具相關資訊回饋能力之貝氏混合式機率檢索模型

Using Relevance Feedback in Bayesian Probabilistic Mixture Retrieval Model

簡仁宗 楊敦淇
國立成功大學資訊工程學系
Email: jtchien@mail.ncku.edu.tw

摘要

本篇論文提出新穎之相關回饋（Relevance Feedback）方法並應用於混合式機率檢索系統（Mixture Probability Model）以提昇檢索效能。相關資訊回饋法以往最常用的技術是查詢句擴充法（Query Expansion），本回饋方式是架構在以混合式機率模型為主的檢索系統上，為了加強檢索效能，我們是在查詢句擴充法中，強調不同查詢詞的重要性，所以提出查詢詞權重重調整（Query Term Reweighting）技術；此外，我們也利用檢索出來的前 *N* 名文件和資料庫的每份文件個別重調成新的文件語言模型，以提供較好的文件語言模型提供檢索時使用。在查詢字權重之重調整部分以最佳相似度（Maximum Likelihood）為估測準則，而文件語言模型之調整部分先後以最佳相似度與最佳事後機率（Maximum *a Posteriori*）為估測準則供我們對照比較，並使用了 EM（Expectation Maximization）演算法去估測出適當的參數。實驗結果顯示使用資訊回饋及貝氏語言模型調整可有效提升文件檢索正確率。

1. 簡介

目前資訊檢索的型態大致可分為[1]：布林式(Boolean)檢索，類神經網路(Neural Network)檢索，向量式(Vector-Based)檢索以及機率式(Probability-Based)檢索等；以上數種檢索式中，目前在搜尋引擎上較為廣泛使用的為布林式檢索，目前常被使用的 Google 搜尋引擎根據網站上的檢索方式說明[19]，整個過程便是從布林運算發展，以比對字串為主的檢索。

資訊檢索的領域裡，有一種能有效地提昇效能的方法稱為相關資訊回饋（Relevance Feedback），它是使用前一次檢索所得到的文件分數中，找出檢索分數較高的前 *N* 篇或是適當的 *N* 篇文件，從其中擷取可用的資訊回饋加入下一次遞迴的檢索中，增強檢索所需要的資訊；其概念是假設某些和查詢句相關的文件檢索後排名很前面，但是某些相關文件（Relevant Document）語意上雖相似，但是也許內容出現了問題，例如：查詢詞出現的比較少，因此檢索的排名會比較後面，所以利用排名前面的相關文件去想辦法拉抬排名於後的相關文件。在過去常用於資訊檢索的相關回饋方式主要為查詢句擴充和查詢詞權重再調整。

一般使用者在搜尋引擎所下的查詢句通常都不長，因此提供的資訊並不多；另外，相關回饋於資訊檢索之研究大部分都是針對向量模型檢索系統，對於以機率為主的 *n*-gram 語言模型檢索系統，只能使用查詢句擴充法來提昇檢索效能，但是觀察整個檢索流程，發現將每一份文件視為一個語言模型時，裡面能提供的資訊其實也不多，會造成不同文件之間的混淆，假若能利用前一次遞迴檢索出排名較高的數篇文件去調整資料庫中的文件，與它們相關的文件提供較多的資訊，與它們不相關的文件便提供少一點的資訊，那麼在下一次遞迴的檢索中，便能減少一些文件與文件之間混淆的程度，而達成有效的自動檢索過程；此外在一些檢索系統上會用到的查詢詞權重的觀念若能引進來，將這些參數額外地加到混合式 *n*-gram 檢索架構中輔助原本的語言模型計算分數，並利用回饋的資訊去重調整權重，如此應可加強一些重要字的分數以提昇檢索效能。所以我們以混合式機率檢索架構為主，於此架構上使用相關資訊回饋。除了沿用先前的查詢句擴充方式外，我們嘗試在檢索式中針對每個查詢詞加入權重的參數，將前一次遞迴檢索分數最高的 *N* 篇文件去做查詢詞的權重重調整，期望以這 *N* 篇文件內的分布情形，去調整出每個查詢詞的重要程度，此外，針對文件內提供資訊過少的問題，我們使用最佳事後機率（Maximum *a Posteriori*）法則將這 *N* 篇文件和資料庫裡的每一份文件調成新的文件混合語言模型，利用這 *N* 篇文件模型適當的補充資訊予資料庫內的文件。

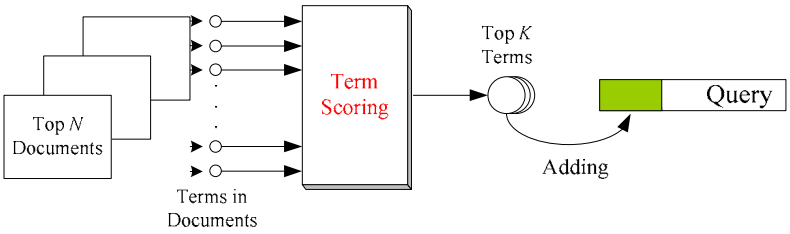
2. 相關研究

2.1 相關資訊回饋

使用者提供給檢索系統的查詢句中，通常句子的長度都偏短，如此能提供之資訊便相對的減少，容易造成檢索時產生混淆的情況[3]。為了此類問題的解決，有很多研究朝著上下文分析，語意分析等自然語言處理以及文件內容標記的定義，如 XML 上來發展。而利用前一次檢索所獲得之相關文件來調整查詢句，對於檢索效果

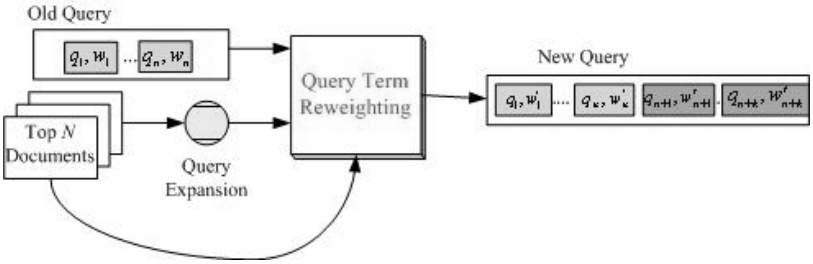
也有相當程度之改善。

在現有的檢索模型架構中，查詢句擴充的目的就是為了能從相關文件內多找些同主題中常會出現的詞，以補充查詢句過短之缺點，所以當查詢句的長度越長，查詢句所含之資訊越多，查詢句擴充所能提供的效果就越有可能降低。其架構如圖一所示：



