文件自我擴展於自動分類之應用

Application of Document Self-Expansion to Text Categorization

曾元顯 ， 莊大衛
輔仁大學 圖書資訊學系
台北縣新莊市中正路 510 號 242
TEL: 02-29031111 ext 2333, FAX: 02-29017405
tseng@lins.fju.edu.tw

摘要：

近幾年自動分類的研究顯示，訓練文件越多，分類效果越好。然而，訓練文件的獲得需要花費相當的人力與時間，此一成本常造成使用單位導入自動分類流程的困擾。針對此問題，本文提出一種文件自我擴展方法，在沒有利用任何額外資源的情況下，全自動的增加訓練文件，以期達到降低成本、提高成效的目的。經由兩種分類測試集以及兩種分類器的實驗驗證，顯示此方法在原始訓練文件數越少時，其改進的效果越明顯。而且此改進方法，乃策略層面上的技巧，與分類器無關，亦即任何一種分類器都可以運用上面的技巧來增強其分類效果。

關鍵詞：文件分類，機器學習、文件擴展、中文、資訊檢索

一、前言

「文件主題分類」或簡稱「文件分類」（document classification or text categorization）是指依文件的「內容主旨」給定「類別」（class or category）的意思。文件分類的目的，在對文件進行分門別類的加值處理，使得文件易於管理、利用。分類後的文件，可提供使用者依主題查找文件而不受文件用詞的限制。另外，文件分類後，還可顯示館藏文件的主題分佈與範圍，對館藏文件的後續徵集與使用者的文件使用情形，提供重要的決策參考。

近年來，拜資訊技術普及運用之賜，各個企業與機構的數位文件不斷累積，數量大到難以有效的管理與利用，文件分類的需求也就遽然而生。為此，如何利用自動化的技術，快速有效的協助人工分類，來應付大量暴增的分類需求，是現今資訊服務與知識管理的重要課題。

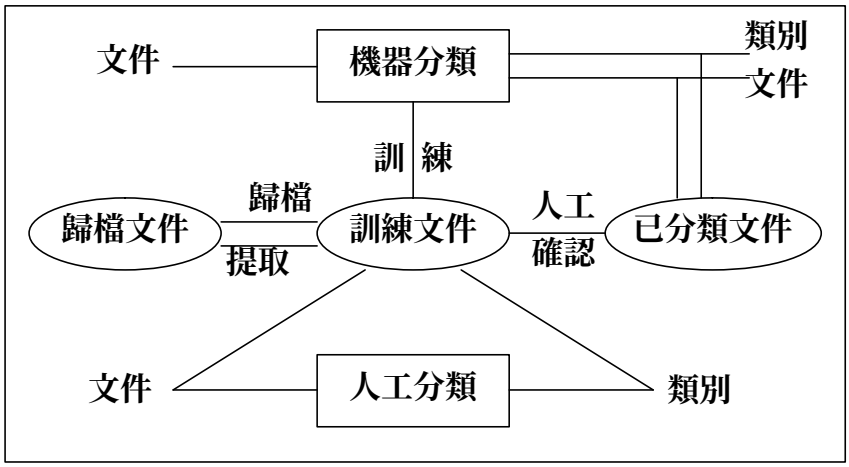
文件分類，需要瞭解文件的主題大意，才能給定類別，因此是相當高階的知識處理工作。要將文件分類自動化，必須先整理出分類時的規則，電腦才能據以執行。然而，有效的分類規則通常難以用人工分析歸納獲得。因此，機器在做

自動分類之前，還必須加以訓練，使其自動學習出人工分類的經驗與知識。

所謂訓練，就是讓機器去分析一堆「訓練文件」，如圖一所示。訓練文件記錄了人工進行文件分類的知識，這種知識相當隱晦，只是一堆（文件=>類別）的對應記錄。機器在反覆的閱讀文件以及其標示的類別後，自動歸納出一些對應規則，使其下次看到類似的文件時，可以給出適當的類別。

機器分類雖然速度快、節省大量人力，缺點則是需要事先準備相當數量的訓練文件，機器才能做出有效的分類。近幾年研究自動分類的經驗顯示，訓練文件越多，分類效果越好。因此，各個機構在導入自動分類時，需要事先準備一定數量的訓練文件。然而，訓練文件的獲得需要花費相當的人力與時間，此一成本常造成了導入自動分類流程的困擾。

