

# 應用文件重排序與局部查詢擴展於中文文件檢索之研究

## Improving Retrieval Effectiveness by Document Re-ranking and Local Expansion

王文祺、林伯慎

國立台灣科技大學資訊管理系

Email: [M9409112@mail.ntust.edu.tw](mailto:M9409112@mail.ntust.edu.tw), [bslin@cs.ntust.edu.tw](mailto:bslin@cs.ntust.edu.tw)

### 摘要

在資訊檢索領域中，查詢擴展(Query Expansion)已成為提升檢索系統效能的重要技術之一。而查詢擴展的技術中又以局部查詢擴展(Local Expansion)對於檢索效能的提昇最為顯著。局部查詢擴展是分析初始檢索結果的前幾篇文件，從中選出擴展詞，但此方法有一項缺點，就是當這些文件中與查詢句不相關的文件較多時，所選出的擴展詞可能與查詢句不相關，若把這些擴展詞加入查詢句做檢索，會造成查詢偏移(Query Drift)的情形，降低檢索效能。

因此，在本論文中針對局部查詢擴展的缺點做改進，使用文件重排序(Document Re-ranking)的方法，在作擴展前先將初始檢索輸出的文件重新排序，讓相關的文件盡量往前排，以提升局部查詢擴展的精確性。本論文研究了三種文件重排序法，實驗結果顯示其皆能有效的提昇檢索效能。且我們更進一步地探討這三種重排序法間的互補性，將其作結合，實驗結果顯示重排序法間的結合能更有效的提昇檢索效能，將檢索的平均精確率(MAP)從 0.3727 提升至 0.3956，前 10 篇文件的精確率(P@10)從 0.4929 提升至 0.5595。

關鍵字：資訊檢索、局部查詢擴展、查詢偏移、文件重排序

### 1. 序論

在現今資訊量極為龐大的時代，要在大量的資料中找出符合使用者需求的資料，是資訊檢索領域困難的課題。目前使用者與檢索系統之間的互動，多是以詞為主，系統也多是以詞作為檢索的單位。然而使用者下達檢索詞時，在表達特定

的檢索概念，可能會因為使用不同的詞，而檢索出不同的結果，有些符合查詢概念的文件，可能會因為字詞的使用不同，而無法被檢索出來。這都是「字詞差異」(Word Mismatch)造成的問題。

查詢擴展(Query Expansion)為解決上述問題的方法，其基本概念是把一些與查詢主題相關的詞—稱作擴展詞(Expanded Words)—加入到原始查詢句中，擴展原查詢句的概念，以提高一些原本未被檢索到的文件被檢索出來的機率，使查詢結果更為精確。在本論文中使用的查詢擴展方法是局部查詢擴展(Local Expansion)[7、10、11、13、14、18]，其概念是先對使用者所下達的查詢句做第一次的檢索，篩選出排名前面的文件進行分析，將這些文件中重要性較高的詞作為擴展詞，加到原查詢句中，再進行檢索。由於擴展詞增加了查詢概念的涵蓋度，因此可以提昇檢索效能。

然而，局部查詢擴展有一樣弱點，就是當第一次的檢索結果不佳時，排名前面的文件中，包含相關文件的比率相對地比較低，如果從這些文件選取擴展詞，很可能會選出與查詢主題不相關的詞，而造成查詢的偏移(Query Drift)。為了彌補局部查詢擴展的弱點，可以使用文件重排序(Document Re-ranking)的技術[7、9、12、15、16、17、18、19、20]，在第一次檢索結果輸出後，用更精確的演算法對輸出的文件重新排序，讓排序在前面的文件中，能涵蓋較多與查詢相關的文件，以改進擴展詞的品質，進而提升檢索的效能。

本論文的研究主要是利用文件重排序來改進局部查詢擴展。在論文中提出了三種文件重排序演算法，分別是概念查詢法、文件分群法與局部鏈結法。而實驗結果顯示，這三種文件重排序都能夠有效地提昇檢索效能，而其中又以局部鏈結法效果最佳。另外，利用重排序法之間的互補性而加以結合，可以進一步達到更好的效能。在同時結合三種方法的情況下，平均精確率可以從 0.3727 提升至 0.3956，前 10 篇文件精確率則可以從 0.4929 提升至 0.5595。

此外，我們也探討擴展詞的過濾，希望藉由過濾的方法，將與查詢句較不相關的擴展詞過濾掉，以提昇檢索效能。實驗結果也顯示了擴展詞的過濾確實對於檢索效能有正面的效益。

本論文的第二章中為實驗語料介紹與基礎檢索模型。第三章中將介紹三種文件重排序的方法與重排序方法間的結合。第四章為擴展詞過濾方法。第五章為為結論與未來研究方向。

## 2. 研究方法

### 2.1 實驗語料與評估準則

本論文使用的實驗語料為中華民國計算語言學學會所發行的「CIRB030 中文資訊檢索測試集」(NTCIR-3 中文語料部份)，此測試集共包含三個部分：問題集、文件集和答案集。文件集皆為一般的新聞文件，包含了七個部份，一共有 381375 篇新聞文件。問題集一共包含了 42 個查詢問題。圖 1 為一個查詢的範例。在本