圖一、查詢句擴充架構圖

查詢詞的權重調整以前一回檢索排名較高之數篇文件裡面的詞分布情形為依據，為了強調某些常出現的詞而設計的計算式，以便在向量等檢索模型的表現中會更趨近同主題文件。架構如下圖所示：



圖二、查詢詞權重重調整架構圖

相對於查詢詞 q 有一個對應的權重 w，經過查詢句擴充及權重之調整後，除了新增詞於查詢句之外，原本的權重也被更新過了。

2.2 向量檢索模型之資訊回饋

以向量檢索模型而言，針對查詢句 Q 和文件 d，利用每個字的出現次數以及在文件間分布的情形去算出特徵向量 $\mathbf{q}$ 和 $\mathbf{d}$，計算查詢句和文件的相似度以內積（Inner Product）運算為主，在此情形下，回饋資訊必然以向量的型態去調整查詢句的向量從 $\mathbf{q}$ 到 $\tilde{\mathbf{q}}$，目前常見之向量型態的資訊回饋略舉兩例[8]：

$$\text{Rocchio：}\quad \tilde{\mathbf{q}} = \mathbf{q} + \frac{\beta}{|V|}\sum_{\mathbf{d}_i \in V}\mathbf{d}_i - \frac{\gamma}{|U|}\sum_{\mathbf{d}_j \in U}\mathbf{d}_j \tag{1}$$

$$\text{Ide dec-hi：}\quad \tilde{\mathbf{q}} = \mathbf{q} + \sum_{\mathbf{d}_i \in V}\mathbf{d}_i - \max_{\mathbf{d}_i \in U}\mathbf{d}_j \tag{2}$$

β 和 γ 是經實驗所找出的經驗值，V 是指和查詢句 Q 相關的文件群，U 是指和查詢句 Q 不相關的文件群，其方法是利用找出相關與不相關的文件群來改善查詢句向量 $\mathbf{q}$，以提昇下一次遞迴的檢索效能。

3. 使用相關資訊回饋於貝氏混合式機率檢索

3.1 *N*-gram 模型的建立

N-gram[6]模型在自然語言處理中是常見的技術，應用的範圍很廣，有資訊檢索、語音辨識、光學文字辨識和文件分類等方向。本論文的主架構混合式機率檢索即是 *n*-gram 模型於資訊檢索上的應用，我們首先針對 *n*-gram 的建立方法與評估作概略的介紹。

語言模型主要的功能是在評估一段文句出現的機率，假設有一查詢句 Q 其長度為 T 並且是由一段詞序列 $q_1, q_2, \ldots, q_T$ 所組成，則 Q 出現的機率可以寫成：

$$\begin{aligned} P(Q) &= P(q_1, q_2, \ldots, q_T) = P(q_1)P(q_2 \mid q_1)\cdots P(q_T \mid q_1, q_2, \ldots, q_{T-1}) \\ &= \prod_{t=1}^{T} P(q_t \mid q_1, q_2, \ldots, q_{t-1}) \end{aligned} \tag{3}$$

但是此種方法的計算量與空間使用量太大而無法實現，為解決這個問題所以有 *n*-gram 模型的產生，在 *n*-gram 模型中，它是假設一個詞出現的機率只跟前面 *n*-1 個詞有關，因此（3）式可以近似為

$$P(Q) = P(q_1, q_2, \ldots, q_T) \cong \prod_{t=1}^{T} P(q_t \mid q_{t-n+1}^{t-1}) \tag{4}$$

其中 q_{t-n+1}^{t-1} 代表 $q_{t-n+1}, q_{t-n+2}, \cdots, q_{t-1}$ 詞序列如此一來使用 *n*-gram 可以大量節省計算時間與記憶體，讓實用性大為提高。而建立 *n*-gram 機率模型 $P(q_t \mid q_{t-n+1}^{t-1})$ 的基本式如下：

$$P(q_t \mid q_{t-n+1}^{t-1}) = \frac{c(q_{t-n+1}^{t})}{c(q_{t-n+1}^{t-1})} = \frac{c(q_{t-n+1}^{t})}{\sum_{q_j} c(q_{j-n+1}^{j})} \tag{5}$$