此外，即便準備好了訓練文件，可能由於各個類別在整個事件機率分佈上的自然現象，個別類別的訓練篇數分佈常有極不平均的現象，亦即訓練篇數多的類別只有少數幾類，而訓練篇數少的類別則佔大多數的類別。以學術界常用做分類研究的 Reuter-21578 測試集（test collection）為例，共 90 個類別、7770 篇訓練文件，最大的 10 類，就佔了 75% 的訓練文件量、平均每類有 719 篇訓練文件，而最小的 20 類，只佔 0.5% 的訓練文件量，平均每類只有 2 篇訓練文件。這種情形在很多真實生活的測試集中常常見到，不是 Reuters 文件獨有的現象 [1]。



圖一：自動分類流程圖。

綜上所述，訓練文件數不足，乃導入自動分類時常碰到的現象。為了維持有效的分類，同時又要降低訓練文件的獲得成本，一個直覺的想法，是以自動的方法來增加訓練文件，使得即便只有少量的訓練文件時，自動分類還能達到一定的效果。這個想法的好處是它跟任何方法都無關，因此可適用於任何既有的分類方法上。

本文便是在少量訓練文件的環境下，探討如何進行有效自動分類的問題。下一節將簡略的分析過去的相關研究。第三節則介紹本文採用的「文件自我擴展」的方法，來增加訓練文件。第四節描述驗證此方法的實驗資料與環境。第五節報告實驗的結果與心得。最後一節總結本文的結論並提出未來可能的研究方向。

二、相關研究

過去數年，國內外有關文件自動分類的研究相當豐富 [2-5]。很多研究嚐試提出不同的方法，讓自動分類達到更高的成效。然而，針對訓練文件量少的情況，來提升自動分類成效的研究，則相對稀少。比較接近的研究題目有 expectation maximization（EM）[6-7] 與 co-training [8]。EM 方法是將人工尚未分類的文件以機器自動分類完後，就視其為已分類文件，而拿來訓練。這過程反覆的進行一直到分類器收斂為止。如此，在沒有人工介入的情形下，用少量人工準備的訓練資料，就可以訓練出初步的分類器，而這個分類器可以用來產生更多的「訓練文件」，來訓練分類器本身。當然這些機器產生的「訓練文件」，其分類錯誤的情形，可能較人工準備的真正訓練文件為高，依此訓練出來的分類器，有可能會不甚準確。然而即便人工準備的訓練文件也不能保證百分之百正確的（不同的人對同一篇文件會給出不同的類別，此種不一致的現象並不少見），因此，只要機器產生的「訓練文件」品質不太差，這種自我訓練大都可以增進分類器的準確度。

Co-training 的方法則是假設文件的特徵可以分成兩組獨立的集合，每一組集合可以訓練出一個分類器。每個分類器都以人工準備好的訓練資料以及個別的特徵集合訓練出初步的分類器。對於尚未分類的文件，每一個分類器都對每一個類別分出一些文件，然後將這些自動分好的文件視為「訓練文件」再去訓練這兩個分類器，如此不斷重複，直到所有未分類的文件都給定類別為止。這個作法是讓某個分類器做出來的訓練文件用來訓練另一個分類器，如此反覆互相訓練，最後彼此的分類準確度可能就越來越好。Co-training 在漸進式地互相自我訓練出兩個分類器後，真正進行文件分類時，再將這兩個分類器的結果融合，作為該文件的分類結果。

上述這兩種方法都可以從少量的訓練文件開始，利用大量多餘的未分類文件，得到不錯的分類成效。CMU 大學的 Nigam 與 Ghani 兩人的實驗中，曾利用 12 篇有標示類別的文件以及 776 篇未標示類別的文件，對 263 篇網頁文件做「課程網頁」與「非課程網頁」的分類，結果 co-training 的錯誤率為 5.4%，EM 的錯誤率為 4.3%，而如果以傳統的方法，且利用到 12+776= 788 篇有標示類別的文件做訓練，則錯誤率為 3.3% [8]。顯見 co-training 與 EM 方法，真的可以用少量訓練文件，就可達到相當好的成效。可惜 EM 與 Co-training 的計算量都很大，每次反覆訓練一次，等於又做了一次傳統分類方法的訓練。

上述兩種方法的另一個缺點，是當只有少量的訓練文件，而沒有大量多餘的未分類文件可利用時，就無法適用。這個問題會發生在前述類別分配不平均的大量小類別上。也就是說，即使想要以人工蒐集、準備資料，也會因文件出現實例太少，只得出少量的訓練文件，而沒有多餘的未分類文件可用。因而，會有無法運用 EM 或 Co-training 的情形。

