電腦輔助推薦學術會議論文評審委員之初探

陳禹勳　　劉昭麟
國立政治大學　資訊科學系
{g9418,chaolin}@cs.nccu.edu.tw

摘要

會議論文評審委員由會議議程主席指派，目的在分配適當且數量平均的論文給評審委員，以求審核論文的公正性與正確性。本研究以系統化的方法讓機器輔助人工，達到避免個人的主觀因素及節省人力的目標，並利用文件分類技術以及 Google 學術搜尋提供的資訊，建構協助議程主席指派論文的環境。我們依照一般學術會議論文的小節結構，將論文切成數個區段，藉由整合論文不同區段的特性，期望得到一個較佳的指派結果。

關鍵詞：文件分類、向量空間模型，社群網路

1. 緒論

投稿學術研討會的論文審核時程，以國內會議人工智慧與應用研討會[1](Taiwan Association for Artificial Intelligence)爲例，由 2004、2006 及 2007 這三年的研討會網頁得知，從截稿日期至通知接受日期，大約需要一個月以上的時間。主要考量在於議程主席指派待審論文給評審委員，以及評審委員研讀待審論文所花的時間。指派論文給評審委員，需要知道評審委員的研究領域與待審論文的研究領域是否相近。由於論文評審委員的領域有所不同，甚至有跨領域的研究，因此待審論文對評審委員的分配不容易決定。通常議程主席對於各教授的領域只有大略的了解，指派評審是從該教授的著作來決定，因此在面對不熟領域的教授著作時，常需要花費大量的時間與精力。加上各領域教授人數眾多，在眾多的議程委員中選取論文評審委員變得窒礙難行。

Peterson[15]的研究指出，由於閱讀論文相當費時，因此研究生及學者閱讀論文時通常不是看全篇論文，而是挑出摘要、簡介、結論及參考文獻區段來看。摘要區段透露比較多論文主題及應用技術的訊息；簡介區段則是大致說明此論文的研究動機、研究背景以及架構流程；結論區段敘述此研究的研究成果，由實驗結果印證研究方法並提出相關研究方向；參考文獻區段提供一個相關領域的查詢。因此本研究認爲對論文的指派，可以細分成各區段的相似度比對，再將其結果整合，使得建議評審委員的正確性較高。

引用共同的參考文獻強烈暗示著領域相近。各種不同領域的論文，不容易引用到同一篇論文。參考文獻區段的相似度比對上，我們採取參考文獻標題以及參考文獻作者比對作爲相似度的考量。參考文獻的部分含有許多的資訊，包含作者、論文名稱、出處及年份。我們應用 Google 學術搜尋及正規表示式取出參考文獻的標題以及作者，藉由找出待審論文及各評審委員著作的共同引用參考文獻數或作者數，作爲參考文獻區段建議評審委員的根據。對於摘要、簡介及結論區段，本研究採用向量空間模型來做相似度比對。向量空間模型[17] (Vector Space Model)是文件分類的重要技術，我們希望應用文件分類的技術，來輔助議程主席指派論文。

文件分類是根據文件內容或主題給定類別的工作，以往文件分類的研究，都是對整篇文件去取出特徵，接著藉由某些分類方法去作分類。文件分類的特徵大多是找出關鍵詞，也就是這篇文章中具有鑒別度的詞。顧皓光等[8]在 1997 年提出網路文件自動分類的方法，採用向量空間模型去對 Yahoo 內部資料庫的網頁進行分類。由於網路文件資料量相當的大，在大量資料的情形下，向量空間模型可以分出相當不錯的結果。

然而在資訊不夠充足的情形下，向量空間模型分類的效果會變的非常的差，錢炳全等[7]在 2002 年提出中文試題自動分類方法，試圖對簡短的試題作分類。在系統自動學習試題的情形下，資訊量越來越多，而分類效果也隨之改善。駱思安等[6]則是在 2006 年提出一個以機率爲主的中文網站分類系統，此系統可自動學習詞彙來改善分類效果。Dow 等[10]在 2007 年利用 DSpace[11]建立了一個論文的查詢網站，不但可從查詢的關鍵詞推薦相關的論文，並提供與查詢的關鍵詞相關的關鍵詞、領域及相關的教授，同時也提供各教授論文領域的分布，使用者可以更容易找出要查詢的資料。

專利文件的自動分類是向量空間模型在文件分類上的另一種應用。爲了避免侵犯智慧財產權，專利文件寫法上較爲格式化且嚴謹，也因此專利文件的篇幅通常相當巨大。專利文件通常分成數個段落，分別是標題、摘要、專利權利範圍、專利技術描述以及總結。Larkey 等[18]建立一個專利文獻的查詢與分類系統，藉由抽取出不同段落及計算詞彙的重要性來分類專利文件。李駿翔等[4]則是嘗試著將標題跟不同段落的分類結果整合，發現標題結合總結與標題結合專利技術描述的分類效果最好。林蘭綺等[5]則是應用標題加上總結段落部分，利用詞彙的不同權重來提高分類效果。