其中 $c(q_{t-n+1}^{t})$ 代表 q_{t-n+1}^{t} 在訓練文集中出現的次數

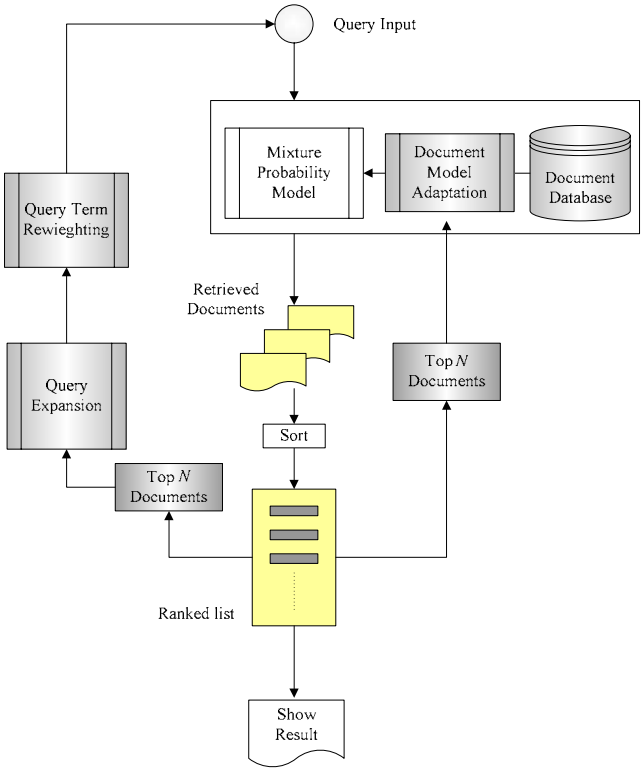
3.2 混合式機率檢索模型

混合式機率檢索是以 *n*-gram 模型為主的資訊檢索技術。此架構原出於[11]，被稱為隱藏式馬可夫模型（Hidden Markov Model）[13]，可是因為此架構只有單一狀態，故稱為"混合式機率模型"較為適當。裡面包含了 $P(q_i \mid d_j)$和 $P(q_i \mid q_{i-1}, d_j)$這種相對於文件 d_j 的 Uni-gram 和 Bi-gram，並且為了表示出查詢句 Uni-gram 和 Bi-gram 一般分布的情形而引入了一個背景語料(*Corpus*)，這個背景語料的語言模型 $P(q_i \mid Corpus)$ 和 $P(q_i \mid q_{i-1}, Corpus)$ 是由大批的文件集合依照（5）所算出來的。而相似度的量測 $P(Q \mid d_j)$，即是由查詢句 *Q* 中每個詞，循序計算的機率值，累計的結果即可視為其相關程度，則查詢句 *Q* 相關於文件 d_i 的機率表示如下：

$$\begin{aligned} P(Q \mid d_j) &= \left[\lambda_1 P(q_1 \mid d_j) + \lambda_2 P(q_1 \mid Corpus)\right] \\ &\times \prod_{t=2}^{|Q|} \begin{bmatrix} \lambda_1 P(q_t \mid d_j) + \lambda_2 P(q_t \mid Corpus) \\ + \lambda_3 P(q_t \mid q_{t-1}, d_j) + \lambda_4 P(q_t \mid q_{t-1}, d_j) \end{bmatrix} \end{aligned} \tag{6}$$

關於混合式 *n*-gram 檢索模型之推導與詳細內容可參考[17]。

本論文的重點便是在混合式機率檢索模型內加入回饋的機制，下方圖三為本論文檢索模型之新穎回饋流程架構，而其中的研究，主要在相關資訊回饋方式有三個改進方向：*查詢句擴充*、*查詢詞權重重調整*和*文件模型調整*（Document Model Adaptation）；於前一回的檢索及文件排序完成後，先使用 Top N 的文件做查詢句擴充、查詢詞權重重調整之後更新 Query 後,於檢索開始時調整文件模型。而每一程序之處理過程，將於接下來之各節做詳細的描述。



圖三、加入相關資訊回饋機制的檢索流程

3.3 查詢句擴充

過去的研究中，都顯示出了查詢句擴充的效果，並發現查詢詞的挑選，應該以接近檢索時的計算式為主，如此較有機會補充適合此檢索系統的查詢詞；在本論文裡，因為採用語言模型的方式檢索，所以擴充詞挑選的

方式便簡單地利用字詞相對於文件的機率，因此我們根據下式為選擇判斷式，對每個出現於排名前 N 名文件 $\{\hat{d}_1,\cdots,\hat{d}_N\}$ 內的詞做排名，並找出名次最高的前幾個詞加入查詢句中：

$$\sum_{j=1}^{N} P(q_t \mid \hat{d}_j) P(\hat{d}_j) P(Q \mid \hat{d}_j) \tag{7}$$

其中 q_t 是出現在前 N 名文件內的詞，$P(q_t \mid \hat{d}_j)$ 為在文件 $\hat{d}_j$ 的 Uni-gram 的機率；$P(\hat{d}_j)$ 為事前（Prior）機率，是文件 $\hat{d}_j$ 長度（詞數）於 N 篇文件長度總和的比例；$P(Q \mid \hat{d}_j)$ 就是前一次遞迴的檢索中，查詢句 Q 和文件 $\hat{d}_j$ 比對的分數。

3.4 查詢詞權重重新調整

在向量檢索的相關資訊回饋中有一種常被應用且變化的 Rocchio 公式，其焦點放在正面的例子（相關文件），而忽略了負面例子的加入（不相關文件）。從這裡，我們可以得到一個想法，有些查詢詞因為有其重要的代表性，所以在某些相關文件中出現的次數比較多，使得這些相關文件的排名會比較前面，但是這些查詢詞在其他的相關文件出現次數比較少，於是使得這些文件就會被排名比較後面；假設我們能夠對這些查詢詞適當地分配一權重，於式子中可改變每個查詢詞所提供之資訊，比較重要的詞給予較高的權重，相反地，對於不重要的詞給予較低的權重，如此，期望對於這些含有具代表性查詢詞比較少的相關文件在計算對查詢句之相似分數時能夠有所提昇。我們把簡化語言模型的機率檢索式來看：

$$P(Q \mid d_j) = \prod_{t=1}^{|Q|} P(q_t \mid d_j) \tag{8}$$

若可以從式子中抽取出一個因子 κ_t 代表查詢詞 q_t 的權重，在第一輪的最初檢索過程中，初始的查詢句因為沒有其他資訊介入，所以每個 κ_t 可視做 1，對原結果不受任何影響，即：

$$P(Q \mid d_j) = \prod_{t=1}^{|Q|} \kappa_t P(q_t \mid d_j) \tag{9}$$

至於下一輪的更新若以 $\kappa_t + \Delta\kappa_t$ 表示，因為在使用者所下的查詢句中，經過調整出來的權重必有一定的代表性，當權重值高時，則此查詢詞可看做檢索之關鍵，在此情形下，舊的查詢詞權重也應該在下一回的回饋程序中保留，並加上一更新權重值 $\Delta\kappa_t$ 以做調整，每一個查詢詞的權重更新值依據此分配的量做正規化，得到以下更新後的機率值：

$$\hat{\kappa}_t = \frac{\kappa_t + \Delta\kappa_t}{\sum_{k=1}^{|Q|} (\kappa_k + \Delta\kappa_k)} \tag{10}$$