三、文件自我擴展

文件自動分類的研究，已觀察到：「訓練文件越多，分類成效越好」的現象。因此在導入自動分類機制的時候，導入單位常常會碰到一個難題：要準備多少訓練文件才夠？準備得太少，效果不好；準備得太多，要投入很多人力、時間成本。如果只需要準備少量訓練文件，就可以得到宛如有很多訓練文件才能獲得的分類成效，豈不兩全其美。

為解決此一問題，本文採用一個策略，就是以自動化的方式，來獲得更多的訓練文件，以達到降低成本、提高成效的目的。跟前述相關研究不同的是，此方法只「擴展」既有的訓練文件，沒有利用到其他的資源，包括未分類文件，因此可跟其他方法一起運用，而不相衝突。

擴展（expansion）的概念，在資訊檢索領域裡常常運用。例如，對查詢而言，有「查詢擴展」（query expansion）的方法 [9]，運用在主題檢索上。其作法是增加一些查詢詞彙或修改原查詢詞彙的權重，來擴增原查詢條件，以期能獲得更佳的查詢結果。對文件而言，也有「文件擴展」的方法，運用在語音文件（如口語播報新聞）的檢索 [10]。其作法是將語音辨識成文字，再以以原查詢條件查詢乾淨的平行文件（與語音文件內容近似的文字文件，如語音新聞文字稿或同一天的新聞文字），以此查詢結果作相關回饋或查詢擴展，再運用到語音辨識過的文件查詢上，以便降低語音辨識錯誤的影響。本文提出的方法，類似資訊檢索的文件擴展法，但不需要額外的平行文件，只從原文件本身擴展，因此稱為「文件自我擴展」法。

這裡的文件自我擴展作法，是對每一個類別，從其現有的訓練文件中，擷取每篇文件的部分片段，組成新的文件，以增加該類別的訓練文件數。理想上，在「擷取每篇文件的部分片段」方面，應該要擷取可以彰顯文件主題的片段，例如利用自動摘要技術擷取文件的重要片段；在「組成新的文件」時，簡單的作法，是像遺傳演算法的基因重組那樣，將同類中數篇文件的標題或摘要，拿來交叉組合，做成新的文件。雖然這樣組出來的新文件，對人而言，也許語句不連貫，沒有實質的義意，但重要的類別用詞，就會重新分佈，而可能有助於分類器的學習、訓練。最簡單的效果，就是重要的詞彙在該類別的不同文件中重複出現了，而不重要的詞彙，則因為較少被選出來而降低其在分類中能夠扮演的角色。

基於上述的想法，本文提出兩種文件擴展法：一是摘要擴展法，另一是詞彙擴展法。

摘要擴展法的構想，是將每篇訓練文件以自動摘要法將其句子按照重要性排序，排序在前面的句子才視為該篇文件的摘要，然後依此摘要再組成新文件。在此，自動摘要法可以選擇只取文件的標題，那麼此方法將簡化成「標題擴展法」。當然，也可以用關鍵詞彙出現的次數、句子本身出現的位置，或句中出現特殊詞彙（如：「因此」、「所以」、「結論是」）的資訊，或以機器學習的方式，來一起決定句子的重要性。

在後面的實驗驗證裡，我們選擇較簡單而運用性較廣的方法，即只計數句子中包含關鍵詞彙的出現次數，來決定該句子的重要性。例如，若某個句子包含A、B、C三個關鍵詞，且他們在整篇文件中出現的次數分別為2、4、3,，那麼該句子的重要性為2+4+3=9。文件中的每個句子都以此計算其重要性，再由大到小排序。

上述所謂關鍵詞彙，是以Tseng的演算法求出的最大重複字串（maximally repeated string）[11]，做為該文件的關鍵詞彙。此方法假設文件的主題詞彙會重複出現，但並非所有的重複字串都是有用的關鍵詞，它們必須是最長的，或是出現頻率最高的，因此稱為最大重複字串。例如前兩句中「最大重複字串」出現了二次，而「重複字串」出現了三次，那麼這兩個詞都會被擷取出來。但「大重複字」此字串也出現二次，但因它是「最大重複字串」的完全子字串，所以不會被擷取出來成為關鍵詞。

一旦每篇文件的句子按上述方式排列後，便隨機的選取同一類別中任一文件的前S個句子，來累積出新文件，一直到該新文件的長度為所有文件的平均長度對為止。在後面的實驗驗證中，我們只取文件的第一個句子，期使其累積出跟舊文件差異較大的新文件。