論文中的實驗均是以<DESC>標籤的內容作為查詢句。在答案集中每份文件均標記了和查詢問題的相關度，分為四個層級：非常相關、相關、部分相關與不相關。檢索輸出文件的判別方式分為兩種，一種為嚴謹相關（Rigid Relevance），也就是把「非常相關」和「相關」視為相關，其它視為不相關；另一種為寬鬆相關（Relax Relevance），則是把「非常相關」、「相關」和「部分相關」均視為相關。本論文中均以“寬鬆相關”作為判別標準。

```

<TOPIC>
<NUM>001</NUM>
<SLANG>CH</SLANG>
<TLANG>CH</TLANG>
<TITLE>漢代文物大展</TITLE>
<DESC>
查詢故宮博物院所舉辦之千禧漢代文物大展相關內容
</DESC>
<NARR>
台灣的故宮博物院是著名的典藏中國寶物的博物館，有關漢朝的典藏品展現了西元前206年到西元220年，中國漢朝的強盛與偉大。對於故宮博物院所舉辦之千禧漢代文物大展之說明，例如展出的文物種類、對於展出文物之介紹、展出時間、故宮的籌畫過程、合作單位等，以及展出後的成果與民眾的反應視為相關。非本次展覽內容之漢代文物介紹，以及其他展覽活動之介紹視為不相關。
</NARR>
<CONC>
漢代，文物大展，故宮博物院，歷史
</CONC>
</TOPIC>

```

圖 1 CIRB030 問題集之範例

在評估方法上本論文採用 TREC 所定的標準評估程式(TREC\_EVAL) [1]來評估檢索效能，使用其中的「平均精確率(MAP)」和「前 N 篇文件的精確率(Precision: At N docs)」作為效能指標，其定義如下：

- 平均精確率(MAP)：先對各個查詢問題，計算檢索出文件的平均精確率(AP)，再對所有查詢問題作平均。其計算方法如下公式：

$$AP = \frac{1}{r} \sum_{i=1}^r \frac{i}{Doc(i)} \quad (1)$$

$$MAP = \frac{1}{Q} \sum_{j=1}^Q AP_j \quad (2)$$

$r$  是檢索系統對該查詢問題，所檢索出文件中相關文件的數目， $Doc(i)$  則是檢索出來的第  $i$  篇相關文件的排名值。 $Q$  是查詢問題的總數。 $AP_j$  是第  $j$