其中 $\sum_{t}^{|Q|} \hat{\kappa}_t = 1$，而 $|Q|$ 意指查詢句的長度。令 D 是一個集合，裡面是檢索分數中排名前 N 名的文件，$D = \{\hat{d}_1, \hat{d}_2, \cdots, \hat{d}_N\}$，$K$ 也是一個集合，裡面是每個查詢詞相對應的權重，$K = \{\kappa_1, \kappa_2, \cdots, \kappa_{|Q|}\}$。因為檢索式之目的是為了提昇與查詢句相關文件的分數，以便增加相關文件與非相關文件之差別，我們為每個查詢詞加入相對應的權重也是為了這個原因，所以我們必須找出一組適當的權重，而這組權重是確定可以提昇與相關文件之相似度：

$$\hat{K} = \arg\max_{K} P(Q \mid D, K) \tag{11}$$

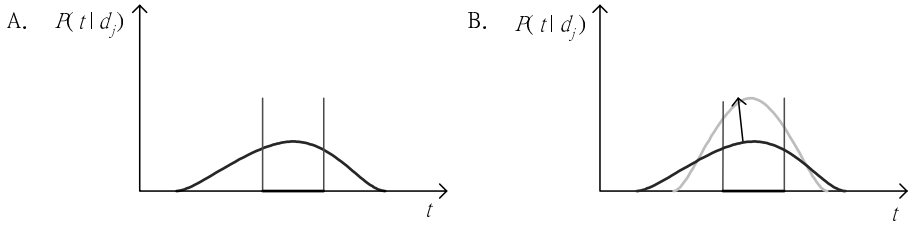
其中能觀察到的資料是我們所使用的語言模型與查詢句 Q，這是不完整的資料集（Incomplete Data），但是權重為未知的參數，如此只能想辦法去近似出適當的權重，所以我們以最佳相似度估測（Maximum Likelihood Estimation，MLE）為標準（Criterion），使用 EM 演算法[5]的步驟去推估出新的估測值 $\hat{\kappa}_t$ 的公式如下：

$$\hat{\kappa}_t = \frac{\sum_{j=1}^{N} \frac{\kappa_t P(q_t \mid \hat{d}_j)}{\sum_{k=1}^{|Q|} \kappa_k P(q_k \mid \hat{d}_j)}}{\sum_{l=1}^{|Q|} \sum_{j=1}^{N} \frac{\kappa_l P(q_l \mid \hat{d}_j)}{\sum_{k=1}^{|Q|} \kappa_k P(q_k \mid \hat{d}_j)}} = \frac{\sum_{j=1}^{N} \frac{\kappa_t P(q_t \mid \hat{d}_j)}{\sum_{k=1}^{|Q|} \kappa_k P(q_k \mid \hat{d}_j)}}{N} \tag{12}$$

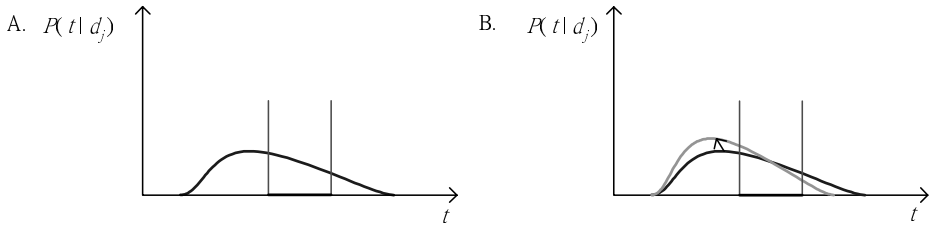
$\hat{\kappa}_t$ 值將會依照（12）遞迴地被訓練出來。

3.5 貝氏混合式機率模型調整

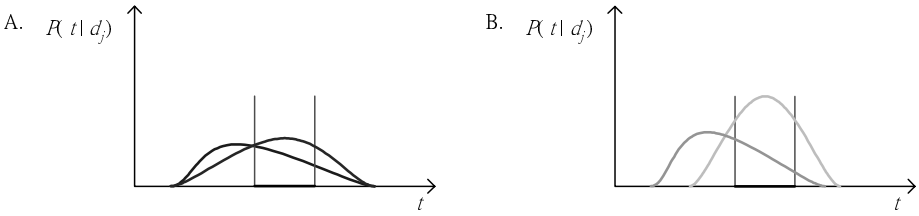
對於文件模型，我們嘗試利用前一回檢索所得到排名前 N 名文件的文件語言模型來補充目前查詢句正要比對的文件 d_j 其文件模型的資訊，使得文件 d_j 的語言模型能適用於目前查詢句的檢索。而調整文件語言模型之目的以下圖來說明：（水平軸：詞彙 t；垂直軸：在文件 d 內的詞彙 t 機率 $P(t \mid d_j)$。）



圖四、與查詢句相關文件的語言模型假想圖，(A)未經過相關資訊回饋的調整和(B)經過相關資訊回饋的調整



圖五、與查詢句不相關文件的語言模型假想圖，(A)未經過相關資訊回饋的調整和(B)經過相關資訊回饋的調整



圖六、相關與不相關文件的模型重疊假想圖，(A)未經過相關資訊回饋的調整和(B)經過相關資訊回饋的調整

在這些語言模型的機率分布裡，水平軸上出現框線的間隔，即代表查詢詞出現的範圍，所以此間隔與曲線圍起來之區域可以說是查詢句於文件模型內可能會用到的機率值，圖五是文件和查詢句相關的情形，圖六是和查詢句不相關的文件機率分布情形，圖七為兩種文件模型對於查詢句的機率分布重疊的比較。當原始的兩篇文件其分布情形為圖五-(A)和圖六-(A)，重疊之後得到圖七-(A)，並由圖七-(A)可得知，對於目前的查詢句經過比對計算得到的分數差距比較小，這有可能會造成檢索排名出現問題。若是在前一回檢索出來的結果，前 N 名的文件其資訊是可以利用的，意即可用來調整每份文件的模型，經過調整後得到圖五-(B)，圖六-(B)以及將這兩種調整過的文件模型分布重疊後得到圖七-(B)的情形，結果是相關與不相關之語言模型差異度變大了，如此一來就可以減少語言模型的模糊情形，並且有利於相關文件檢索分數的提昇。