在摘要擴展法裡，可能會引入不相干的詞彙在新文件裡，因此，在詞彙擴展法的構想裡，便希望只擷取出跟該類別有關的特徵詞彙來擴展。Yang等人曾比較了五種特徵詞選取方式，其實驗結果顯示，在五種方法裡，Chi-square與Information Gain同樣為最有效的特徵詞選取方法 [12]。若以表一中詞彙T在類別C的出現篇數分佈表示，Chi-square計算某個詞T與某類別C的相關性如下：

$$c^2(T,C) = \frac{(\mathrm{TP}\times\mathrm{TN} - \mathrm{FN}\times\mathrm{FP})^2}{(\mathrm{TP}+\mathrm{FN})(\mathrm{FP}+\mathrm{TN})(\mathrm{TP}+\mathrm{FP})(\mathrm{FN}+\mathrm{TN})}$$

對某一類別用Chi-square選詞，就是將所有的詞彙依照Chi-square值做排序，然後選出其中Chi-square值最大的前N個詞。

表一：詞彙T在類別C中的出現篇數分佈表

	詞彙 T		
類別 C		出現篇數	沒出現篇數
	是	TP	FN
	否	FP	TN

然而，Ng等人的研究觀察顯示 [13]，Chi-square會同時選出正相關與負相關的詞彙，因為正相關與負相關的詞都因Chi-sqaure的二次方計算，使得其值都變成正數，造成不出現在類別C中的詞彙，也會被選為類別C的特徵詞。這對文件分類是沒有幫助的。因為大部分的分類方法，都是依賴文件中出現某個詞，來計算其權重，而將文件分為某個類別，而不是依賴文件中沒有出現某個詞，來將文件分為該類別。因此，Ng等人提倡改用相關係數（即單邊的Chi-square）

來選詞：

$$Co(T,C) = \frac{(\mathrm{TP} \times \mathrm{TN} - \mathrm{FN} \times \mathrm{FP})}{\sqrt{(\mathrm{TP} + \mathrm{FN})(\mathrm{FP} + \mathrm{TN})(\mathrm{TP} + \mathrm{FP})(\mathrm{FN} + \mathrm{TN})}}$$

如此，與類別負相關的詞，會因為變成負數，被降低排序，而比較不可能被選為類別的特徵詞。以某一只有「營建類」與「非營建類」兩類的分類文件集為例，表二顯示其 Chi-square 與相關係數選出的前六個詞。Chi-square 選出的詞彙中，對「營建類」而言，「設備」、「公告」兩詞事實上與「營建類」為負相關，即此兩個詞在營建類的文件中極少出現，反而在「非營建類」的文件中極常出現。同理「工程」、「改善」與「非營建類」為負相關，且其相關係數分別為 -0.7880 與 -0.3169，平方後，恰為 Chi-square 值：0.6210 與 0.1004。表二顯示，相關係數選出的正相關詞彙，較符合分類需要的特徵詞彙。

表二：Chi-square 與相關係數選出的詞彙比較表

Chi-square 選詞		相關係數選詞	
營建類	非營建類	營建類	非營建類
工程 0.6210	**工程 0.6210**	工程 0.7880	設備 0.2854
改善 0.1004	**改善 0.1004**	改善 0.3169	電腦 0.2231
設備 0.0815	設備 0.0815	路面 0.2009	採購 0.2231
公告 0.0425	電腦 0.0498	道路 0.1764	公告 0.2062
路面 0.0404	採購 0.0498	新建 0.1629	系統 0.1764
道路 0.0311	公告 0.0425	土城市 0.1563	購置 0.1484

當選出的類別詞彙很少時，相關係數與 Chi-square 選出來的詞彙會有較大的差異。但當選出的詞彙數較多時，此兩種方法得到的特徵詞彙，差異就縮小了。這可以解釋為何 Yang 等人利用 Chi-sqaure 選詞，還可以得到不錯的結果。但這裡我們要用類別特徵詞來增加文件數，因為擴增的文件不會太多，因此選擇以相關係數來選詞較為妥當。

在詞彙擴展法裡，類別的特徵詞以相關係數計算、排序後，取前 K 個詞，以每個詞就視為一份新文件的方式，來增加該類別的訓練文件。

四、實驗設計

為了瞭解上述想法的效果，本文以中文文件的分類來驗證。過去的分類研究顯示，不同的分類法對不同的測試集有不同的表現。單獨以某種分類法在某種測試集上做實驗，容易產生偏向（bias）的實驗結論。因此，本文特別以兩種分類法在兩種測試集上進行交叉驗證。

