名，例：馬英九，被拆解成(馬 | 英 | 九)三個一字詞。我們嘗試不將人名做拆解，可以預期的是 unknown word 會增加，OOV rate 會上升。如下表所示。

表 3-14 OOV 統計表

	OOV 詞數	OOV rate
54k(人名有斷詞)	2838	0.430%
54k(人名未斷詞)	3335	0.820%

接著重新訓練 Language model，並加入辨識器中。下表為 characters 和 syllable 辨識率的結果。

表 3-15 characters 辨識率

Outside test : Total 4948 characters				
	Deletion	Substitution	Insertion	Accuracy
54k(人名斷開)	0.26%	18.53%	0.31%	80.78%
54k(人名未斷詞)	0.30%	19.26%	0.38%	80.05%

表 3-16 syllable 辨識率

Outside test : Total 4948 syllable				
	Deletion	Substitution	Insertion	Accuracy
54k(人名斷開)	0.24%	6.71%	0.31%	92.68%
54k(人名未斷詞)	0.25%	6.72%	0.33%	92.70%

實驗分析

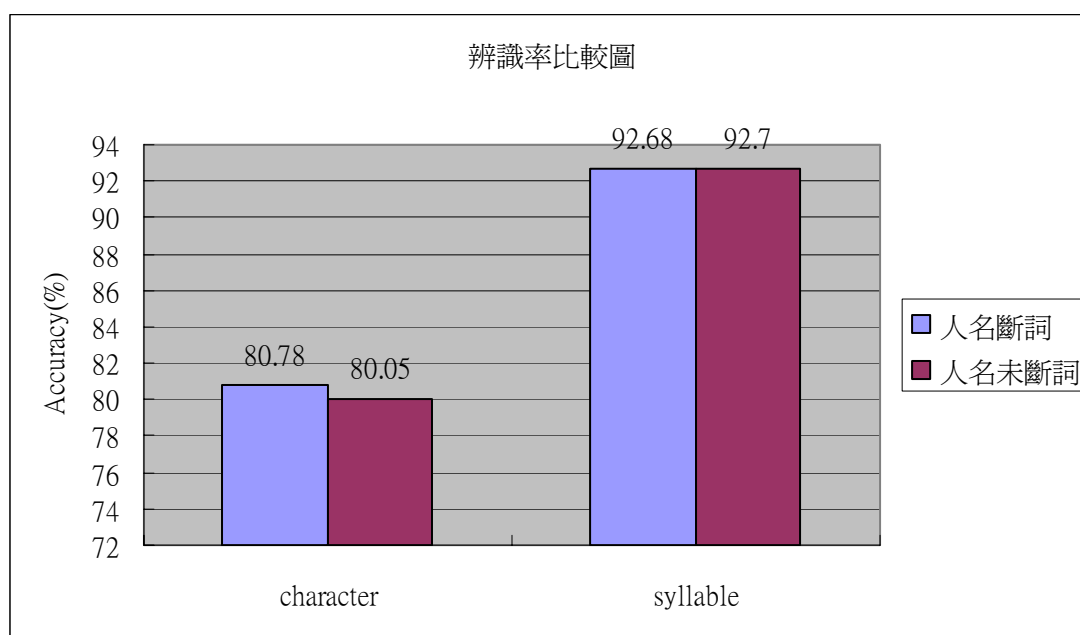


圖 3-5 character 和 syllable 辨識率比較圖

若未將人名做斷詞動作，則人名在訓練語言模型的時候，則會屬於 OOV，因此在辨識的時候。如果在測試語料，會出現人名，則有可能會辨識成同音義字的組合，若是人名做斷詞作用，例：馬英九，被拆解成(馬 | 英 | 九)三個一字詞，所以在訓練語言模型的時候會有(馬)接(英)和(英)接(九)的串接機率。因此在測試語料中，如果有出現(馬英九)這個姓名，就有可能能夠辨識出來。所以人名做斷

詞，其 characters 的辨識率才會比未做斷詞，表現稍微好一點。



第四章 POS-based Language Model 建立

前兩章節所用的語言模型，是以統計的方式求出的 word-based LM，但一個好的 word-based 語言模型，所需要的訓練語料是非常龐大的，如果我們手邊沒有龐大的語料庫，那我們可以考慮用只需少量語料就可以訓練出來的 POS-based language model。

4.1 POS Database 產生

使用五百萬詞平衡語料庫，在這語料庫中，不但有斷詞的結果，而且每個詞還保留著該詞所屬的詞類。經過以下的處理動作，我們可以得到以詞類所組合而成的句子。

Ex1 :

總辦事處(Nc) 秘書組(Nc) 主任(Na) 戴政(Nb) 先生(Na) 請辭(VA) 獲准(VF)

↓

(Nc) (Nc) (Na) (Nb) (Na) (VA) (VF)

Ex2 :

陳省身(Nb) 院士(Na) 即為(VG) 其(Nep) 指導(VC) 教授(Na)

↓

(Nb) (Na) (VG) (Nep) (VC) (Na)

下表為做完 POS 抽取動作 total POS 統計表，發現到總數少於平衡語料庫的總詞數，原因是在於只要在文章中 POS 標示為(FW)的詞，其所屬的句子，內容皆刪除。

表 4-1 POS 統計表

	Total POS
平衡語料庫	4,796,163

4.2 POS-based Language Model

POS-based LM，其實算是 class-based LM 的一種，它也是將 word 做分類，只是分類的依據，則是根據語言學中每個詞的 POS 資訊，將詞分到所屬的 POS class 底下，因此我們可以利用 class-based LM 的訓練方式，訓練出 POS-based 的語言模型。

