國立臺灣師範大學
資訊工程研究所碩士論文

指導教授：陳柏琳 博士

語言模型訓練與調適技術於中文大詞彙連續語音辨識之初步研究

An Initial Study on Language Model Estimation and Adaptation Techniques for Mandarin Large Vocabulary Continuous Speech Recognition

研究生：蔡文鴻 撰

中華民國 九十四 年 七 月

摘　　要

在過去三十年間，統計式語言模型在各種與自然語言相關的應用上一直是一個重要的研究議題，它的功能是擷取自然語言中的各種資訊，諸如前後文資訊（contextual information）、語意資訊（semantic information）等，再利用這些資訊以機率量化來決定一個詞序列（word sequence）發生的可能性。例如，在語音辨識中，語言模型扮演的角色是要解決聲學混淆（acoustic confusion）的問題，將正確的辨識結果從有可能的候選詞序列中挑選出來。

近年來，語音辨識在我們生活中已有越來越多的應用，例如語音聽寫（voice dictation）、電話轉接（call routing）系統等等。但是語音辨識效能的好壞，通常會隨著辨識任務的詞彙或語意的不同，而受到嚴重的影響，於是誕生了語言模型調適的研究。語言模型調適是要利用辨識任務中固有的詞彙和語意資訊來彌補訓練語料與測試語料間的不一致性（mismatch）。

在本論文中，提出了原本應用在機率式資訊檢索上的主題混合模型法（topic mixture model, TMM）來動態的利用長距離的主題資訊，並且運用在語言模型調適上得到了不錯的效果。此外，本論文對最大熵值法（maximum entropy, ME）亦做了深入的研究，最大熵值法是一種將不同資訊來源（information sources）整合的方法，在此方法中，每一個資訊來源都會引發一群限制（constraints），限制合併後的語言模型要滿足所有的資訊。然而，這些限制的交集（intersection），是滿足所有資訊的機率分佈的集合，在這個集合中，擁有最大熵值（highest entropy）的機率分佈即為此方法的解。初步的實驗結果顯示以最大熵值法來合併一連詞、二連詞與三連詞所得到的語言模型，比用傳統最大相似度估測法（maximum likelihood）所訓練的語言模型，在中文廣播新聞轉寫上的字錯誤率（character error rate, CER）與語言模型複雜度（perplexity）都達到較好的效果。

Abstract

Statistical language modeling, which aims to capture the regularities in human natural language and quantify the acceptance of a given word sequence, has continuously been an important research issue in a wide variety of applications of natural language processing (NLP) over the past three decades. For example, in speech recognition, the principal role of the language models is to help resolve the acoustic confusion and thus separate the correct hypothesis from the competing ones.

In the recent past, there were quite many applications of speech recognition technology being developed, such as voice dictation and call routing systems, etc. However, speech recognition performance is often seriously affected by the varying lexical and semantic characteristics among different application tasks. Thus, there is always a need for language model adaptation, which has the goal to exploit the specific lexical and semantic information inherent in the recognition domain, so as to compensate the mismatch between training and testing conditions.

In this thesis, a topical mixture model (TMM) previously proposed for probabilistic information retrieval was investigated to dynamically explore the long-span latent topical information for language model adaptation. Moreover, we also studied the use of the Maximum Entropy (ME) principle for language modeling. ME is a principle for efficient combination of a variety of information sources. Under the ME criterion, each information source gives rise to a set of constraints that can be futher imposed on the resultant language model. The intersection of these constraints is the set of language model probability distributions which can satisfy all of these constraints. The probability distribution which has highest entropy is thus the solution of the ME principle. The preliminary experimental results show that the ME-based language modeling approach can achieve superior performance over the conventional Maximum Likelihood (ML) based approach in both character error rate and perplexity reductions on the Mandarin broadcast news transcription task.

誌　　謝

感謝我的爸媽以及弟弟，你們從小到大對我的支持讓我得以順利完成學業，取得碩士學位。在我這一段求學的路上真是辛苦你們了，若今後有任何小小的成就都是屬於你們的。

感謝指導教授陳柏琳博士在碩士班求學過程中的諄諄教誨，不論是作研究的態度、治學方法，甚至在為人處世上都給予學生莫大的影響與幫助。從剛進實驗室對語音的完全陌生，到完成這本論文，這都要感謝您不厭其煩的指導，謝謝老師。

感謝成功大學的簡仁忠教授、中研院的王新民博士，還有成功大學和中研院語音實驗室的諸位學長，在我碩一升碩二的暑假於中研院 meeting 時的指導，讓我得以對語音相關的知識更加完備。

感謝口試委員林順喜博士、劉昭麟博士與林伯慎博士對學生論文的指正，使得這本論文能夠更臻完全。

感謝士傑學長、人瑋、耀民及信維，和你們一起作研究總是那麼的令人愉快，每每都有意想不到的收穫；感謝志豪、惠銘和成韋，在研究之餘總是為實驗室帶來歡樂的氣氛；感謝炫盛、燦輝、士弘、怡婷、芳輝、家豪、庭瑋與鴻彬，你們的加入將成為語音實驗室的生力軍，希望大家都可以在這個領域中闖出一片天來。

感謝民生社區禮拜堂的弟兄姐妹，你們有聲無聲的禱告，我想　神都聽見了，願祂也一樣眷顧你們。

感謝我的女友凰婷，從我準備研究所考試到取得畢業證書，妳都一直在身旁鼓勵我、支持我，還記得妳在實驗室陪我熬夜寫程式、寫論文，真的是辛苦妳了。

感謝那些曾經幫助過我的老師、同學與朋友，謹將這本論文獻給你們。

文鴻　民國 94 年秋

目　　錄

圖目錄

表目錄

第1章 緒論

1.1 研究動機

記得小時候看過的電視影集「霹靂遊俠李麥克」，影片中的主角和他的搭檔「霹靂車」一起衝鋒陷陣，他們之間的溝通是透過人類最直接也最簡單的方式—語音。但是霹靂車是一部機器，要如何和人們溝通呢？

隨著科技的進步，語音相關技術也越來越趨向成熟，要讓機器聽懂人們說的話進而和人們溝通已經不是什麼大問題了，坊間已經有許多語音相關的應用，像是打電話到某某公司的總機，會有語音詢問要找哪位員工，這時我們只要說出該名員工的姓名，電腦便會自動幫我們轉接，這中間涉及了語音辨識、語音合成、…等技術。

語言模型在自動語音辨識上扮演一個重要的角色。它的功能是要擷取自然語言中各式各樣的資訊，諸如前後文的資訊、語意資訊、文法資訊等。在語音辨識的過程中，語言模型透過上述所擷取的資訊，將一個詞序列（word sequence）以機率量化（quantify）來決定是否被接受（acceptable）[Rosenfeld 2000]，這可用來決定輸入語音的最有可能之詞序列。

語音辨識的基本架構是輸入一語音訊號，然後系統輸出最有可能的詞序列，如下式所示：

$$\begin{aligned}\hat{W} &= \arg\max_{W} P(W \mid X) \\ &= \arg\max_{W} \frac{P(X \mid W)P(W)}{P(X)} \\ &\approx \arg\max_{W} P(X \mid W)P(W)\end{aligned} \tag{1.1}$$

式(1.1)中，X 代表輸入的語音訊號，W 代表可能的詞序列，$\hat{W}$ 是最後輸出的詞

序列，且對於所有的W而言，$P(X)$都是相同的，故可以將之省略；另外，$P(X|W)$為聲學模型（acoustic model），$P(W)$為語言模型。在人類的語言中，詞的發音數往往都是遠少於詞的數目，以中文為例，發音是「ㄩˇ」的有非常多，如「與」、「語」、「雨」、「羽」等，在這樣的情況下，即使語音辨識器已將發音辨識正確，但是卻在決定詞的時候選錯詞，那麼此辨識的結果還是錯誤的，這個情形稱之為聲學混淆（acoustic confusion），於是，我們必須要有足夠的資訊，來決定哪一個詞才是正確的，這就是語言模型所背負的任務。在上述的概念下，可以把語言模型想像成是在訓練語料中蒐集自然語言的資訊，再利用這些資訊來估測每個一詞序列發生的機率，最後輸出機率最大者當作辨識的結果。

語言模型不僅是用在語音辨識上，舉凡資訊檢索（information retrieval）、機器翻譯（machine translation）、文字辨識（optical character recognition, OCR）以及自然語言處理（natural language processing, NLP）等應用，都可以發現語言模型的蹤跡，在這些應用裡，語言模型都扮演著重要的角色[Chou et al. 2003]。

然而自然語言（natural language）是一直不斷地在改變的：隨著人類文明的發展，一直有新的詞或片語被創造出來，如「冷笑話」、「草莓族」等，另外，在不同的領域，相同的詞或片語也有可能表示不同的意思，如「機械」在工業方面可以解釋成機器，在形容人的方面可以解釋成呆板，不知變化；此外，在不同的情況下，一個人的遣詞用語也會有所不同，例如在寫正式書信和寫筆記時的文法和用詞就會有明顯的不同，或是因為說話當時的社會、政治、經濟情況或說話者當時的情緒不同，也會有不同的用語。

由於這些變化性，造成了訓練語料（training corpus）和測試語料（testing corpus）之間詞典和語意的不一致（mismatch）。這使得統計式語言模型（statistical language model, SLM）在跨領域的應用上顯得很脆弱，就算是相同領域，也可能因為語料蒐集時間的差異，而造成相當的傷害，於是便有了語言模型調適的研究產生，這也是造成吾人研究語言模型以及調適的動機。

1.2 研究內容

自然語言中所包含的資訊是相當豐富的，諸如前後文的資訊、語意資訊、主題資訊、句法資訊等。語言模型就是要將這些資訊擷取出來，常見的如統計式語言模型（statistical language model, SLM）、觸發對（trigger pair）[Rosenfeld 1996]、潛藏語意分析（latent semantic analysis, LSA）[Bellegarda 2000]、主題混合模型（topic mixture model）[Chen et al. 2004c]等。

一般而言，在語音辨識的研究中，會先蒐集一訓練語料，此語料所包含的內容相當多，且涵蓋了許多的領域與主題，如運動、社會科學、自然科學、政治、藝文等，都包含在其中，並不偏頗某個方向，為的是要能夠從中擷取出一般性（general）的資訊，通常會利用此訓練語料來訓練一個 N 連語言模型（N-gram language model），當作背景語言模型使用。但是，如同 1.1 節中所敘述的訓練語料和測試語料間不一致的問題，所以只利用背景語言模型來辨識測試語料通常是不足夠的，因此，會另外蒐集一調適語料（adaptation corpus），此語料的大小相對於訓練語料小很多，但是其所包含的內容是與測試語料相關的，例如領域或主題相同（in-domain），或是蒐集時間相同（contemporary）等。利用擷取自調適語料的資訊，如前後文資訊、主題資訊、語意資訊等，將之與背景語言模型結合，使得結合之後的新模型對於測試語料有更好的預測（prediction）能力，而達到較佳的辨識結果，這個動作稱之為語言模型調適（language model adaptation）。換句話說，語言模型調適的目的，便是要利用調適語料中與測試語料相關的資訊，來彌補訓練語料與測試語料的不一致性。

目前較為人熟知的語言模型調適方法大致上可以區分為下列兩種：

1.以最大事後機率法為基礎（MAP-based）[Sasaki et al. 2000; Moriya et al. 2001; Bacchiani et al. 2003]，包含了詞頻數混合法（count merging）、模型插補法（model interpolation） [Jelinek 1991]以及動態快取模型法（dynamic caching model）

[Kuhn et al. 1990]。

2.以限制為基礎（constraint-based），包含了最大熵值法（maximum entropy, ME）[Jaines 1957; Rosenfeld 1996; Burger et al. 1996; Chueh et al. 2004]與最小鑑別資訊法（minimum discrimination information, MDI）[Federico 1999; Chen et al. 2003]。

1.3 研究成果

本論文對於語言模型的調適，提出了主題混合模型法（topic mixture model, TMM）[Chen et al. 2004c]。主題混合模型原是應用在資訊檢索中[Chen 2005; Chen et al. 2004b]，在此模型中，每一篇文件被表示成一個混合模型，模型中定義了 K 個主題，各由一個主題一連語言模型（topic unigram）所表示，且每一篇文件對這 K 個主題一連語言模型都有不同的權重。在給定一個查詢詞序列（query word sequence），要計算某個文件與此查詢詞序列的檢索機率便可以利用主題一連語言模型與其在該篇文件中所擁有的權重來計算之。應用在語言模型訓練上，我們將詞 w 視為只擁有一個詞的查詢，而其歷史詞序列 h（word history）視為一篇文件，但因歷史詞序列會隨著語音辨識的過程而改變，所以各主題在歷史詞序列所形成的文件中之權重必須動態地計算，再配合上語言模型調適的技術與背景語言模型結合，進而達到了動態的語言模型調適（dynamic language model adaptation）。

同時，本論文對最大熵值法（maximum entropy, ME）做了深入的探討，此方法與常用的模型插補法都是要將多個所蒐集到資訊來源（information sources）合併成一個新的模型的方法，差別在於模型插補法是將各個資訊來源所訓練的模型乘以個別的權重之後相加合併（weighted sum）；然而，最大熵值法是將每一個資訊來源當成一種限制（constraint），限制合併後的模型要滿足每一個所蒐集到

資訊，根據此概念訓練出一個單一的、整合的模型。不過滿足所有資訊來源的模型可能有無窮多個，在最大熵值法的概念下，取擁有最大熵值的模型當作此方法的解。在本論文的實驗可以看到，採用最大熵值法合併一連詞、二連詞與三連詞所得到的模型，比用傳統方法所訓練的後撤式三連語言模型（back-off trigram language model）之複雜度（perplexity）與字錯誤率（word error rate, WER）皆有改善。

1.4 論文架構

本論文的章節安排如下：

第 2 章簡介統計式語言模型以及數種擷取自自然語言中的資訊。此外，對於語言模型調適的基本架構做一介紹，並介紹以最大事後機率法為基礎（maximum a posterior, MAP）的數種語言模型調適的方法。

第 3 章介紹以限制為基礎的語言模型調適方法，包含了最大熵值法（maximum entropy, ME）以及最小鑑別資訊法（minimum discrimination information, MDI）。

第 4 章介紹主題混合模型（topic mixture model, TMM），這是一個由資訊檢索（Information retrieval）中延伸而來的方法，可用來擷取自然語言中的主題資訊（topic information）。

第 5 章為本篇論文的實驗部分，比較各種擷取自自然語言中的資訊以及不同的語言模型調適方法對語音辨識效果的影響。

第 6 章為本論文結論及未來展望。

第2章 背景簡介

本章首先將介紹語音辨識中所使用的統計式語言模型，以及由語料中所擷取的語意資訊（semantic information）、主題資訊（topic information）；接下來介紹語言模型調適的架構以及常用的方法，在本章中介紹以最大事後機率法（maximum a posterior, MAP）為基礎的詞頻數混合法（count merging）、模型插補法（model interpolation）、以及動態快取模型法（dynamic caching model），另外在第三章將特別介紹以限制（constraint）為基礎的最大熵值法（maximum entropy, ME）以及最小鑑別資訊法（minimum discrimination information, MDI）。

2.1 統計式語言模型（Statistical Language Model, SLM）

給定一長度為 n 之詞串 W，$W = w_1, w_2, ..., w_n$，以及一語言模型 P，要估測 W 在 P 中的機率（P 對 W 的量化接受度）$P(W)$ [Rosenfeld 2000]，可以利用連鎖律（chain rule）將其分解成：

$$\begin{aligned} P(W) &= P(w_1, w_2, ..., w_n) \\ &= P(w_1)P(w_2, ..., w_n \mid w_1) \\ &= P(w_1)P(w_2 \mid w_1)P(w_3, ..., w_n \mid w_1, w_2) \\ &= \prod_{i=1}^{n} P(w_i \mid w_1, ..., w_{i-1}) \\ &= \prod_{i=1}^{n} P(w_i \mid h_i) \end{aligned} \tag{2.1}$$

式(2.1)中 h_i 是詞 w_i 的歷史詞序列（word history），$h_i = w_1, ..., w_{i-1}$。

假設 V 表示我們所使用的詞典而 $|V|$ 是其所包含的詞的數量，則式(2.1)中 $P(w_i \mid h_i)$ 之參數量為 $|V|^i$，這個數量即使在一般的 $|V|$ 與 i 值之下，仍然是一個相當龐大的值，要完全估測非常的困難，勢必要做簡化。

在語音辨識中，N 連語言模型廣泛的被使用來處理這個問題，N 連語言模型是帶入 $N-1$ 階馬可夫模型假設（Markovian assumption），即假設詞 w_i 的出現只跟其前面 $N-1$ 個詞有關聯，而與前第 $N-1$ 個詞之前的詞沒有關聯。也就是說歷史詞序列可簡化成 $h_i = w_{i-N+1},...,w_{i-1}$，則式(2.1)可以改寫成：

$$\begin{aligned} P(W) &= \prod_{i=1}^{n} P(w_i \mid h_i) \\ &= \prod_{i=1}^{n} P(w_i \mid w_{i-N+1},...,w_{i-1}) \end{aligned} \tag{2.2}$$

常用的三連語言模型（tri-gram language model）可以表示成：

$$P(W) = P(w_1)P(w_2 \mid w_1)\prod_{i=3}^{n} P(w_i \mid w_{i-2}, w_{i-1}) \tag{2.3}$$

要估測式(2.3)中的 $P(w_i \mid w_{i-2}, w_{i-1})$，可以依據最大相似度估測法（maximum likelihood estimation, MLE）得到。令 Θ 為一語言模型，其參數 $\theta_{h_k,w_i} = P(w_i \mid h_k)$，則式(2.1)可以改寫成：

$$\begin{aligned} P(W) &= \prod_{i=1}^{n} P(w_i \mid h_i) \\ &= \prod_{k=1}^{K}\prod_{i=1}^{|V|} P(w_i \mid h_k)^{C_{h_k,w_i}} \\ &= \prod_{k=1}^{K}\prod_{i=1}^{|V|} \theta_{h_k,w_i}^{C_{h_k,w_i}} \end{aligned} \tag{2.4}$$

$|V|$ 為詞典大小，K 為所有歷史詞序列的種類，C_{h_k,w_i} 是事件 (h_k, w_i) 在 W 中出現的次數，(h_k, w_i) 是指 h_k 和 w_i 相鄰且 h_k 出現在 w_i 之前的事件。最大相似度估測法是要找出一語言模型 Θ^{MLE}，使得式(2.4)中 $P(W)$ 有最大值：

$$\Theta^{MLE} = \arg\max_{\Theta} P(W \mid \Theta) \tag{2.5}$$

語言模型 Θ 滿足

$$0 \le \theta_{h_k,w_i} \le 1 \tag{2.6}$$

以及

$$\sum_{i=1}^{|V|} \theta_{h_k,w_i} = 1 \quad \text{for all } h_k \tag{2.7}$$

我們以某個歷史詞序列 h_k 來看，所有事件 (h_k, w_i) 的出現次數 C_{h_k,w_i} 之分佈為一個多項式分佈（multinominal distribution），則可以得到：

$$P\left(C_{h_k,w_1},...,C_{h_k,w_{|V|}}\right) = \frac{C_{h_k}}{\prod_{i=1}^{|V|} C_{h_k,w_i}!} \prod_{i=1}^{|V|} \theta_{h_k,w_i}^{C_{h_k,w_i}} \tag{2.8}$$

式(2.8)中，$C_{h_k} = \sum_{i=1}^{|V|} C_{h_k,w_i}$。接下來將式(2.4)取對數：

$$\log P(W) = \sum_{k=1}^{K} \sum_{i=1}^{|V|} C_{h_k,w_i} \log \theta_{h_k,w_i} \tag{2.9}$$

則原本要對式(2.4)求最大值也相當於對式(2.9)求最大值，我們可以利用拉格朗日乘數（lagrange multiplier）將式(2.7)代入式(2.9)：

$$\log P(W) = \sum_{k=1}^{K} \sum_{i=1}^{|V|} C_{h_k,w_i} \log \theta_{h_k,w_i} + \sum_{k=1}^{K} l_{h_k} \left(\sum_{i=1}^{|V|} \theta_{h_k,w_i} - 1 \right) \tag{2.10}$$

再將式(2.10)對某個 θ_{h_k,w_i} 作偏微分並令其等於 0：

$$\frac{\partial \log P(W)}{\partial \theta_{h_k,w_i}} = \frac{\partial \left[\sum_{k=1}^{K} \sum_{i=1}^{|V|} C_{h_k,w_i} \log \theta_{h_k,w_i} + \sum_{k=1}^{K} l_{h_k} \left(\sum_{i=1}^{|V|} \theta_{h_k,w_i} - 1 \right) \right]}{\partial \theta_{h_k,w_1}} \tag{2.11}$$

$$\Rightarrow \frac{C_{h_k,w_i}}{\theta_{h_k,w_i}} + l_{h_k} = 0$$

我們可以將式(2.11)延伸到所有的θ_{h_k,w_i}，則得到：

$$\frac{\theta_{h_k,w_1}}{C_{h_k,w_1}}=\frac{\theta_{h_k,w_2}}{C_{h_k,w_2}}=\cdots=\frac{\theta_{h_k,w_{|V|}}}{C_{h_k,w_{|V|}}}=-l_{h_k} \tag{2.12}$$

又因為有$\frac{B_1}{A_1}=\frac{B_2}{A_2}=\frac{B_3}{A_3}=\frac{B_1+B_2+B_3}{A_1+A_2+A_3}$的關係式，則可以得到：

$$\frac{\sum_{j=1}^{|V|}C_{h_k,w_j}}{\sum_{s=1}^{|V|}\theta_{h_k,w_s}}=-l_{h_k} \quad \Rightarrow \quad l_{h_k}=-\sum_{j=1}^{|V|}C_{h_k,w_j}=-C_{h_k} \tag{2.13}$$

將l_{h_k}代入式(2.12)中得到：

$$\theta_{h_k,w_i}^{MLE}=\frac{C_{h_k,w_i}}{\sum_{j=1}^{|V|}C_{h_k,w_j}} \tag{2.14}$$

式(2.14)可以推廣到所有的歷史詞序列，所以最大相似度估測所求得的語言模型Θ^{MLE}為：

$$P^{MLE}\left(w_i \mid h_k\right)=\theta_{h_k,w_i}^{MLE}=\frac{C_{h_k,w_i}}{\sum_{j=1}^{|V|}C_{h_k,w_j}}=\frac{C_{h_k,w_i}}{C_{h_k}} \tag{2.15}$$

因此，式(2.3)的三連語言模型經由最大相似度估測法所求得為：

$$P\left(w_i \mid w_{i-2},w_{i-1}\right)=\frac{C\left(w_{i-2},w_{i-1},w_i\right)}{C\left(w_{i-2},w_{i-1}\right)} \tag{2.16}$$