這兩種測試集都於 2001 年時得自 PC home Online 的線上文件。一是 PC home 蒐集的新聞，共 12 類，為方便爾後的討論，稱其為 News 測試集，其每篇

文件的平均長度為 9.87 個句子。表三顯示其類別名稱、訓練文件與測試文件的篇數。另一測試集是 PC home 製作的網頁分類描述，共 26 類，全都是「網路與電腦」類別底下的細類，稱其為 WebDes 測試集，其每篇文件的平均長度為 2.10 個句子。表四顯示其類別名稱、訓練文件與測試文件篇數。

News 測試集全部的文件數有 914 篇，最大類與最小類的篇數相差約 30 倍。WebDes 測試集全部文件數有 1686 篇，最大類與最小類的篇數相差約 90 倍。此兩測試集的每一篇文件都是單一分類，亦即沒有任何一篇文件分在兩個或兩個以上的類別，且每一類的訓練篇數與測試篇數大多維持在 7：3 的比例，只有當該類實例太少時，才無法維持此比例。

表三：News 測試集類別名稱與文件篇數

編號	類別	訓練	測試	合計	編號	類別	訓練	測試	合計
1	產業	232	99	331	7	地方	29	12	41
2	財經	117	50	167	8	科技	18	7	25
3	政治	78	33	111	9	體育	12	4	16
4	社會	53	22	75	10	醫藥	10	4	14
5	生活	40	17	57	11	文教	10	4	14
6	娛樂	38	15	53	12	休閒	7	3	10

表四：WebDes 測試集類別名稱與文件篇數

編號	類別	訓練	測試	合計	編號	類別	訓練	測試	合計
1	網頁設計工作室	262	112	374	14	搜尋引擎連結	25	10	35
2	網頁設計教學	194	82	276	15	網站宣傳	24	10	34
3	電子賀卡	140	59	199	16	搜尋引擎	17	6	23
4	國內網站網頁搜尋	66	27	93	17	網路文化	16	6	22
5	駭客	53	22	75	18	Proxy	14	5	19
6	主題搜尋	53	22	75	19	Plug-in	12	5	17
7	ISP	49	21	70	20	固接專線	11	4	15
8	網域註冊	45	18	63	21	瀏覽器	11	4	15
9	網路資訊討論	40	17	57	22	電子商務	10	4	14
10	國外網站網頁搜尋	36	15	51	23	檔案搜尋	6	2	8
11	網路調查	35	14	49	24	BBS 文章搜尋	5	2	7
12	網路安全	33	14	47	25	Intranet	4	2	6
13	網站評鑑	27	11	38	26	電子郵件搜尋	2	2	4

在分類方法方面，近年來常被驗證效果最好的分類法為：SVM（Support Vector Machine）與 KNN（K-Nearest Neighbor），本文選擇此兩方法來實驗。我們選擇 Thorsten Joachims 製作的 SVMlight 作為 SVM 分類器 [14-15]。經過一些

測試，我們以 SVMlight 的預設環境做分類（線性分類），因為這樣效果最好，並根據 SVMlight 的使用說明以其類別輸出值的正負號做為類別分類的依據。這是因為 SVM 是二元分類器（binary classifier），因此有 C 個類別要分類時，就要做出 C 個 SVM 分類器，當某一類別的分類器其輸出值為正時，就將文件分為該類別。但我們發現很多文件都沒有任何類別分出來（所有的類別其輸出值均為負數），因此我們改變分類方法，即取類別輸出值最大者為分類的類別。在我們的實驗中，這樣改變之後，都能夠增進 SVMlight 的分類成效。

至於 KNN 分類器方面，我們實作了一套 KNN 分類系統。由於 KNN 分類器的成效非常依賴於文件相似度的正確計算，我們特別以下面公式計算：

$$Sim(d_i, q_j) = \frac{\sum_{k=1}^{T} d_{i,k} q_{j,k}}{(bytesize_{d_i})^{0.375} \sqrt{\sum_{k=1}^{T} q_{j,k}^2}}$$

其中 q_j 是輸入文件，d_i 是訓練文件，$q_{j,k}$（$d_{i,k}$）是詞彙 k 的權重，以詞頻（term frequency）及反相篇數（inverse document frequency）來加權計算，而 $bytesize_{dj}$ 為文件的長度，以位元（byte）為單位。此相似度公式乃 Singhal 等人為改進 Cosine 相似度公式的缺點而提出的，在 Singhal 等人以及某些中文 OCR 文件的（主題）檢索實驗中，bytesize 的確比 Cosine 的成效更好 [16-18]。