個查詢問題的平均精確率。 $MAP$  表示所有查詢問題的平均精確率的平均。

- Precision: At N docs：表示在檢索出 N 篇文件時的精確率。

對所有查詢問題的前 N 篇文件精確率計算如下式：

$$P @ N = \frac{1}{Q} \times \sum_{j=1}^Q p_j \quad (3)$$

$p_j$  表示第  $j$  個查詢問題的前  $N$  篇文件精確率。

## 2.2 檢索系統架構

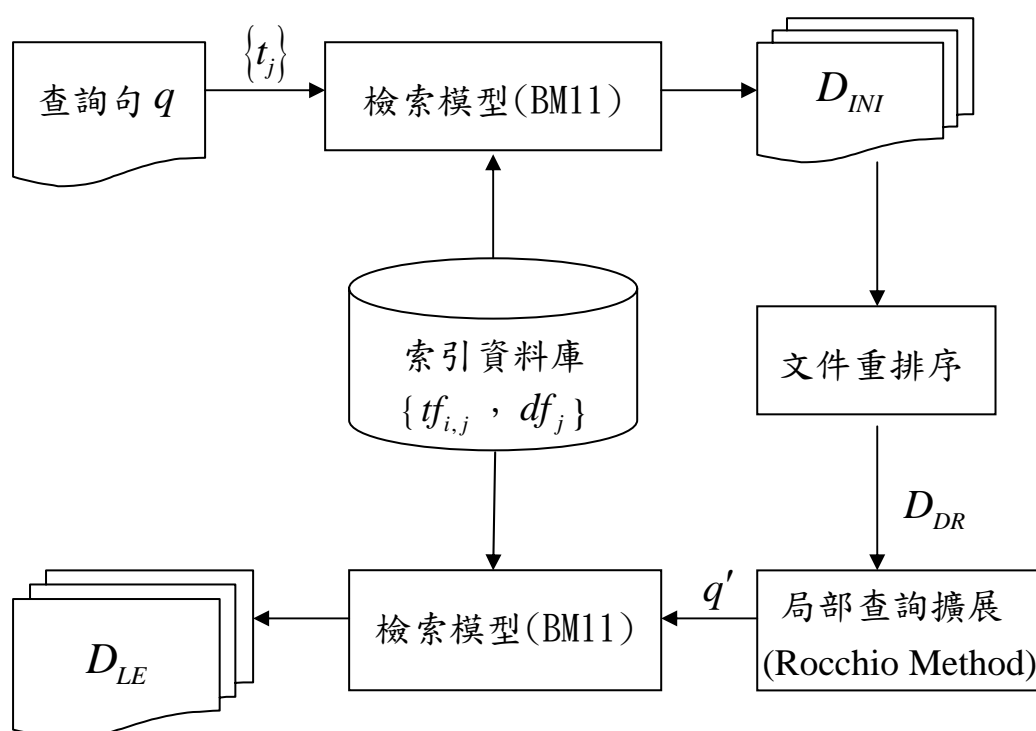


圖 2 檢索系統架構圖

本論文的檢索系統架構如圖 2 所示。由於中文的詞與詞之間並沒有明顯的間隔，而檢索系統的演算法是以「詞」為基礎單位，因此所有檢索文件或查詢句都必須要先經過斷詞處理。本實驗中的斷詞工具是採用中研院的線上斷詞系統[8]。我們把新聞文件庫中的每份新聞文件進行斷詞後，只保留表 1 中所示特定詞類的詞，並計算詞頻( $tf_{i,j}$ )和文件頻( $df_j$ )，存入索引資料庫中。

檢索時，查詢句  $q$  經斷詞後可以得到查詢詞集( $\{t_j\}$ )，再和索引資料庫中各篇文件以 BM11 模型進行相似度比對，可初步篩選出和查詢句最相關的前 N 篇文

件( $D_{INI}$ )。接著進行文件重排序，重排序後的文件( $D_{DR}$ )挑選前  $R$  篇作局部查詢擴展，找出擴展詞加入原查詢句，再用擴展後的查詢句  $q'$  進行檢索，而產生最後的文件的排序( $D_{LE}$ )。

本論文所使用的檢索模型為 Robertson and Walker 提出的 OKAPI BM11 檢索模型[2]，其計算式如下：

$$BM11(d_i, q) = \sum_{j=1}^T c_j \times tf'_{i,j} \times idf'_j$$

$$tf'_{i,j} \square \frac{tf_{i,j}}{tf_{i,j} + \frac{dl_i}{dl}}$$

$$idf'_j \square \log\left(\frac{n - df_j + 0.5}{df_j + 0.5}\right) \quad (4)$$

$c_j$  是關鍵詞  $j$  在查詢句中出現的次數， $n$  是文件總數， $dl_i$  是文件  $d_i$  的長度， $\overline{dl}$  是所有文件的平均長度。上式的  $tf'_{i,j}$  和  $idf'_j$  是 BM11 模型對詞頻和反文件頻的修正。

在局部查詢擴展的方法上，本論文採用「Rocchio Blind Feedback」[3][4]。Rocchio 演算法中將關鍵詞的權重定義如下：

$$w_j = \frac{1}{R} \times \sum_{i=1}^R tf'_{i,j} - \beta \times \frac{1}{S} \times \sum_{i=1}^S tf'_{i,j} \quad (5)$$

$R$  指的是排名前  $R$  篇的文件， $S$  則是其餘的文件 ( $S = n - R$ ,  $n$  代表總文件數)， $\beta$  為可調整的參數。如果第  $j$  個關鍵詞在前  $R$  篇文件中出現頻率高且在其餘文件中出現頻率低，此時  $w_j$  高，表示這個關鍵詞對於檢索出前  $R$  篇文件具有鑑別力。因此，權重  $w_j$  可視為關鍵詞鑑別力的度量，局部查詢擴展即可用它來挑選最具鑑別力的詞作為擴展詞。

普通名詞	Na
專有名詞	Nb

地方詞	Nc
時間詞	Nd
外文標記	FW
動作不及物動詞	VA
動作類及物動詞	VB
動作及物動詞	VC
動作接地方賓語動詞	VCL
動作句賓動詞	VE
狀態不及物動詞	VH
狀態使動動詞	VHC
狀態類及物動詞	VI
狀態及物動詞	VJ

表 1 斷詞後保留的詞類標記

### 3. 文件重排序

本論文共研究了三種文件重排序的方法，分別是：

1. 概念查詢法。
2. 文件分群法。
3. 局部鏈結法。

下面章節將詳細介紹此三種方法。

#### 3.1 概念查詢法

概念查詢的方法 Qiu[5]提出，最初是應用在全域查詢擴展上。其想法是：不應該將每個查詢關鍵詞個別獨立地看待，而應該把整個查詢句當成整體的查詢概念。如圖 3 的這個例子所示，如果用查詢關鍵詞「黑澤明」和「電影」個別地去擴展，所擴展出的詞很有可能偏離整個查詢的主題。但是，若把查詢句中的所有關鍵詞合併成爲一個查詢概念，所擴展出的關鍵詞就會比較接近查詢主題。