式(2.16)中，$C(\bullet)$表示事件・出現的次數。

在實作上，雖然使用了 N 連語言模型，但還是沒有辦法蒐集到所有的 N 連

詞事件，以本論文的實驗 Set 2 為例（實驗語料設定請參閱第 5 章），我們蒐集了約一億五千萬個中文字的訓練語料，經過斷詞後得到約八千八百萬個中文詞，用此語料來訓練三連語言模型。經過統計，在此語料中出現的相異三連詞共有 9,344,284 個，但本論文所使用的詞典共包含 71,696 個詞，隱含的三連詞共有 $(71{,}696)^3$ 個，所以絕大部分的三連詞都沒有出現在訓練語料中。此外，在這九十三萬多個三連詞當中，有 5,019,151 個三連詞是只出現一次，1,623,608 個三連詞只出現兩次。在如此樣本數量缺乏的情況下所估測出來的三連語言模型是欠缺代表性與可靠度的，這會導致估測詞序列發生的機率產生錯誤，例如某個長度為 n 的詞序列 W 內含有訓練語料中未出現的三連詞 (w_{k-1}, w_{k-2}, w_k)，即 $P(w_k \mid w_{k-2}, w_{k-1}) = 0$，則 $P(W)$ 的機率為：

$$\begin{aligned} P(W) &= P(w_1)P(w_2)\prod_{i=3}^{n} P(w_i \mid w_{i-2}, w_{i-1}) \\ &= P(w_1)P(w_2)\prod_{i=3}^{k-1} P(w_i \mid w_{i-2}, w_{i-1}) \cdot P(w_k \mid w_{k-2}, w_{k-1}) \cdot \prod_{j=k+1}^{n} P(w_j \mid w_{j-2}, w_{j-1}) \\ &= P(w_1)P(w_2)\prod_{i=3}^{k-1} P(w_i \mid w_{i-2}, w_{i-1}) \cdot 0 \cdot \prod_{j=k+1}^{n} P(w_j \mid w_{j-2}, w_{j-1}) \\ &= 0 \end{aligned} \tag{2.17}$$

由式(2.17)得知詞序列 W 發生的機率為 0，這顯然是估測產生了錯誤，這將使得詞序列 W 永遠不可能被辨識出來。這樣的情形稱為 N 連語言模型的資料稀疏性（data sparseness），為了克服這個問題，語言模型的平滑化（smoothing）技術[Goodman 2001]被廣泛的應用在 N 連語言模型的訓練上，如本論文所採用的 Katz 模型平滑化技術[Katz 1987]。

2.2 語意資訊（Semantic information）

N 連語言模型因其本質的緣故，只能擷取詞 w_i 附近的資訊，即其歷史詞序列 $h_i = w_{i-N+1},...,w_{i-1}$ 中的資訊，超出這範圍以外的資訊就無法取得，所以決定 N 值的大小牽涉到了此模型的預測能力及可靠度，但是這兩者之間是成反比關係的；較大的 N 值可以擷取範圍較廣的資訊，但是卻也會使得潛在的相異歷史詞序列數量變得非常龐大（有 $|V|^{N-1}$ 種可能），而此模型的參數量共有 $|V|^N$ 個，越大的 N 值所需的訓練語料量越大，且並非每一種 (h_i, w_i) 都可以在訓練語料中觀察到，或者是出現的次數非常少，不具代表性，這造成了在估測此 N 連語言模型的缺失，使得所訓練出來的 N 連模型變得不可靠；另一方面，雖然較小的 N 值所訓練出的語言模型較為可靠，但所能夠擷取到的資訊也相對的變得很有限。舉例來說，下面兩個意思相同且組成的詞也相同的句子：

「今天 下 很大 的 雨」 (2.18)

「今天 雨 下 的 很大」 (2.19)

若我們要擷取『雨』和『很大』之間的資訊，在式(2.18)中，$N = 3$ 即可；但在式(2.19)中，N 要等於 4 才有辦法，這是 N 連語言模型的缺點。語意資訊由於其擷取資訊的方法與 N 連語言模型不同，所以沒有上述的問題，N 連語言模型是統計訓練語料中 N 連詞出現的次數（count），而語意模型是擷取詞 w_i 與其歷史詞序列 h_i 中的語意相依資訊，可以是歷史詞序列中的某個詞 w_j 與詞 w_i 之間的語意相依資訊，如觸發對（trigger pair），也可以是整個歷史詞序列 h_i 與詞 w_i 之間的語意資訊，如潛藏語意資訊（latent semantic information, LSA）[Bellegarda 2000, 2005]。換句話說，語意資訊擷取資訊的能力範圍可以延伸到整個歷史詞序列中，不像 N 連語言模型被 N 值所限定。

2.2.1 觸發對（Trigger pair）

觸發對[Rosenfeld 1996]是擷取較長距離的詞與詞之間的語意相依資訊，舉個本論文所蒐集到的新聞為例子說明：

『交通部』『電信總局』『從』『去年』『十一月』『成立』『了』『風險』『管理』『小組』『針對』『手機』『簡訊』『不法』『的』『案件』『進行』『取締』『兩』『個』『多』『月』『來』『已經』『斷』『話』『了』『一』『千』『多』『件』『不過』『電信』『警察局』『表示』『這』『類』『詐財』『色情』『的』『簡訊』『捉不勝捉』『而且』『有』『氾濫』『的』『趨勢』 (2.20)

在式(2.20)中的（『電信總局』，『手機』）、（『案件』，『警察局』）、（『電信』，『簡訊』）等詞組在這則新聞中是擁有相同語意的詞，換句話說，當一個詞序列有『電信總局』出現時，之後會出現『手機』的機率很高，也就是說『電信總局』會觸發（trigger）『手機』的出現。評估兩個詞 w_i 語 w_j 是否為觸發對的方法通常採用平均交互資訊（average mutual information, AMI）：

$$\begin{aligned} AMI(w_i, w_j) = {} & P(w_i, w_j)\log\frac{P(w_i, w_j)}{P(w_i)P(w_j)} + P(w_i, \overline{w}_j)\log\frac{P(w_i, \overline{w}_j)}{P(w_i)P(\overline{w}_j)} \\ & + P(\overline{w}_i, w_j)\log\frac{P(\overline{w}_i, w_j)}{P(\overline{w}_i)P(w_j)} + P(\overline{w}_i, \overline{w}_j)\log\frac{P(\overline{w}_i, \overline{w}_j)}{P(\overline{w}_i)P(\overline{w}_j)} \end{aligned} \tag{2.21}$$

其中 $P(w_i)$ 是詞 w_i 的一連語言模型機率，$P(w_i, w_j)$ 是詞 w_i 與詞 w_j 的聯合機率，$\overline{w}_i$ 代表非 w_i 的意思，也就是除了 w_i 之外的其他所有詞。要估測 $P(w_i, w_j)$ 可藉由貝氏定理將其分解成 $P(w_i, w_j) = P(w_i)P(w_j \mid w_i)$，再由最大相似度估測法可得 $P(w_j \mid w_i) = \frac{C(w_i, w_j)}{C(w_i)}$，$C(w_i)$ 是指詞 w_i 在訓練語料出現的次數，而計算 $C(w_i, w_j)$ 可設定一個長度為 *n* 的窗（window），將此窗由訓練語料的第一個詞移動到最後

一個詞，詞w_i與詞w_j在此窗移動的過程中一起出現的次數即為$C(w_i, w_j)$，與計算N連語言模型時的定義不同，根據[Rosenfeld 1996]中對詞典中的每一個詞（w_i）透過平均交互資訊所計算出的$AMI(w_i, w_j)$，發現分數最高的大部分皆為$AMI(w_i, w_i)$，也就是自我觸發（self-trigger），如式(2.20)中的（『簡訊』,『簡訊』）。根據自我觸發的現象發展出了動態快取一連語言模型（dynamic caching unigram language model），我們將在之後的語言模型調適方法中介紹。

2.2.2 潛藏語意分析（Latent Semantic Analysis, LSA）

潛藏語意分析[Bellegarda 2000, 2005]是一個利用線性代數中之奇異值分解（singular value decomposition, SVD）降維的技術，將原本高維度且不相關的詞向量與文件（document）向量投影到一個維度較低的潛藏語意空間。應用在語言模型上，將歷史詞序列h_i視為一個新文件，並計算其在前藏語意空間中的向量，進而利用餘弦估測（cosine measure）來估測詞w_i與其歷史詞序列h_i在此語意空間中的相似度$P(w_i \mid h_i)$。

2.2.2.1 詞與文件矩陣（Term-Document Matrix）

在進行奇異值分解之前，我們要將訓練語料製作成詞與文件矩陣（term-document matrix）。辨識詞典V的大小是$|V|$，令$M = |V|$；此外訓練語料包含了N篇文件。所以詞與文件矩陣W的維度是$M \times N$，矩陣中每個元素的值計算方法如下：

$$w_{i,j} = (1 - \varepsilon_i)\frac{c_{i,j}}{n_j} \tag{2.22}$$

$c_{i,j}$是詞w_i在第j篇文件d_j中出現的次數；n_j是第j篇文件的大小；ε_i是詞w_i在此N篇文件中的正規化熵值（normalized entropy）：

$$\varepsilon_i = -\frac{1}{\log N}\sum_{j=1}^{N}\frac{c_{i,j}}{t_i}\log\frac{c_{i,j}}{t_i} \tag{2.23}$$

式(2.23)中，$t_i = \sum_{j=1}^{N} c_{i,j}$，表示詞$w_i$在所有$N$篇文件中出現的次數，則$\frac{c_{i,j}}{t_i}$即表示詞$w_i$出現在文件$d_j$中的機率，我們以$P_{w_i}(d_j)$來表示，所以式(2.23)可以改寫成：

$$\varepsilon_i = -\frac{1}{\log N}\sum_{j=1}^{N} P_{w_i}(d_j)\log P_{w_i}(d_j) \tag{2.24}$$

其中$-\sum_{j=1}^{N} P_{w_i}(d_j)\log P_{w_i}(d_j)$代表詞$w_i$在此$N$篇文件間之機率分佈的熵值，且熵值的上下限為（參閱本論文第 3.1 節）：

$$0 \le -\sum_{j=1}^{N} P_{w_i}(d_j)\log P_{w_i}(d_j) \le \log N \tag{2.25}$$

所以式(2.23)乘以$\frac{1}{\log N}$所得到的是正規化後的熵值，即$0 \le \varepsilon_i \le 1$。$\varepsilon_i$越接近 0 表示詞$w_i$在越少的文件中出現，也越具有代表性；$\varepsilon_i$越接近 1 表示詞$w_i$在越多的文件中出現，也越不具有代表性，所以式(2.22)中才以$(1-\varepsilon_i)$來當作計算詞與文件矩陣的權重。

在建立好詞與文件矩陣W之後，每一個文件為W中一個維度是M的行向量（column vector），以$\widetilde{d}_j$表示，而每一個詞為W中一個維度是N的列向量（row vector），以$\widetilde{w}_i$表示。至此，我們還是無法估測詞與文件之間的關係，因為$\widetilde{w}_i$與$\widetilde{d}_j$的維度不同，而且其所代表的意義也不相同。在下一節中，我們將介紹如何透過線性代數中的奇異值分解，將$\widetilde{w}_i$與$\widetilde{d}_j$投影到一低維度且可以比較的潛藏語意空間。

2.2.2.2 奇異值分解（Singular Value Decomposition, SVD）

奇異值分解的式子如下所示：

$$W = USV^T \tag{2.26}$$

U 是 $M \times R$ 維的左奇異矩陣，S 是 $R \times R$ 維的對角奇異矩陣，V 是 $N \times R$ 維的右奇異矩陣，T 是距陣轉置的意思，奇異值分解如圖 2.1 所示：

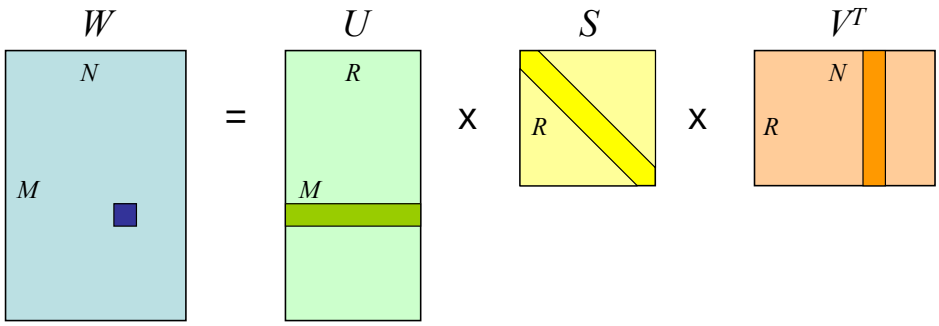


圖 2.1、奇異值分解圖示。

經過奇異值分解後，詞和文件都被投影到新的空間，稱為潛藏語意空間（LSA space），此空間的維度是 R 維，R 遠小於 M 與 N。換句話說，奇異值分解透過降維的方式，將原本高維的（N 與 M 維）且互不相關的詞和文件向量，投影到低維的同一個空間內，原本在 W 的列向量 $\tilde{w}_i$ 可用 U 的列向量 $\tilde{u}_i$ 來表示，而行向量 $\tilde{d}_j$ 可用 V 的行向量 $\tilde{v}_j$ 來表示，且 $\tilde{u}_i$ 與 $\tilde{v}_j$ 的每一維度都有一對一的對應關係，代表著某種潛藏的語意或主題[Bellegarda 2000, 2005]。第 j 篇文件 d_j 在原本 M 維空間的向量可以表示成（如圖 2.2）：

$$\tilde{d}_j = U S \tilde{v}_j^T \tag{2.27}$$

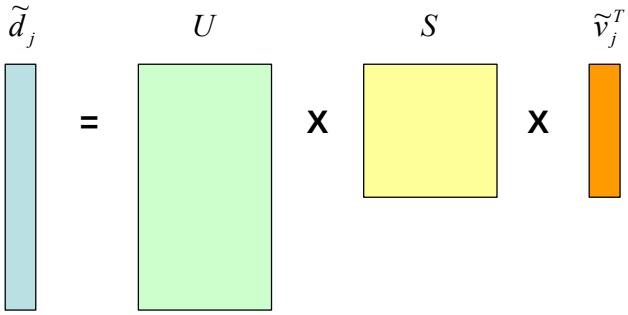


圖 2.2、第 j 篇文件經奇異值分解的示意圖。

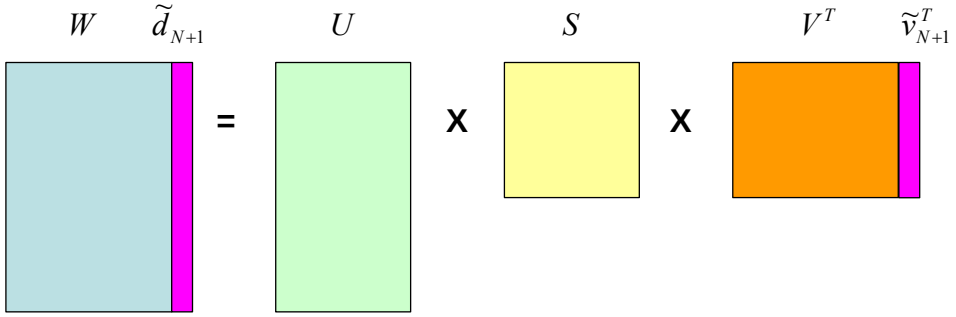


圖 2.3、摺入新的文件示意圖。

則文件 d_j 在潛藏語意空間中的向量可表示為：

$$\tilde{\tilde{v}}_j = \tilde{v}_j S = \tilde{d}_j^T U \tag{2.28}$$

若今有一新的文件加入，以 d_{N+1} 表示，假設此文件的加入並不影響原本經過奇異值分解所得到的三個矩陣（U、S 與 V），則 $\tilde{d}_{N+1}$ 可透過摺入（fold-in）的方式在此潛藏語意空間以式(2.27)來表示，摺入的示意圖如圖 2.3 所示。

應用在語言模型估測上，我們可以將歷史詞序列 h_i 視為一個新的文件，透過上述的觀念將其摺入到潛藏語意空間中，便可以估測歷史詞序列 h_i 在潛藏語意空

間（以 S 表示）內預測詞 w_i 的機率，以下式表示之：

$$P(w_i \mid h_i, \mathbf{S}) = P(w_i \mid \tilde{d}_{i-1}) \tag{2.29}$$

其中，$\tilde{d}_{i-1}$ 是由 $w_1, \cdots, w_{i-1}$ 所構成的向量。在此，$P(w_i \mid \tilde{d}_{i-1})$ 是將 w_i 與 $\tilde{d}_{i-1}$ 作餘弦估測（cosine measure）運算，求得的是 w_i 與 $\tilde{d}_{i-1}$ 的接近程度（closeness）。應用在語音辨識上，因歷史詞序列 h_i 會隨著辨識過程一直改變，所以 $\tilde{d}_{i-1}$ 也需要一直更新，更新的式子如下：

$$\tilde{d}_i = \frac{n_i - 1}{n_i}\tilde{d}_{i-1} + \frac{1-\varepsilon_i}{n_i}[0 \cdots 1 \cdots 0]^T \tag{2.30}$$

$[0 \cdots 1 \cdots 0]^T$ 是一個 M 維的向量，1 的地方代表該編號的詞新增到 $\tilde{d}_{i-1}$ 內；n_i 代表 $\tilde{d}_i$ 的長度。有了新的 $\tilde{d}_i$ 後便可以計算 h_i 在潛藏語意空間中的向量了：

$$\tilde{\tilde{v}}_i = \tilde{v}_i S = \frac{1}{n_i}\left[(n_i - 1)\tilde{\tilde{v}}_{i-1} + (1-\varepsilon_i)u_i\right] \tag{2.31}$$

$\tilde{\tilde{v}}_i$ 表示 $\tilde{d}_i$ 在潛藏語意空間中的向量，u_i 是矩陣 U 的第 i 列，表示詞 w_i 在潛藏語意空間中的向量，則我們可以得到：

$$\begin{aligned} P(w_i \mid \tilde{d}_{i-1}) &= \cos(u_i S^{1/2}, \tilde{\tilde{v}}_{i-1} S^{1/2}) \\ &= \frac{u_i S \tilde{\tilde{v}}_{i-1}^T}{\|u_i S^{1/2}\| \cdot \|\tilde{\tilde{v}}_{i-1} S^{1/2}\|} \end{aligned} \tag{2.32}$$

如此一來，我們就得到詞 w_i 與其歷史詞序列 h_i 在潛藏語意空間中的相似度了，下一節我們將介紹如何將此在潛藏語意空間中估測到的值與 N 連語言模型結合。

2.2.2.3 潛藏語意機率與 N 連語言模型的結合

將詞 w_i 在 N連語言模型中的歷史詞序列表示為 $h_i^{(N)}$，而在潛藏語意分析法中的歷史詞序列表示為 $h_i^{(L)}$，則：

$$\begin{aligned} P(w_i \mid h_i) &= P\left(w_i \mid h_i^{(N+L)}\right) \\ &= P\left(w_i \mid h_i^{(N)}, h_i^{(L)}\right) \\ &= \frac{P\left(w_i, h_i^{(L)} \mid h_i^{(N)}\right)}{\sum_w P\left(w, h_i^{(L)} \mid h_i^{(N)}\right)} \end{aligned} \tag{2.33}$$

式(2.33)中分子的部分可再展開成：

$$\begin{aligned} P\left(w_i, h_i^{(L)} \mid h_i^{(N)}\right) &= P\left(w_i \mid h_i^{(N)}\right) \cdot P\left(h_i^{(L)} \mid w_i, h_i^{(N)}\right) \\ &= P\left(w_i \mid w_{i-N+1}, \ldots, w_{i-1}\right) \cdot P\left(\tilde{d}_{i-1} \mid w_{i-N+1}, \ldots, w_{i-1}, w_i\right) \end{aligned} \tag{2.34}$$

這時我們加入一個假設，$\tilde{d}_{i-1}$ 只跟詞 w_i 有關，與詞 w_i 之前的詞都沒有關係，也就是說：

$$P\left(\tilde{d}_{i-1} \mid w_{i-N+1}, \ldots, w_{i-1}, w_i\right) = P\left(\tilde{d}_{i-1} \mid w_i\right) \tag{2.35}$$

將式(2.34)與式(2.35)代入式(2.33)中可以得到：

$$P\left(w_i \mid h_i^{(N+L)}\right) = \frac{P\left(w_i \mid w_{i-N+1}, \ldots, w_{i-1}\right) \cdot P\left(\tilde{d}_{i-1} \mid w_i\right)}{\sum_w P\left(w \mid w_{i-N+1}, \ldots, w_{i-1}\right) \cdot P\left(\tilde{d}_{i-1} \mid w\right)} \tag{2.36}$$

經由貝氏定理：

$$P\left(w_i \mid h_i^{(N+L)}\right) = \frac{P\left(w_i \mid w_{i-N+1}, \ldots, w_{i-1}\right) \cdot \frac{P\left(w_i \mid \tilde{d}_{i-1}\right)}{P\left(w_i\right)}}{\sum_w P\left(w \mid w_{i-N+1}, \ldots, w_{i-1}\right) \cdot \frac{P\left(w \mid \tilde{d}_{i-1}\right)}{P(w)}} \tag{2.37}$$

式(2.37)是將 N 連語言模型與潛藏語意分析合併，但當我們只使用一連語言模型的時候，式(2.37)將變成：

$$\begin{aligned} P\left(w_i \mid h_i^{(N+L)}\right) &= \frac{P\left(w_i\right)\cdot\frac{P\left(w_i \mid \tilde{d}_{i-1}\right)}{P\left(w_i\right)}}{\sum_w P\left(w\right)\cdot\frac{P\left(w \mid \tilde{d}_{i-1}\right)}{P\left(w\right)}} \\ &= \frac{P\left(w_i \mid \tilde{d}_{i-1}\right)}{\sum_w P\left(w \mid \tilde{d}_{i-1}\right)} \\ &= P\left(w_i \mid \tilde{d}_{i-1}\right) \end{aligned} \tag{2.38}$$

式(2.38)是潛藏語意分析的值。也就是說，當 N 連語言模型採用 $N = 1$ 時，式(2.37)將退化成只有潛藏語意分析的部分，所以合併 N 連語言模型和潛藏語意分析法只能適用於 $N > 1$ 的情形。

2.3 主題資訊（Topic information）

在此方法中，蒐集到的語料被分成數個主題（topic），例如體育、政治、經濟等等。將文件分成各主題的方法，分成監督式（supervised）與非監督式（unsupervised），監督式是在蒐集語料的同時即對每一個文件以人工的方式分類（classification），或是直接使用已經分類好的語料，如新聞網站；然而，非監督式分類是將蒐集到的每個文件表示成一個向量，再依據每個文件向量的距離（distance）來分類，常用的如 K-means 演算法[Ball et al. 1967; Duda et al. 1973]。非監督式分類法概念上認為在同一主題中的文件，其語意可視為是類似的、相近的，換句話說，主題資訊可視為是語意分類（semantic classification）的應用。

最簡單的方法便是採用線性模型插補法（參照 2.4.2.2 節）的概念。假設用人工的方式決定 K個主題，$\{t_k\}$表示這 K個主題的集合，這些主題涵蓋了整個訓練語料，即將訓練語料分成 K份，所以此主題模型可以視為是由 K個子模型所組成，每一個子模型都對應到一個主題。在這 K個子模型中，通常會包含一個「普遍」模型（"general" model），是使用全部的訓練語料所訓練而來，為的是要避免將訓練語料分成 K份後，造成資料缺乏（data sparseness）的問題產生，其餘的子模型則是由該主題所包含的訓練語料所訓練出來的。以線性插補法合併 K個子模型如下式所示：

$$P(w_i \mid h_i) = \sum_{k=1}^{K} \lambda_k P_k(w_i \mid h_i) \tag{2.39}$$

$P_k(w_i \mid h_i)$代表由訓練語料所訓練出第 k 個主題的子模型；λ_k 代表第 k個子模型的權重。

2.4 語言模型調適方法

本節將介紹語言模型調適的基本架構以及調適的方法。

2.4.1 語言模型調適的架構

統計式語言模型調適的一般架構如圖 2.4 所示。其中包含了兩種語料：第一種是訓練語料，包含了大量的語料，用來訓練背景語言模型用，這些語料涵蓋了許多領域與主題，可以從中訓練或求得一般性（general）的自然語言規則；第二種是調適語料，包含了較少量的語料（相對於訓練語料而言），但是是與辨識任務相關的語料。