KNN 分類法中每一個類別的分數，以下面公式計算：

$$y(q, c_j) = \sum_{d_j \in KNN} Sim(d_i, q) y(d_i, c_j)$$

其中 y(d,c)為 1 或 0 的值，代表訓練文件 d 是否為類別 c，而 KNN 表示跟文件 q 最相近的 K 篇訓練文件的集合。在後面的實驗中，K 都取 20，並以 y 值中最大者的類別 c，做為文件 q 的類別。

文件分類的成效，受很多因素影響 [2-3]，其中之一就是用來分類的特徵詞彙個數與選擇方式。我們試驗了數種選擇方式以及詞彙個數，把效果最好的用在後面的實驗中。對 SVM 分類器，在兩個測試集中，我們都取文件篇數大於 1 且 Chi-square 值大於 0 者為特徵詞。對 KNN 分類器，在 News 測試集中，所有的詞都來拿分類，在 WebDes 測試集中，則只有文件篇數大於 1 的詞彙，才用來分類。至於這些文件詞彙，是以 Tseng 描述的方法從文件中所取出來的詞彙 [19]，包括字典裡的詞、不在字典裡的最大重複詞，以及少數無法斷字的單字詞。

為了測試少量訓練文件時，本文方法的效果，我們對這兩個測試集的訓練文件做 5%、10%、20%、40%以及 100%的縮減取樣。亦即，對每一個類別，將其訓練文件分成 100/p 個等份，其中 p 代表前述的百分比，然後取其第一等份的文件作為縮減後的訓練資料。若原訓練文件本就不多，則第一等份至少必須包含 1 篇訓練文件。因此，縮減後每一類都還有訓練文件，而測試文件則保持原來的不變動。

在前述文件擴展的方法裡，對每一類別擴增多少文件數，乃一實驗參數。在此，我們試驗三種文件擴增的數量，分別求其成效，加總平均後，再與原來沒

有做文件擴展的成效作比較，以得到比較穩定的結果。

此擴展的新文件數量與原訓練文件數有關。對每一個縮減的測試集，此三個數量為 $np^{0.5}$，其中 p 代表前述的縮減百分比，而 n 在 News 測試集中分別為 10、20、30，在 WebDes 測試集中，分別為 40、60、80。這是因為 News 與 WebDes 訓練文件數不同，因此選用的擴增文件數也不同。例如，對縮減成 5%的 News 訓練文件，用來實驗的三種擴增文件數分別為 2（$=10*0.05^{0.5}$）、4、6，對 WebDes 而言，其擴增文件數分別為 8、13、17。

根據上述方式決定擴增文件數 E 之後，只有當該類的文件數不足 E 時，才擴展其文件數達 E 篇文件。

在成效評估方面，不同的度量方式有各自不同的強調對象，因而容易導致偏向（bias）的結論。本文以 MicroF 以及 MacroF 值同時呈現分類的效果，其計算方式如下：

$$MicroF = \frac{2\times\sum_{i=1}^{C} TP_i}{2\times\sum_{i=1}^{C} TP_i + \sum_{i=1}^{C} FP_i + \sum_{i=1}^{C} FN_i}$$

$$MacroF = \frac{1}{C}\sum_{i=1}^{C} \frac{2\times TP_i}{2\times TP_i + FP_i + FN_i}$$

其中 C 是類別總數，i 代表某一類別，而 TP_i（True Positive）、FP_i（False Positive）、FN_i（False Negative）類似表一的意義，分別代表：是類別 i 而且也正確分為類別 i 的篇數、不是 i 類卻分為 i 類的篇數、是 i 類卻沒有分為 i 類的篇數。此 F 值乃從精確率（P）與召回率（R）的常見公式：F = 2PR/(P+R) = 2TP/(2TP+FP+FN) 演化而來，其中 P = TP/(TP+FP)、R = TP/(TP+FN)。

由於 MicroF 是全部文件一起累加統計，不分類別，因此容易受到大類別（佔大多數文件）表現好壞的影響。相對的，MacroF 考慮每個類別的成效後再做平均，因此容易受到大量的小類別影響。將兩種平均數據都報告出來，可以瞭解大多數文件的分類效果（MicroF），以及大多數類別的分類效果（MacroF）。