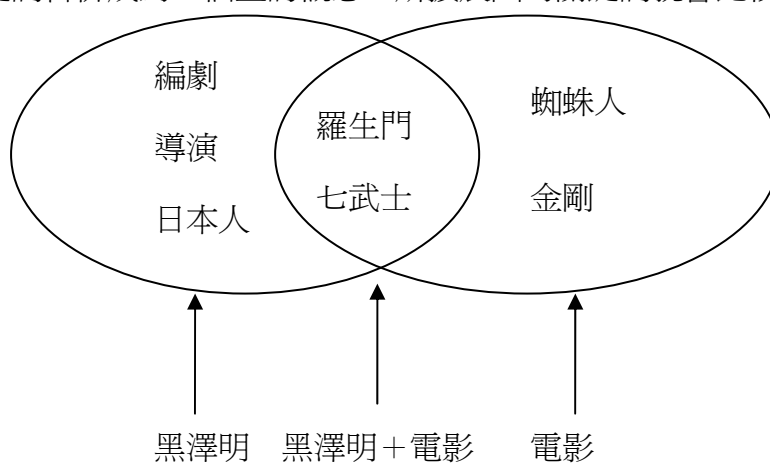


圖 3 概念查詢擴展示意圖

我們應用概念查詢的演算法，篩選出與查詢概念關聯度較高的一些關鍵詞來建立動態的關聯詞典，並利用此關聯詞典來進行文件重排序。步驟如下：

1. 對初步篩選的前  $N$  篇文件中所有的詞，建立詞向量  $\bar{t}_j$

$$\begin{aligned}\bar{t}_j &= \bar{d}_j / |\bar{d}_j| \\ \bar{d}_j &= (d_{1,j}, d_{2,j}, \dots, d_{N,j})^T \\ d_{i,j} &= (0.5 + 0.5 \frac{tf_{i,j}}{\max_i(tf_{i,j})}) \times \log(\frac{m}{|d_i|})\end{aligned}\quad (6)$$

$N$  表示重排序文件數， $tf_{i,j}$  表示詞  $j$  在文件  $i$  的詞頻， $\max_i(tf_{i,j})$  表示詞  $j$  在此  $N$  篇文件中出現過的最大次數， $m$  表示在此  $N$  篇文件中的詞的總數， $|d_i|$  表示在文件  $i$  中相異詞的數目(並非詞的總個數)。

2. 建立「概念查詢向量」(Concept Query Vector)。

$$\bar{q}_c = \sum_{t_j \in q} q_j \cdot \bar{t}_j \quad (7)$$

$q$  為查詢句， $\bar{t}_j$  為公式(6)所定義的詞向量， $q_j$  為關鍵詞  $t_j$  的權重，在本實驗中以詞頻作為權重。

3. 計算「概念查詢向量」與各關鍵詞向量的關聯度。

$$r(\bar{q}_c, \bar{t}_j) = \frac{\bar{q}_c^T \cdot \bar{t}_j}{\sum_{t_j \in q} q_j} \quad (8)$$

$\bar{t}_j$  表示公式(6)所定義的詞向量， $q_j$  為關鍵詞  $t_j$  的權重， $\bar{q}_c$  為公式(7)所定義的概念式查詢向量。

4. 選出與查詢概念關聯度較高的一些關鍵詞，建立動態關聯詞典  $C$ 。關聯詞典中必須記錄每個關聯詞與查詢概念的關聯度  $r(\bar{q}_c, \bar{t}_j)$ 。

有了關聯詞典後，我們便可根據下式修正排序分數。

$$s'_i = \alpha \times s_i + (1 - \alpha) \times \sum_{t_j \in (C \cap d_i)} r(\bar{q}_c, \bar{t}_j) \quad (9)$$

$s_i$  表示文件  $d_i$  在第一次檢索得到的分數， $s'_i$  表示更新後的文件  $d_i$

的分數， $C$  代表關聯詞典， $c \cap d_i$  代表同時出現在關聯詞典  $C$  與文件  $d_i$  的關鍵詞。 $\alpha$  代表原始排序分數  $s_i$  在重排序分數  $s_i'$  中所佔的比率， $0 \leq \alpha \leq 1$ 。