在經過語言模型調適後，估測 $P(w_i \mid h_i)$ 便受兩個不同的資訊所影響，一個是用訓練語料所訓練的具有一般性的背景語言模型，如圖 2.4 中的 $P_B(w_i \mid h_i)$；另一個是由調適語料中擷取出來的資訊，這些資訊是和測試語料有關係的。接下來我們可以看到，這些由調適語料所擷取出來的資訊可以是各種不同的型態，諸如條件限制（constraint）、主題資訊（topic information）等。基本的概念是利用從調適語料中所取得的資訊來調整背景語言模型。

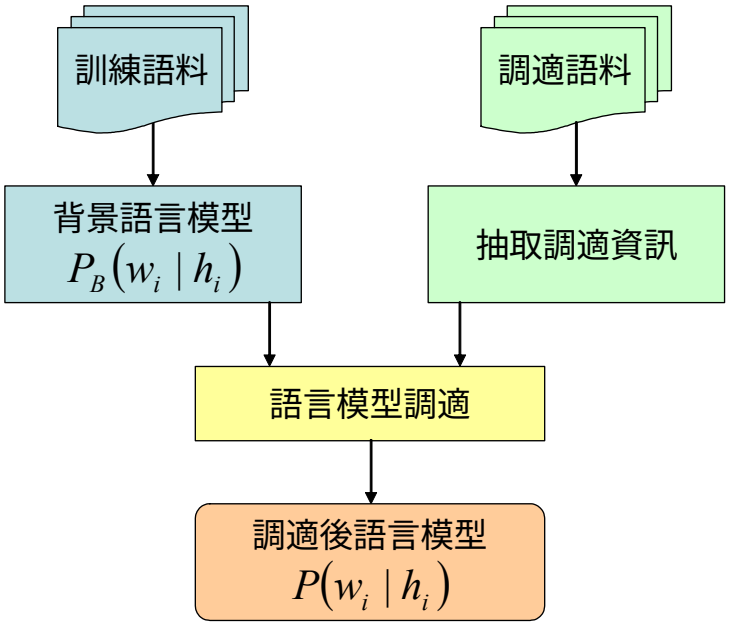


圖 2.4、語言模型調適架構圖。

2.4.2 最大事後機率法

由調適語料中取得的 N 連詞頻（N-gram count）可以透過不同的方式來調適背景語言模型，常見的有詞頻數合併法（count merging）以及模型插補法（model interpolation），這兩種方法都可視為是最大事後機率（maximum a posteriori, MAP）[Bacchiani et al. 2003]調適法的一種。

給定一個觀察樣本$\overline{W} = w_1 ... w_n$（observation sample），最大事後機率法便是要找到一個機率模型θ^*，使得$\overline{W}$在此機率模型上的事後機率分佈為最大值，式子如下：

$$\theta^* = \arg\max_{\theta} P(\theta \mid \overline{W}) = \arg\max_{\theta} P(\overline{W} \mid \theta) P(\theta) \tag{2.40}$$

其中N連語言模型$P(\overline{W} \mid \theta)$是一個多項式分佈（multinomial distribution）：

$$\begin{aligned} P(\overline{W} \mid \theta) &= P(w_1, ..., w_n \mid \phi_{h_1,w_1}, ..., \phi_{h_1,w_{|V|}}, ..., \phi_{h_k,w_1}, ..., \phi_{h_k,w_{|V|}}) \\ &\propto \prod_{k=1}^{K} \prod_{i=1}^{|V|} \phi_{h_k,w_i}^{c_{h_k,w_i}} \end{aligned} \tag{2.41}$$

$P(\theta)$是此模型的事前機率，其參數為$\phi_{h_1,w_1}, ..., \phi_{h_1,w_{|V|}}, ..., \phi_{h_k,w_1,...,}\phi_{h_k,w_{|V|}}$，$h_k$是$N$連模型中所有可能的歷史詞序列，總共有 K 個，$|V|$是辨識詞典的大小，則$\sum_{i=1}^{|V|} \phi_{h_k,w_i} = 1$。$c_{h_k,w_i}$是事件$(h_k, w_i)$在$\overline{W}$中出現的次數。此處的$(h_k, w_i)$不是指$h_k$與$w_i$的聯合出現次數，而是指$h_k$與$w_i$相鄰出現的次數。

令此機率模型為一個 Dirichlet 分佈，則：

$$P(\phi_{h_1,w_1,...,}\phi_{h_k,w_{|V|}} \mid \nu_{h_1,1}, ..., \nu_{h_k,w_{|V|}}) \propto \prod_{k=1}^{K} \prod_{i=1}^{|V|} \phi_{h_k,w_i}^{\nu_{h_k,w_i}-1} \tag{2.42}$$

$\nu_{h_k,w_i} > 0$，是 Dirichlet 分佈的參數；將式(2.41)和式(2.42)帶到式(2.40)中可以得到：

$$\begin{aligned} & P(\overline{W} \mid \theta) P(\theta) \\ &= P(w_1, ..., w_n \mid \phi_{h_1,w_1}, ..., \phi_{h_k,w_{|V|}}) \cdot P(\phi_{h_1,w_1}, ..., \phi_{h_k,w_{|V|}} \mid \nu_{h_1,w_1}, ..., \nu_{h_k,w_{|V|}}) \\ &\propto \prod_{k=1}^{K} \prod_{i=1}^{|V|} \phi_{h_k,w_i}^{\nu_{h_k,w_i}-1+c_{h_k,w_i}} \end{aligned} \tag{2.43}$$

將式(2.43)取 log 再加上 Lagrange multiplier，得到：

$$\sum_{k=1}^{K}\sum_{i=1}^{|V|}\log\phi_{h_k,w_i}^{\nu_{h_k,w_i}-1+c_{h_k,w_i}}$$
$$=\sum_{k=1}^{K}\sum_{i=1}^{|V|}\left(\nu_{h_k,w_i}-1+c_{h_k,w_i}\right)\log\phi_{h_k,w_i}+\sum_{k=1}^{K}l_{h_k}\left(\sum_{i=1}^{|V|}\phi_{h_k,w_i}-1\right) \tag{2.44}$$

l_{h_k} 是歷史詞序列 h_k 的 Lagrange multiplier。再來將式(2.44)對 ϕ_{h_k,w_i} 作偏微分後令其等於 0，可求得極值：

$$\left(\nu_{h_k,w_i}-1+c_{h_k,w_i}\right)\frac{1}{\phi_{h_k,w_i}}+l_{h_k}=0$$
$$\Rightarrow\phi_{h_k,w_i}=-\frac{\nu_{h_k,w_i}-1+c_{h_k,w_i}}{l_{h_k}} \tag{2.45}$$

又

$$\sum_{i=1}^{|V|}\phi_{h_k,w_i}=-\sum_{i=1}^{|V|}\frac{\nu_{h_k,w_i}-1+c_{h_k,w_i}}{l_{h_k}}=1$$
$$\Rightarrow l_{h_k}=-\sum_{i=1}^{|V|}\left(\nu_{h_k,w_i}-1+c_{h_k,w_i}\right)$$

將 l_{h_k} 帶入式(2.45)中，得到：

$$\phi_{h_k,w_i}=\frac{\nu_{h_k,w_i}-1+c_{h_k,w_i}}{\sum_{j=1}^{|V|}\left(\nu_{h_k,w_j}-1+c_{h_k,w_j}\right)} \tag{2.46}$$

式(2.46)即為最大事後機率法的解，若給予 ν_{h_k,w_i} 不同的值，會得到不同的結果，如下面要介紹的詞頻數合併法（count merging）和線性模型插補法（linear model interpolation）。

2.4.2.1 詞頻數合併法

若令式(2.46)中的$v_{h_k,w_i} = C_B(h_k)\frac{\alpha}{\beta}P_B(w_i \mid h_k)+1$，可以得到：

$$\begin{aligned} P(w_i \mid h_k) &= \frac{C_B(h_k)\frac{\alpha}{\beta}P_B(w_i \mid h_k)+C_A(h_k w_i)}{\sum_{j=1}^{|V|}\left(C_B(h_k)\frac{\alpha}{\beta}P_B(w_i \mid h_k)+C_A(h_k w_i)\right)} \\ &= \frac{\alpha C_B(h_k w_i)+\beta C_A(h_k w_i)}{\sum_{j=1}^{|V|}\alpha C_B(h_k w_j)+\sum_{j=1}^{|V|}\beta C_A(h_k w_j)} \\ &= \frac{\alpha C_B(h_k w_i)+\beta C_A(h_k w_i)}{\alpha C_B(h_k)+\beta C_A(h_k)} \end{aligned} \tag{2.47}$$

$P_B(\bullet)$代表背景語言模型機率，$C_B(\bullet)$代表某個詞或詞序列在訓練語料中出現的次數，而$C_A(\bullet)$則是在調適語料中出現的次數。$\frac{\alpha}{\beta}$是一個常數，可藉由期望值最大化（expectation-maximization, EM）演算法[Dempster et al. 1977]來求得。從上式可以觀察到，語言模型調適是作用在詞頻數階級（frequency count level）。

2.4.2.2 線性模型插補法

若令式(2.46)中的 $v_{h_k,w_i} = C_A(h_k)\frac{\lambda}{1-\lambda}P_B(w_i \mid h_k)+1$，可以得到：

$$
\begin{aligned}
P(w_i \mid h_k) &= \frac{C_A(h_k)\frac{\lambda}{1-\lambda}P_B(w_i \mid h_k)+C_A(h_k w_i)}{\sum_{j=1}^{|V|}\left(C_A(h_k)\frac{\lambda}{1-\lambda}P_B(w_i \mid h_k)+C_A(h_k w_i)\right)} \\
&= \frac{C_A(h_k)\left(\frac{\lambda}{1-\lambda}P_B(w_i \mid h_k)+P_A(w_i \mid h_k)\right)}{C_A(h_k)\frac{\lambda}{1-\lambda}\sum_{j=1}^{|V|}P_B(w_i \mid h_k)+\sum_{j=1}^{|V|}C_A(h_k w_j)} \\
&= \frac{C_A(h_k)\left(\frac{\lambda}{1-\lambda}P_B(w_i \mid h_k)+P_A(w_i \mid h_k)\right)}{C_A(h_k)\frac{\lambda}{1-\lambda}+C_A(h_k)} \\
&= \frac{\frac{\lambda}{1-\lambda}P_B(w_i \mid h_k)+P_A(w_i \mid h_k)}{\frac{\lambda}{1-\lambda}+1} \\
&= \lambda P_B(w_i \mid h_k)+(1-\lambda)P_A(w_i \mid h_k)
\end{aligned}
\tag{2.48}
$$

(2.48)中的 λ 和(2.47)中的 λ 同樣是一個常數，都可藉由期望值最大化演算法求得。在此式中，λ 扮演著線性插補法的係數，可以視為是各個機率分佈的權重。在(2.48)中，語言模型調適是作用在模型階級（model level）上的。

2.4.2.3 動態快取模型法（Dynamic caching model）

當一個語者說了一個詞 w_i，在不久之後，再用到這個詞的機會相當大，利用這個現象所發展的技術就是所謂的快取（caching）[Kuhn et al. 1990]。

在實作上，訓練快取模型的資料是由先前辨識的結果而來，通常蒐集自同一個辨識口語文件（spoken document）或是辨識文句（sentence），或者是蒐集前面

固定長度的詞。但由於辨識所得到的資料量有限，無法估測較高階的 N 連語言模型，所以通常只採用一連語言模型（unigram)。此一連語言模型是在辨識的過程當中，一邊將辨識所得到的結果記錄下來當作調適語料，一邊估測新的快取一連語言模型，$P_{cache}(w_i)$，所以稱為動態。

有了快取一連語言模型之後，我們可以利用線性插補法將之與背景語言模型合併，如式(2.49)所示，所以動態快取模型也可看成是線性模型插補法的一種，差別在於估測 $P_A(w_i \mid h_i)$ 的調適語料來源不同。

$$P(w_i \mid h_i) = \lambda P_B(w_i \mid h_i) + (1-\lambda) P_{chche}(w_i) \tag{2.49}$$

若想要採用較高階的 N 連語言模型也是可行的，只要加入分類的概念即可：

$$P(w_i \mid h_i) = \sum_{\{c_i\}} P(w_i \mid c_i) P(c_i \mid h_i) \tag{2.50}$$

$\{c_i\}$ 是詞 w_i 有可能所屬的類別（class）的集合。式(2.50)包含了兩個部分，一個是類別 N 連模型（class N-gram）：$P(c_i \mid h_i)$；另一個是類別指派機率模型（class assignment）：$P(w_i \mid c_i)$。

一般而言，類別 N 連模型都假設與任何辨識任務沒有關係（task independent)，所以我們可以使用背景訓練語料來估測類別 N 連模型，也就是說，$P(c_i \mid h_i) = P_B(c_i \mid h_i)$。另一方面，類別指派機率模型便是動態快取模型主要動態更新的部分了：

$$P(w_i \mid c_i) = (1-\lambda) P_A(w_i \mid c_i) + \lambda P_B(w_i \mid c_i) \tag{2.51}$$

其中 $P_A(w_i \mid c_i)$ 會隨著辨識過程而動態的計算，進而調適用背景訓練語料所訓練出來的背景類別指派模型，$P_B(w_i \mid c_i)$，以得到新的類別指派模型 $P(w_i \mid c_i)$；而 λ 和線性模型插補法中的 λ 是相同的，都可用期望值最大化演算法求得。

2.4.2.4 線性模型插補法的延伸

經由最大事後機率法所求得的式(2.48)中 $P_A(w_i \mid h_i)$及 $P_B(w_i \mid h_i)$都是 N 連語言模型，但若以其他的型式代替，例如觸發對模型（trigger pair model)、類別 N 連語言模型（class-based n-gram model)，便可得到另外一種模型的差補法。而且也可將組成元素由兩個擴充到更多個，如：

$$P(w_i \mid h_i) = \lambda_1 P_1(w_i \mid h_i) + \lambda_2 P_2(w_i \mid h_i) + \ldots + \lambda_n P_n(w_i \mid h_i) \tag{2.52}$$

其中，$\sum_{i=1}^{n} \lambda_i = 1$。

換句話說，在這個架構下，根據調適語料蒐集的方式以及 $P_A(w_i \mid h_i)$和 $P_B(w_i \mid h_i)$的估測方式，甚至是模型平滑化（smoothing）的採取與否，可以得到許多不同變化的結果。

2.5 語言模型評估

本論文採用兩個方法來評估語言模型的效能，一為語言模型複雜度 (perplexity)，另一為語音辨識字錯誤率（character error rate, CER)。語言模型複雜度是最常被使用來評估一語言模型之效能的方法，此方法是拿要評估的語言模型，計算測試語料的正確參照轉寫（transcription）在此模型上發生的機率，此機率之倒數的幾何平均值即為此語言模型的複雜度，如下式所示：

$$Perplexity = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i \mid w_1, \cdots, w_{i-1})}} \tag{2.53}$$

在資訊理論（information theory）中的熵值（entropy)，代表了一個機率模型對測

試語料中每一個詞的平均編碼長度（bits），且 $perplexity = 2^{entropy}$ [Jelinek et al. 1977]，此式為一單調函數（monotonic function），即 entropy 越大 perplexity 越高，因為 entropy 越小代表此模型越好，也就是說，perplexity 值越低代表此語言模型越好。

另外一個評估的方法是採用語音辨識的字錯誤率，本論文使用美國標準與科技組織所訂立的評估標準（US NIST F.O.M metric）[NIST]來進行正確參照轉寫與辨識結果的字串比對，其中 H 代表兩者相同（match）的數量，I 代表字串插入（insertion）的數量，N 代表正確參照轉寫的字元數（character），則辨識率（accuracy）的計算方式為 $\frac{H-I}{N}\times 100\%$，則錯誤率為(1－辨識率)。

第3章 最大熵值法

本章將介紹以限制為基礎（constraint based）的語言模型調適方法，從最大熵值法（maximum entropy, ME）來切入，接著討論最小鑑別資訊法（minimum discrimination information, MDI）。

3.1 最大熵值法簡介

最大熵值法最早是由學者 E. T. Jaines 在 1957 年所提出[Jaines 1957]。此方法應用在語言模型調適上[Rosenfeld 1996; Berger et al. 1996; Chueh et al. 2004]和模型插補法（model interpolation）的精神相似，都是嘗試將多個資訊來源（information sources）合併成一個新的模型的方法。差別在於模型插補法是將各個資訊來源個別訓練出一個語言模型之後，再將這些模型乘以個別的權重之後合併，如式（3.1）：

$$P(w_i \mid h_i) = \lambda_1 P_1(w_i \mid h_i) + \lambda_2 P_2(w_i \mid h_i) + \ldots + \lambda_n P_n(w_i \mid h_i) \tag{3.1}$$

其中 $\sum_{i=1}^{n} \lambda_i = 1$。在模型插補法中，通常同一個語言模型中的所有模型參數共用一個權重，例如 $P_1(w_i \mid h_i)$ 和 $P_1(w_j \mid h_j)$ 皆使用 λ_1 當作權重。然而，相反地，最大熵值法並不對每一個資訊來源作個別的訓練，而是直接訓練出一個單一的、整合的模型，這個模型將每一個資訊來源所提供的訊息都包含進來。

如上所述，最大熵值法是一種整合各種資訊來源（information sources）的方法。在最大熵值方法中，每一個資訊來源都會引發一群限制 (a set of constraints)，而這些限制的交集區域（intersection）便是代表滿足所有的限制的機率分佈的集

合，此集合內的機率分佈可能有無窮多個，然而在這些機率分佈中，擁有最高熵值（highest entropy）的便是此方法的解。

舉個簡單的例子，假設現有一個骰子，若在僅知丟擲此骰子有可能出現 1 點、2 點、3 點、4 點、5 點以及 6 點，以及擲出這六種點數的機率和是 1 的情況下，我們得到第一個限制（constraint），即：

$$P(1)+P(2)+P(3)+P(4)+P(5)+P(6)=1 \tag{3.2}$$

明顯地，滿足這個限制的機率分佈有無窮多個，例如 $p(3)=1$，即每一次丟擲這個骰子都是出現 3 點，但這顯然不是一個好的機率分佈。直覺上，在沒有其他額外資訊的情況下，我們可以將機率平均分給這六種點數：

$$P(1)=P(2)=P(3)=P(4)=P(5)=P(6)=\frac{1}{6} \tag{3.3}$$

會採用平均分配是因為我們至此只知道丟擲此骰子會出現六種點數，這樣子分配既滿足已知的限制又合乎公平性；所謂的公平性是指每種點數出現的機率相同。假設我們又獲得了其他的資訊，例如，丟擲此骰子出現 1 點或 2 點的機率是 0.5，則我們得到第二個限制：

$$P(1)+P(2)=\frac{1}{2} \tag{3.4}$$

當然，滿足此第一及第二限制的機率分佈還是無窮多個。在缺乏其他資訊的情況下，我們仍可以合理的選擇一個滿足所有限制且最平均分配的機率分佈：

$$\begin{aligned} &P(1)=P(2)=\frac{1}{4} \\ &P(3)=P(4)=P(5)=P(6)=\frac{1}{8} \end{aligned} \tag{3.5}$$

假設我們再獲得了其他的資訊，例如，丟擲此骰子出現 2 點或 5 點的機率是 0.3，則第三個限制可表示成：

$$P(2)+P(5)=\frac{3}{10} \tag{3.6}$$

同樣地，滿足此三個限制的機率分佈仍然有無窮多個，但我們還是選擇滿足所有限制且最平均分配的分佈，不過這時候可能的機率分佈就沒有那麼明顯可以直接觀察出來了。換句話說，隨著越多的資訊加入，限制也越來越多，要找出滿足所有限制且最平均分配的機率分佈也就越困難。

這個問題可以用最大熵值法來解決。我們從熵值（entropy）的定義來看，給定一個由事件 $w_1, w_2, ..., w_N$ 所形成的機率模型 P，其熵值可表示成：

$$H(P)=-\sum_{i=1}^{N} P(w_i)\log_2 P(w_i) \tag{3.7}$$

當某個事件 w_i 的機率為 1，$P(w_i)=1$，而其他事件機率皆為 0 的時候，即 $P(w)=\begin{cases}1, & w=w_i\\ 0, & otherwise\end{cases}$，$H(P)$ 有最小值等於 0；而 $H(P)$ 的最大值出現在每個事件間為平均分佈（uniform distribution）的情況，即：

$$P(w_1)=P(w_2)=...=P(w_N)=1/N \tag{3.8}$$

此時熵值為 $H(P)=-\log_2(1/N)$。所以說，要找到滿足所有限制且最平均分配的機率模型，便等同於在滿足所有限制的機率模型當中，挑選出擁有最大熵值的機率模型。

3.1.1 特徵與限制（Features and Constraints）

假設要估測 $P(h,很好)$ 的機率值，其中 h 是『很好』這個詞的歷史詞序列（word history），且已知有一個資訊來源是二連語言模型（bigram language model）。二連語言模型根據歷史詞序列 h 的最後一個詞來分割 $P(h,很好)$ 的機率分佈，如圖 3.1 所示[Rosenfeld 1996]：

*h*最後是『天氣』	*h*最後是『心情』	…	…	…
· · · · ·	· · · · ·	· · · · ·	· · · · ·	· · · · ·

圖 3.1、二連語言模型將機率分佈 $P(h,\text{很好})$ 切割成每一行（column）的等價類，在同一個等價類中的歷史詞序列 h 的最後一個詞皆相同。

圖 3.1 中的每一行（column）是一個等價類（equivalence class），此二連語言模型指派相同的機率值給屬於同一類中的事件。我們拿（h 最後是『天氣』）這一行來說，當歷史詞序列 h 是（『今天』,『天氣』）、（『明天』,『天氣』）或是（『昨天』,『天氣』）…等等的時候，都會落在這一類中，當滿足此二連語言模型且最平均分配的情況下，有以下的關係：

$$P(\text{今天},\text{天氣},\text{很好})+P(\text{明天},\text{天氣},\text{很好})+P(\text{後天},\text{天氣},\text{很好})+\cdots = P(\text{天氣},\text{很好}) \tag{3.9}$$

且

$$P(\text{今天},\text{天氣},\text{很好})=P(\text{明天},\text{天氣},\text{很好})=P(\text{後天},\text{天氣},\text{很好})=\cdots \tag{3.10}$$

假設我們另外有一個主題模型（topic model），且令每個歷史詞序列都只能夠歸類到某一個主題中。則此模型根據歷史詞序列所歸屬的主題來分割機率分佈 $P(h,\text{很好})$ 如圖 3.2 所示。

$h \in$主題一	. . . . .
$h \in$主題二	. . . . .
$h \in$主題三	. . . . .
…	. . . . .