4.2.1 Nonoverlap:

建立 POS-based LM，如果限定了一個 word 只能存在於一個 POS 之中，不能有 overlap 的情形產生。計算 word sequence 的數學式如下【8】：

$$\begin{aligned} P(W) &= \prod_i P_{class}(w_i | w_{i-1}) \\ &= \prod_i P(w_i | c_i)P(c_i | c_{i-1}) \end{aligned} \quad (4.1)$$

其中

$$P(w_i | c_i) = \frac{\text{Count}(\text{word } i \text{ belong pos } i)}{\text{Count}(\text{pos } i)}$$

$P(c_i | c_{i-1})$: POS bi-gram

W : word sequence

c_i : pos i

4.2.2 Overlap

由於一個詞的 POS，可以有多數個 POS，所以每個 word 有可能存在於不同的 POS class 底下，而產生 overlap。尤其是短詞需要注意，短詞的 POS 並不會只有一個。可以參照下表，詞典中一字詞和二字詞的 POS 種類表。

表 4-2 詞典一字詞 POS overlap 統計表

出現種類	一	二	三	四	五	六	七
比例	55.66%	23.22%	10.1%	5.73%	2.84%	1.18%	0.06%

表 4-3 詞典二字詞 POS overlap 統計表

出現種類	一	二	三	四
比例	94.81%	4.72%	0.38%	0.06%

因此，計算 word sequence 機率的數學式要有所變動，數學式如下【8】：

$$P(W) = \sum_{c_1, c_2, \dots, c_n} \prod_i P(w_i | c_i) P(c_i | c_{i-1}) \quad (4.2)$$

其中

$$P(w_i | c_i) = \frac{\text{Count}(\text{word } i \text{ belong pos } c_i)}{\text{Count}(\text{pos } c_i)}$$

$P(c_i | c_{i-1})$: POS bi-gram

W : word sequence

c_i : pos of word i

$\text{Count}(\cdot)$: **count in the training text**。



4.2.3 建立 POS-based Language Model

在建立POS-based LM之前，我們必須先統計好(4.2)式中 $P(w_i | c_i)$ 的機率。 $P(w_i | c_i)$ 機率的計算，是根據在平衡語料庫出現的次數為參考資料，而統計出來的。 $\text{Count}(\text{pos } i)$ 為詞性 $\text{pos } c_i$ 在平衡語料庫中出現的次數。 $\text{Count}(\text{word } i \text{ belong pos } c_i)$ 則是word i 其詞性是屬於 $\text{pos } c_i$ ，在平衡語料庫中出現的次數。 $P(c_i | c_{i-1})$ 則是POS的bigram機率，建立方法如同建立word-based LM。訓練POS-based LM所使用的軟體為SRI Language Modeling Toolkit【2】。下圖為POS-based LM的建立流程。

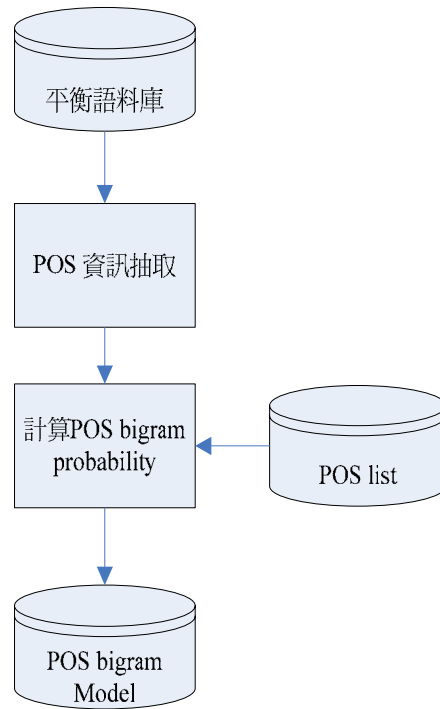


圖 4-1 POS bigram model 建立流程

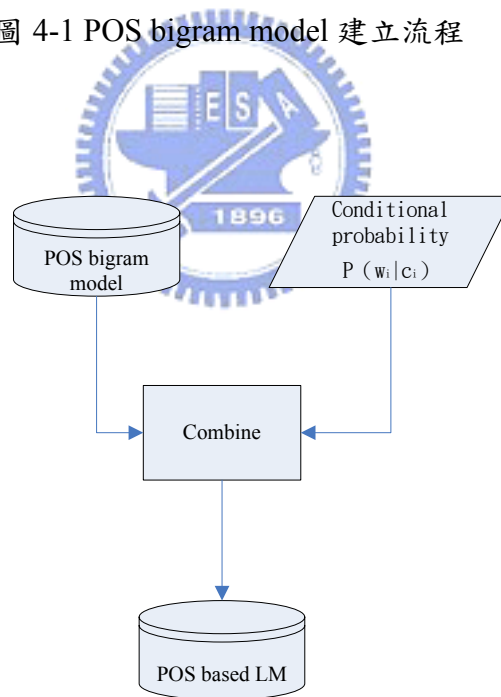


圖 4-2 POS-based LM 建立

實驗結果

表 4-4 word 辨識率

Outside test : Total 2975 words				
	Deletion	Substitution	Insertion	Accuracy
POS	5.21%	37.01%	7.33%	50.45%

表 4-5 character 辨識率

Outside test : Total 4948 characters				
	Deletion	Substitution	Insertion	Accuracy
POS	0.46%	28.40%	0.46%	70.68%

表 4-6 syllable 辨識率

Outside test : Total 4948 syllables				
	Deletion	Substitution	Insertion	Accuracy
POS	0.32%	8.12%	0.32%	91.23%