假設資料庫裡的文件 d_j 和相對應的文件語言模型，可以和排序列表（Ranked List）內排名前 N 名的文件一起作用，來產生出新的文件語言模型，此時即是把排名前 N 名的文件語言模型和文件 d_j 的文件語言模型混合成一個新的文件語言模型，我們將導入權重參數作語言模型的合併，原（8）在加入了相關資訊回饋的機制，於第二次以及之後的遞迴檢索過程，將會變成

$$\tilde{P}(Q \mid d_j) = \prod_{t=1}^{|Q|} \tilde{P}(q_t \mid d_j) \tag{13}$$

（13）之變換過程描述於下：

1.令 M_j 為一個相對應於 $D_j = \{d_j, \hat{d}_1, \cdots, \hat{d}_N\}$ 之權重參數（Mixture Weight）集合,裡面放置相對應的合併權重參數，$M_j = \{m_{j,0}, m_{j,1}, \cdots, m_{j,N}\}$，$m_{j,0} + \sum_{k=1}^{N} m_{j,k} = 1$，$D_j$ 和 M_j 皆是針對文件 d_j 用到的資訊。而 $m_{j,k}$ 意指混合數 k 之是語言模型權重，會隨著文件 d_j 有所不同。

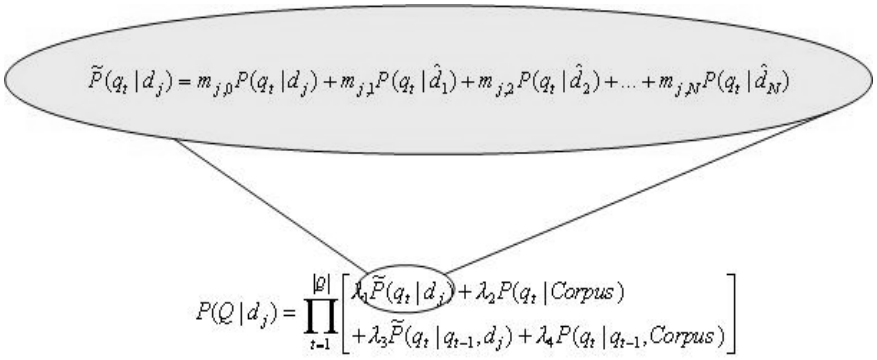
2.因為查詢詞是和文件內容相關，而文件內容和文件模型相關，若是文件模型產生文件內容之機率能更適當，則一個和此文件相關之查詢句 Q，使用此文件模型產生出來的機率也會更適當；所以文件語言模型權重的訓練過程即是以文件內容為估測之主要內容，目的是要針對文件內容 d_j，利用回饋的資訊去調出最適當的文件語言模型，因此最後依照我們所選擇之標準去找出適當的合併參數。

綜合上述三點及原本的想法，我們得到以下的式子：

$$\tilde{P}(q_t \mid d_j) = m_{j,0} P(q_t \mid d_j) + m_{j,1} P(q_t \mid \hat{d}_1) + m_{j,2} P(q_t \mid \hat{d}_2) + \ldots + m_{j,N} P(q_t \mid \hat{d}_N) \tag{14}$$

此式對原檢索模型之影響如圖八，圖中$\{\lambda_1,\ldots,\lambda_4\}$是混合式檢索模型的參數，是在建立混合式機率模型時就已經計算好了，這裡是強調在做模型參數$P(q_t \mid d_j)$之調整。若將d_j當成$\hat{d}_0$,結合前N名文件$\hat{d}_1,\hat{d}_2,\cdots,\hat{d}_N$，於是此式可轉換為

$$\tilde{P}(q_t \mid d_j) = \sum_{k=0}^{N} m_{j,k} P(q_t \mid \hat{d}_k) \tag{15}$$



圖七、混合式機率模型之調整

A.最佳相似度估測

若使用最佳相似度（Maximum Likelihood，ML）估測法則，其最佳參數M_j^{ML}計算如下[17]

$$M_j^{ML} = \arg\max_{M_j} P(Q \mid D_j, M_j) \tag{16}$$

在此$\sum_{k=0}^{N} m_{j,k} = 1$為一限制（Constraint）。我們必須執行有限制的最佳化（Constraint Optimization），利用文件本身與回饋之文件調整出一個更符合該文件之模型出來，因為參數 M_j未知，已知的觀察資料為文件集合與語言模型集合，資料並不完全，所以依照 EM 演算法去推出合併參數的式子，其結果如下

$$m_{j,k}^{ML} = \frac{\sum_{t=1}^{|Q|} \frac{m_{j,k} P(q_t \mid \hat{d}_k)}{\sum_{l=0}^{N} m_{j,l} P(q_t \mid \hat{d}_l)}}{\sum_{v=0}^{N}\sum_{t=1}^{|Q|} \frac{m_{j,v} P(q_t \mid \hat{d}_v)}{\sum_{l=0}^{N} m_{j,l} P(q_t \mid \hat{d}_l)}} = \frac{\sum_{t=1}^{|Q|} \frac{m_{j,k} P(q_t \mid \hat{d}_k)}{\sum_{l=0}^{N} m_{j,l} P(q_t \mid \hat{d}_l)}}{|Q|} \tag{17}$$

B.最佳事後機率估測

雖然使用最佳相似度為標準可估出一組參數去調整文件的語言模型，不過以最佳事後機率（Maximum *a Posteriori*，MAP）為標準，和 ML 比起來，在估測過程中多加入了事前機率通常是有助於在稀疏（Sparse）資料條件下的估測[7]。在進行 MAP 的推導之前，我們定義所需的參數 Ω_j如下，$\Omega_j = \{m_{j,k}, P(q_t \mid \hat{d}_k), 0 \le k \le N, 1 \le t \le |Q|\}$

$$\Omega_j^{MAP} = \arg\max_{\Omega_j} g(\Omega_j \mid Q, D_j) = \arg\max_{\Omega_j} P(Q \mid D_j, \Omega_j) g(\Omega_j) \tag{18}$$