圖 3.2、主題分類語言模型將機率分佈 $P(h,\text{很好})$ 切割成每一列（row）的等價類，在同一個等價類中的歷史詞序列 h 皆歸屬於相同的主題。

圖 3.2 中的每一列（row）是一個等價類，主題模型指派相同的機率值給屬於同一類中的事件。我們拿（$h \in$主題一）這一列來說，會歸屬於主題一的不同歷史詞序列，如 $h_1, h_2, h_3, \cdots$ 等等，都會落在這一類中，當滿足此主題模型且最平均分配的情況下，有以下的關係：

$$P(h_1,\text{很好}) + P(h_2,\text{很好}) + P(h_3,\text{很好}) + \cdots = P(h \in T_1,\text{很好}) \tag{3.11}$$

且

$$P(h_1,\text{很好}) = P(h_2,\text{很好}) = P(h_3,\text{很好}) = \cdots \tag{3.12}$$

式(3.11)中，T_1 代表主題一。接下來，我們討論同時滿足上述之二連語言模型與主題模型的情況，此模型會依據歷史詞序列 h 的最後一個詞以及其所歸屬的主題來分割機率分佈 $P(h,\text{很好})$，如圖 3.3 所示。

	h最後是『天氣』	h最後是『心情』	…	…
$h\in$主題一	⋮ ⋮	⋮ ⋮	⋮ ⋮	⋮ ⋮
$h\in$主題二	⋮ ⋮	⋮ ⋮	⋮ ⋮	⋮ ⋮
…	⋮ ⋮	⋮ ⋮	⋮ ⋮	⋮ ⋮

圖 3.3、機率分佈 $P(h,$很好$)$ 被二連語言模型與主題模型所切割，圖中每一個方格為一個等價類，在同一個等價類中的歷史詞序列 h 的最後一個詞皆相同且歸屬於相同的主題。

圖 3.3 中的每一個方格是一個等價類，經由滿足二連語言模型與主題分類語言模型的機率模型指派相同的機率值給屬於同一類中的事件。我們拿滿足（h 最後是『天氣』）與（$h\in$主題一）的這一個方格來說，滿足此兩個限制的歷史詞序列，如 $h_1,h_2,h_3,...$ 等等，都會落在這一類中，在最平均分配的情況下有以下關係：

$$\begin{aligned} & P(h_1\text{最後是『天氣』} \,\&\, h_1\in T_1,\text{很好}) \\ =\ & P(h_2\text{最後是『天氣』} \,\&\, h_2\in T_1,\text{很好}) \\ =\ & P(h_3\text{最後是『天氣』} \,\&\, h_3\in T_1,\text{很好}) \\ =\ & \cdots \end{aligned} \tag{3.13}$$

要表示我們所擁有的資訊可以利用指示函數（indicator function）f 來完成，例如要表示二連語言模型中的（『天氣』,『很好』）：

$$f_{(\text{『天氣』,『很好』})}(h,w)=\begin{cases}1, & h\text{最後的詞是『天氣』，且}w\text{是『很好』}\\ 0, & otherwise\end{cases} \tag{3.14}$$

從式(3.14)可以得知，必須要（h 最後的詞是『天氣』）和（w 是『很好』）都成立的情況下，$f_{(『天氣』,『很好』)}(h,w)$的值才會是 1，其他情形則為 0。我們又可稱 f 為特徵函數（feature function）或簡稱為特徵（feature）。

我們對特徵 $f_{(『天氣』,『很好』)}(h,w)$在訓練語料中求期望值：

$$\widetilde{P}(f)=\sum_{h,w}\widetilde{P}(h,w)f_{(『天氣』,『很好』)}(h,w)=\widetilde{P}(天氣,很好) \tag{3.15}$$

式(3.15)中的(h,w)是由訓練語料中所蒐集到的歷史詞序列 h，與詞典中的詞 w 所組成的所有配對，$\widetilde{P}(\bullet)$是由訓練語料經由最大相似度估測法（MLE）所求得的二連語言模型。式(3.15)中，$\widetilde{P}(f)=\widetilde{P}(天氣,很好)$代表了 $f_{(『天氣』,『很好』)}(h,w)$在訓練語料中的期望值，此期望值就是在式(3.9)中將所有歷史詞序列最後是『天氣』的機率值，$P(h,很好)$，相加起來所得到的值。

假設有一個任意的聯合機率函數 $P(h,w)$，則特徵 $f_{(『天氣』,『很好』)}(h,w)$在此機率函數的期望值可以表示成：

$$P(f)=\sum_{h,w}P(h,w)f_{(『天氣』,『很好』)}(h,w) \tag{3.16}$$

在 3.1 節中提到，最大熵值法所要求的機率函數，必須要滿足所有蒐集到的資訊來源，即限定此聯合機率函數 $P(h,w)$滿足在訓練語料中各種資訊來源所引發的限制。換句話說，對每個特徵（以下用 $f(h,w)$表示）而言，其在訓練語料中期望值要等於使用機率函數 $P(h,w)$所計算出的期望值：

$$P(f)=\widetilde{P}(f) \tag{3.17}$$

式(3.17)稱之為限制方程式（constraint equation）或簡稱為限制（constraint）。

將式(3.15)和式(3.16)代入式(3.17)中，可以得到對於某一特徵 $f(h,w)$ 而言，必須滿足：

$$\sum_{h,w} P(h,w) f(h,w) = \sum_{h,w} \widetilde{P}(h,w) f(h,w) \tag{3.18}$$

由於在語音辨識中，語言模型是採用條件機率函數（conditional probability distribution），$P(w \mid h)$，而式(3.18)中所求得的 $P(h,w)$ 為聯合機率函數（joint probability distribution），我們可以由貝氏定理得到：

$$P(w \mid h) = \frac{P(h,w)}{P(h)} \tag{3.19}$$

不過要計算所有可能的歷史詞序列 h 與詞典中的詞 w 的機率 $P(h,w)$，其隱含的參數量總共有 $\left(|V|^{L+1}\right)$，V 表示所使用的詞典，$|V|$ 則為此詞典所包含的詞的數量，L 是歷史詞序列 h 的長度，這即使在一個合理的詞典大小及 L 值之下，也是一個很龐大的數量。然而，對於某個詞 w_i 而言，在我們所蒐集的訓練語料中，並非所有的歷史詞序列都會與其相鄰著出現，換句話說，絕大部分 $P(h,w_i)$ 的值都等於 0，只有少數的歷史詞序列會使得 $P(h,w_i)$ 為非 0 的值，則我們假設：

$$P(h) \approx \widetilde{P}(h) \tag{3.20}$$

即歷史詞序列在 P 的事前機率 $P(h)$ 等於其在訓練語料中的事前機率 $\widetilde{P}(h)$。我們將式(3.19)與式(3.20)代入式(3.18)中得到：

$$\sum_{h,w} \widetilde{P}(h) P(w \mid h) f(h,w) = \sum_{h,w} \widetilde{P}(h,w) f(h,w) \tag{3.21}$$

我們再對特徵（feature）與限制（constraint）做個定義：特徵是 (h,w) 配對的二元值函數（binary-valued function），$f(h,w)$；限制是特徵函數在某一機率分佈下與訓練語料中之期望值「相等」的動作。

3.1.2 指數型（Exponential form）

今假設我們從訓練語料中找出 n 個特徵，$f_1, f_2, \cdots, f_n$，根據最大熵值法的概念，首先我們所要求的機率模型必須滿足這 n 個特徵，令 $\mathbf{C}$ 是所有滿足此 n 個特徵的機率模型的集合：

$$\mathbf{C} \equiv \left\{ P \in \mathbf{P} \mid P(f_i) = \tilde{P}(f_i) \ \text{ for } i = 1,2,...,n \right\} \tag{3.22}$$

$\mathbf{P}$ 是所有機率模型的集合。在統計上，要估測一個條件機率分佈 $P(w \mid h)$的熵值可以利用條件熵值（conditional entropy）：

$$H(P) \equiv -\sum_h P(h) \sum_w P(w \mid h) \log P(w \mid h) \tag{3.23}$$

同式(3.20)，我們將式(3.23)中的 $P(h)$置換成 $\tilde{P}(h)$得到：

$$H(P) \equiv -\sum_h \tilde{P}(h) \sum_w P(w \mid h) \log P(w \mid h) \tag{3.24}$$

接下來，就是要從集合 $\mathbf{C}$ 中找出擁有最大熵值的機率模型：

$$P^* = \arg\max_{P \in \mathbf{C}} H(P) \tag{3.25}$$

P^* 即為最大熵值法所要求的機率模型。

將式(3.24)代入式(3.25)中，可以得到：

$$\begin{aligned} P^* &= \arg\max_{P \in \mathbf{C}} H(P) \\ &= \arg\max_{P \in \mathbf{C}} \left(-\sum_{h,w} \tilde{P}(h) P(w \mid h) \log P(w \mid h) \right) \end{aligned} \tag{3.26}$$

到目前為止，我們獲得了下列的訊息：

(A) $0 \le P(w \mid h) \le 1 \quad \forall h, w$，且 $\sum_{w} P(w \mid h) = 1 \quad \forall h$，

(A)保證 P 是一個條件機率分佈。

(B) $\sum_{h,w} \tilde{P}(h) P(w \mid h) f_i(h, w) = \sum_{h,w} \tilde{P}(h, w) f_i(h, w) \quad \text{for } i = 1, ..., n$，

(B)是最大熵值法中令所求的機率函數 P 要滿足所有特徵的限制。

(C) $H(P) = -\sum_{h,w} \tilde{P}(h) P(w \mid h) \log P(w \mid h)$

我們使用 Lagrange multiplier Λ 和 γ，分別將(B)與(A)引入(C)中來估測 $P(w \mid h)$，並令其為 $\xi(P, \Lambda, \gamma)$：

$$\begin{aligned} \xi(P, \Lambda, \gamma) = & -\sum_{h,w} \tilde{P}(h) P(w \mid h) \log P(w \mid h) \\ & + \sum_{i} \lambda_i \left(\sum_{h,w} \tilde{P}(h) P(w \mid h) f_i(h, w) - \sum_{h,w} \tilde{P}(h, w) f_i(h, w) \right) \\ & - \gamma \left(\sum_{w} P(w \mid h) - 1 \right) \end{aligned} \tag{3.27}$$

λ_i 是特徵 $f_i(h, w)$ 的參數，$\Lambda = (\lambda_1, ..., \lambda_n)$。將式(3.27)對 $P(w \mid h)$ 作偏微分後令其等於 0，可以求得 $P^*(w \mid h)$，使得 $\xi(P, \Lambda, \gamma)$ 有最大值：

$$\begin{aligned} & \frac{\partial \xi}{\partial P(w \mid h)} = -\tilde{P}(h)(1 + \log P(w \mid h)) + \sum_{i} \lambda_i \tilde{P}(h) f_i(h, w) - \gamma = 0 \\ & \Rightarrow \tilde{P}(h)(1 + \log P^*(w \mid h)) = \sum_{i} \lambda_i \tilde{P}(h) f_i(h, w) - \gamma \\ & \Rightarrow \log P^*(w \mid h) = \sum_{i} \lambda_i f_i(h, w) - \frac{\gamma}{\tilde{P}(h)} - 1 \\ & \Rightarrow P^*(w \mid h) = \exp\left(\sum_{i} \lambda_i f_i(h, w) \right) \exp\left(-\frac{\gamma}{\tilde{P}(h)} - 1 \right) \end{aligned} \tag{3.28}$$

我們再藉由(A)進一步的推導：

$$
\begin{aligned}
&\because \text{(A)}\ \forall h, \sum_w P(w \mid h) = 1 \\
&\Rightarrow \sum_w P(w \mid h) = \sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right) \exp\left(-\frac{\gamma}{\widetilde{P}(h)} - 1\right) = 1 \\
&\Rightarrow \exp\left(-\frac{\gamma}{\widetilde{P}(h)} - 1\right) \cdot \sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right) = 1 \\
&\Rightarrow \exp\left(-\frac{\gamma}{\widetilde{P}(h)} - 1\right) = \frac{1}{\sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right)}
\end{aligned}
\tag{3.29}
$$

將式(3.29)代入式(3.28)，且令 $Z(h) = \sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right)$，則式(3.28)可以改寫成：

$$
\begin{aligned}
P^*(w \mid h) &= \frac{1}{\sum_{\hat{w}} \exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right)} \cdot \exp\left(\sum_i \lambda_i f_i(h,w)\right) \\
&= \frac{1}{Z(h)} \exp\left(\sum_i \lambda_i f_i(h,w)\right)
\end{aligned}
\tag{3.30}
$$

式(3.30)即為指數型，其中 $Z(h)$ 是正規化因子（normalizing factor），確保 $P^*(w \mid h)$ 是一個條件機率分佈。

3.1.3 最大熵值法與最大相似法的關係

給一個由訓練語料所訓練出來的聯合機率函數 $\widetilde{P}(h,w)$，及某個條件機率函數 $P(w \mid h)$，則以 $P(w \mid h)$ 來預測訓練語料之對數相似值（log-likelihood）可定義成：

$$
L_{\widetilde{P}}(P) \equiv \log \prod_{h,w} P(w \mid h)^{\widetilde{P}(h,w)} = \sum_{h,w} \widetilde{P}(h,w) \log P(w \mid h) \tag{3.31}
$$

將式(3.30)中的 $P^*(w|h)$ 代入式(3.31)中的 $P(w|h)$：

$$\begin{aligned} L_{\tilde{P}}(P) &= \sum_{h,w} \tilde{P}(h,w)\left(\sum_i \lambda_i f_i(h,w)\right) - \sum_{h,w} \tilde{P}(h,w)\log \sum_{\hat{w}} \exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right) \\ &= \sum_{h,w} \tilde{P}(h,w)\sum_i \lambda_i f_i(h,w) - \sum_h \tilde{P}(h)\log \sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right) \end{aligned} \tag{3.32}$$

將式(3.32)對 λ_i 作偏微分，得到：

$$\begin{aligned} \frac{\partial L_{\tilde{P}}(P)}{\partial \lambda_i} &= \sum_{h,w} \tilde{P}(h,w) f_i(h,w) - \sum_h \tilde{P}(h) \frac{\sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right)\cdot f_i(h,w)}{\sum_{\hat{w}} \exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right)} \\ &= \sum_{h,w} \tilde{P}(h,w) f_i(h,w) - \sum_{h,w} \tilde{P}(h) P^*(w|h) f_i(h,w) \\ &= \tilde{P}(f_i) - P(f_i) \end{aligned} \tag{3.33}$$

式(3.33)中，$\tilde{P}(f_i)$ 代表特徵 f_i 在 empirical distribution $\tilde{P}(h,w)$ 的期望值，$P(f_i)$ 代表特徵 f_i 在 $\tilde{P}(x)P^*(y|x)$ 的期望值。令式(3.33)等於 0，可以得到一組 $\Lambda = (\lambda_1, \lambda_2, \cdots \lambda_n)$，使得式(3.31)有最大值，即 $P^*(w|h)$ 預測訓練語料有最大的對數相似值（Maximum log-likelihood）。然而，令式(3.33)等於 0 得到：

$$\begin{aligned} & P(f_i) = \tilde{P}(f_i) \\ & \Rightarrow \sum_{h,w} \tilde{P}(h) P^*(w|h) f_i(h,w) = \sum_{h,w} \tilde{P}(h,w) f_i(h,w) \end{aligned} \tag{3.34}$$

式(3.34)與 3.1.2 節中所得到的(B)相同，即最大熵值法中限制所求的機率模型必須滿足所有特徵的式子。換句話說，最大熵值法所求得的機率函數 P^* 也會使得預測訓練語料之對數相似值為最大。根據這個結果，我們可以將最大熵值法所要解的問題重新定義如下：

找到一個機率函數 $P^*(w|h)$ 滿足所有的特徵，並且使預測訓練語料之對數相似值為最大。

接下來，我們便利用這個定義來求 $P^*(w|h)$。

3.1.4 IIS（Improved Iterative Scaling）演算法

假設已經有一個機率模型是以指數型表示的，其參數 $\Lambda \equiv \{\lambda_1, \lambda_2, ..., \lambda_n\}$，我們想要找到一組新的參數 Λ'，此參數所形成的機率模型比以參數 Λ 所形成的機率模型擁有較高的對數相似值（log-likelihood）值，新參數 Λ' 與舊參數 Λ 之間的變化量以 Δ 表示，則它們之間的關係為 $\Lambda' = \Lambda + \Delta \equiv \{\lambda_1 + \delta_1, \lambda_2 + \delta_2, ..., \lambda_n + \delta_n\}$。當參數從 Λ 改變成 $\Lambda + \Delta$，對數相似值的變化量如下：

$$
\begin{aligned}
& L_{\tilde{P}}(\Lambda + \Delta) - L_{\tilde{P}}(\Lambda) \\
&= \sum_{h,w} \tilde{P}(h,w) \log P_{\Lambda+\Delta}(w \mid h) - \sum_{h,w} \tilde{P}(h,w) \log P_{\Lambda}(w \mid h) \\
&= \sum_{h,w} \tilde{P}(h,w) \log \left[\frac{1}{Z_{\Lambda+\Delta}(h)} \exp\left(\sum_i (\lambda_i + \delta_i) f_i(h,w) \right) \right] - \sum_{h,w} \tilde{P}(h,w) \log \left[\frac{1}{Z_{\Lambda}(h)} \exp\left(\sum_i \lambda_i f_i(h,w) \right) \right] \\
&= \sum_{h,w} \tilde{P}(h,w) \left[\sum_i (\lambda_i + \delta_i) f_i(h,w) + \log\left(\frac{1}{Z_{\Lambda+\Delta}(h)} \right) \right] - \sum_{h,w} \tilde{P}(h,w) \left[\sum_i \lambda_i f_i(h,w) + \log\left(\frac{1}{Z_{\Lambda}(h)} \right) \right] \\
&= \sum_{h,w} \tilde{P}(h,w) \left[\sum_i \delta_i f_i(h,w) + \log\left(\frac{1}{Z_{\Lambda+\Delta}(h)} \right) - \log\left(\frac{1}{Z_{\Lambda}(h)} \right) \right] \\
&= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) - \sum_{h,w} \tilde{P}(h,w) \log\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)} \right) \\
&= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) - \sum_{h} \tilde{P}(h) \log\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)} \right)
\end{aligned}
\tag{3.35}
$$

再來我們利用不等式 $-\log a \geq 1 - a$，此式在 $a > 0$ 時皆成立。在式(3.35)中，$Z_{\Lambda+\Delta}(h) > 0,\ Z_{\Lambda}(h) > 0 \Rightarrow \frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)} > 0$，符合此不等式的條件，所以我們可以利用它來找出對數相似值變化量的下限值（lower bound）：

$$
\begin{aligned}
&L_{\tilde{P}}(\Lambda+\Delta)-L_{\tilde{P}}(\Lambda)=\sum_{h,w}\tilde{P}(h,w)\sum_i\delta_i f_i(h,w)-\sum_h\tilde{P}(h)\log\frac{Z_{\Lambda+\Delta}(h)}{Z_\Lambda(h)}\\
&\geq\sum_{h,w}\tilde{P}(h,w)\sum_i\delta_i f_i(h,w)+\sum_h\tilde{P}(h)\left(1-\frac{Z_{\Lambda+\Delta}(h)}{Z_\Lambda(h)}\right)\\
&=\sum_{h,w}\tilde{P}(h,w)\sum_i\delta_i f_i(h,w)+1-\sum_h\tilde{P}(h)\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_\Lambda(h)}\right)\\
&=\sum_{h,w}\tilde{P}(h,w)\sum_i\delta_i f_i(h,w)+1-\sum_h\tilde{P}(h)\frac{\sum_w\exp\left(\sum_i(\lambda_i+\delta_i)f_i(h,w)\right)}{\sum_{\hat{w}}\exp\left(\sum_i\lambda_i f_i(h,\hat{w})\right)}\\
&=\sum_{h,w}\tilde{P}(h,w)\sum_i\delta_i f_i(h,w)+1-\sum_h\tilde{P}(h)\frac{\sum_w\exp\left(\sum_i\lambda_i f_i(h,w)\right)\cdot\exp\left(\sum_i\delta_i f_i(h,w)\right)}{\sum_{\hat{w}}\exp\left(\sum_i\lambda_i f_i(h,\hat{w})\right)}\\
&=\sum_{h,w}\tilde{P}(h,w)\sum_i\delta_i f_i(h,w)+1-\sum_h\tilde{P}(h)\sum_w\frac{\exp\left(\sum_i\lambda_i f_i(h,w)\right)}{\sum_{\hat{w}}\exp\left(\sum_i\lambda_i f_i(h,\hat{w})\right)}\cdot\exp\left(\sum_i\delta_i f_i(h,w)\right)\\
&=\sum_{h,w}\tilde{P}(h,w)\sum_i\delta_i f_i(h,w)+1-\sum_h\tilde{P}(h)\sum_w p_\Lambda(w\mid h)\exp\left(\sum_i\delta_i f_i(h,w)\right)
\end{aligned}
\tag{3.36}
$$

在式(3.36)中，令

$$
A(\Delta\mid\Lambda)=\sum_{h,w}\tilde{p}(h,w)\sum_i\delta_i f_i(h,w)+1-\sum_h\tilde{p}(h)\sum_w p_\Lambda(w\mid h)\exp\sum_i\delta_i f_i(h,w) \tag{3.37}
$$

於是我們得到 $L_{\tilde{p}}(\Lambda+\Delta)-L_{\tilde{p}}(\Lambda)\geq A(\Delta\mid\Lambda)$，現在問題變成要找到一個 Δ 使得 $A(\Delta\mid\Lambda)>0$，這樣就能保證新的參數 $\Lambda+\Delta$ 是有改進的。接下來再引入一個函數：

$$
f^{\#}(h,w)=\sum_{i=1}^{n}f_i(h,w) \tag{3.38}
$$

因為 $f_i(h,w)$ 是一個二元值函數（binary value function），其值不是 1 就是 0，所以 $f^{\#}(h,w)$ 代表某一個 (h,w) 配對能夠滿足的特徵（features）個數。我們將式(3.37)

改寫：

$$A(\Delta \mid \Lambda) = \sum_{h,w} \widetilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \widetilde{P}(h) \sum_w p_\Lambda(w \mid h) \exp\left(f^{\#}(h,w) \sum_i \delta_i \frac{f_i(h,w)}{f^{\#}(h,w)} \right) \tag{3.39}$$

$\frac{f_i(h,w)}{f^{\#}(h,w)}$ 對特徵函數 $f_i(h,w)$ 而言是一個機率密度函數（p.d.f），所以滿足 $\forall i,\ 0 \le \frac{f_i(h,w)}{f^{\#}(h,w)} \le 1$ 且 $\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} = 1$，因此我們可以利用 Jensen 不等式（Jensen's inequality）[1]——對於任一個機率密度函數 $P(x)$，滿足下式：

$$\exp\left(\sum_x P(x) Q(x) \right) \le \sum_x P(x) \exp(Q(x)) \tag{3.40}$$

利用式(3.40)改寫式(3.39)：

$$\begin{aligned} & A(\Delta \mid \Lambda) \\ &= \sum_{h,w} \widetilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \widetilde{P}(h) \sum_w P_\Lambda(w \mid h) \exp\left(f^{\#}(h,w) \sum_i \delta_i \frac{f_i(h,w)}{f^{\#}(h,w)} \right) \\ &= \sum_{h,w} \widetilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \widetilde{P}(h) \sum_w P_\Lambda(w \mid h) \exp\left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} \left(\delta_i f^{\#}(h,w) \right) \right) \\ &\ge \sum_{h,w} \widetilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \widetilde{P}(h) \sum_w P_\Lambda(w \mid h) \left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} \exp\left(\delta_i f^{\#}(h,w) \right) \right) \end{aligned} \tag{3.41}$$