實驗分析

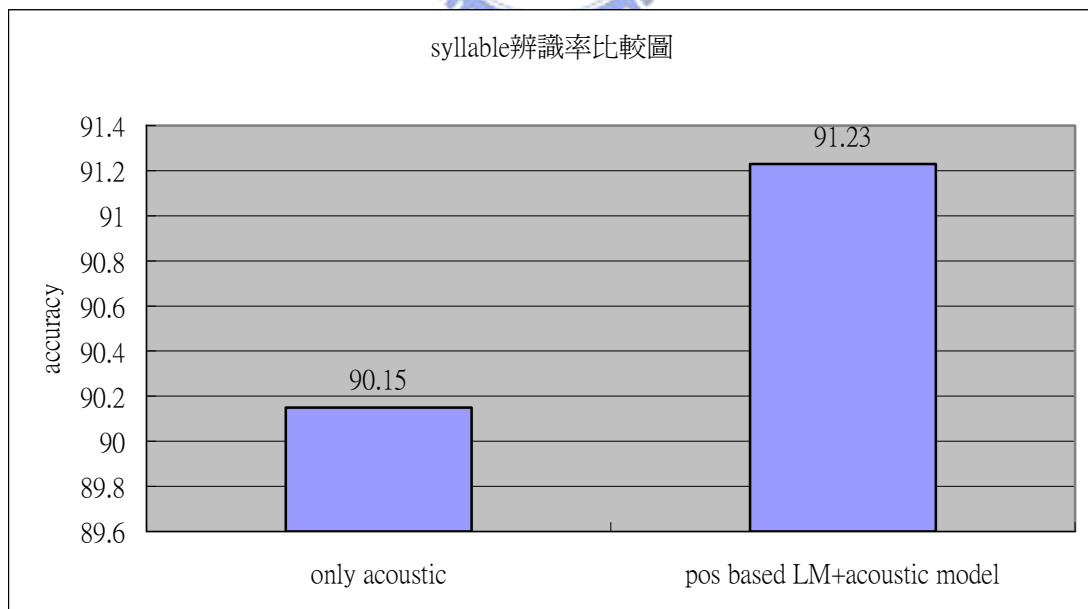
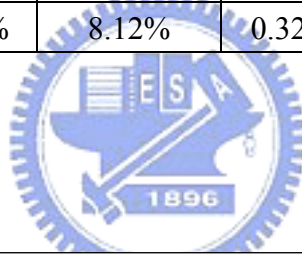


圖 4-3 syllable 辨識率比較圖

將 POS-based LM 加入辨識器中，從 syllable 的辨識率的比較圖，加入 POS-based LM，可以使 syllable 的辨識率提昇一個百分比以上。主要是因為 POS-based LM 可以在辨識系統種加入文法規則，來達到提高辨識率的效果。但以 characters 的角度來看，辨識率比使用 word-based LM 還要低。

4.3 詞長資訊

由上述的實驗中，發現到如果只用 POS 的 information 來做分類的依據，其 characters 的辨識的結果，不盡理想，因此加入詞長的 information，來拓展 POS class 的個數，舉例如下：

Ex：

總辦事處(Nc) 秘書組(Nc) 主任(Na) 戴政(Nb) 先生(Na) 請辭(VA) 獲准(VF)

↓

(Nc)_4 (Nc)_3 (Na)_2 (Nb)_2 (Na)_2 (VA)_2 (VF)_2



在抽取 POS information 時，也統計該詞的字數並加到 POS 之後，如上所示。從平衡語料庫中抽取出 POS 和詞長的資訊，可以用來訓練新的 POS bigram。因為增加了詞長的資訊，所以 POS 的個數增加，從原本 46 種 POS 拓展成 149 種。所以必須重新計算式 (4.2) 中的 $P(w_i | c_i)$ ，尤其是 $Count(pos c_i)$ 的出現次數將重新做計算。建立方式如同圖 4-1 和圖 4-2 建立 POS-based LM 的方法

實驗結果

表 4-7 character 辨識率

Outside test : Total 4948 characters				
	Deletion	Substitution	Insertion	Accuracy
POS	0.46%	28.09%	0.51%	70.94%

表 4-8 syllable 辨識率

Outside test : Total 4948 syllables				
	Deletion	Substitution	Insertion	Accuracy
POS	0.28%	8.04%	0.32%	91.35%