其中$g(\Omega_j)$是參數 Ω_j的事前機率，我們假設為 Dirichlet 機率分佈，而 j 是指目前查詢到第 j 篇文件 d_j。事前機率$g(\Omega_j)$如下所示

$$g(\Omega_j) \propto \prod_{k=0}^{N} m_{j,k}^{v_{j,k}-1} \prod_{t=1}^{|Q|} P(q_t \mid \hat{d}_k)^{l_{j,k,t}-1} \tag{19}$$

$v_{j,k}$ 和 $l_{j,k,t}$ 是 Dirichlet 機率分佈的 Hyperparameter。k 代表回饋文件的編號，k=0 時，代表資料庫裡正被查詢到的文件，$k = 1,\cdots,N$ 為前一回找出排名前 N 的文件。

針對混合參數$m_{j,k}$推導出來的結果為

$$m_{j,k}^{MAP}=\frac{\sum_{t=1}^{|Q|}\frac{m_{j,k}P(q_t\mid\hat{d}_k)}{\sum_{l=0}^{N}m_{j,l}P(q_t\mid\hat{d}_l)}+(v_{j,k}-1)}{\sum_{v=0}^{N}\left[\sum_{t=1}^{|Q|}\frac{m_{j,v}P(q_t\mid\hat{d}_k)}{\sum_{l=0}^{N}m_{j,l}P(q_t\mid\hat{d}_l)}+(v_{j,v}-1)\right]} \tag{20}$$

其語言模型參數 $P(q_t\mid\hat{d}_k)$ 部分，其最後推導結果如下

$$P^{MAP}(q_t\mid\hat{d}_k)=\frac{n_{t,k}\left(\frac{m_{j,k}P(q_t\mid\hat{d}_k)}{\sum_{l=0}^{N}m_{j,l}P(q_t\mid\hat{d}_l)}+(l_{j,k,t}-1)\right)}{\sum_{v=1}^{|Q|}\left[n_{v,k}\left(\frac{m_{j,k}P(q_v\mid\hat{d}_k)}{\sum_{l=0}^{N}m_{j,l}P(q_v\mid\hat{d}_l)}+(l_{j,k,v}-1)\right)\right]} \tag{21}$$

其中 $n_{t,k}$ 為詞組 q_t 在第 k 個混合數出現的次數。

C. Hyperparameter 的初始化與更新方式

在 Hyperparameter 的初始化的部分，我們參考[10]並採用以下的公式做初始化

$$v_{j,k}^{(0)}=1+\varepsilon\cdot\overline{m}_{j,k} \tag{22}$$

$$l_{j,k,t}^{(0)}=1+\varepsilon\cdot\overline{P}(q_t\mid\hat{d}_k) \tag{23}$$

其中 $\overline{m}_{j,k}$ 及 $\overline{P}(q_t\mid\hat{d}_k)$ 的計算方式是將訓練資料估測出來的最佳相似度值 $m_{j,k}^{ML}$ 及 $P^{ML}(q_t\mid\hat{d}_k)$，進行取平均值的運算而得到的。$0<\varepsilon<1$ 是一個加權的係數，目的是去調整事前資料的權重。而 Hyperparameter 的更新公式是根據 Dirichlet 事前機率分布是屬於 Conjugate Prior 的特性推導出如下的結果[10]

$$v_{j,k}^{new}=v_{j,k}^{old}+\sum_{t=1}^{|Q|}\frac{m_{j,k}P(q_t\mid\hat{d}_j)}{\sum_{l=0}^{N}m_{j,l}P(q_t\mid\hat{d}_l)} \tag{24}$$

$$l_{j,k,t}^{new}=l_{j,k,t}^{old}+\frac{m_{j,k}P(q_t\mid\hat{d}_j)}{\sum_{l=0}^{N}m_{j,l}P(q_t\mid\hat{d}_l)} \tag{25}$$

把前一次遞迴算出 Hyperparameter 的 $v_{j,k}^{old}$ 及 $l_{j,k,t}^{old}$，用此公式更新到 $v_{j,k}^{new}$ 及 $l_{j,k,t}^{new}$。

4. 實驗

4.1 實驗環境-斷詞工具與實驗文集說明

為了將本論文方法實現在中文新聞資訊檢索系統中，首先必須製作了一套詞典，這個詞典之功用是為了能將文件裡得句子斷成更小的詞單位，同時將每一個斷出來的詞轉成詞典中對應的編號，詞典中主要部分是來自 CKIP (Chinese Knowledge Information Processing)中文詞庫[18]，主要是利用國語日報辭典中約四萬目詞的原始資料加以分類，並且附加部分的語法及語意訊息在其中，但在本論文，只使用到詞出現的頻率，並無使用到語法與語意資訊，我們只有取出其中一、二、三、四詞的部分作為基本辭典。

實驗過程所使用的文集為 TDT2（Topic Detection and Tracking Phase 2），是由 LDC（Linguistic Data Consortium）所收集的新華社新聞文件。總計有 11,161 篇西元 1998 年 1 月 1 日到 6 月 30 日的新聞，總共有 20 個主題，1,183 篇新聞文件，剩下 9,978 篇無標出主題文件，從其中取出 4,815 篇來算出檢索模型中所需要的背景語言模型，由於我們所使用之新華社新聞並無經過分類（國際、政治、財經及體育…等類別），為了保持背景語言模型之平衡性，針對每月每日之新聞以隨機方式抽出，平均每月取八百篇文件。標明主題之 1,183 篇文件與尚未使用無標主題的 5,163 篇文件合併為本實驗中的測試文集，總共 6,346 篇。而實驗測試時所需之查詢句，