在式(3.41)中，我們又得到一個新的下限值，令其為 $B(\Delta \mid \Lambda)$：

$$\begin{aligned} & B(\Delta \mid \Lambda) \\ &= \sum_{h,w} \widetilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \widetilde{P}(h) \sum_w P_\Lambda(w \mid h) \left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} \exp\left(\delta_i f^{\#}(h,w) \right) \right) \end{aligned} \tag{3.42}$$

[1] Jensen 不等式請參閱附錄 A。

$B(\Delta|\Lambda)$是新得到的下限值，即$L_{\tilde{p}}(\Lambda+\Delta)-L_{\tilde{p}}(\Lambda)\geq B(\Delta|\Lambda)$。接著再將式(3.42)對$\delta_i$作偏微分：

$$\frac{\partial B(\Delta|\Lambda)}{\partial \delta_i}=\sum_{h,w}\tilde{P}(h,w)f_i(h,w)-\sum_{h}\tilde{P}(h)\sum_{w}P_{\Lambda}(w|h)f_i(h,w)\exp\left(\delta_i f^{\#}(h,w)\right) \quad (3.43)$$

令式(3.43)等於 0，可解得$\Delta=\{\delta_1,...,\delta_n\}$，於是便可以求得新的參數$\Lambda'=\Lambda+\Delta\equiv\{\lambda_1+\delta_1,\lambda_2+\delta_2,...,\lambda_n+\delta_n\}$。

實作上，我們採用迭代演算法來求$\Lambda+\Delta$，稱之為 IIS（Improved Iterative scaling）演算法[2]。將透過式(3.43)所求得之新的參數$\Lambda'=\Lambda+\Delta\equiv\{\lambda_1+\delta_1,\lambda_2+\delta_2,...,\lambda_n+\delta_n\}$代入到式(3.30)，可解得新的條件機率$P_{\Lambda+\Delta}(w|h)$，再將此條件機率代入式(3.43)，反覆求得新的$\Lambda+\Delta$以及$P_{\Lambda+\Delta}(w|h)$，直到所有$\lambda_i$都收斂或是$\Delta$小於一個門檻值。圖 3.4 為 IIS 演算法。

輸入 : n個特徵 $f_1,f_2,\cdots,f_n$ 與訓練語料的機率分布 $\tilde{P}(h,w)$
輸出:最佳參數 $\Lambda=(\lambda_1,\lambda_2,\cdots,\lambda_n)$
1.所有 λ_i 的初始值設為 0
2.對每一個 λ_i 進行以下運算
  a.由下式中解得 δ_i

$$\sum_{h,w}\tilde{P}(h)P_{\Lambda}(w|h)f_i(h,w)\exp\left(\delta_i f^{\#}(h,w)\right)=\sum_{h,w}\tilde{P}(h,w)f_i(h,w)$$

  其中 $f^{\#}=\sum_{i=1}^{n}f_i(h,w)$ $P_{\Lambda}(w|h)$可由式(3.30)求得
  b.將 λ_i 的值更新：$\lambda_i=\lambda_i+\delta_i$
3.返回步驟 2直到所有的 λ_i 都收斂

圖 3.4、IIS 演算法。

[2] 實作 IIS 演算法請參閱附錄 B。

3.1.5 GIS（Generalized Iterative Scaling）演算法

除了 IIS 演算法之外，在[Darroch et al. 1972]中提出另外一個演算法來解最大熵值的問題，稱為 GIS 演算法[Ratnaparkhi 1997]，本節將簡單地介紹此演算法。在此演算法中，所求得的聯合機率函數 $P(h,w)$也是一個指數型[Rosenfeld 1996]：

$$P(h,w) \equiv \prod_i \lambda_i^{f_i(x,y)} \tag{3.44}$$

式(3.44)中，λ_i 是特徵 f_i 的參數。 $P(h,w)$初始值為：

$$P^{(0)}(h,w) = \prod_i \lambda_i^{(0)f_i(h,w)} \tag{3.45}$$

特徵 f_i 在每 k 個迭代所計算出之機率模型中的期望值為：

$$\sum_{h,w} P^{(k)}(h,w) f_i(h,w) \tag{3.46}$$

特徵 f_i 在訓練語料中的期望值為：

$$\sum_{h,w} \widetilde{P}(h,w) f_i(h,w) \tag{3.47}$$

GIS 演算法更新 λ_i 的式子如下：

$$\lambda_i^{(k+1)} = \lambda_i^{(k)} \cdot \frac{\sum_{h,w} \widetilde{p}(h,w) f_i(h,w)}{\sum_{h,w} p^{(k)}(h,w) f_i(h,w)} \tag{3.48}$$

圖 3.5 為 GIS 演算法迭代示意圖。

輸入：n個特徵 $f_1, f_2, ..., f_n$ 與訓練語料的機率分佈 $\tilde{P}(h, w)$

輸出：最佳參數 $\Lambda = (\lambda_1, \lambda_2, ..., \lambda_n)$

1. 所有 λ_i 的初始值設為 1

2. 利用下式更新每一個 λ_i

$$\lambda_i^{(k+1)} = \lambda_i^{(k)} \frac{\sum_{h,w} \tilde{P}(h,w) f_i(h,w)}{\sum_{h,w} P^{(k)}(h,w) f_i(h,w)}$$

3. 返回步驟 2 直到所有的 λ_i 都收斂

圖 3.5、GIS 演算法。

3.2 最小鑑別資訊法（Minimum Discrimination Information, MDI）

給定兩個機率分佈 $P(h,w)$ 和 $Q(h.w)$，要求它們之間的『距離』可以用 Kullback-Liebler distance [Kullback 1959]來估測：

$$D(P(h,w), Q(h,w)) \stackrel{def}{=} \sum_{h,w} P(h,w) \log \frac{P(h,w)}{Q(h,w)} \tag{3.49}$$

而最小鑑別資訊法[Della Pietra et al. 1992; Federico 1999; Chen et al. 2003]是要找到一個機率分佈 $P^*(h,w)$ 與 $Q(h.w)$ 的距離最小，如下式所示：

$$P^*(h,w) = \arg\min_{P(h,w)} D(P(h,w), Q(h,w)) \tag{3.50}$$

3.2.1 最大熵值法與最小鑑別資訊法

若式(3.50)中的$P^*(h,w)$滿足在最大熵值法中從訓練語料所蒐集的 n 個特徵，且$Q(h.w)$是平均分佈（uniform distribution）的情況下，那麼式(3.50)所求得的$P^*(h,w)$便與最大熵值法的解相同。因為平均分佈是擁有最大熵值的機率模型，即$Q(h.w)$擁有最大熵值，且$P^*(h,w)$滿足所有的特徵又與$Q(h.w)$的距離最小，所以$P^*(h,w)$是滿足所有特徵當中擁有最大熵值的機率模型，也就相當於最大熵值法的解。換句話說，最大熵值法可以視為是最小鑑別資訊法的一個特例，即當$Q(h.w)$是平均分佈的機率模型時。

3.2.2 一連語言模型限制（unigram constraints）

另外一個特殊的情況是當從語料中所蒐集到的特徵皆為一連語言模型的時候，我們拿一連語言模型特徵$f_{w_i}(h,w)$為例子，可得到下式的限制（constraint）：

$$\sum_{h,w} P(h,w) f_{w_i}(h,w) = \sum_{h,w} \widetilde{P}(h,w) f_{w_i}(h,w) = \widetilde{P}(w_i) \tag{3.51}$$

此特殊情況通常發生在語料量較少的時候，因為語料量少的緣故，使得訓練高階的 N 連語言模型較不可靠，所以只蒐集一連語言模型當作特徵。這可以應用在當我們只擁有少量的調適語料時，利用此調適語料所訓練出來的一連語言模型來調適背景語言模型。在 GIS 演算法中要計算第$(k+1)$個迭代的$P^{(k+1)}(h,w)$如下式所示：

$$P^{(k+1)}(h,w) = \prod_i \lambda_i^{(k+1) f_i(h,w)} \tag{3.52}$$

將式(3.48)代入式(3.52)中得到：

$$\begin{aligned} P^{(k+1)}(h,w) &= \prod_i \lambda_i^{(k+1)f_i(h,w)} \\ &= \prod_i \left(\lambda_i^{(k)} \cdot \frac{\sum_{h,w} \tilde{P}(h,w) f_i(h,w)}{\sum_{h,w} P^{(k)}(h,w) f_i(h,w)} \right)^{f_i(h,w)} \\ &= \prod_i \lambda_i^{(k)f_i(h,w)} \cdot \prod_i \left(\frac{\sum_{h,w} \tilde{P}(h,w) f_i(h,w)}{\sum_{h,w} P^{(k)}(h,w) f_i(h,w)} \right)^{f_i(h,w)} \\ &= P^{(k)}(h,w) \cdot \prod_i \left(\frac{\sum_{h,w} \tilde{P}(h,w) f_i(h,w)}{\sum_{h,w} P^{(k)}(h,w) f_i(h,w)} \right)^{f_i(h,w)} \end{aligned} \tag{3.53}$$

因為現在是要從調適語料中蒐集特徵來調適背景語言模型，所以 $P(h,w)$ 的初始值即為背景語言模型，$P^{(0)}(h,w) = P_B(h,w)$（B 表示 *Background*），且 $\tilde{P}(h,w) = P_A(h,w)$，（$A$ 表示 *Adaptation*）；又因為只從調適語料中蒐集一連語言模型當特徵，則對於某個特徵 f_{w_i} 可以得到：

$$\prod_i \left(\frac{\sum_{h,w} \tilde{P}(h,w) f_{w_i}(h,w)}{\sum_{h,w} P^{(k)}(h,w) f_{w_i}(h,w)} \right)^{f_{w_i}(h,w)} = \frac{\tilde{P}(w_i)}{P^{(k)}(w_i)} \tag{3.54}$$

即特徵函數為

$$f_{w_i}(h,w) = \begin{cases} 1, & w = w_i \\ 0, & \text{otherwise} \end{cases} \tag{3.55}$$

接著再假設 GIS 演算法只執行一次迭代，即將 $k=0$ 代入式(3.53)和式(3.54)中：

$$\begin{aligned} & P^{(1)}(h,w) = P^{(0)}(h,w) \cdot \frac{\tilde{P}(w)}{P^{(0)}(w)} \\ \Rightarrow\ & P^*(h,w) = P_B(h,w) \cdot \frac{P_A(w)}{P_B(w)} \end{aligned} \tag{3.56}$$

式(3.56)就是在一連語言模型限制下的最小鑑別資訊（unigram constraint MDI）

所得到的解[Federico 1999]。式(3.56)所得到的是聯合機率 $P(h,w)$，但因為語言模型的應用大部分採用條件機率，所以要做轉換，先令：

$$\alpha(w)=\frac{P_A(w)}{P_B(w)} \tag{3.57}$$

式(3.56)可以改寫成：

$$P(h,w)=P_B(h,w)\cdot\alpha(w) \tag{3.58}$$

代入貝氏定理：

$$\begin{aligned} P(w\mid h)&=\frac{P_B(h,w)\cdot\alpha(w)}{P(h)}\\ &=\frac{P_B(w\mid h)P_B(h)\alpha(w)}{\sum_{\hat{w}}P(h,\hat{w})}\\ &=\frac{P_B(w\mid h)P_B(h)\alpha(w)}{\sum_{\hat{w}}P_B(\hat{w}\mid h)P_B(h)\alpha(\hat{w})}\\ &=\frac{P_B(w\mid h)\alpha(w)}{\sum_{\hat{w}}P_B(\hat{w}\mid h)\alpha(\hat{w})} \end{aligned} \tag{3.59}$$

式(3.59)的分母部分可視為一個正規化因子（normalizing factor），使得 $\sum_{w}P(w\mid h)=1$，這使得機率模型 P 為一正確的條件機率分佈。

此外，學者 Kneser 在 [Kneser 1997] 中提到可以對 $\alpha(\bullet)$ 採用指數型平滑化：

$$\alpha(w)=\left(\frac{P_A(w)}{P_B(w)}\right)^{\gamma}\qquad 0<\gamma\le 1 \tag{3.60}$$

其中γ為最小鑑別資訊權重，則式(3.59)可以改寫成：

$$P(w\mid h)=\frac{P_B(w\mid h)(\alpha(w))^{\gamma}}{\sum_{\hat{w}}P_B(\hat{w}\mid h)(\alpha(w))^{\gamma}} \tag{3.61}$$

式(3.61)中的γ用來改變從調適語料中所蒐集到的一連語言模型特徵對背景語言

模型的調適能力，當$\gamma = 1$時式(3.61)等同於式(3.59)；當$\gamma = 0$時式(3.61)會變成$P(w \mid h) = P_B(w \mid h)$，即等於使用原本的背景語言模形而不調適，所以限定$\gamma$值的範圍為$0 < \gamma \le 1$。在本論文實驗的部分，可以看到不同的$\gamma$值對語言模型調適的影響力。

第4章 主題混合模型

在此章中，本論文先簡介資訊檢索中的主題混合模型（topic mixture model, TMM），接著再介紹如何將主題混合模型應用於語言模型調適上。

4.1 主題混合模型簡介

在一個資訊檢索的架構中，使用者輸入一個長度為 N 的查詢詞序列（query word sequence）$Q = q_1 q_2 ... q_n ... q_N$，系統會根據給定的查詢詞序列計算資料庫中每一篇文件的機率，$P(D_i \mid Q)$，經由貝氏定理可得到：

$$P(D_i \mid Q) = \frac{P(Q \mid D_i)P(D_i)}{P(Q)} \tag{4.1}$$

式(4.1)中，$P(Q \mid D_i)$是文件 D_i 產生查詢詞序列 Q 的機率，$P(D_i)$與$P(Q)$分別是文件 D_i 與查詢詞序列 Q 的事前機率 (prior probability)。但對於所有的文件來說，$P(Q)$都是相同的，將其從式(4.1)中消去並不會影響到$P(Q \mid D_i)$最後的排名；此外，進一步假設$P(D_i)$是平均分佈（uniform distribution）或者每個文件的事前機率皆相同，$P(D_1) = P(D_2) = P(D_i)$ [Miller et al. 1999; Liu et al. 2003]，這樣一來，便可以用$P(Q \mid D_i)$來近似$P(D_i \mid Q)$：

$$P(D_i \mid Q) \approx P(Q \mid D_i) \tag{4.2}$$

另外一方面，在給定文件 D_i 的條件下，假設查詢詞序列中每個詞的發生互為獨立事件，所以要計算 $P(Q \mid D_i)$可以視為文件 D_i 產生每一個查詢詞序列中的詞之機率的連乘積：

$$P(Q \mid D_i) \approx \prod_{n=1}^{N} P(q_n \mid D_i) \tag{4.3}$$

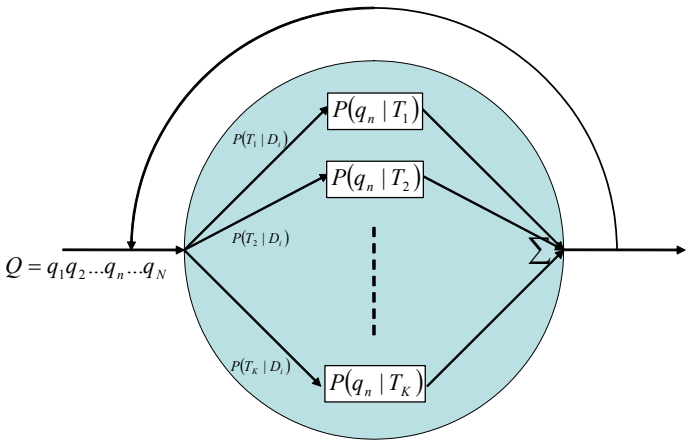


圖 4.1、文件 D_i 的主題混合模型示意圖。

在主題混合模型的研究中[Chen et al. 2004b]，每一個文件 D_i 被表示成一個混合模型，如圖 4.1 所示。在這個模型中定義了 K 個主題，每一個主題模型是由一連語言模型所表示成的主題一連語言模型（topic unigram），例如主題 k 產生詞 w_i 的機率表示成 $P(w_i \mid T_k)$；另外，這 K 個主題一連語言模型在每一個文件中皆擁有不同的權重，主題 k 在文件 D_i 中的權重表示成 $P(T_k \mid D_i)$。所以說，在某個文件 D_i 中，我們利用這 K 個主題一連語言模型以及其在 D_i 中的權重，來估測文件 D_i 產生使用者所輸入的查詢詞序列的機率，我們將 $P(w_i \mid T_k)$ 與 $P(T_k \mid D_i)$ 代入式(4.3)中：

$$P(Q \mid D_i) \approx \prod_{n=1}^{N} \sum_{k=1}^{K} P(q_n \mid T_k) P(T_k \mid D_i) \tag{4.4}$$

式(4.4)中，$P(q_n \mid T_k)$ 表示主題 k 產生查詢詞 q_n 的機率，$P(T_k \mid D_i)$ 表示主題 k 在文件 D_i 中的權重，且須滿足 $\sum_{k=1}^{K} P(T_k \mid D_i) = 1$，這也意味著文件 D_i 是可以屬於多個主題的。

總結來說，主題混合模型中的主題一連語言模型，如 $P(w_i \mid T_k)$，是由全部的文件所訓練而來，且每個文件對於 K 個主題都有其所屬的權重，如 $P(T_k \mid D_i)$。

在主題混合模型的架構下，估測文件 D_i 產生查詢詞 q_n 的機率，不再是利用 q_n 出現在文件 D_i 的次數，而是以 q_n 出現在主題 k（T_k）中的次數以及 T_k 在 D_i 中的權重來代替。因此，若有一個查詢詞序列 Q 與某個文件 D_i 可能有很高的相關性，但是查詢詞序列中的詞並沒有出現在 D_i 裡面，利用主題混合模型還是可以給予 $P(Q \mid D_i)$ 較高的值，即文件 D_i 能夠被檢索出來。如此一來，主題混合模型可以達到概念比對（concept matching）的目的，而非逐字比對（literal term matching）。

4.2 主題混合模型訓練

我們使用 K-means 演算法[Ball et al. 1967; Duda et al. 1973]將搜集到的所有文件分成 K 群，將之視為 K 個主題，則只要統計每個詞在各個主題的文件中出現的次數，即可得到主題一連語言模型的初始值。另一方面，文件產生各個主題的機率，可以用文件與各群的中心點（centroid）的距離來估測，估測此距離可採用餘弦估測法（cosine measure）。要使用餘弦估測法之前，必須先將文件 D_i 以向量的型式表示成 $\vec{D}_i$，此向量的維度與詞典的大小相同，向量中的元素，$\vec{D}_i(w)$，以下式表示[Baeza-Yates et al. 1999]：

$$\vec{D}_i(w) = (1 + \ln(c(w)))\ln(N / N_w) \tag{4.5}$$

式(4.5)中，$c(w)$表示詞 w 在文件 D_i 中出現的次數，即詞頻率（term frequency, TF），N_w 是所有的文件中，詞 w 有出現在其中的文件數，而 N 代表所有文件的數量，(N / N_w)是倒文件頻率（inverse document frequency, IDF）。詞頻率是要評估詞 w 在文件內（intra-document）權重，而倒文件頻率是要評估詞 w 在文件間（inter-document）的鑑別力，詞 w 在越多的文件中出現，即 N_w 越大，其鑑別力就越小。

經由式(4.5)的運算，我們可以將所有的文件都轉換成向量，而要計算由 K-means 演算法所求得的 K 個主題的中心點，便可以利用屬於該主題的文件之向量來計算：

$$\vec{C}_k = \frac{1}{N_{T_k}} \sum_{D_i \in T_k} \vec{D}_i \tag{4.6}$$

式(4.6)中，$\vec{C}_k$ 代表主題 k 的中心點向量，N_{T_k} 代表屬於主題 k 的文件數量。在得到 $\vec{D}_i$ 與 $\vec{C}_k$ 之後，我們便可以利用餘弦估測來計算文件 D_i 與主題 k 之間的距離（或相關性）：

$$R\left(\vec{D}_i, \vec{C}_k\right) = \frac{\vec{D}_i \cdot \vec{C}_k}{\left\|\vec{D}_i\right\| \cdot \left\|\vec{C}_k\right\|} \tag{4.7}$$

式(4.7)所計算出來的值越大，代表文件 D_i 與主題 k 的距離越近（越相關），反之則距離越遠（越不相關）。不過，我們所要求的是文件 D_i 產生主題 k 的機率，$P(T_k \mid D_i)$，這可以利用正規化因子（normalizing factor）來達成：

$$P(T_k \mid D_i) = \frac{R\left(\vec{D}_i \cdot \vec{C}_k\right)}{\sum_{r=1}^{K} R\left(\vec{D}_i \cdot \vec{C}_r\right)} \tag{4.8}$$

利用式(4.8)，便可以求得每個文件產生各個主題的機率之初始值。有了主題一連語言模型與文件產生主題機率的初始值之後，我們可以把每篇文件視為一個查詢詞序列，以非監督（unsupervised）的方式來訓練文件本身的混合模型，透過期望值最大化演算法（EM algorithm）所求得的三個式子（式(4.9)－式(4.11)）來計算混合模型的參數：

$$\hat{P}(w_n \mid T_k) = \frac{\sum_{D_i \in [D]} n(w_n, D_i) P(T_k \mid w_n, D_i)}{\sum_{D_i \in [D]} \sum_{w_s \in D_i} n(w_s, D_i) P(T_k \mid w_s, D_i)} \tag{4.9}$$

$$\hat{P}(T_k \mid D_i) = \frac{\sum_{w_s \in D_i} n(w_s, D_i) P(T_k \mid w_s, D_i)}{|D_i|} \tag{4.10}$$

$$P(T_k \mid q_n, D_i) = \frac{P(T_k \mid D_i) P(q_n \mid T_k)}{\sum_{r=1}^{K} P(T_r \mid D_i) P(q_n \mid T_r)} \tag{4.11}$$

其中$[D]$代表所有文件的集合，$n(w_n, D_i)$表示詞w_n出現在文件D_i中的次數，$|D_i|$是文件D_i的大小。

4.3 主題混合模型應用在語言模型調適

給定一詞w_i以及其歷史詞序列h_i，將詞w_i視為在主題混合模型中的查詢詞序列（query），差別在其長度只有一個詞；另外，將歷史詞序列h_i視為一個文件。這樣一來, 我們便可以根據式(4.4)來計算歷史詞序列h_i與詞w_i之間的主題混合模型機率：

$$P_{TMM}(w_i \mid h_i) = \sum_{k=1}^{K} P(w_i \mid T_k) P(T_k \mid h_i) \tag{4.12}$$

式(4.12)中的主題一連語言模型，$P(w_i \mid T_k)$，是用調適語料透過期望值最大化演算法（即式(4.9)－式(4.11)）所求得，在本論文中，我們令主題一連語言模型在辨識的過程中不會改變，即固定已訓練好的$P(w_i \mid T_k)$的值；另一方面，因為在語音辨識中，正在辨識的詞（current word）的歷史詞序列會隨著辨識過程而改變，所以各主題在歷史詞序列h_i中的權重，是採用動態計算的方式來完成。以圖 4.2 的詞圖（word graph）為例，正在辨識的詞是『隱藏』，而其歷史詞序列有兩個，$h_i^1 = (\text{SIL}, \text{肺炎}, \text{疫情})$以及$h_i^2 = (\text{SIL}, \text{飛蛾}, \text{疫情})$，我們可利用期望值最大化演算法分別為每個歷史詞序列求得其對應的$P(T_k \mid h_i)$（式(4.13)與式(4.14)）：