### 3.2 文件分群法

本方法的概念是希望藉由分群的分法把內容相似的文件分在同一群集中，再利用群集與查詢句的關聯度來修正排序分數[20]。在圖 4 的查詢範例中，我們看到文件 A 因為包含了“漢代、文物、大展”這三個查詢關鍵詞，所以會得到較高的檢索分數。而文件 B 雖然看起來是相關的文件，卻因沒有包含查詢句中的關鍵詞，以致於檢索分數為 0。然而，文件 A 和文件 B 其實相當地類似，也包含一些相同的關鍵詞(例如：東漢、馬王堆)。利用文件分群的方法，文件 A 和文件 B 因為內容相似，可能會被分在同一群集中。如果此群集和查詢句的關聯度高，文件 B 就可以靠著所屬群集的關聯度分數而提升其排名。

本實驗的分群演算法是採用 K-means 分群法[6]，在大量高維度的資料中，找出具有代表性的 K 個資料點，稱為群中心(centroids)。之後每份文件就可以計算其所屬群集和查詢句的關聯度(群中心和查詢向量的 cosine 夾角)，用來修正排序分數。

實驗演算法如下：

1. 建立查詢句向量  $\vec{q}$  與每份文件的文件向量。
2. 利用 K-means 演算法，計算出各群集的群中心向量  $\vec{c}_k$ 。K-means 演算法步驟：
  - a. 隨機選取 K 個文件向量，這 K 個文件向量就為初始的群中心向量。
  - b. 對每一文件向量計算其與 K 個群中心向量的距離，找出距離最接近的群中心，並分群進此群集中。
  - c. 全部文件分群完畢後重新計算各群集的群中心向量。
  - d. 重複 b、c 步驟直到所有群集內的資料皆不再變動為止。

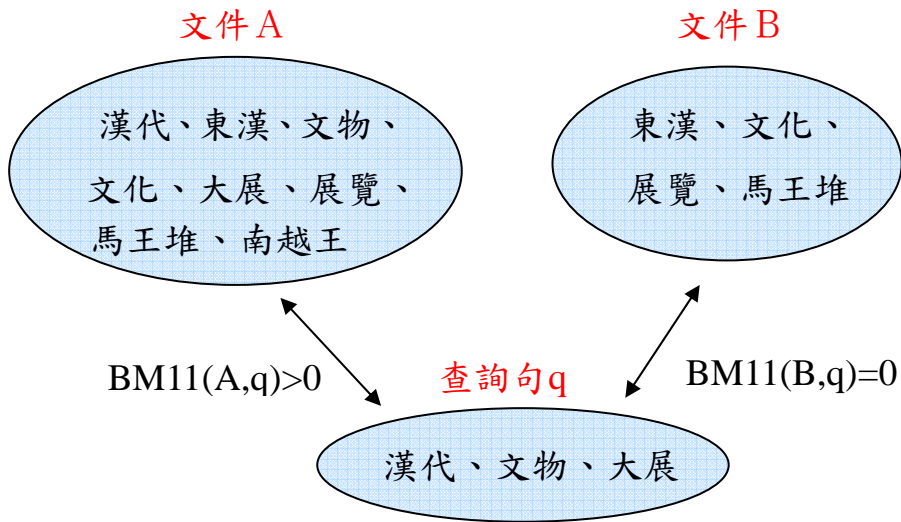


圖 4 文件檢索分數示意圖(BM11)

3. 計算查詢句與群集  $k$  的關聯度。

$$\cos(\bar{q}, \bar{c}_k) = \frac{\bar{q} \cdot \bar{c}_k}{|\bar{q}| \times |\bar{c}_k|} \quad (10)$$

$\bar{c}_k$  表示第  $k$  群的群中心的向量。特別注意的是公式(10)中，屬於同一群集的文件，並不個別計算其與查詢句的關聯度，而是共用所屬群集的關聯度。

4. 每篇文件重排序的分數則可以用原排序分數和群集關聯度兩者加權得到，如下公式：

$$s'_i = \alpha \times s_i + (1 - \alpha) \times \cos(\bar{q}, \bar{c}_k) \quad (11)$$

$s'_i$ 、 $s_i$ 、 $\alpha$  之定義和公式(9)中相同。文件  $i$  屬於群集  $k$ 。

### 3.3 局部鏈結法

在 BM11 的檢索模型中，分析查詢句與各篇文件的關聯時，僅是計算個別查詢關鍵詞對於文件的權重並加總起來。但是，各個關鍵詞之間可能有語意上的關聯或限制，若分別去計算權重，可能導致只包含部份查詢概念的文件其關聯度反而會比包含全部查詢概念的文件來得高。例如：我們欲查詢「漢代文物大展」的相關文件時，可能會有「汽車大展」的文件因「大展」這個詞出現很多次而排名較前面之情形，造成檢索精確率的下降。為了修正這種不合理的現象，我們利用