為了模擬使用者使用檢索之情形，於是從標明主題之 1,183 篇文件中，挑選出 102 篇新聞文件的標題來當做查詢短句樣本，其平均長度約為 17 個字。

4.2 檢索效能的評估方法

Non-Interpolated Average Precision Rate（NAP）以單一數值來作效能評估，是文件檢索效能相當普遍的評估方式，其式子如下：

$$NAP = \frac{\sum_{i=1}^{N} \frac{i}{Rank}}{N} \tag{26}$$

舉例來說，在檢索出來的文件中實際相關的文件被排名在第一名、第二名、第四名及第六名，則 NAP 的值為 0.854 ($NAP = \frac{\frac{1}{1}+\frac{2}{2}+\frac{3}{4}+\frac{4}{6}}{4} = 0.854$)。

4.3 實驗結果

關於實驗結果表達所用的符號，以 QE 代表 Query Expansion，QTR 代表 Query Term Reweighting，MA 代表 Model Adaptation，而 ALL 代表 QE+QTR+MA。本實驗基礎架構為混合式機率檢索，以不加入任何回饋方式之檢索正確度作為我們比較的基本系統（Baseline），並使用 NAP 做評估量測。

A.不同資訊回饋法在檢索效能之影響

本部分實驗取前一回檢索分數排名於前 6 名之文件（*N*=6）做回饋，在查詢句擴充裡，每次找出分數最高之前 6 個詞（*K*=6），並對查詢句做比對，刪除重複部分，剩下的詞便可加入查詢句成為一擴充之新查詢句。我們得到基本系統的 NAP 為 66.4%，不同資訊回饋法得到的文件檢索 NAP 如下表所示

表一、不同資訊回饋演算法之實驗結果比較

回饋方式	QTR	MA	QE	QTR+QE	QTR+MA	QE+MA	ALL
NAP (%)	66.1	72.1	72.2	72.3	77.7	78.3	81.1

本實驗比較各資訊回饋演算法之效果，從圖表中可以看出各方法對於檢索之準確度皆有提昇。在查詢詞權重重調整方面，比較 QTR、QTR+QE 與 QTR+MA 這三組實驗發現，詞權重之調整雖然單獨使用之改善不甚明顯，但是若有較好的語言模型調整，則檢索效果的提昇會更顯著。此外，我們對各 QTR、QE 與 MA 這三種方式做不同的合併，亦有不同之提昇效果顯現出來，而全部合併時，檢索之準確度提昇最多。接下來，比較文件模型調整時使用 ML 與 MAP 之效果，我們以 QTR、QE 與 MA 三者合併之實驗來比較。

表二、混合式機率檢索模型調整使用 ML 及 MAP 之實驗結果比較

回饋方式	ALL (ML)	ALL (MAP)
NAP (%)	81.1	82.4

從上表中可以看出在改用 MAP 去調整文件模型後，其檢索效果的確有改進。

B.不同資訊回饋量之影響

本部分實驗將設定不同之查詢詞增加數與回饋時的文件數，並且針對文件模型調整的 ML 與 MAP 做比較，同樣地，實驗是以三者合併(ALL)的效果來觀察。

表三、不同資訊回饋量之實驗結果比較

回饋方式	ML(*K*=10, *N*=6)	ML(*K*=10, *N*=10)	ML(*K*=6, *N*=10)	MAP(*K*=6, *N*=10)
NAP (%)	80.7	81.3	82	83.5

表中 *K* 是指查詢句擴充裡，算完詞分數後所挑選的詞數量；*N* 是指使用於回饋程序中的文件數。我們很明顯的看出，增加詞的個數，造成不適當的詞加進查詢句的機會提昇，如此會造成檢索效果的降低；而提昇回饋之文件數，可以補充更多的資訊於系統內，並進一步地提高檢索效能。

C. MAP 調整之不同參數初始比較

在貝氏的文件模型調整中，對於 Hyper parameter 的初始，需使用一係數 ε 做調整，其範圍 $0 < \varepsilon < 1$，我們針對不同的 ε，以合併 QTR、QE 與 MA 的實驗結果找出可能最佳值。

表四、不同 ε 值之實驗結果比較

ε	0.2	0.5	0.8
NAP (%)	82.44	82.41	82.38

由表中的結果可看出，不同的係數雖然結果不同，但相差量是很少的，不過在其他的實驗比較中，仍以 $\varepsilon = 0.2$ 為主。

D.較短查詢句與較長查詢句之比較

在這一小節裡，我們將實驗樣本的長度分成兩群，每一樣本大於 15 個中文字的分成一群，小於或等於十五個字的分成另一群，以觀察不同長度對實驗結果之影響，同樣以 QTR、QE 與 MA 合併的實驗觀察。

表五、長句與短句之實驗結果（NAP（%））比較

平均長度(字)	基本系統	本論文方法
12.64	63.4	84.5
20.47	68.7	80.9

從表中看出，較長查詢句可用的資訊量包含較多，所以在基本系統可表現較好，但是同時參雜了一些多餘字出現，所以在回饋之後的效果容易低於較短的查詢句，不過，在這些實驗中發現到，檢索最後效果的好壞不在於查訊句的短或長，使用者所下的查詢句，其意思的表達是否明確，才是檢索效果的關鍵。

E.臺灣電子報之實驗結果

本小節實驗目的在做一組對照的結果，其資料來源為 YAHOO 奇摩網站上搜得之電子報，作為被查詢的新聞文件其範圍從西元 2002 年 1 月 25 日至 5 月 21 日與西元 2002 年 11 月 12 日至西元 2003 年 1 月 11 日總共有 7,800 篇，並取出 110 新聞文件之標題作為查詢句之樣本，背景語言模型為 CKIP 平衡語料庫。