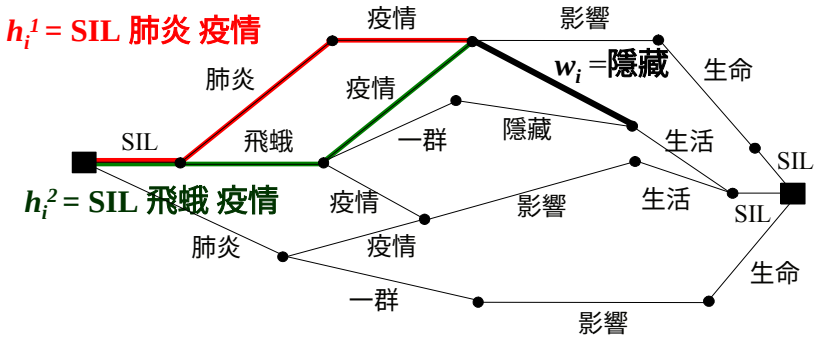


圖 4.2、詞圖示意圖。

$$P(T_k \mid h_i) = \frac{\sum_{w_s \in h_i} n(w_s, h_i) P(T_k \mid w_s, h_i)}{|h_i|} \tag{4.13}$$

$$P(T_k \mid w_n, h_i) = \frac{P(T_k \mid h_i) P(w_n \mid T_k)}{\sum_{r=1}^{K} P(T_r \mid h_i) P(w_n \mid T_r)} \tag{4.14}$$

在得到歷史詞序列 h_i 與詞 w_i 的主題混合模型機率（$P_{TMM}(w_i \mid h_i)$）之後，我們可以利用模型插補法（model interpolation）或機率調整法（probability scaling）將之與背景語言模型（如三連語言模型）結合：

$$\begin{aligned} P(w_i \mid h_i) &= \lambda P_{TMM}(w_i \mid h_i) + (1-\lambda) P_B(w_i \mid h_i) \\ &\approx \lambda P_{TMM}(w_i \mid h_i) + (1-\lambda) P_B(w_i \mid w_{i-2}, w_{i-1}) \end{aligned} \tag{4.15}$$

$$\begin{aligned} P(w_i \mid h_i) &= \frac{P_B(w_i \mid h_i) \left(\frac{P_{TMM}(w_i \mid h_i)}{P_B(w_i)} \right)^{\gamma}}{\sum_{j=1}^{|V|} \left[P_B(w_j \mid h_i) \left(\frac{P_{TMM}(w_j \mid h_i)}{P_B(w_j)} \right)^{\gamma} \right]} \\ &\approx \frac{P_B(w_i \mid w_{i-2}, w_{i-1}) \left(\frac{P_{TMM}(w_i \mid h_i)}{P_B(w_i)} \right)^{\gamma}}{\sum_{j=1}^{|V|} \left[P_B(w_j \mid w_{i-2}, w_{i-1}) \left(\frac{P_{TMM}(w_j \mid h_i)}{P_B(w_j)} \right)^{\gamma} \right]} \end{aligned} \tag{4.16}$$

式(4.15)為模型插補法，式(4.16)為機率調整法，在此兩式中，$P_{TMM}(w_i \mid h_i)$表示主題混合模型機率，$P_B(w_i \mid w_{i-2}, w_{i-1})$表示背景三連語言模型機率，$\lambda$是模型插補法的參數，$\gamma$是機率調整法的參數。

第5章 實驗介紹

本章先簡介師大廣播新聞轉寫系統[Chen et al. 2004a, 2005]以及本論文實驗所使用的語料，最候介紹各種語言模型調適的實驗結果。

5.1 師大廣播新聞轉寫系統

在本章中將介紹台灣師範大學資工所目前所發展的語音辨識系統，這是一套大詞彙連續語音辨識系統，主要包含了前端處理、聲學模型訓練（acoustic model training）、詞典建立（lexicon construction）、語言模型訓練與詞彙樹複製搜尋（tree-copy search）等部分。

5.1.1 前端處理與聲學模型訓練

本論文中使用梅爾倒頻譜特徵向量作為語音訊號的特徵參數。在求取梅爾倒頻譜特徵向量時，將語音資料切割成一連串部分重疊的音框（frame），每一個音框由13 維的梅爾倒頻譜特徵加上其一階與二階的時間軸導數（time derivatives）所形成的 39 維特徵向量所組成。其中 13 維的梅爾倒頻譜特徵是由 18 個梅爾頻譜上濾波器組（filter banks）的輸出經餘弦轉換求得。同時，為了降低通道效應對語音辨識的影響，在此使用倒頻譜平均消去法（ceptral mean subtraction, CMS）。另外，在中文的語音結構中，每個中文音節（syllable）都是由一個子音（INITIAL）與一個母音（FINAL）所組成，故基本的聲學模型由 22 個 INITIAL 模型、38 個 FINAL 模型以及一個靜音（SIL）模型所組成，不過 INITIAL 模型可在依其右邊可能接的 FINAL 模型種類細分成 112 個 INITIAL 模型[Chen et al. 2002]，最後總

共使用了 151 個隱藏式馬可夫模型（hidden Markov model, HMM）來作為這些 INITIAL-FINAL 聲學模型的統計模型。在隱藏式馬可夫模型中，每個狀態（state）則依據其對應到的聲學模型訓練語料多寡，以 2 至 128 個高斯混合（Gaussian mixture）來表示，不論男女性別皆使用同一套聲學模型。

5.1.2 詞典建立

在中文裡約有 7,000 個單字詞，新詞可由這些單字詞合併產生，本系統根據字詞在語料中的統計特性，以自動化的方式產生新的複合詞（compound words），方式如下所述：對於語料中任意相鄰的兩個詞(w_i, w_j)，分別計算它們的前二連（forward bigram）機率$P_f(w_j \mid w_i)$與後二連（backward bigram）機率$P_b(w_i \mid w_j)$，在以前後二連（forward and backward bigrams）的機率幾何平均$FB(w_i, w_j) = \sqrt{P_f(w_j \mid w_i) P_b(w_i \mid w_j)}$，作為$(w_i, w_j)$是否合併的依據。文字語料先經由一個含有一至四字詞約六萬八千個詞的原始詞典斷詞，再利用上述的機算方法，經過數次的迭代以及不同的基準閥值（threshold）設定，產生約五千個二至十字詞的複合詞，將這五千個新詞加入原始詞典中，得到一含有約七萬兩千個詞的新詞典。

5.1.3 詞彙樹複製搜尋

本系統的大詞彙連續語音辨識方法是採用由左至右（left-to-right）、音框同步（frame-synchronous）的詞彙樹複製搜尋方式[Aubert 2002]。在詞彙樹中每個分枝（arc）代表一個 INITIAL 或 FINAL 的隱藏式馬可夫模型，由根節點（root）到任一個葉節點（leaf）的路徑代表一個詞或一些發音相同的詞，路徑上的分枝

就是代表這個詞或這些詞會使用到的隱藏式馬可夫模型。具體來說，所採用的詞彙樹複製搜尋演算法，在搜尋時每個音框會同時存在數棵詞彙樹複製（tree copies），每個詞彙樹代表不同的語言模型歷史詞序列（language model history）。實際上，搜尋時產生的不完全路徑（partial path）如果擁有相同的歷史詞序列會被歸類在同一棵詞彙樹複製裡，以進行隱藏式馬可夫模型狀態層次（state-level）維特比（Viterbi）動態規劃搜尋。在每個音框中，若有不完全路徑已到達葉節點時，代表一個完整詞已可被產生；同時，不同棵詞彙樹複製間已抵達葉節點的不完全路徑，若具有相同的語言模型歷史詞序列，則會進行再結合（recombination），保留最大分數者，並以它們的歷史詞序列為標註，產生一棵新的詞彙樹複製，或加入到一棵已存在且具有相同歷史詞序列的詞彙樹複製中。值得注意的是，在實作時並不需要真的建立如此多的詞彙樹複製，僅需建立一棵詞彙樹作為搜尋時路徑展開參考之用即可，並分別記錄搜尋時存活下來的隱藏式馬可夫模型狀態節點（也就是不完全路徑目前拜訪到的節點）的相關資訊。另一方面，由於存活的隱藏式馬可夫模型狀態節點可能會隨音框數呈指數倍增加，因此採用光束搜尋（beam search）技術，適當地剪裁分數較低的狀態節點或不完全路徑。在執行剪裁動作時會同時考量每一個詞彙樹複製之內部節點（internal node）下涵蓋的可能拜訪葉節點所代表之所有詞對應的語言模型機率，並以其中最大者當作每一個詞彙樹複製內部狀態節點的語言模型前看分數（language model look-ahead score）[Aubert 2002]，再加上內部狀態節點本身搜尋時所累積的解碼分數（decoding score）及聲學模型前看分數[Chen et al. 2004a, 2005]來當成剪裁比較的依據。本系統採用一連語言模型前看（unigram language model look-ahead）技術，對每一個詞彙樹複製內部狀態節點，會以其所在分枝（或隱藏式馬可夫模型）之可能拜訪葉節點中之最大一連語言模型機率，作為該內部狀態節點的語言模型前看分數。此外，每個音框會記錄存活的詞彙樹複製葉節點中分數較高者的相關資訊（這些葉節點本身代表著可能的候選詞），諸如它們的語言模型歷史詞序列、候選詞所對應的開始與結束的音框以及搜尋時聲學模型解碼

的分數，然後再依此資訊建立詞圖（word graph），如圖 4.2，並在詞圖上使用更高階的語言模型，重新進行一次詞圖動態規劃搜尋（word graph rescoring），找出最佳的辨識詞序列。在本系統中，詞彙樹複製搜尋階段是使用二連語言模型，而在詞圖搜尋階段是使用三連語言模型。

5.2 實驗語料

本篇論文的實驗語料包含兩個類型，一個是廣播新聞語料（Set 1），另一個是公視新聞語料（Set 2）。

Set 1：廣播新聞語料是 1998 年到 2002 年之間在台北錄製的廣播新聞，包含了 112 個小時的語料[Chen et al. 2004a]。這中間的 7.7 個小時的語料經過人工轉寫，其中 4 個小時是在 1998 年至 1999 年之間錄製的，被拿來當作聲學模型訓練語料，而另外的 3.7 個小時是在 2002 年 9 月錄製的，拿來當作測試語料。

Set 2：公視新聞語料是採用 MATBN 新聞語料[Wang et al. 2005]，此語料是由中央研究院資訊所口語小組[SLG]與公共電視台[PTS]，在 2001 年到 2003 年之間合作錄製的；其中 2001 年包含了 30 小時，2002 年包含了 146 小時，2003 年包含了 24 小時，總共是 200 小時的新聞語料。這 200 小時的語料包含了 28,913 個句子（sub-term），每個句子都經過人工轉寫且標註了額外的資訊，如各種背景雜訊、語者性別、停頓、呼吸、語助詞等。

MATBN 新聞語料包含了內場以及外場兩個部分，內場為主播（Studio Anchors）語料，外場又可分成採訪記者（Field Reporters）與受訪者（Interviewees）語料。但是由於內場主播語料大部分為同一個主播所錄製的因素，為了避免語料的偏差性讓實驗偏向語者相依（speaker dependent），故不採用內場主播語料；又發現外場的受訪者語料，包含了太多的語助詞，故此篇論文的實驗初步採用外場

記者語料。我們從外場記者語料中挑選了約 27 個小時的語料，其中 25.5 個小時（包含了 5,774 個句子）拿來當作聲學模型訓練語料，1.5 個小時（包含了 292 個句子）拿來當作測試語料。聲學模型訓練語料是從 2001 年及 2002 年挑選出來的，其中男女生的語料各半，且都為不包含語助詞（particles）的語料。測試語料是由中央研究院所選定的 MATBN Evaluation Set 中，挑選出採訪記者語料並過濾掉含有語助詞的句子而來，此測試語料涵蓋的時間是 2003 年的。語料分佈資訊如表 5.1 所示：

	訓練語料長度（分）	測試語料長度（分）
男生	766.68	21.69
女生	766.78	65.23

表 5.1、Set 2 聲學模型訓練語料和測試語料分佈資訊。

Set 1 之背景語言模型（background language model）訓練語料蒐集自中央通訊社[CNA news]，蒐集期間包含了 2000 年以及 2001 年，大約包含一億七千萬（170M）個中文字（characters）。而調適語料也蒐集自中央通訊社，蒐集期間為 2002 年 8 月到 10 月，大約包含五千萬（50M）個中文字，此調適語料的搜集時間涵蓋了測試語料的搜集時間（2002 年 9 月）。

Set 2 之背景語言模型的訓練語料也是蒐集自中央通訊社，蒐集期間包含了 2001 年以及 2002 年，大約包含一億五千萬（150M）個中文字。調適語料選自 MATBN 新聞語料，時間為 2001 年以及 2002 年，共包含 3,528 則新聞，大約包含一百九十三萬（1.9M）個中文字。

背景語言模型為一三連語言模型（trigram language model），為彌補資料缺乏（data sparseness）的問題，Set 1 與 Set 2 皆採用 Katz 語言模型平滑技術[Katz 1987]，且採用 cutoff 值等於 3，即 N 連詞頻數要大於等於 3 次的 N 連詞才會被

訓練。本論文的語言模型訓練是使用 SRI Language Modeling Toolkit （SRILM）[SRILM]。

5.3 基礎實驗

在 Set 1 中，背景語言模型訓練語料蒐集自中央通訊社 2000 年以及 2001 年的新聞語料，透過 Katz 語言模型平滑技術來訓練背景語言模型，其字錯誤率（character error rate, CER）以及語言模型複雜度（Perplexity）如表 5.2 所示：

	字錯誤率	複雜度
Baseline	15.22%	752.49

表 5.2、Set 1 背景語言模型基礎實驗之字錯誤率以及語言模型複雜度。

在 Set 2 中，背景語言模型訓練語料蒐集自中央通訊社 2001 年以及 2002 年的新聞語料，採用 Katz 語言模型平滑技術來訓練背景語言模型，其字錯誤率以及語言模型複雜度如表 5.3 所示：

	字錯誤率	複雜度
Baseline	25.72%	667.23

表 5.3、Set 2 背景語言模型基礎實驗之字錯誤率以及語言模型複雜度。

5.4 語言模型調適實驗

在本節中，本論文將討論各種語言模型調適法的實驗結果。首先將前後文資訊（contexture informaion）應用在以最大事後機率法為基礎的調適方法中，包含了詞頻數合併法、模型插補法以及動態快取模型法；接著將潛藏語意分析資訊以及主題混合模型資訊應用到語言模型調適中；再來討論最小鑑別資訊法；最後討論合併其中數個方法的結果。

5.4.1 詞頻數合併法

詞頻數混合法的式子如下所示：

$$P(w_i \mid h_i) = \frac{\alpha C_B(h_i w_i) + \beta C_A(h_i w_i)}{\alpha C_B(h_i) + \beta C_A(h_i)} \tag{5.1}$$

在本論文中皆將 α 設定為 1，來改變 β 的值，即拿一倍的訓練語料混合不同倍數的調適語料來訓練語言模型。

在 Set 1 中，調適語料蒐集自中央通訊社 2002 年 8 月到 10 月之間的新聞語料，字錯誤率以及語言模型複雜度如表 5.4 所示：

	字錯誤率	複雜度
β = 1	13.70% (9.99%)	458.79 (39.03%)

表 5.4、Set 1 詞頻數混合法之字錯誤率以及語言模型複雜度，括號中數值代表相對改進量。

在 Set 1 中，詞頻數混合法的字錯誤率達到 13.70%，相對改進 9.99%，語言模型複雜度達到 458.79，相對改進 39.03%。

在 Set 2 中，調適語料蒐集自 MATBN 新聞語料中 2001 年以及 2002 年，在實驗中發現，當 β 設成 1 的時候，語言模型複雜度會比基礎實驗來的差，這可能與在訓練語言模型時，採用 cutoff 值為 3 的關係。因為就本質上來說，Set 2 之調適語料的大小不大（153 萬個中文詞），造成 *N* 連詞頻數很小的情況，cutoff 值為 3 可能會將許多有用的資訊給刪除，所以再比較將 β 設成 3 的結果，便得到較好的效果，字錯誤率以及語言模型複雜度如表 5.5 所示：

	字錯誤率	複雜度
β = 1	25.20% (2.02%)	675.46 (-1.23%)
β = 3	24.98% (2.88%)	634.43 (4.92%)

表 5.5、Set 2 詞頻數混合法之字錯誤率以及語言模型複雜度，括號中數值代表相對改進量。

在 Set 2 中，當 β 設成 3 時，詞頻數混合法的字錯誤率達到 24.98%，相對改進 2.88%，語言模型複雜度達到 634.43，相對改進 4.92%。

相較於 Set 2，在 Set 1 中因為其調適語料較為豐富（約 5000 萬的中文詞），所以將 β 設成 1 便有不錯的效果。

5.4.2 線性模型插補法

線性模型插補法的式子如下所示：

$$P(w_i \mid h_i) = (1-\lambda)P_A(w_i \mid h_i) + \lambda P_B(w_i \mid h_i) \tag{5.2}$$

在 Set 1 中，由調適語料所訓練的語言模型（$P_A(w_i \mid h_i)$）也是採用 Katz 語言模型平滑技術，cutoff 值設定為 3 所訓練的三連語言模型。式(5.2)中將 λ 設成 0.5 時的字錯誤率以及語言模型複雜度如表 5.6 所示：

	字錯誤率	複雜度
$\lambda = 0.5$	13.74% (9.72%)	430.59 (42.78%)

表 5.6、Set 1 模型插補法之字錯誤率以及語言模型複雜度，括號中數值代表相對改進量。

在 Set 1 中，模型插補法的字錯誤率降低到 13.74%，相對改進 9.72%，語言模型複雜度為 430.59，相對改進 42.78%。

在 Set 2 中，因為調適語料較少的關係，所以本論文將 cutoff 值設成 1，即保留所有出現的 N 連詞，並採用 Katz 語言模型平滑技術訓練三連語言模型。此外，在 Set 2 中我們另外比較了不同 λ 值對字錯誤率以及語言模型複雜度的影響，如表 5.7 與圖 5.1 所示。我們可以觀察到，當 λ 值為 0.7 時，可以得到最低的字錯誤率 24.11%，相對改進 6.28%；當 λ 值為 0.6 時，可以得到最低的語言模型複雜度 430.26，相對改進 35.52%。

λ	字錯誤率	複雜度
0.1	25.44%	560.35
0.2	24.81%	496.58
0.3	24.62%	463.31
0.4	24.37%	444.06
0.5	24.33%	433.67
0.6	24.24%	**430.26 (35.52%)**
0.7	**24.11% (6.26%)**	433.71
0.8	24.3%	445.79
0.9	24.54%	473.27

表 5.7、Set 2 模型插補法之字錯誤率以及語言模型複雜度，括號中數值代表相對改進量。

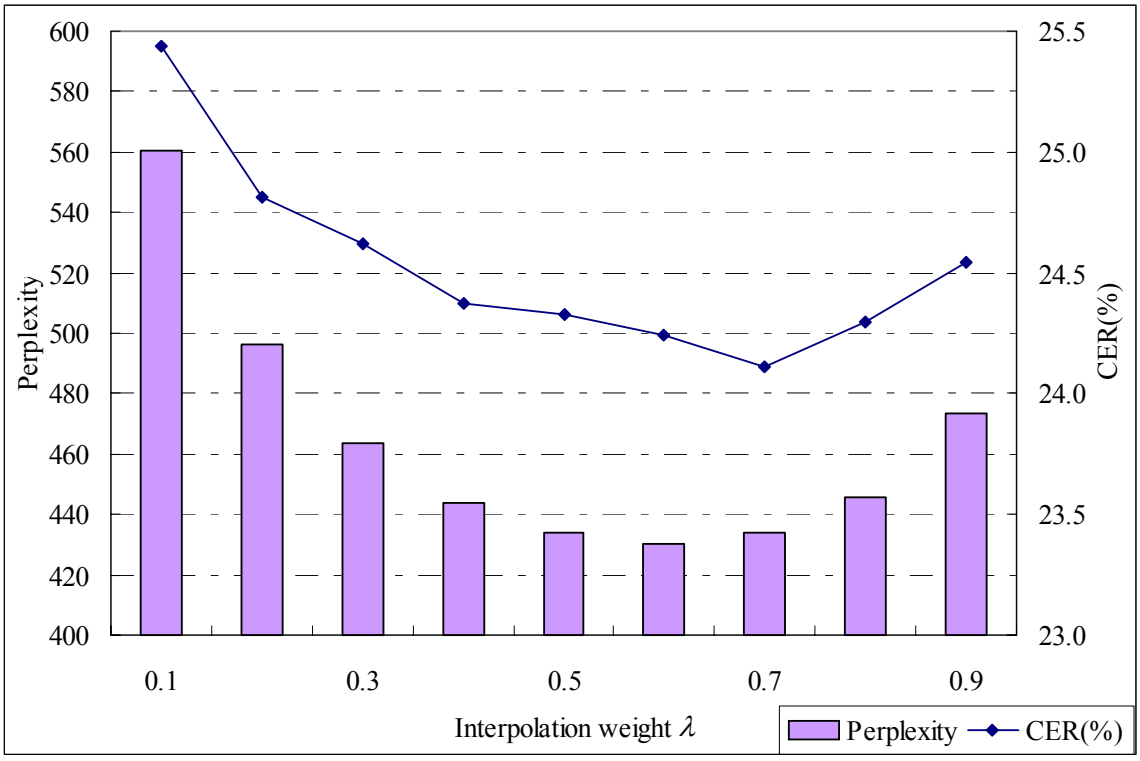


圖 5.1、Set 2 模型插補法在不同 λ 值之字錯誤率以及語言模型複雜度。

5.4.3 動態快取模型法

動態快取模型法的式子如下所示：

$$P(w_i \mid h_i) = \lambda P_B(w_i \mid h_i) + (1-\lambda) P_{chche}(w_i) \tag{5.3}$$

在動態快取模型法的實驗中，由於估測快取一連語言模型的語料是由辨識的結果蒐集而來的，且由較多的語料所訓練出來的快取一連語言模型也較可靠的關係，所以本論文將式(5.3)中的 λ 值設定成會隨著辨識過程而改變，即根據已辨識完成的詞數量，也就是估測快取一連語言模型的語料量，共分成四個階段：0　9 個詞、10　19 個詞、20　49 個詞以及 50 個詞以上。

動態快取模型法的實驗初步只實作在 Set 2 的語料上。經由實驗，我們得到將 λ 值依照表 5.8 的設定，可以得到最低的語言模型複雜度 572.68，相對改進 14.17%：

快取模型訓練語料量	λ
0　9 個詞	0.00
10　19 個詞	0.03
20　49 個詞	0.07
50 個詞以上	0.11

表 5.8、Set 2 動態快取模型法對不同的快取模型訓練語料量（即辨識的結果）採用不同的 λ 值。

在字錯誤率的實驗上，本論文除了表 5.8 的設定之外，也另外比較了其他的 λ 值設定，如表 5.9 所示：

0　9 詞 λ值	10　19 詞 λ值	20　49 詞 λ值	50 詞以上 λ值	字錯誤率	複雜度
0.00	0.03	0.07	0.11	25.94%	**572.68 (14.17%)**
0.00	0.01	0.05	0.10	25.72%	573.90
0.00	0.05	0.05	0.05	25.74%	576.84
0.00	0.04	0.04	0.04	25.65%	578.91
0.00	0.03	0.03	0.03	25.54%	582.50
0.00	0.02	0.02	0.02	25.44%	588.69
0.00	0.01	0.01	0.01	25.42%	600.61
0.00	0.001	0.001	0.001	**25.41% (1.21%)**	636.17