五、實驗結果

實驗結果顯示於表五與表六。第一欄顯示縮減後的訓練文件量，第二欄與第五欄為運用 KNN 與 SVM 對測試文件做分類得到的數值，其他欄為運用 KNN 或 SVM 再加上文件擴展而獲得的分類成效值，其中 s 與 t 分別代表摘要擴展法與詞彙擴展法。表中粗體的數值表示文件擴展法比原始方法效果較好的情形。從表中可知：

1. 訓練文件越多，效果越好。
2. 訓練文件越少時，文件擴展法的改進成效越明顯。
3. 文件擴展法對 MacroF 的改進效果，比 MicroF 高。
4. 摘要擴展法與詞彙擴展法的效果各有優劣，但詞彙擴展法計算量較低，

因為它只增加少數幾個詞到原始的訓練文件中而已（每一篇新增的文件裡只包含一個詞）。

5. 訓練文件夠多時，再怎們運用改進策略，成效有限。必須根本的改變分類方法，才有可能大幅度地提昇成效。
6. 不同的分類方法在不同的分類問題上，有不同的表現。例如：在新聞文件的例子中，KNN 比 SVM 好，但在網頁描述的文件上則是 SVM 比 KNN 好（MicroF 雖相同，但 MacroF 方面 SVM 比 KNN 好）。
7. 相同的分類方法，在不同的分類問題上，其成效不盡相同。例如，網頁描述文件的分類數據顯示，SVM 的 MicroF 有 0.78 的成效，但在新聞文件中，只有 0.71 的成效。
8. 單獨一種分類方法，在單獨一種分類文件集上獲得的成效改進，難以保證其在另一種分類文件集上，也會有相同好的效果。

表五(a)：News 測試集的 MicroF 值

Sample	KNN	KNNs	KNNt	SVM	SVMs	SVMt
5%	0.47	**0.51**	**0.48**	0.40	**0.45**	**0.41**
10%	0.58	**0.64**	**0.60**	0.57	**0.60**	**0.59**
20%	0.70	0.67	0.70	0.63	0.62	**0.68**
40%	0.72	0.72	0.72	0.63	**0.65**	**0.71**
100%	0.79	0.78	0.77	0.71	**0.72**	**0.74**

表五(b)：News 測試集的 MacroF 值

Sample	KNN	KNNs	KNNt	SVM	SVMs	SVMt
5%	0.30	**0.35**	0.28	0.19	**0.29**	**0.27**
10%	0.32	**0.49**	**0.40**	0.31	**0.42**	**0.42**
20%	0.50	**0.54**	**0.52**	0.45	**0.49**	**0.54**
40%	0.62	0.61	**0.65**	0.49	**0.55**	**0.61**
100%	0.73	**0.76**	0.70	0.64	**0.66**	**0.69**

表六(a)：WebDes 測試集的 MicroF 值

Sample	KNN	KNNs	KNNt	SVM	SVMs	SVMt
5%	0.64	**0.69**	**0.67**	0.67	**0.68**	0.67
10%	0.69	**0.70**	**0.70**	0.71	0.70	**0.72**
20%	0.67	**0.73**	**0.74**	0.65	**0.73**	**0.75**
40%	0.75	0.75	**0.76**	0.78	0.77	**0.79**
100%	0.78	0.78	0.78	0.78	0.78	0.78

表六(b)：WebDes 測試集的 MacroF 值

Sample	KNN	KNNs	KNNt	SVM	SVMs	SVMt
5%	0.32	**0.43**	**0.39**	0.35	**0.43**	**0.37**
10%	0.38	**0.46**	**0.45**	0.42	**0.49**	**0.47**
20%	0.45	**0.49**	**0.51**	0.46	**0.52**	**0.54**
40%	0.55	**0.57**	**0.58**	0.61	**0.63**	0.61
100%	0.58	**0.63**	**0.61**	0.67	0.66	0.67

六、結論

文件分類在知識管理、資訊組織與檢索的服務上，是很重要的工作。由於需要大量而密集的知識加工，傳統上文件分類大都由人力進行。但其耗費的時間與成本相當可觀。自動分類系統近年來雖有研究，然而導入自動分類流程仍有相當的障礙，其中之一就是要準備足夠數量的訓練文件。

本文提出文件自我擴展法，在沒有利用任何額外資源的情況下，全自動的增加訓練文件，以期能提升分類的效果。在原始訓練文件數越少時，其改進的效果越明顯。而且此改進方法，乃策略層面上的技巧，與分類器無關，亦即任何一種分類器都可以運用上面的技巧來增強其分類效果。