表六、臺灣電子報實驗結果

回饋方式	基本系統	QTR	MA	QE	QTR+QE	QTR+MA	QE+MA	ALL	ALL (MAP)
NAP (%)	85.3	84.0	89.5	91.1	90.4	91.6	92.8	93.7	93.8

從實驗結果可以看出兩種實驗文集的差異，這是因為實驗資料只利用詞典去斷詞，並無做其他的處理，並且兩種文集之書寫表達方式有很多差異的存在，交互影響所造成的結果。雖然如此，但本方法之效果大致上的表現是差不多的。

5. 結論與未來研究方向

本論文於混合式檢索的架構上研究相關資訊回饋的效果，並證明我們的方法可使檢索之最後效能提昇許多；此外，在實驗中發現到，檢索系統的關鍵有二：文件模型的好壞與相關文件的回饋數；當我們調出較好的文件模型時，對於查詢詞權重、查詢句擴充或者是合併來檢索，結果都會更加優秀；相關文件的回饋數量增加時，能夠補充的資訊也會相對的增加，這有助於檢索效果的提昇。文件模型的調整方面，我們從實驗中發現了 MAP 這種加入事前資訊的準則比使用 ML 的準確度多出 1.33%左右，是可以進一步地調出更好的文件模型。在查詢句擴充方面，我們從實驗發現到“回饋有如兩面刃”這個事實，當加入查詢句的詞無法控制時，便有可能出現不適當的詞加入查詢句，使得原本句子的表達走樣，這在自動的回饋裡是不可避免的現象，所以在一個實踐的系統中必須讓使用者能夠自行判斷與干涉，如此或許才可確實將使用者想要閱讀的文件或網頁拉抬其檢索的分數。

未來，在查詢句擴充中，可嘗試不同之挑選詞的方式。我們亦可改良本回饋方式，以配合加入潛在語意資訊與增加混合式檢索系統的混合數，以提昇查詢句所能提供的資訊，使其有可能再次提高系統檢索的能力。另外，對於查詢句與文件來說，這兩者便是檢索的主角，我們目前檢索的實驗中，對於這兩者相似度的計算，就只是利用到詞頻的變化，若可以加入自然語言處理的相關技術，針對這兩者做語意、語法等結構的分析，使檢索時能夠使用之資訊量增加，並進而改良本論文內相關資訊回饋的方法，也將是提昇檢索效能之方向。一般而言，文件模型的好壞影響著檢索效能，不管是以 ML 或是以 MAP 方式去調整文件，最後都可以有明顯的改善，這說明了文件模型若有更好的調整方法，則檢索系統便有機會提供給使用者更好的搜尋結果。

參考文獻

[1] Ricardo Baeza-Yates and Berthier Ribeiro-Neto, *Modern Information Retrieval*, Addison-Wesley Longman, pages 118-123, May 1999.

[2] Claudio Carpineto, Renato De Mori, Glovanni Romano and Brigitee Bigi, "An Information-Theoretic Approach to Automatic Query Expansion", *ACM Transactions on Information Systems*, Vol.19, No. 1, pages 1–27, January 2001.

[3] Claudio Carpineto, Glovanni Romano and Vittorio Giannini, "Improving Retrieval Feedback with Multiple Term-Ranking Function Combination", *ACM Transactions on Information Systems*, Vol. 20, No. 3, pages 259–290, July 2002.

[4] Berlin Chen, Hsin-min Wang, and Lin-shan Lee, "An HMM/N-gram-based Linguistic Approach for Mandarin Spoken Document Retrieval", *In Proceedings of the 7th European Conference on Speech Communication and Technology* (*Eurospeech*2001), Aalborg Demark, Sept. 2001.

[5] 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*, Vol. 39, No. 1, pages 1-38, 1977.

[6] Jelinek Frederick, *Statistical Methods for Speech Recognition*, The MIT Press, Cambridge, Massachusetts, 1997.

[7] Jean-Luc Gauvain and Chin-Hui Lee, "Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observation of Markov Chains", *IEEE Transactions on Speech And Audio Processing*, Vol. 2, No. 4, pages 291-298, April 1994.

[8] Donna Harman, "Relevance Feedback Revisited", *In Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1-10, 1992.

[9] Xuedong Huang, Alex Acero and Hsiao-Wuen Hon, *Spoken Language Processing-A Guide to Theory, Algorithm, and System Development*, Microsoft Research, Prentice Hall PTR, pages 73-132, 2001.

[10] Qiang Huo and Chin-Hui Lee, "On-Line Adaptive Learning of the Continuous Density Hidden Markov Model Based on Approximate Recursive Bayes Estimate", *IEEE Transactions on Speech And Audio Processing*, Vol. 5, No. 2, pages 161-172, March 1997.

[11] David R. H. Miller, Tim Leek and Richard M. Schwartz, "A Hidden Markov Model Information Retrieval System ", *In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 214-221, 1999.

[12] Jay M. Ponte and W. Bruce Croft, "A Language Modeling Approach to Information Retrieval", *In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275-281, 1998.

[13] L. Rabiner and Biing-Hwang Juang, "An introduction to hidden Markov models", *IEEE Signal Processing Magazine*, Vol. 3, Issue: 1, pages 4 –16, Jan 1986.

[14] S. E. Robertson, S. Walker, and M. Beaulieu, "Okapi at TREC-7: Automatic Ad Hoc, Filtering, VLC and Interactive Track", *In Proceedings of the 7th Text Retrieval Conference* (*TREC*-7), pages 253-264, 1999.

[15] S. E. Robertson, S. Walker, "Okapi/Keenbow at TREC-8", *In Proceedings of the 8th Text Retrieval Conference* (*TREC*-8), pages 151-162, 1999.

[16] F. Song and W. Bruce Croft, "A General Language Model for Information Retrieval", *In Proceedings of the 8th International Conference on Information and Knowledge Management* (*CIKM*99'), ACM Press, pages 93-96, 1999.

[17]李建志,"應用混合式機率模型於新聞資訊檢索之研究", 碩士論文, 成功大學資訊工程學系, 2002.

[18] CKIP, http://godel.iis.sinica.edu.tw, 中央究院資訊科學研究所詞庫小組。

[19] Google 搜尋說明,http://www.google.com.tw/intl/zh-TW/help.html。