表 5.9、Set 2 動態快取模型法在不同 λ 值的字錯誤率與語言模型複雜度比較，括號中數值代表相對改進量。

由表 5.9 中可以發現，在動態快取模型法中，擁有較低語言模型複雜度的 λ 值並非擁有較佳的辨識效果。這可能是因為此方法的調適語料來源是之前的辨識結果，然而在計算語言模型複雜度的時候，是以正確的轉寫來進行估測，所以調適語料所包含的詞都是正確的，訓練出來的快取一連語言模型也較正確，而且調適語料越多，λ 值可以設的越大；但是在辨識時所蒐集到的調適語料會因為語音辨識錯誤的緣故而蒐集到錯誤的語料，換句話說，之前的辨識錯誤會連帶影響到之後的辨識結果，所以 λ 都維持在很小的值，才有好的辨識效果。

5.4.4 潛藏語意分析法

在潛藏式語意分析的實驗中，我們將 Set 1 與 Set 2 所蒐集到的調適語料，依照 2.2.2 節中所介紹的方法，製作詞與文件矩陣 W，再設定分解維度為 100（$R=100$）對矩陣 W 進行奇異值分解，便可計算潛藏語意分析機率，$P_{LSA}\left(w_i \mid \tilde{d}_{i-1}\right)$，將之與背景三連語言模型機率，$P_B\left(w_i \mid w_{i-2}, w_{i-1}\right)$，經由式(5.4)合併，並設$\gamma$值為 0.1，其字錯誤率以及語言模型複雜度如表 5.10 所示。

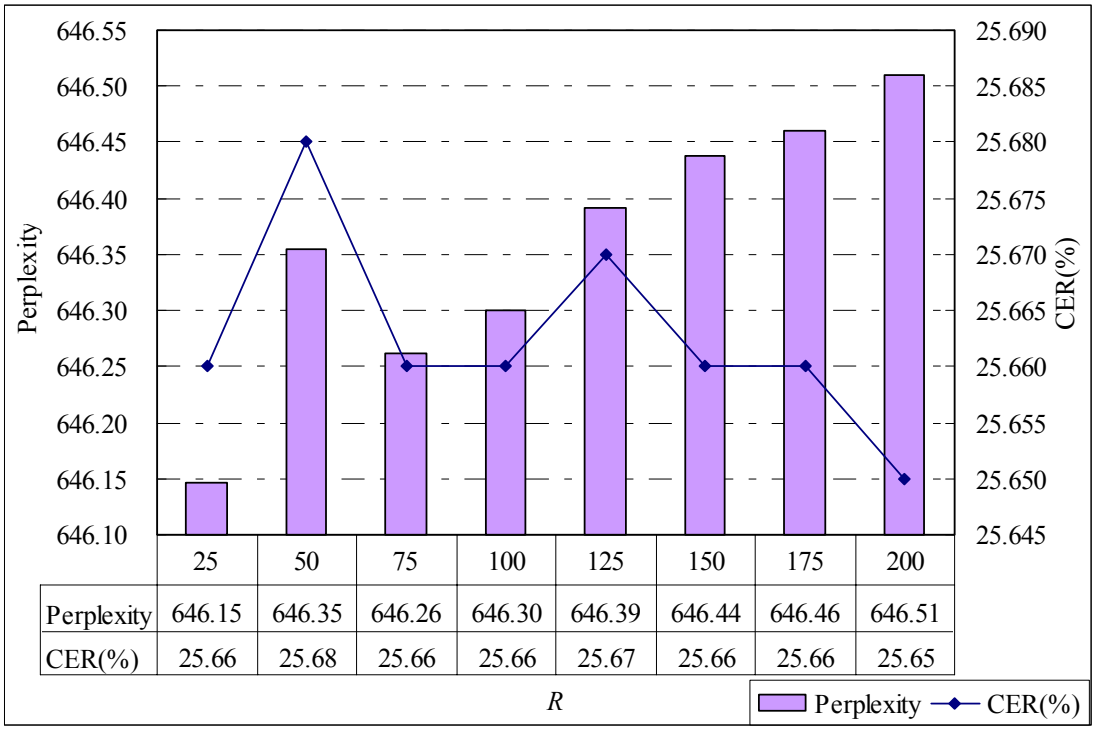
$$P\left(w_i \mid h_i\right)=\frac{P_B\left(w_i \mid w_{i-2}, w_{i-1}\right)\cdot\left(\frac{P_{LSA}\left(w_i \mid \tilde{d}_{i-1}\right)}{P_B\left(w_i\right)}\right)^{\gamma}}{\sum_{w} P_B\left(w \mid w_{i-2}, w_{i-1}\right)\cdot\left(\frac{P_{LSA}\left(w \mid \tilde{d}_{i-1}\right)}{P_B\left(w\right)}\right)^{\gamma}} \tag{5.4}$$

	字錯誤率	複雜度
Set 1	15.10% (0.79%)	531.95 (29.31%)
Set 2	25.66% (0.23%)	646.30 (3.14%)

表 5.10、加入潛藏式語意分析資訊之字錯誤率以及複雜度，括號中數值代表相對改進量。

由表 5.10 中可以觀察到，Set 1 的語言模型複雜度有明顯改進（相對改進 29.31%），但是在字錯誤率上只相對改進了約 0.8%；然而 Set 2 在維度為 100 時的改進並沒有十分顯著，字錯誤率相對改進 0.23%，語言模型複雜度相對改進 3.14%，所以我們對 Set 2 再分析其他的分解維度對字錯誤率以及語言模型複雜

度的影響，如圖 5.2 所示。



	25	50	75	100	125	150	175	200
Perplexity	646.15	646.35	646.26	646.30	646.39	646.44	646.46	646.51
CER(%)	25.66	25.68	25.66	25.66	25.67	25.66	25.66	25.65

圖 5.2、Set 2 在不同維度之潛藏語意分析比較。

在圖 5.2 中我們發現不同的分解維度之間並沒有顯著的差異性，不過在各個維度下，字錯誤率與語言模型複雜度都較基礎實驗有小幅度的改進。根據推測，可能是因為調適語料與測試語料的蒐集時間並沒有重疊，也就是它們並非同時期的語料，所以擷取自調適語料中的語意資訊，沒有辦法提供太大的幫助來改進辨識結果與複雜度。在維度為 25 的時候有最低的語言模型複雜度 646.15，相對改進 3.16%，維度為 200 的時候有最低的字錯誤率 25.65%，相對改進 0.27%。

5.4.5 主題混合模型

主題混合模型機率，$P_{TMM}(w_i \mid h_i)$，的式子如下：

$$P_{TMM}(w_i \mid h_i) = \sum_{k=1}^{K} P(w_i \mid T_k) P(T_k \mid h_i) \tag{5.5}$$

將之與背景三連語言模型（$P_B(w_i \mid w_{i-2}, w_{i-1})$）利用模型插補法，如式(5.6)，以及機率調整法，如式(5.7)，來合併：

$$P(w_i \mid h_i) = \lambda P_{TMM}(w_i \mid h_i) + (1-\lambda) P_B(w_i \mid w_{i-2}, w_{i-1}) \tag{5.6}$$

$$P(w_i \mid h_i) = \frac{P_B(w_i \mid w_{i-2}, w_{i-1}) \left(\frac{P_{TMM}(w_i \mid h_i)}{P_B(w_i)} \right)^{\gamma}}{\sum_{j=1}^{|V|} \left[P_B(w_j \mid w_{i-2}, w_{i-1}) \left(\frac{P_{TMM}(w_j \mid h_i)}{P_B(w_j)} \right)^{\gamma} \right]} \tag{5.7}$$

在本論文中，我們比較了不同的主題數（式(5.5)中的 K 值），再利用模型插補法與機率調整法合併不同主題數之主題混合模型與背景三連語言模型。在 Set 1 上的字錯誤率與語言模型複雜度，分別以表 5.11 與表 5.12 表示：

主題數（K）	字錯誤率	複雜度
16	14.83%	589.32
32	14.73%	572.25
64	14.58%	553.56
128	14.53%	523.14
256	**14.47% (4.93%)**	**510.90 (32.11%)**

表 5.11、Set 1 中利用模型插補法合併主題混合模型與背景三連語言模型之字錯誤率與語言模型複雜度，括號中數值代表相對改進量。

主題數（K）	字錯誤率	複雜度
16	14.41%	514.76
32	14.23%	495.64
64	14.02%	490.67
128	13.89%	**478.10 (36.46%)**
256	**13.87% (8.87%)**	493.26

表 5.12、Set 1 中利用機率調整法合併主題混合模型與背景三連語言模型之字錯誤率與語言模型複雜度，括號中數值代表相對改進量。

由表 5.11 及表 5.12 可以觀察到，主題數越多可以獲得較佳的效果。在以模型插補法合併的實驗中，當主題數為 256 時有最低的字錯誤率 14.47%，相對改進 4.93%，以及最低的語言模型複雜度 510.90，相對改進，相對改進 32.11%；在以機率調整法合併的實驗中，當主題數為 256 時有最低的字錯誤率 13.87%，相對改進 8.87%，當主題數為 128 時有最低的語言模型複雜度 478.10，相對改進 36.46%。

此外，我們比較在 Set 1 上使用主題混合模型與潛藏語意分析來調適背景語言模型的效果, 主題混合模型無論使用模型插補法或機率調整法來與背景語言模型合併，在字錯誤率以及語言模型複雜度的改進量都比潛藏語意分析法來的顯著。另外，在 Set 2 上，由於其調適語料與測試語料為非同時期的語料，且在潛藏式語意分析的實驗中，Set 2 的字錯誤率以及語言模型複雜度並沒有顯著的改善，所以本論文初步只將主題混合模型應用在 Set 1 上，在未來將再研究如何從非同時期的調適語料中擷取有用的資訊。

5.4.6 最小鑑別資訊法

在最小鑑別資訊法中，本論文採用一連語言模型限制中所得到的調適方法，如式(5.8)所示：

$$P(w \mid h) = \frac{P_B(w \mid h)\alpha(w)}{\sum_{\hat{w}} P_B(\hat{w} \mid h)\alpha(\hat{w})} \tag{5.8}$$

其中 $\alpha(w) = \frac{P_A(w)}{P_B(w)}$，$P_B(\ )$為背景語言模型，$P_A(w)$則為調適語料中所訓練的一連語言模型。利用此方法的字錯誤率以及語言模型複雜度如表 5.13 所示：

	字錯誤率	複雜度
Set 1	15.12% (0.66%)	590.37 (21.54%)
Set 2	25.58% (0.54%)	592.93 (11.14%)

表 5.13、最小鑑別資訊法之字錯誤率與語言模型複雜度，括號中數值代表相對改進量。

此外，本論文對 Set 2 又比較了不同的最小鑑別資訊權重（即式(5.9)中的γ）對字錯誤率以及語言模型複雜度的影響，如圖 5.3 與表 5.14 所示：

$$\alpha(w)^{\gamma} = \left(\frac{P_A(w)}{P_B(w)}\right)^{\gamma} \tag{5.9}$$

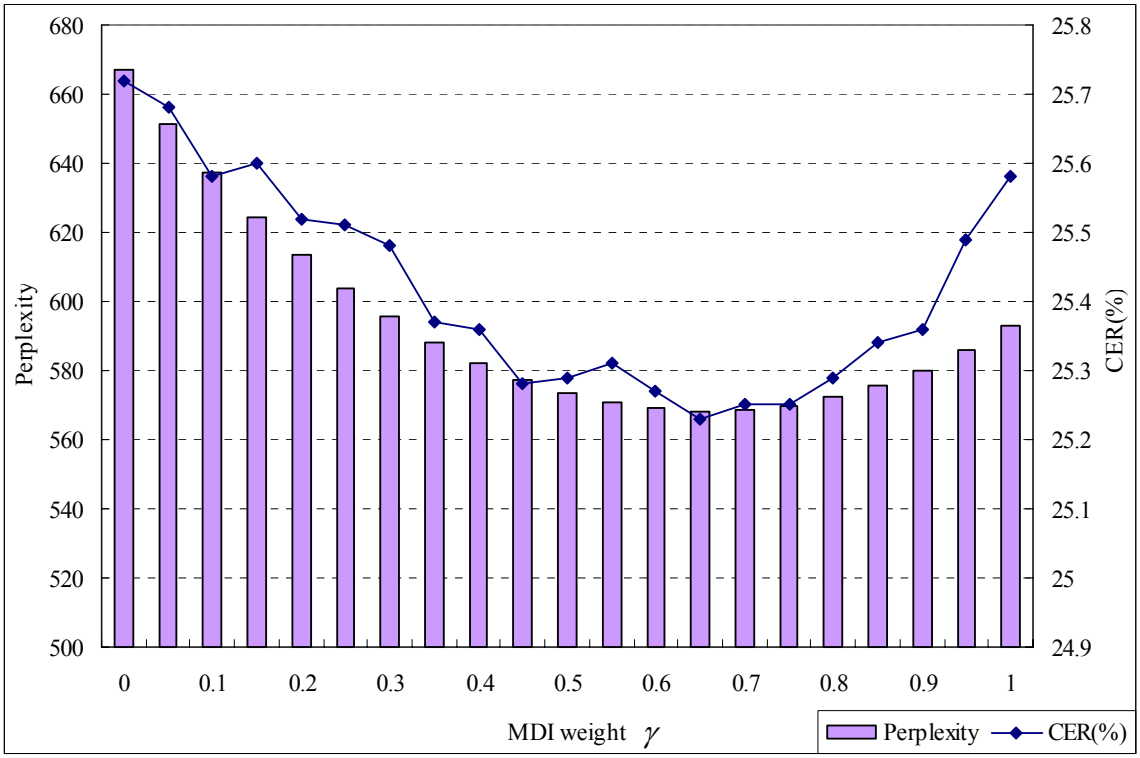


圖 5.3、Set 2 在不同的最小鑑別資訊權重（MDI weight, γ）下字錯誤率與語言模型複雜度的變化。

γ	字錯誤率	複雜度	γ	字錯誤率	複雜度
0.05	25.68%	651.25	0.55	25.31%	570.82
0.10	25.58%	637.06	0.60	25.27%	569.07
0.15	25.60%	624.51	0.65	**25.23%** **(1.91%)**	**568.33** **(14.82%)**
0.20	25.52%	613.46	0.70	25.25%	568.61
0.25	25.51%	603.80	0.75	25.25%	569.92
0.30	25.48%	595.43	0.80	25.29%	572.28
0.35	25.37%	588.28	0.85	25.34%	575.71
0.40	25.36%	582.28	0.90	25.36%	580.25
0.45	25.28%	577.40	0.95	25.49%	585.97
0.50	25.29%	573.59	1.00	25.58%	592.93

表 5.14、Set 2 在不同的最小鑑別資訊權重（γ）下之字錯誤率與語言模型複雜度，括號中數值代表相對改進量。

在圖 5.3 中，最小鑑別資訊權重為 0 的數據即為基礎實驗所得到的數據，我們可以發現，在不同的最小鑑別資訊權重下，最小鑑別資訊法都比基礎實驗來的好，當權重為 0.65 時，有最低的字錯誤率 25.23%，相對改進 1.91%，以及最低的語言模型複雜度 568.33，相對改進 14.82%。

5.4.7 主題混合模型與詞頻數合併法結合

結合主題混合模型與詞頻數合併法可以依據式(5.6)與式(5.7)，將兩式中的 $P_B(w_i \mid w_{i-2}, w_{i-1})$經由式(5.1)稍作變更即可：

$$\hat{P}_B(w_i \mid w_{i-2}, w_{i-1}) = \frac{\alpha C_B(w_{i-2}, w_{i-1}, w_i) + \beta C_A(w_{i-2}, w_{i-1}, w_i)}{\alpha C_B(w_{i-2}, w_{i-1}) + \beta C_A(w_{i-2}, w_{i-1})} \tag{5.10}$$

式(5.10)為詞頻數混合法的式子。換句話說，先將訓練語料與調適語料合併後訓練出語言模型 $\hat{P}_B$，再與 $P_{TMM}(w_i \mid h_i)$合併，合併方法也是有模型插補法與機率調整法兩種，其字錯誤率與語言模型複雜度如表 5.15 所示：

合併方法	字錯誤率	複雜度
模型插補法	13.27% (12.81%)	337.31 (55.17%)
機率調整法	13.03% (14.39%)	349.12 (53.60%)

表 5.15、主題混合模型與詞頻數合併法結合。

由表 5.15 觀察到主題混合模型與詞頻數合併法結合後的效果比單獨使用其中一種都要來的好，代表其資訊是有相輔相成關係的。

5.4.8 主題混合模型與模型插補法結合

要將主題混合模型結合模型插補法就和結合詞頻數合併法一樣，將式(5.6)與式(5.7)兩式中的 $P_B(w_i \mid w_{i-2}, w_{i-1})$ 經由式(5.2)稍作修改即可：

$$\hat{P}_B(w_i \mid w_{i-2}, w_{i-1}) = (1-\lambda)P_A(w_i \mid w_{i-2}, w_{i-1}) + \lambda P_B(w_i \mid w_{i-2}, w_{i-1}) \tag{5.11}$$

也就是說，先將背景語言模型與調適語料訓練的語言模型插補後，再和主題混合模型合併，其字錯誤率與語言模型複雜度如表 5.16 所示：

合併方法	字錯誤率	複雜度
模型插補法	13.37% (12.16%)	338.96 (54.95%)
機率調整法	13.08% (14.06%)	346.36 (53.97%)

表 5.16、主題混合模型與模型插補法結合。

5.5 最大熵值法應用於語言模型上

本論文採用 IIS 演算法來求最大熵值法的解，在 IIS 演算法中，對於每個特徵 $f_i(\hat{h}, \hat{w})$ 都要解以下的式子：

$$\sum_{h,w} \tilde{P}(h) P_\Lambda(w \mid h) f_i(h,w) \exp\left(\delta_i f^{\#}(h,w)\right) = \sum_{h,w} \tilde{P}(h,w) f_i(h,w) \tag{5.12}$$

等號右邊可以直接從訓練語料來統計，且只需計算一次即可，但是等號左邊潛在

的計算複雜度為 $O\left(|V|^{l+1}\right)$，l 為歷史詞序列 h 的長度。不過因為特徵函數 $f_i\left(\hat{h},\hat{w}\right)$ 中有限定 w 要等於 $\hat{w}$，所以可以將等號左邊改寫：

$$\begin{aligned}&\sum_{h,w}\widetilde{P}(h)P_\Lambda(w\mid h)f_i(h,w)\exp\left(\delta_i f^{\#}(h,w)\right)\\&=\sum_{h}\widetilde{P}(h)P_\Lambda(\hat{w}\mid h)f_i(h,\hat{w})\exp\left(\delta_i f^{\#}(h,\hat{w})\right)\end{aligned}\tag{5.13}$$

此外，$\widetilde{P}(h)$ 為 h 在訓練語料中的機率，如果 h 沒有出現在訓練語料中，則 $\widetilde{P}(h)=0$，所以可再將式(5.13)改寫：

$$\begin{aligned}&\sum_{h,w}\widetilde{P}(h)P_\Lambda(\hat{w}\mid h)f_i(h,\hat{w})\exp\left(\delta_i f^{\#}(h,\hat{w})\right)\\&=\sum_{h\in\text{訓練語料}}\widetilde{P}(h)P_\Lambda(\hat{w}\mid h)f_i(h,\hat{w})\exp\left(\delta_i f^{\#}(h,\hat{w})\right)\end{aligned}\tag{5.14}$$

透過式(5.14)，對每一個特徵而言，只要計算訓練語料中有出現的歷史詞序列即可，但是這依然是一個很龐大的數量。假設我們要使用最大熵值法來合併一連語言模型、二連語言模型與三連語言模型，則歷史詞序列最少要看兩個詞，也就是歷史詞序列的長度為 2，以 Set 2 的訓練語料為例，這樣的歷史詞序列總共有 5,173,627 個，意味著每一個特徵都要應過這麼多次的計算再加總，這使得 IIS 演算法執行的速度非常的緩慢。故本論文在最大熵值法的實驗部分，初步拿 Set 2 的調適語料來當作另外一組實驗的訓練語料，稱之為 Set 3，Set 3 的測試語料和 Set 2 相同。Set 3 的基礎實驗也是將訓練語料透過 Katz 模型平滑化技術所訓練的三連語言模型，其字錯誤率與語言模型複雜度如表 5.17 所示：

	字錯誤率	複雜度
Baseline	36.72%	1208.50

表 5.17、Set 3 之基礎實驗數據。

經過統計分析，Set 3 長度為 2 的歷史詞序列共約有 55 萬個，另外，我們挑選三連詞頻與二連詞頻大於等於 3 次的三連詞與二連詞以及所有出現的一連詞當作特徵，其統計量如表 5.18 所示：

特徵	個數
一連詞	21,931
二連詞	65,245
三連詞	30,440

表 5.18、最大熵值法中特徵的分佈。

我們把這三種特徵透過 IIS 演算法將其整合成一個新個模型，並用此模型來估測字錯誤率以及語言模型複雜度，圖 5.4 為每一次迭代（iteration）所產生的新模型之語言模型複雜度與基礎實驗的比較圖；表 5.19 與圖 5.5 為字錯誤率與語言模型複雜度之數據及圖示。

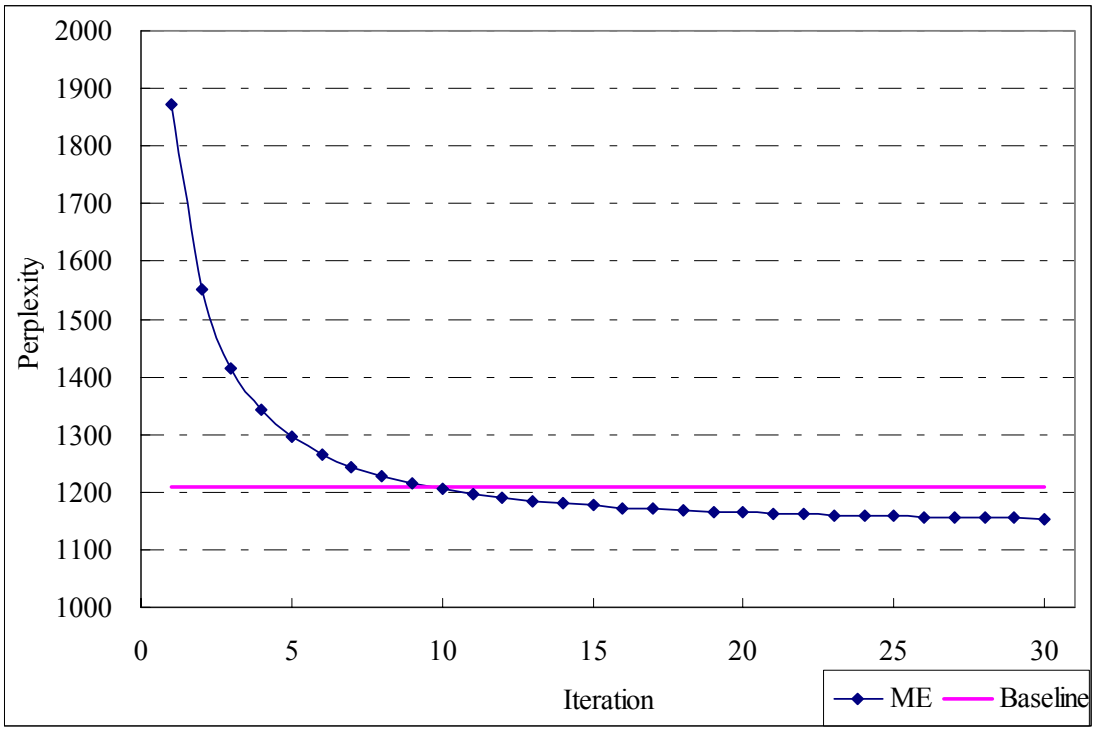


圖 5.4、最大熵值法不同迭代次數之語言模型複雜度與基礎實驗比較圖。

從圖 5.4 可以觀察到，以最大熵值法將一連詞、二連詞與三連詞合併所訓練出的語言模型，比用傳統最大相似度估測法（maximum likelihood）所訓練的語言模型擁有較低的複雜度。