實驗分析

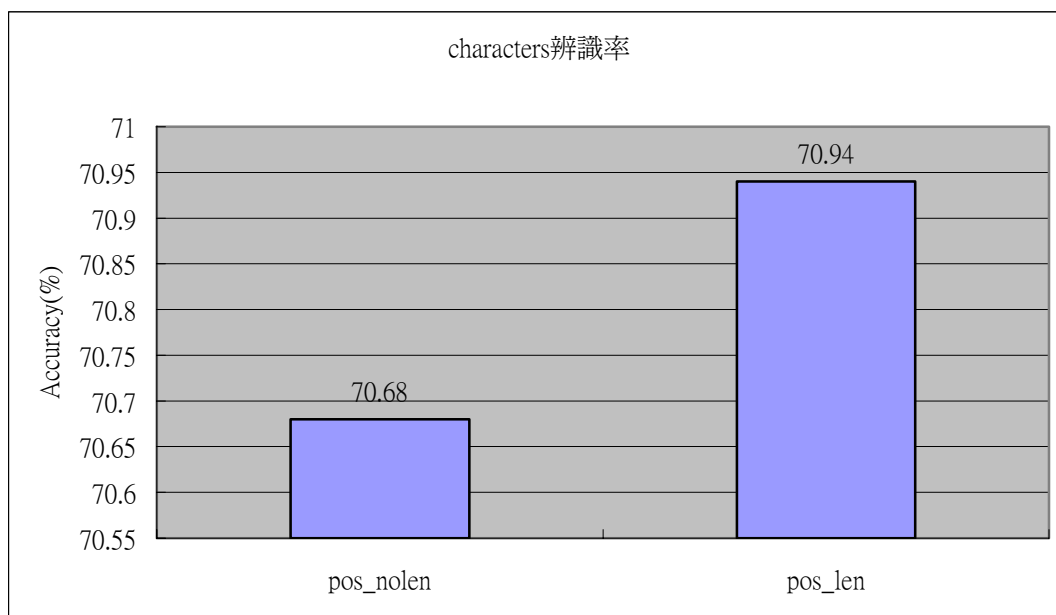


圖 4-4 character 辨識率比較圖

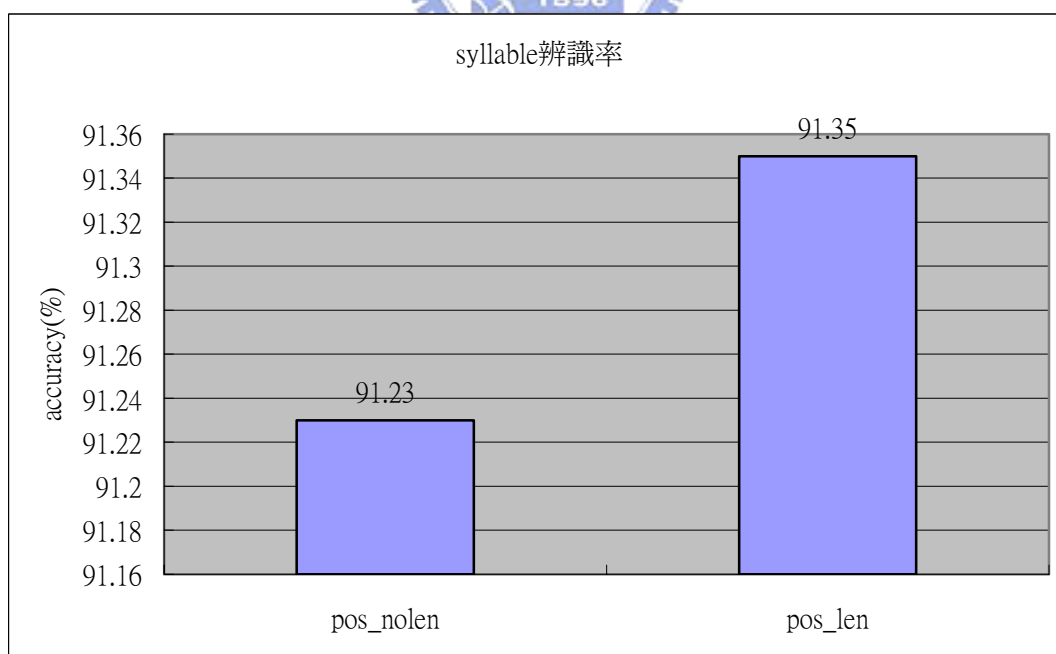


圖 4-5 syllable 辨識率比較圖

加入詞長 information，使得原本 pos 分類從 46 類，擴展到了 149 類，擴展

後的效果由上兩張 character 和 syllable 的比較圖，可以看出，character 的辨識率約提昇了 0.3%，syllable 的辨識率約提昇了 0.12%。對於 character 的辨識率，只能有少許的改進，這是因為加了詞長的資訊，只是分群更為細密，詞長的資訊，並沒有帶任何的語言學上的訊息，所以辨識率只提昇一點是可以預期的

4.4 POS and Word Mixture LM

從上一章節中我們發現到即時加入了詞長的資訊，character 的辨識率只能提昇少許。在這章節中，我們將訓練出一個 words 和 POS 混合的語言模型。有了 word 的資訊，預估會有較大的幫助。

4.4.1 Database 建立

在建立 Database 之前，我們先統計詞典中的每個詞在平衡語料庫中出現的次數並給予排名。接著重複上述抽取 POS 資訊的動作，一但碰到出現次數超過一千次以上的詞，則保留 word 的資訊，如下所示：

Ex1 :

總辦事處(Nc) 秘書組(Nc) 主任(Na) 戴政(Nb) 先生(Na) 請辭(VA) 獲准(VF)

↓

(Nc)_4 (Nc)_3 (Na)_2 (Nb)_2 先生 (VA)_2 (VF)_2

(先生) 這個二字詞，因為在平衡語料庫中出現次數超過一千次以上，所以在產生 database 的時候，原本是要儲存(Na)這個 POS，改為儲存(先生)這二字詞。另外，我們也加入詞長資訊，但是將詞長超過三以上的 POS class 合併，使 POS class 的個數減少，其原因是多字詞的詞性很相近。我們不需要特別的去照顧多字詞。由上一節加入詞長資訊的辨識結果中可以看到，辨識率只有改善少許，因此減少 POS class 的個數，並不會對辨識結果造成多大的影響。

4.4.2 訓練 POS And Word Mixture LM

參考圖 4-1 和圖 4-2 的流程圖，使用新的 database，訓練出 $P(c_i | c_{i-1})$ 的 bigram 機率。 $P(w_i | c_i)$ 的機率，要全部重新計算。而出現超過一千次以上的詞，其 $P(w_i | c_i)$ 等於 1，且這些詞將不包含在其他 POS class 底下。

實驗結果

表 4-9 character 辨識率

Outside test : Total 4948 characters				
	Deletion	Substitution	Insertion	Accuracy
POS	0.36%	25.52%	0.49%	73.63%

表 4-10 syllable 辨識率

Outside test : Total 4948 syllables				
	Deletion	Substitution	Insertion	Accuracy
POS	0.26%	7.76%	0.38%	91.59%