查詢詞之間「局部鏈結」的統計來修正排序分數。局部鏈結的統計以查詢句中相鄰的兩個關鍵詞形成的「關鍵詞組」(keyword pair)做為統計的單位，例如，查詢句“漢代、文物、大展”就可被分為“漢代、文物”和“文物、大展”兩個詞組。每一詞組中的兩個詞(例如“漢代”與“文物”)如果在同一篇文件中出現的位置足夠接近，稱之為一個局部鏈結(Local Link)。這兩個詞在文件中的距離以區間(frame)來限制，例如，區間數設為 50 就表示同一組內的兩個詞距離必須小於 50 才算是一個局部鏈結。

實驗演算法為：

1. 對查詢句  $q$  中每個關鍵詞組  $(t_j, t_k)$ ，統計其在文件  $i$  中的局部鏈結數  $L_{j,k}$

以及局部鏈結文件頻  $df_{j,k}$ 。局部鏈結數  $L_{j,k}$  的計算方式為，若  $(t_j, t_k)$  出現在文件  $i$  中的區間一次，則局部鏈結數加 1。局部鏈結文件頻  $df_{j,k}$  則是計算  $(t_j, t_k)$  之局部鏈結共在多少文件中出現過。

2. 對原始文件分數作加權。

$$s'_i = \alpha \times s_i + (1 - \alpha) \times \left( \sum_{(t_j, t_k) \in q} L_{j,k} \times idf_{j,k} \right)$$

$$idf_{j,k} = \log \frac{n}{df_{j,k}} \quad (\text{公式 12})$$

$(t_j, t_k)$  表示查詢句中某一關鍵詞組。 $n$  為總文件數。

$s'_i$ 、 $s_i$ 、 $\alpha$  之定義和公式(9)中相同。

### 3.4 文件重排序方法之結合

本論文的三種文件重排序方法運用了不同的概念：概念查詢法是先利用查詢概念產生出關聯詞(非原查詢詞)，並計算其關聯度以改進文件的排序；文件分群法是利用「群集的關聯度」來改進文件的排序；局部鏈結法則是利用「查詢詞局部共現」特性，讓查詢語意更準確的文件可以提升排名。前兩種方法重在查詢概念的「擴展」，將一些相關但可能不包含查詢詞的文件有機會往前排，其差異是擴展的方式不同。第三種方法則重在查詢語意的「準確」，所影響的文件是那些含有查詢詞的文件。由於這些方法的設計理念和作用的範圍各有不同，我們想進一步探討這些方法是否能彼此互補，並適當地結合。我們將重排序方法結合的方式如下：

$$R_i = \beta \times R_{i,1} + (1 - \beta) \times R_{i,2}$$

$$R'_i = \gamma \times R_i + (1 - \gamma) \times R_{i,3}$$

$$s'_i = \alpha \times s_i + (1 - \alpha) \times R'_i \quad (13)$$

$R_{i,1}$ 、 $R_{i,2}$ 、 $R_{i,3}$  為文件  $d_i$  在任兩種文件重排序方法所得到的修正部份分數(公式 9、11、12)。 $\beta$  為權重參數。

### 3.5 實驗結果

首先對於檢索系統進行實驗，發現在局部查詢擴展中取前 10 篇文件當作是相關文件(R=10)，並取權重排名前 80 個詞當作擴展詞加入查詢句(E=80)，可達到最佳的檢索效能，故本節實驗數據皆以此設定。實驗之系統架構圖參考圖 2。

表 2 文件重排序方法間之比較(無局部查詢擴展)

		MAP	P@10	P@100
Baseline(BM11 only)		0.2429	0.4476	0.1907
加入文件 重排序	概念查詢法	0.2426	0.4643	0.1950
	文件分群法	0.2600	0.4643	<b>0.2195</b>
	局部鏈結法	0.2520	0.4690	0.2040
	三種結合	<b>0.2723</b>	<b>0.4786</b>	0.2150

表 3 文件重排序方法間之比較(有局部查詢擴展)

		MAP	P@10	P@100
BM11 + Rocchio		0.3727	0.4929	0.2767
加入文件 重排序	概念查詢法	0.3821	0.5262	<b>0.2838</b>
	文件分群法	0.3831	0.5238	0.2795
	局部鏈結法	0.3855	0.5405	0.2757
	三種結合	<b>0.3956</b>	<b>0.5595</b>	0.2767

表 2 顯示了文件重排序方法間之比較。對 Baseline 而言，應用文件分群的重排序方法有最佳的平均精確率。而應用局部鏈結的重排序方法，則有最佳的前 10 篇文件的精確率。應用概念式查詢的重排序法和另外兩種方法比較起來則相對比較差。表 3 結合局部查詢擴展後，三種文件重排序法的平均精確率，與 Baseline 比較起來，都有不錯的提昇效果，而提升的效能則以局部鏈結的重排序法最佳。對於前 10 篇文件的精確率來說，局部鏈結的重排序法，有相當明顯的提昇效果，為三種文件重排序方法中最佳的。