迭代次數	字錯誤率	複雜度	迭代次數	字錯誤率	複雜度
1	38.60	1871.70	16	36.57	1172.82
2	38.02	1550.10	17	36.53	1170.02
3	37.55	1415.57	18	36.55	1167.61
4	37.39	1342.40	19	36.5	1165.51
5	37.10	1296.82	20	36.48	1163.70
6	37.01	1266.00	21	36.49	1162.10
7	36.90	1243.90	22	36.49	1160.70
8	36.83	1227.38	23	36.45	1159.49
9	36.79	1214.66	24	36.45	1158.43
10	36.70	1204.63	25	36.43	1157.49
11	36.69	1196.56	26	36.41	1156.65
12	36.63	1189.96	27	36.42	1155.91
13	36.58	1184.51	28	36.38	1155.26
14	36.52	1179.95	29	36.32	1154.68
15	36.52	1176.09	30	36.29	1154.19

表 5.19、最大熵值法各迭代次數之字錯誤率與語言模型複雜度。

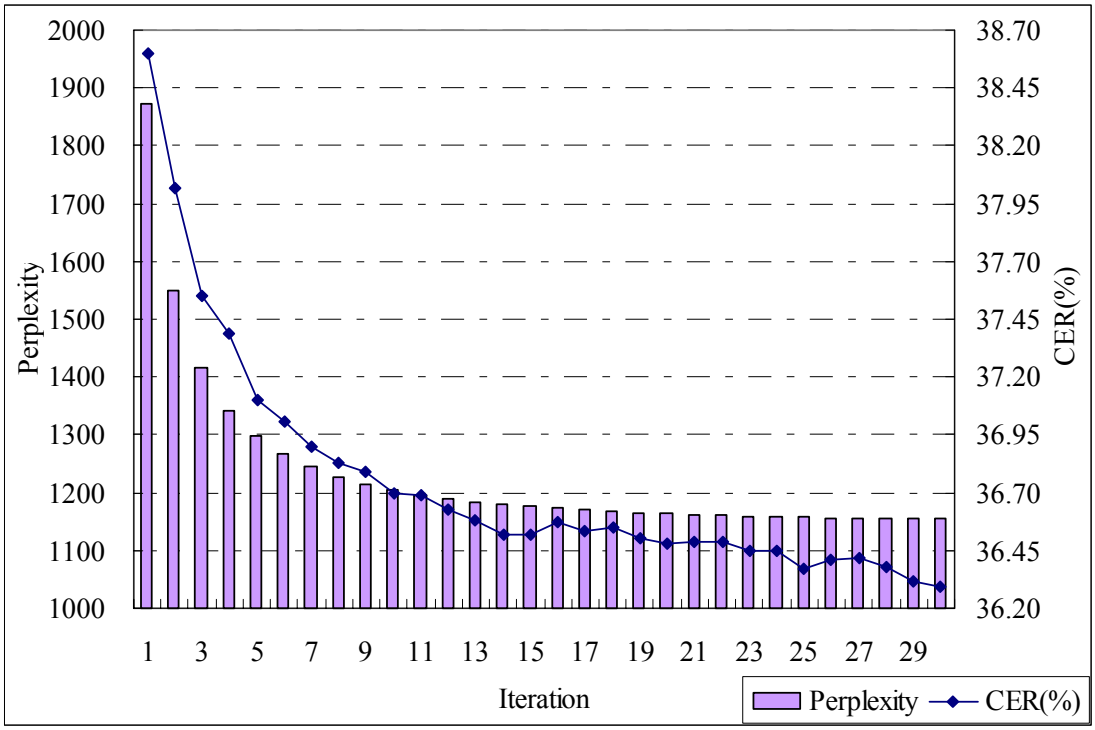


圖 5.5、最大熵值法在不同迭代次數之字錯誤率與語言模型複雜度。

從表 5.19 可以觀察出，在第 10 次迭代之後，最大熵商法在語言模型複雜度與字錯誤率已低於基礎實驗。以語言模型複雜度來觀察，在超過 15 次之後的迭代已經開始趨於收斂，超過 25 次之後已經沒什麼改變了；而以字錯誤率來觀察的話，大致上還是呈現迭代越多次，字錯誤率越低的趨勢，惟下降的幅度也趨於平緩。最大熵值法最佳的字錯誤率是 36.29%，相對進步 1.17%，最佳的語言模型複雜度為 1154.19，相對進步 4.49%。

此外，值得一提的是，本實驗所使用的電腦是 Pentium 4 XEON 2.8G Hz，記憶體 2GB，在最大熵值法的合併上述三種特徵，一次迭代約需 16 個小時，這遠比用最大相似度估測法訓練三連語言模型要來的費時，不過實驗結果也證實最大熵值是一個比傳統最大相似度估測還好的方法。

第6章 結論與未來展望

在過去的三十年間，語言模型在自然語言相關的應用上一直扮演著重要的角色，它被用來擷取自然語言中各式各樣的資訊，進而以量化的方式來決定一個詞序列是否被接受，例如幫助解決語音辨識中聲學混淆的問題。然而，隨著自然語言的演進，以早先搜集到的訓練語料所訓練的語言模型，漸漸地與要辨識的領域在詞彙或語意上發生的不一致性（mismatch），而造成不佳的辨識效果，於是便需要語言模型調適的技術。

語言模型調適的目的，是要從與辨識任務同時期或同領域的調適語料中擷取相關的資訊，如前後文資訊、語意資訊和主題資訊等，將這些資訊加入原有的語言模型中，使之對測試語料有更好的預測能力，進而達到較佳的辨識效果。

在本論文中，我們提出了主題混合模型法來調適背景語言模型。此方法原是應用在資訊檢索中，其中，每一個文件被表示成一個混合模型，模型中定義了 K 個主題，各由一主題一連語言模型所表示，另外每一個文件對這 K 個主題都有不同的權重，利用這兩個機率可以來計算查詢詞序列與此文件的檢索機率。今將詞 w 視為只擁有一個詞的查詢，而其歷史詞序列 h 視為一個文件，便可以計算它們之間的相關性，但因歷史詞序列會隨著辨識過程而改變，故需要動態地來計算各主題在歷史詞序列上的權重。將此主題混合模型與背景語言模型結合便可達到動態的語言模型調適。在實驗中，顯示了主題混合模型調適的效果，甚至比潛藏語意分析來的有效，且在與其他方法結合的情況下，也達到相輔相成的辨識效果。

另外，本論文對最大熵值法作了深入的探討。最大熵值法是一個以限制為基礎來合併各種資訊來源的方法，在此方法下，每一個資訊來源會引起一群限制，這些限制的交集代表了滿足所有限制的機率分佈的集合，這當中擁有最大熵值的機率分佈便是此方法的解。本論文利用最大熵值法將一連、二連與三連語言模型

合併，利用 IIS 演算法來求得最佳的模型參數。實驗顯示，利用此方法所訓練出來的模型，比用傳統最大相似度估測法所訓練的三連語言模型，在辨識字錯誤率與語言模型複雜度上都有較好的表現。

在往後的研究中，將嘗試以最大熵值法來合併 N 連語言模型、主題分類模型等更多樣的資訊，以期能有較佳的辨識效果。此外，語言模型調適的另外一個議題是如何挑選適當的調適語料，就如同本論文 Set 2 所使用的調適語料，因為與測試語料並非同時期的語料，所以在語意或主題資訊方面就沒有辦法有顯著的效果。但是，如果透過調適語料的篩選技術，將真正與測試語料相關的語料留下來，不相關的去除，再利用篩選下來的資料擷取資訊，如此一來，對語言模型調適也會有相當的幫助。

附錄A　Jensen 不等式

如果函數 f 在 $[a,b]$ 區間中是一個凸函數（convex function），則

$$f\left(\sum_{k=1}^{n}\lambda_k x_k\right)\le\sum_{k=1}^{n}\lambda_k f(x_k) \tag{A.1}$$

其中 $0\le\lambda_k\le 1$ 且 $\lambda_1+\lambda_2+\ldots+\lambda_n=1$，以及每個 $x_k\in[a,b]$。

令 $\lambda_k=\frac{1}{n}$，$f(\,)=\exp(\,)$，$x_k=\log y_k$ 且 $y_k>0$，則

$$\begin{aligned}
&f\left(\frac{1}{n}\sum_{k=1}^{n}x_k\right)\le\frac{1}{n}\sum_{k=1}^{n}f(x_k)\\
&\Rightarrow\exp\left(\frac{1}{n}\sum_{k=1}^{n}\log y_k\right)\le\frac{1}{n}\sum_{k=1}^{n}\exp(\log y_k)\\
&\Rightarrow\left(\exp\left(\log\left(\prod_{k=1}^{n}y_k\right)\right)\right)^{\frac{1}{n}}\le\frac{1}{n}\sum_{k=1}^{n}y_k\\
&\Rightarrow\left(\prod_{k=1}^{n}y_k\right)^{\frac{1}{n}}\le\frac{1}{n}\sum_{k=1}^{n}y_k
\end{aligned} \tag{A.2}$$

式(A.2)即為算術平均數大於等於幾何平均數。

因為 $0\le\lambda_k\le 1$ 且 $\lambda_1+\lambda_2+\ldots+\lambda_n=1$，所以 λ_k 是一個機率密度函數（p.d.f），令 $P(z_k)=\lambda_k$，$Q(z_k)=x_k$，$f(\,)=\exp(\,)$，則：

$$\exp\left(\sum_{k=1}^{n}P(z_k)Q(z_k)\right)\le\sum_{k=1}^{n}P(z_k)\exp(Q(z_k)) \tag{A.3}$$

式(A.3)即為式(3.40)。

附錄B　實作 IIS 演算法

在 IIS 演算法中，對於所有的特徵 f_i 要解下式中的 δ_i ：

$$\sum_{h,w} \widetilde{P}(h) P_\Lambda(w \mid h) f_i(h,w) \exp\left(\delta_i f^{\#}(h,w)\right) = \sum_{h,w} \widetilde{P}(h,w) f_i(h,w) \tag{B.1}$$

式(B.1)中，等號右邊可由訓練語料中求得，為一個數值，等號左邊是將所有 (h,w) 算出的值加總，我們可以稍微將它改寫：

$$\sum_{h,w} \widetilde{P}(h) P_\Lambda(w \mid h) f_i(h,w) \cdot \aleph_i^{f^{\#}(h,w)} = \sum_{h,w} \widetilde{P}(h,w) f_i(h,w) \tag{B.2}$$

其中 $\aleph_i = \exp(\delta_i)$ ，即轉變成解 $\aleph_i$ 。在式(B.2)中， $f^{\#}(h,w)$ 表示 (h,w) 所滿足的特徵個數，為一大於等於 0 的整數值，且若當 $f^{\#}(h,w) = 0$ 時， $f_i(h,w)$ 也會等於 0，即該 (h,w) 對特徵 f_i 而言在式(B.2)中是沒有貢獻的。所以式(B.2)等號左邊對所有 (h,w) 加總，相當於對 $f^{\#}(h,w)$ 大於 0（即 $f_i(h,w) = 1$）的 (h,w) 加總，則每一項的 $f^{\#}(h,w)$ 皆為正整數。接下來，我們可以根據 $f^{\#}(h,w)$ 的值，將式(B.2)等號左邊的各項分開來加總；令所有 $f^{\#}(h,w) = 1$ 的各項加起來等於 $\alpha_1 \cdot \aleph_i^1$ ，即：

$$\sum_{h,w \mid f^{\#}(h,w)=1} \widetilde{P}(h) P_\Lambda(h,w) f_i(h,w) = \alpha_1 \tag{B.3}$$

接著若將 $f^{\#}(h,w) = 2$ 、 $f^{\#}(h,w) = 3 \ldots$ 等各項都加總起來，並令 $\aleph_i^2$ 的係數為 α_2 、 $\aleph_i^3$ 的係數為 $\alpha_3 \ldots$ 依此類推，且令 $f^{\#}(h,w)$ 的最大值為#，則式(B.2)可以改寫成：

$$\alpha_1 \aleph_i^1 + \alpha_2 \aleph_i^2 + \ldots + \alpha_{\#} \aleph_i^{\#} = \sum_{h,w} \widetilde{P}(h,w) f_i(h,w) \tag{B.4}$$

則原本 IIS 演算法所要解的式(B.1)轉變成式(B.4)，式(B.4)為$\aleph_i$的#次多項式，解得$\aleph_i$之後再取對數即可得到δ_i：

$$\delta_i = \log(\aleph_i) \tag{B.5}$$

參考文獻

[Aubert 2002] X. L. Aubert. “An Overview of Decoding Techniques for Large Vocabulary Continuous Speech Recognition,” *Computer Speech and Language*, January 2002.

[Bacchiani et al. 2003] M. Bacchiani and B. Roark. Unsupervised Language Model Adaptation. *In Proceedings of the International Conference on Acoustics*, *Speech*, *and Signal Processing*, 2003.

[Baeza-Yates et al. 1999] R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval. Addison Wesley Longman, 1999.

[Ball et al. 1967] G. H. Ball, and D. J. Hall. A Clustering Technique for Summarizing Multivariate Data. *Behavioral Science*, Volume 12, pages 153-155, 1967.

[Bellegarda 2000] J. R. Bellegarda. Exploiting latent semantic information in statistical language modeling. *Proceedings of the IEEE*, Volume 88, pages 1279-1296, August 2000.

[Bellegarda 2004] J. R. Bellegarda. Statistical language model adaptation: review and perspectives. *Speech Communication*, 42, 2004.

[Bellegarda 2005] J. R. Dellegarda. Latent Semantic Mapping: Dimensionality Reduction via Globally Optimal Continuous Parameter Modeling. to appear in *IEEE Signal Processing Magazine*, September 2005.

[Berger et al. 1996] A. Berger, S. Della Pietra, and V. Della Pietra. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, 22 (1), pages 39-71, 1996.

[Berger 1997] A. Berger. The Improved Iterative Scaling Algorithm: A gentle Introduction. December, 1997.

[Chang et al. 2003] P-C Chang and L-S Lee. Improved Language Model Adaptation Using Existing and Derived External Resources. *In Proceedings of ASRU*, pages 531-536, December, 2003.

[Chen 2005] B. Chen. Exploring the Use of Latent Topical Information for Statistical Chinese Spoken Document Retrieval. Accepted for publication *In Pattern Recognition Letters*, 2005, (SCI Expanded, EI).

[Chen et al. 2002] B. Chen, H-M Wang, and L-S Lee. Discriminating Capabilities of Syllable-Based Features and Approaches of Utilizing Then for Voice Retrieval of Speech Information in Mandarin Chinese. *IEEE Trans. On Speech and Audio Processing*, Volume 10 (5), pages 303-314, July 2002.

[Chen et al. 2004a] B. Chen, J-W Kuo, and W-H Tsai. Lightly Supervised and Data-Driven Approaches to Mandarin Broadcast News Transcription. *In Proceedings of ICASSP*, Volume 1, pages 777-780, May 2004.

[Chen et al. 2004b] B. Chen, J-W Kuo, Y-M Huang, and H-M Wang. Statistical Chinese Spoken Document Retrieval Using Latent Topical Information. *In Proceedings of ICSLP* Volume 11, pages 1621-1625, October 2004.

[Chen et al. 2004c] B. Chen, W-H Tsai, and J-W Kuo. Statistical Language Model Adaptation for Mandarin Broadcast News Transcription. *In Proceedings of ISCSLP04*, pages 313-316.

[Chen et al. 2005] B. Chen, Jen-Wei Kuo, Wen-Huang Tsai. "Lightly Supervised and Data-Driven Approaches to Mandarin Broadcast News Transcription," *International Journal of Computational Linguistics and Chinese Language Processing*, Vol. 10, No. 1, pp.1-18, March 2005.

[Chen et al. 1999] S. F. Chen, J. Goodman. An Empirical Study of Smoothing Techniques for Language Modeling. *Computer Speech and Language*, 13, 1999.

[Chen et al. 2003] L. Chen, J-L Gauvain, L. Lamel, and G. Adda. Unsupervised Language Model Adaptation for Broadcast News. *In Proceedings of ICASSP*, Volume 1, pages 220-223, April 2003.

[Chou et al. 2003] W. Chou (editor), B. H. Juang (editor). *Pattern Recognition in Speech and Language Processing*, CRC Press, 2003.

[Chueh et al. 2004] C-H Chueh, J-T Chien, and H-M Wang. A Maximum Entropy Approach for Integrating Semantic Information in Statistical Language Models. *In Proceedings of ISCSLP04*, pages 309-312.

[CNA news] Central News Agency news. http://www.cna.com.tw.

[Darroch et al. 1972] J. N. Darroch, D. Ratcliff. Generalized iterative scaling for log-linear models. *The Annals of Mathematical Statistics*, Volume 43, pages 1470-1480, 1972.

[Della Pietra et al. 1992] S. Della Pietra, V. Della Pietra, R. Mercer, and S. Roukos. Adaptive Language Model Estimation Using Minimun Discrimination Estimation. *In Proceedings of the International Conference on Acoustics*, *Speech*, *and Signal Processing*, pages 633-636, April 1992.

[Dempster et al. 1977] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, Series B, Volume 39, no. 1, pages 1-38, 1977.

[Duda et al. 1973] R. O. Duda, and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley Sons.

[Federico 1999] M. Federico. Efficient Language Model Adaptation Through MDI Estimation. *In Proceedings of EUROSPEECH*, Volume 4, pages 1583-1586, September 1999.

[Gildea et al. 1999] D. Gildea, and T. Hofmann. Topic-Based Language Models Using EM. *In Proceedings of EUROSPEECH*, Volume 5, pages 2167-2170, September 1999.

[Good 1963] I. J. Good. Maximum Entropy for Hypothesis Formulation, Especially for Multidimensional Contingency Tables. *The Annals of Mathematical Statistics*, Volume 34, No. 3, pages 911-934, September, 1963.

[Goodman 2001] J. Goodman. A Bit of Progress in Language Modeling Extended Version. Microsoft Research, Machine Learning and Applied Statistics Group, Technique Report, 2001.

[Jaynes 1957] E. T. Jaynes. Information Theory and Statistical Mechanics. *Physics Reviews*, Volume 106, no. 4, pages 620-630, 1957.

[Jelinek 1977] F. Jelinek, R. L. Mercer, L. R. Bahl, and J. K. Baker. Perplexity—a measure of difficulty of speech recognition tasks. *94th Meeting of the Acoustic Society of America*, December 1977, Miami Beach, FL.

[Jelinek 1991] F. Jelinek. Up from Trigrams! The Struggle for Improved Language Models. *In Proceedings of EUROSPEECH*, page 1037-1040, 1991.

[Katz 1987] S. M. Katz. Estimation of Probabilities from Sparse Data for the Language Model Component of A Speech Recognizer. *IEEE Trans. On Acoustics, Speech and Signal Processing*, Volume 35 (3), pages 400-401, March 1987.

[Kim et al. 2004] W. Kim, and S. Khudanpur. Cross-Lingual Latent Semantic Analysis for Language Modeling. *In Proceedings of ICASSP*, Volume 1, pages 257-260, May 2004.

[Kneser et al. 1997] R. Kneser, J. Peters, and D. Klakow. Language model adaptation using dynamic marginals. *In Proceedings of EUROSPEECH*, pages 1971-1974, Rhodes, Greece, 1997.

[Kuhn et al. 1990] R. Kuhn, and R. De Mori. A cache-based natural language model for speech recognition. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, Volume 12, pages 570-582, June 1990.

[Kullback 1959] S. Kullback. Information Theory in Statistics. Wiley, New York, 1959.

[Liu et al. 2003] X. Liu, and W. B. Croft. Statistical Language Modeling for Information Retrieval. To appear in *the Annual Review of Information Science and Technology*, Volume 39 (2005).

[Miller et al. 1999] D. R. H. Miller, T. Leek, and R. Schwartz. A Hidden Markov Model Information Retrieval System. *In Proceedings of ACM SIGIR Conference on R&D in Information Retrieval*, pages 214-221, 1999.

[Mori et al. 1999] R. De Mori, and M. Federico. Language model adaptation. *In Computational Models of Speech Pattern Processing*. K. Ponting, Ed., Volume 169 of *F*: *Computer and Systems Sciences*, pages 280-303, 1999.

[Moriya et al. 2001] T. Moriya, K. Hirose, N. Minematsu, and H. Jiang. Enhanced MAP Adaptation of N-gram Language Models Using Indirect Correlation of Distant Words. *In Proceedings of ASRU*, pages 397-400, Italy, December 2001.

[Mrva et al. 2004] D. Mrva, and P. C. Woodland. A PLSA-based Language Model for Conversational Telephone Speech. *In Proceedings of ICSLP*, pages 2257-2260, October 2004.

[Nanjo et al. 2003] H. Nanjo, and T. Kawahara. Unsupervised Language Model Adaptation for Lecture Speech Recognition. *In Proceedings of ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, pages 75-78, 2003.

[NIST] National Institute of Standards and Technology. http://www.nist.gov/ .

[PTS] Public Television Service Foundation. http://www.pts.org.tw .

[Ratnaparkhi 1997] A. Ratnaparkhi. A simple introduction to maximum entropy models for natural language processing. Technical Report 97-08, Institute for Research in Cognitive Science, University of Pennsylvania, 1997.

[Rosenfeld 1996] R. Rosenfeld. A Maximum Entropy Approach to Adaptive Statistical Language Modeling. *Computer Speech and Language*, Volume 10, pages 187-228, 1996.

[Rosenfeld 2000] R. Rosenfeld. Two Decades of Statistical Language Modeling: Where Do We Go from Here. *In Proceedings IEEE*, Volume 88, no. 8, pages 1270-1278, 2000.

[Sasaki et al. 2000] K. Sasaki, H. Jiang, and K. Hirose, Rapid Adaptation of N-gram Language Models Using Inter-word Correlation for Speech Recognition. *In Proceedings of ICSLP*, pages 508-511, Beijing, October 2000.

[SLG] Spoken Language Group at Chinese Information Processing Laboratory, Institute of Information Science, Academia Sinica. http://sovideo.iis.sinica.edu.tw/SLG/index.htm .

[SRILM] A. Stolcke. SRI Language Modeling Toolkit. version 1.3.3, http://www.speech.sri.com/projects/srilm/ .

[Valsan et al. 2003] Z. Valsan and M. Emele. Thematic Text Clustering for Domain Specific Language Model Adaptation. *In Proceedings of ASRU*, pages 513-518, December 2003.

[Wang et al. 2005] Hsin-min Wang, Berlin Chen, Jen-Wei Kuo, and Shih-Sian Cheng. "MATBN: A Mandarin Chinese Broadcast News Corpus," accepted to appear in *International Journal of Computational Linguistics and Chinese Language Processing*.

[郭人瑋 等 2004] 郭人瑋、蔡文鴻、陳柏琳. “非監督式學習於中文電視新聞自動轉寫之初步應用,” 第十六屆自然語言與語音處理研討會 (*Proc. ROCLING XVI*).