實驗分析

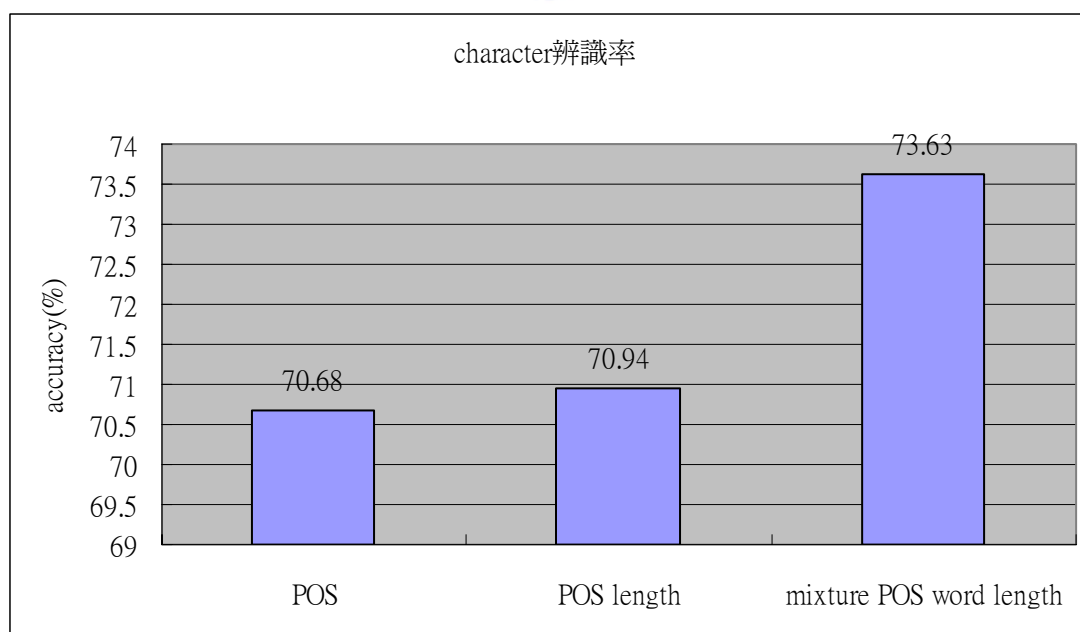


圖 4-6 character 辨識率比較圖

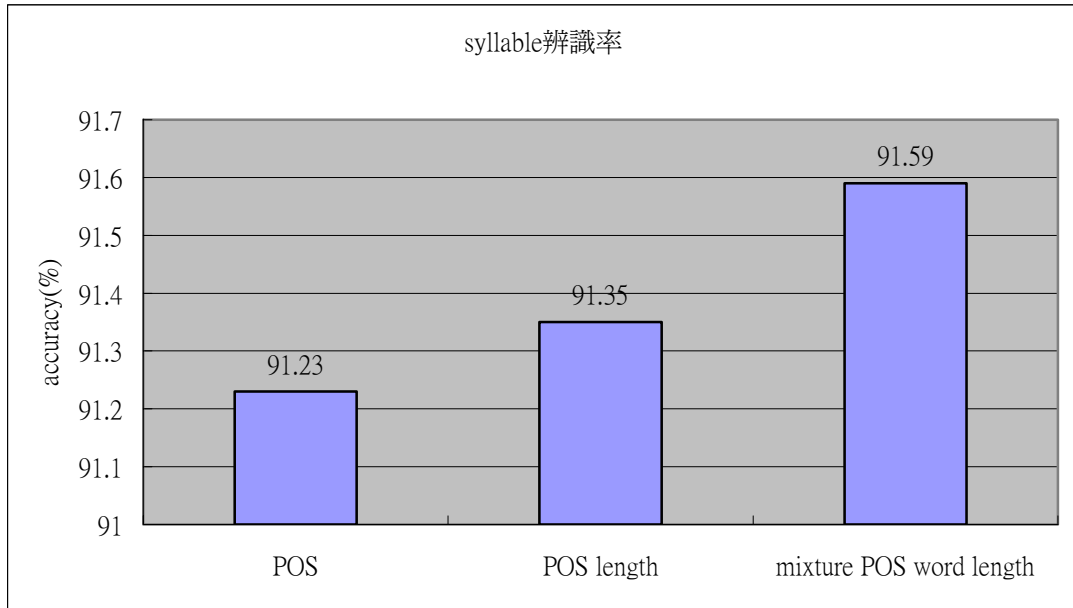


圖 4-7 syllable 辨識率比較圖

由圖 4-6，character 的辨識率在使用 mixture POS 和 word 的語言模型，比只使用 POS 資訊的語言模型，提昇了 3 個百分比左右，由此可見，混和 pos 和 word 所建立出來的語言模型，可以大幅的提昇辨識率。

從圖 4-7 中，發現到 syllable 的辨識率增加的幅度就沒有像 character 這麼多，但是當我們每次增加一個 information，syllable 的辨識率都能夠有些微的提昇。

第五章 Linear Interpolation Word-based And POS-based LM

POS-based LM 概括於從未看過的 word sequence 能夠表現的較 robustness，而 word-based LM 如果要有較好的 performance，需要大量的訓練語料，如果我們能有一個語言模型，能夠結合這兩種語言模型的優點，那對辨識會有很大的幫助。而最常用的結合各種語言模型的方法是 linear interpolation 【9】

5.1 Linear Interpolation

將 word-based 和 POS-based 兩種 language model 做 linear interpolation，數學式如下：

$$P_I(w_i | w_{i-1}) = \lambda_w P_w(w_i | w_{i-1}) + \lambda_c P_{pos}(w_i | w_{i-1}) \quad (5.1)$$