且由表 2 得知，文件重排序方法間的結合確實能夠有效的提昇檢索平均精確率和前 10 篇文件的精確率，且結合的方法確實比單用任一種文件重排序的方法

達到的效果要來的好。表 3 也顯示出結合局部查詢擴展後更能大幅提昇檢索的精確率，且全部文件重排序方法的結合所展現的檢索效能比單用任一種文件重排序法或是任兩種文件重排序方法的結合都要來的好，為所有方法中最佳的，也證明了文件重排序方法之間確實具有互補的效果。

## 4. 擴展詞過濾方法

### 4.1 擴展詞的過濾

Rocchio Method 的擴展詞權重是計算對於前 R 篇文件的「鑑別力」，並未考量這些詞是否真的與查詢主題相關，可能因而擴展出了與查詢主題不相關的詞。因此，我們進一步地研究擴展詞的過濾方式，期望藉由過濾法來篩選掉與查詢主題不相關的擴展詞，以提升檢索效能。我們定義了擴展詞和查詢句的相關度，過濾掉相關度較低的擴展詞，以降低查詢偏移(Query Drift)的可能性，提高檢索的效能。其演算法如下：

1. 計算擴展詞  $e_j$  與查詢關鍵詞  $q_k$  的相關係數(Correlation Coefficient)。

$$r_{e_j, q_k} = \frac{P_{e_j, q_k} - P_{e_j} \times P_{q_k}}{\sqrt{P_{e_j} \times (1 - P_{e_j})} \times \sqrt{P_{q_k} \times (1 - P_{q_k})}}$$

$$P_{e_j, q_k} = df_{e_j, q_k} / n$$

$$P_{e_j} = df_{e_j} / n$$

$$P_{q_k} = df_{q_k} / n \quad (14)$$

$P_{e_j, q_k}$  表示擴展詞  $e_j$  與查詢關鍵詞  $q_k$  共同出現的機率。 $P_{e_j}$ 、 $P_{q_k}$  表示擴展詞  $e_j$ 、查詢關鍵詞  $q_k$  在所有文件中出現的機率。 $df_{e_j, q_k}$  表示擴展詞  $e_j$  與查詢關鍵詞  $q_k$  共同出現的文件數。 $df_{e_j}$  表示擴展詞  $e_j$  的文件頻。 $df_{q_k}$  表示查詢關鍵詞  $q_k$  的文件頻。 $n$  表示總文件數。

2. 計算擴展詞  $e_j$  對查詢句的相關度。

$$r_{e_j} = \sum_{q_k \in q} r_{e_j, q_k} \quad (15)$$

3. 設立一門檻值來過濾掉與查詢句相關度( $r_{e_j}$ )較低的擴展詞。

## 4.2 實驗結果

表6 為使用<DESC>查詢內容作過濾對於平均精確率(MAP)及前10 篇文件精確率(P@10)的影響。No-Filter 表示未將擴展詞作過濾的動作。Filter 表示有將擴展詞過濾。由表中數據可知，用<DESC>的查詢句來過濾，所得到的MAP、P@10 和未作過濾時差不多。造成這種差別是因<DESC>中的查詢句較長，且包含一些與查詢主題不相關的詞。

表6 DESC 查詢內容作過濾對於MAP 及P@10 的影響

		No-Filter		Filter	
		MAP	P@10	MAP	P@10
BM11 + Rocchio		0.3727	0.4929	0.3727	0.5024
加入文件 重排序	概念查詢法	0.3821	0.5262	0.3820	0.5238
	文件分群法	0.3831	0.5238	0.3830	0.5214
	局部鏈結法	0.3855	0.5405	0.3846	0.5429
	三種結合	0.3956	0.5595	0.3951	0.5619

表7 為使用<TITLE>查詢內容作過濾對於平均精確率及前10 篇文件精確率的影響。與DESC 的結果作個對照，我們發現雖然作擴展詞過濾後，並不能提升平均精確率，但是對於前10 篇文件的精確率卻可以有效地提升，且以局部鏈結重排序法及三種重排序法間的結何提升幅度最大。由此可知本實驗的擴展詞過濾方法在查詢句沒有贅詞或是較少贅詞時，對於檢索效能的提升確實有幫助。

表7 TITLE 查詢內容作過濾對於MAP 及P@10 的影響

		No-Filter		Filter	
		MAP	P@10	MAP	P@10
BM11 + Rocchio		0.3341	0.4762	0.3350	0.4857
加入文件 重排序	概念查詢法	0.3377	0.4595	0.3380	0.4667
	文件分群法	0.3366	0.4881	0.3377	0.4929
	局部鏈結法	0.3381	0.4833	0.3399	0.5048
	三種結合	0.3425	0.4905	0.3449	0.5094