雖然訓練文件數夠多時，此方法改進的成效不明顯，但這並非超乎預期。就好像利用壓縮法，將文件壓縮一遍，可以得到不錯的壓縮率，但再壓縮第二遍時，就得不到壓縮效果。如同上一節第 5 點所示，訓練文件夠多時，再怎們運用改進策略，其成效都有限。必須根本的改變分類（壓縮）方法，才可能大幅度地提升成效。

在少量訓練文件時就能提升成效是有價值的，這意味著我們能夠更快地提升自動分類系統的效益。如果將此方法與 EM 法結合，相當於我們在少量訓練文件時，即可獲得較佳的初始成效，這比起單獨運用 EM 法似乎效果更好。未來的研究將設計類似的實驗環境，以印證這個論點。

誌謝

本研究由國科會專題研究計畫補助，編號：NSC 92-2213-E-030-017-。

參考文獻

[1] Yiming Yang, "A Study on Thresholding Strategies for Text Categorization," Proceedings of the 23rd Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, 2001, pp. 137-145.

[2] Fabrizio Sebastiani, “Machine Learning in Automated Text Categorization,” *ACM Computing Surveys*, 34(1):1-47, 2002.

[3] 曾元顯, "文件主題自動分類成效因素探討", 「中國圖書館學會會報」, 2002 年 6 月, 第 68 期, 頁 62-83.

[4] Fabrizio Sebastiani, Alessandro Sperduti and Nicola Valdambrini, "An Improved Boosting Algorithm and its Application to Text Categorization," Proceedings of the 9th International Conference on Information and Knowledge Management CIKM 2000, Pages 78 - 85.

[5] Yiming Yang and Xin Liu, "A Re-Examination of Text Categorization Methods," Proceedings of the 22nd Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, 1999, Pages 42 - 49.

[6] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," Machine Learning, 39(2/3):103-134, 2000.

[7] 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, 39(1):1-38, 1977.

[8] Kamal Nigam and Rayid Ghani, "Analyzing the Effectiveness and Applicability of Co-training," Proceedings of the ninth international conference on information and knowledge management CIKM 2000, McLean, Virginia, United States, pp. 86 – 93.

[9] William B. Frakes and Ricardo Baeza-Yates, *Infomation Retrieval: Data Structure and Algorithms*, Prentice Hall, 1992.

[10] Amit Singhal and Fernando Pereira, “Document Expansion for Speech Retrieval,” Proceedings of the 22th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, 1999, pp.34-41.

[11] 曾元顯, *第一章數位文件關鍵特徵之自動擷取, 數位文件之資訊擷取與檢索*, 269 頁, 2000 年 9 月, ISBN 957-99750-3-2 , 全壘打文化事業有限公司出版.

[12] Yiming Yang and J. Pedersen, "A Comparative Study on Feature Selection in Text Categorization," Proceedings of the International Conference on Machine Learning (ICML'97), 1997, pp. 412-420.

[13] Hwee Tou Ng, Wei Boon Goh and Kok Leong Low, "Feature Selection, Perception Learning, and a Usability Case Study for Text Categorization," Proceedings of the 20th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, 1997, Pages 67 - 73.

[14] Thorsten Joachims, SVMlight: Support Vector Machine, version 5, http://svmlight.joachims.org/, 2002/03/07.

[15] Thorsten Joachims, "A Statistical Learning Model of Text Classification for Support Vector Machines," Proceedings of the 23rd Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, 2001, pp. 128-136.

[16] Amit Singhal, Gerard Salton, and Chris Buckley, "Length Normalization in Degraded Text Collections," Proceedings of Fifth Annual Symposium on Document Analysis and Information Retrieval, April 15-17, 1996, pp. 149-162.

[17] Yuen-Hsien Tseng, "Automatic Cataloguing and Searching for Retrospective Data by Use of OCR Text", Journal of American Society for Information Science and Technology, Vol. 52, No. 5, 2001, pp. 378-390.

[18] Yuen-Hsien Tseng and Douglas W. Oard, "Document Image Retrieval Techniques for Chinese" Proceedings of the Fourth Symposium on Document Image Understanding Technology, Columbia Maryland, April 23-25th, 2001, pp. 151-158.

[19] Yuen-Hsien Tseng, "Automatic Thesaurus Generation for Chinese Documents", Journal of the American Society for Information Science and Technology, Vol. 53, No. 13, 2002, pp. 1130-1138.