其中

$P_w(w_i | w_{i-1})$: word-based language model prediction probability

$P_{pos}(w_i | w_{i-1})$: POS-based language model prediction probability

$$P_{pos}(w_i | w_{i-1}) = P(w_i | pos(w_i)) \times P(pos(w_i) | pos(w_{i-1}))$$

5.2 產生 linear interpolation LM

word-based language model 是使用光華雜誌和 NTCIR 兩個 database 分別以六萬詞、五萬四千詞及五萬兩千詞詞典訓練而成，POS-based language model 則是之前利用第四章所訓練出來的語言模型。所使用的軟體為 SRI Language Modeling Toolkit 【4】，來將兩個語言模型做 linear interpolation，流程圖如下：

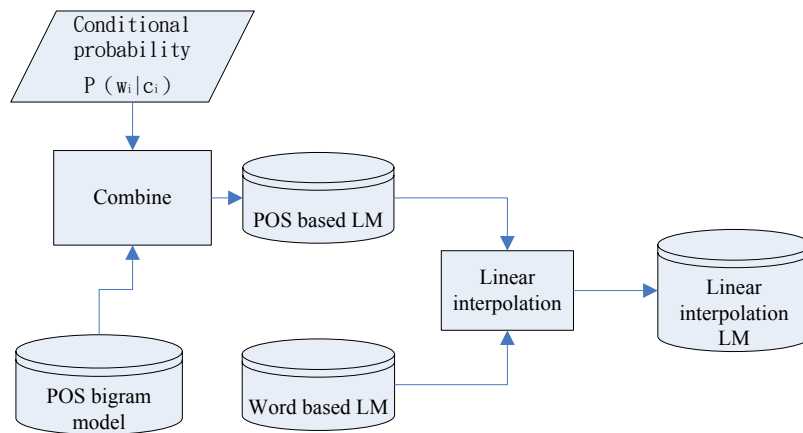


圖 5-1 建立 linear interpolation LM 流程圖

我們先用 perplexity 來測試結合兩個不同型態的 model 是否能夠使 perplexity 下降。測試的語料仍是五百萬詞平衡語料庫。

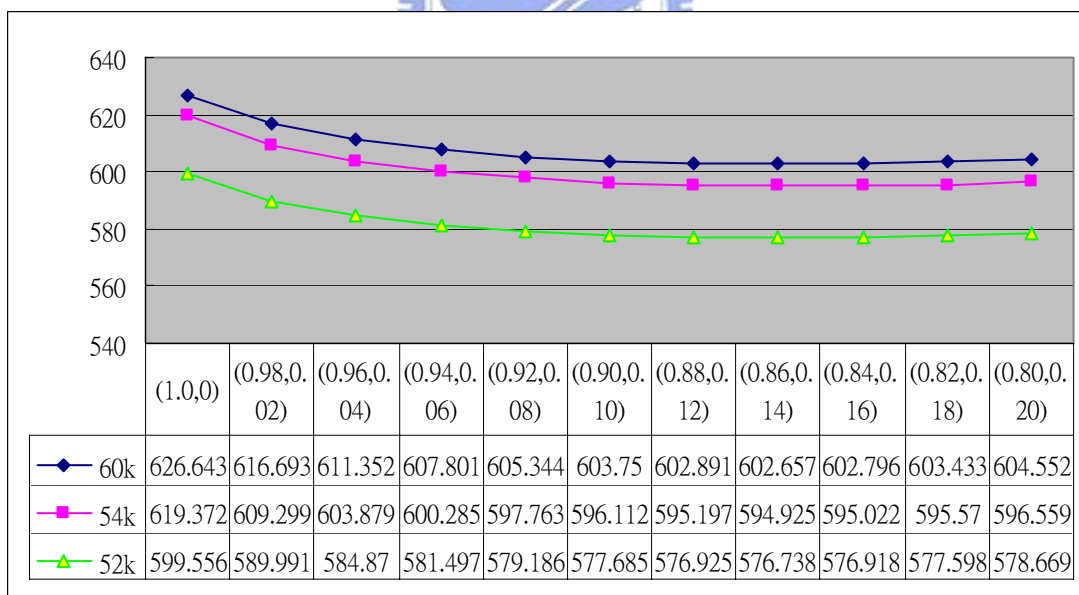


圖 5-2 Test-set word perplexity as a function of the interpolation coefficient λ_w, λ_c

表 5-1 perplexity 比較表

LM	Perplexity
Word-based(60k)	626.643
Linear-inter	602.657

$$(\lambda_w, \lambda_c) = (0.86, 0.14)$$

表 5-2 perplexity 比較表

LM	Perplexity
Word-based(54k)	619.372
Linear-inter	594.925

$$(\lambda_w, \lambda_c) = (0.86, 0.14)$$

表 5-3 perplexity 比較表

LM	Perplexity
Word-based(52k)	599.556
Linear-inter	576.738

$$(\lambda_w, \lambda_c) = (0.86, 0.14)$$

做 word-based 和 POS-based 兩個語言模型的 linear interpolation，發現到每個 word-based 的語言模型，perplexity 平均約下降 20 左右，這是因為 POS-based LM 的功用是用來微調 word-based LM 的參數，所以在做測試的時候，能夠稍微的調降 perplexity 數值。

實驗結果

加入辨識器中， (λ_w, λ_c) 參數的設定為(0.86,0.14)，下表為 characters 和 syllables 的辨識結果。

表 5-4 character 辨識率

Outside test : Total 4948 characters				
	Deletion	Substitution	Insertion	Accuracy
60k	0.34%	17.86%	0.38%	81.41%
54k	0.34%	18.14%	0.38%	81.12%
52k	0.34%	18.59%	0.38%	80.68%