## 5. 結論與未來研究方向

### 5.1 結論

本論文主要研究三種文件重排序的方法，希望藉由重排序的演算法將相關文件往前排序，以改良局部查詢擴展的缺點，進一步地提升檢索效能，而實驗結果也證明此三種重排序法確實對於檢索效能有所貢獻。在 3.4 節中，我們加深探討這三種重排序法之間是否具有互補的特性，將重排序法之間作結合，以期望能更進一步地提升檢索效能，而實驗結果也證明重排序法間確實互補，能有效提升檢索效能，將平均精確率從 0.3727 提升至 0.3956，前 10 篇文件精確率從 0.4929 提升至 0.5595。

在第四章中，我們對於擴展詞的選取方式作更加深入的探討。對擴展詞作過濾的方法，實驗結果顯示其確實對於檢索效能有所提升。

### 5.2 未來研究方向

本論文的局部鏈結重排序法僅探討查詢句中相鄰的兩個查詢詞之間的語意關係，將包含較多語意鏈結的文件往前排序，但此方法容易將沒有語意關係的兩個關鍵詞作配對，降低重排序的效能。未來實驗可利用句法剖析(Parsing)，更精確地抽取出查詢句語意，以預期能達到最佳的效能。

而本論文結合的方式是利用人工的方式做參數調整，以達到最佳化，若能訂立一套自動的參數估測方式，對於檢索系統的改進將會更有幫助。

## 參考文獻

- [1] 陳光華(2004). “資訊檢索的績效評估”. 2004年現代資訊組織與檢索研討會.
- [2] S. E. Roberson & S. Walker. “Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval”. Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1994.
- [3] Rocchio, J.J. Jr.(1971).“Relevance feedback in informationRetrieval”. In the Smart system – experiments in automaticdocument processing, 313-323. Englewood Cliffs, NJ : Prentice Hall Inc.
- [4] Gerard Salton and Chris Buckley. “Improving retrieval performance by relevance feedback.” Journal of the American Society for Information Science. 1990.
- [5] Qiu, Y. & Frei, H. P. “Concept based query expansion”. Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1993, pp.160-169.
- [6] J. B. MacQueen (1967). “Some methods for classification and analysis of

- multivariate observations”, Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability”, Berkeley, University of California Press.
- [7] L.P. Yang, D.H. Ji(2005). “Chinese information retrieval based on terms and relevant terms.” ACM Transactions on Asian Language Information Processing. Vol.4,Issue 3(2005). pp.357-374
- [8] <http://ckipsvr.iis.sinica.edu.tw/>
- [9] Xu J., Croft W.B., “Query expansion using local and global document analysis.” Proceeding of the 19th annual international ACM SIGIR conference on research and development in information retrieval, 1996, pp.4-11.
- [10] Yuen-Hsien Tseng, Da-Wei Juang and, Shiu-Han Chen. “Global and Local Expansion Term Expansion for Text Retrieval.” Proceedings of the Fourth NTCIR Workshop on Evaluation of Information Retrieval, Automatic Text Summarization and Question Answering, June 2-4,2004,Tokyo,Japan.
- [11] Yuen-Hsien Tseng, Yu-Chin Tsai, and Chi-Jen Lin.“Comparison of Global Term Expansion Methods for Text Retrieval.” Proceedings of NTCIR-5 Workshop Meeting, Deceber 6-9,2005,Tokyo,Japan.
- [12] K.S. Lee, Y.C. Park, and K.S Choi. “Document Re-ranking Model Using Clusters.” Information Processing & Management, v37 n1 p1-14 Jan 2001.
- [13] Harman, D. (1992 June). “Relevance feedback revisited.” Paper presented at the Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval, New York.
- [14] ZHANG, M., SONG, R., LIN, C., MA, S., JIANG, Z., LIU, Y., et al. (2002). “Expansion-based technologies in finding relevant and new information.” Paper presented at the TERC.
- [15] Xu J., Croft, W.B. “Improving the Effectiveness of Information Retrieval with Local Context Analysis.” ACM Transactions on Information Systems, 2000.
- [16] M. Mitra., A. Singhal. And C. Buckley. “Improving Automatic Query Expansion.” In Proc. ACM SIGIR’98.
- [17] Qu, Y.L., Xu, G.W., Wang J.2000. “Rerank Method Based on Individual Thesaurus.” Proceedings of NTCIR2 Workshop.
- [18] Kamps, J. 2004. “Improving Retrieval Effectiveness by Reranking Documents Based on Controlled Vocabulary.” The 21th European Conference on Information Retrieval.
- [19] Yang Lingpeng, Ji Donghong, TangLi. 2004. “Document Re-ranking Based on Automatically Acquired Key Terms in Chinese Information Retrieval.” In Proceedings of the COLING’2004, pp. 480-486
- [20] Luk, R.W.P., Wong, K.F.2004. “Pseudo-Relevance Feedback and Title Re-ranking for Chinese IR.” In Proceedings of NTCIR4 Workshop.