表 5-5 syllable 辨識率

Outside test : Total 4948 syllable				
	Deletion	Substitution	Insertion	Accuracy
60k	0.28%	6.55%	0.32%	92.85%
54k	0.28%	6.61%	0.32%	92.78%
52k	0.28%	6.63%	0.32%	92.76%

5.3 實驗分析

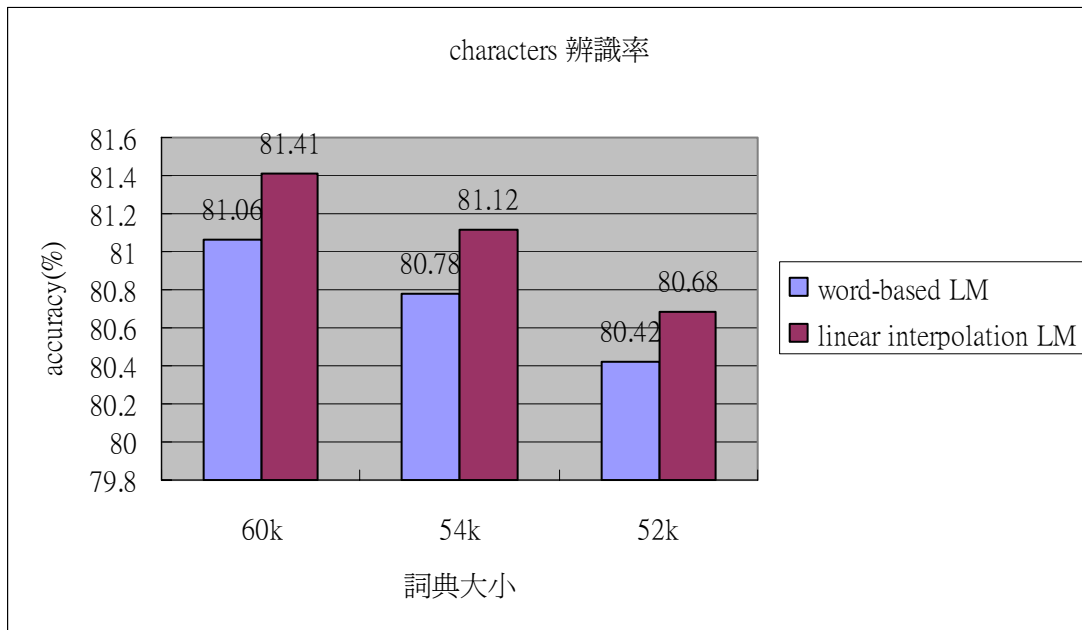


圖 5-3 character 辨識率比較圖

由圖表發現到，加入 linear interpolation LM 的辨認器，在 characters 的辨認率，能夠表現的比使用 word-based LM 還要好，因此雖然 POS-based LM 的 characters 的辨識率只有 70%左右，但是如果用來輔助 word-based LM，則能夠對辨識率的提昇有所貢獻。

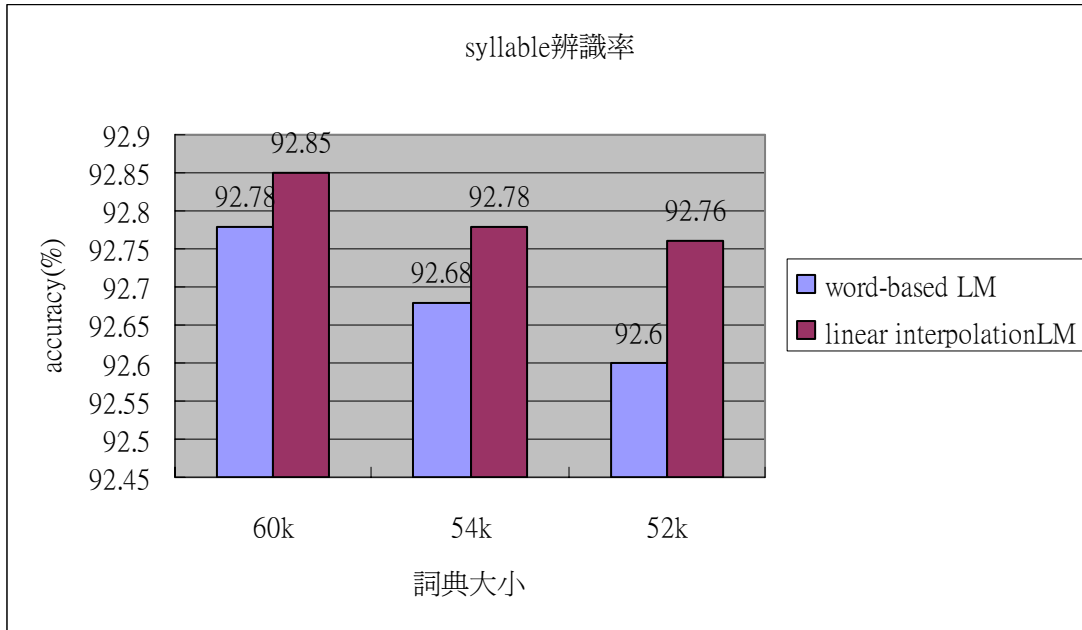


圖 5-4 syllable 辨識率比較圖

由上比較圖，syllable 的辨識率，使用 linear interpolation LM，皆能夠提高 syllable 的辨識率，只是提昇的幅度並不夠多。

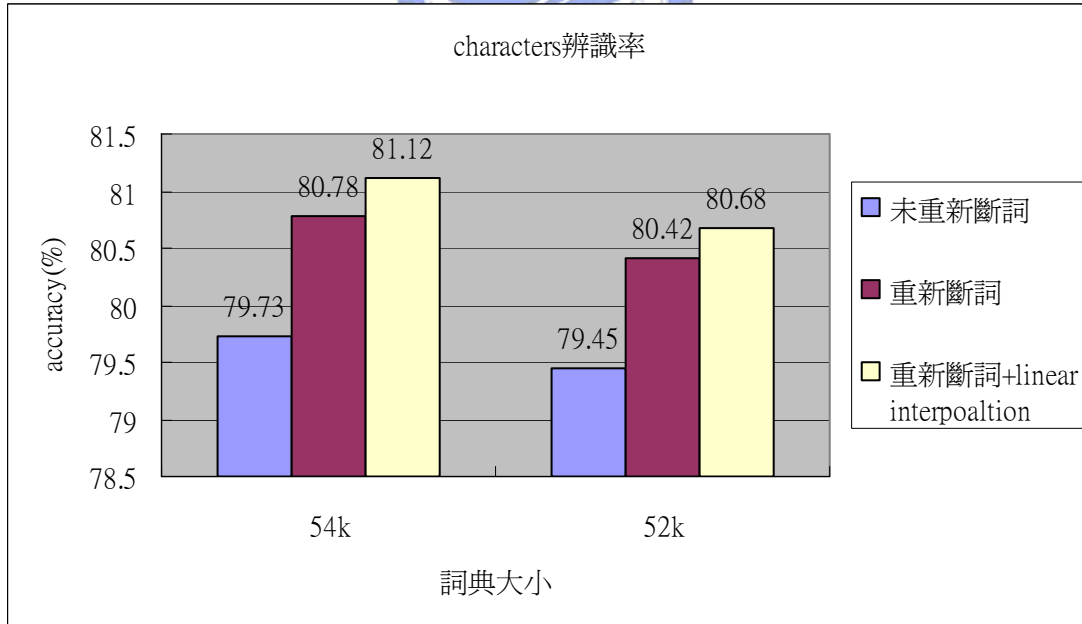


圖 5-5 character 辨識率比較圖

當詞典的詞總數變少之後，連帶著 OOV rate 的上升，而導致辨識率的下降，但將訓練語料做重新斷詞之後，character 辨識率約可上升將近 1%，若是重新斷

詞並且加上 word-based LM 和 POS-based LM 做 linear interpolation，則可使 characters 辨識率提昇約 1.5%。



第六章 結論與展望

6.1 結論

在本論文中，我們使用單一語者的 treebank 語料庫來進行語音辨識的相關研究，從基本系統的建立，詞典的刪減，OOV 的探討和處理，POS based LM 的建立以及改進及使用 linear interpolation 結合 word-based 和 POS-based LM 兩個語言模型的優點，有一個循序漸進的說明，在此我們將幾個重點分列如下：

- (1) Treebank 為單一語者的語料庫，因此即使在辨識系統中採取無文法規則 (free grammar)，其 syllable 的辨識率都能有很好的表現，但 character 的辨識率就無法保證，因此加入語言模型則可使辨識器更完善。所以語言模型是辨識基本要求。
- (2) 由於我們的辨識器主要不是要辨識人名、地名、路名和不常見的一字詞，因此要做刪減的動作。當我們刪減了詞典的內容，可以預見的是 OOV rate 會上升，因此我們將詞做拆解的動作，來解決 OOV 的問題。
- (3) 建立了 POS-based LM，並加入詞長的資訊來做辨識率的改進，但是改善的幅度不大，因此我們採用混和 POS 和 word 的語言模型，能夠大幅度的提昇辨識率。
- (4) 最後我們使用 linear interpolation 來結合 word-based 和 POS-based 這兩種不同類型的語言模型，來改善辨識率。

6.2 展望

國內外的許多基本系統，均已經採用前後文相關模型 (Context Dependent Model) 列為基本條件，但是實驗中加入語言模型，使得運算量變得相當龐大，辨識工作相當費時，因此就沒有嘗試前後文的相關實驗，若在未來的研究中，我

們希望可以把部份加入辨識系統。

POS-based 的語言模型雖然辨識成果不如預期，但是如果和 word-based LM 做結合，能夠稍微改善辨識結果，如果我們能夠在訓練語言模型時同時考慮 POS 和 word，相信這能夠有大幅度的提昇。



參考文獻

- 【1】 B.H.Juang and S.Furui,“Automatic recognition and understanding of spoken language—A first step towards natural human-machine communication,”in Proc IEEE,88,8,pp.1142-1165,2000
- 【2】 L.R.Rabiner and B.H.Juang,“Fundamental of speech Recognition,”New Jersey,Prentice-Hall,Inc.,1993
- 【3】 S.Young, G.Evermann, T.Hain, D.Kershaw, G,Moore, J,Odell,D.Ollan, D.Povey, V.Valtchev, P.Wooland,“*The HTK Book(for HTK version 3.4)*”
- 【4】 . Andreas Stolcke, “SRILM - An Extensible Language Modeling Toolkit”, in Proc.Intl. Conf. Spoken Language Processing, Denver, Colorado, September 2002
- 【5】 Slava M. Katz,“Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer,”IEEE Transactions on Acoustic,Speech and Signal Processing,Vol.ASSP-35,NO.3,MARCH 1987
- 【6】 張隆勳,“國語廣播新聞語音基本系統之建立”,國立交通大學電信工程學系碩士論文,民國九十四年七月
- 【7】 P.Geutner,“Using Morphology Towards Better Large-Vocabulary Speech Recognition Systems” in: Proc .Int. Conf. on Acoustics, Speech, and Signal Processing, Detroit, pp. 445-448 ,1995
- 【8】 Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, “Spoken Language Processing”
- 【9】 F.Jelinek,R.L.Mercer.“Interpolated Estimation of Markov Source Parameters from Sparse Data”.In Pattern Recognition in Practice, pp 381-397.North Holland,1980