本研究介紹順序如下: 第二節描述系統架構、第三節說明研究方法、第四節爲實驗結果以及第五節爲結論。

2. 系統架構

此節介紹本研究的整體流程以及所需要的資料及來源出處。

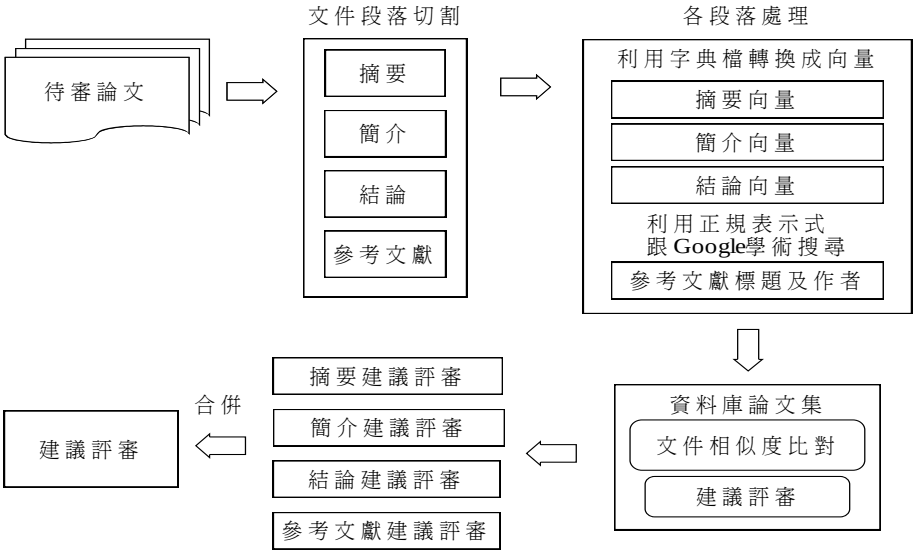

圖 1、系統架構流程圖

2.1 系統流程

本研究推薦中文論文評審委員，研究流程如圖 1 所示。將待審論文切成各個文件區段，使用向量空間模型等方法，進行待審論文各區段與資料庫論文集各區段的相似度比對。再藉由最相似論文來找出該區段的評審委員，最後整合各區段建議評審委員結果，得出待審論文的建議論文評審委員。

2.2 資料來源

本研究爲了處理的方便，論文一律從 PDF 檔轉爲文字檔，檔案轉換工具是使用 Acrobat Professional 版裡的批次處理功能來進行轉換，從 PDF 檔轉爲文字檔的成功率約略爲 74.85%。這些論文的資料來源，除了從網路上下載，還有選自於人工智慧與應用研討會 2002 年、2003 年、2004 年及 2005 年論文集的論文以及全國計算機會議 (National Computer Symposium) 2001 年、2003 年及 2005 年論文集論文共 1089 篇。測試資料則是選自 2007 年的人工智慧與應用研討會論文集，共 71 篇論文。

中文文件分詞的部分，本研究採取使用字典檔分詞的作法，以 HowNet[13]辭典作爲基礎來處理中文分詞。由於 HowNet 辭典是收納一般生活常用的中文詞彙，未必能對論文作精確的分詞，因此我們從九二八電腦股份有限公司[2]的網站，收集了兩岸三地較常見的電腦詞彙字庫，刪除重複詞，分別加入到現有詞庫中。詞庫共有總數量五萬一千多個詞，我們發現五萬一千多個詞中，只有八千多個詞彙出現在訓練資料論文過。因此，我們將沒出現過的四萬多個詞彙刪除，對剩下這八千多個詞彙依照詞的長度作分類，分成二字詞、三字詞與四字詞等，建立出一個較精簡的字典檔作爲中文分詞的依據。

3. 研究方法

本節描述處理論文區段的方法及流程。一般來說論文可分成數個區段，分別是摘要、簡介、研究方法、實驗結果及結論等等。研究方法與實驗結果區段描述研究過程，用詞以解釋清楚爲目的，站在文件分類關鍵詞爲特徵的角度來看，文件關鍵詞應具有代表性而非只是詞頻高，而這兩個區段的詞彙多爲描述研究過程，作爲關鍵詞較爲不適當。摘要、簡介及結論等區段常精簡的描述研究，很有機會出現重要的關鍵字。因此不同於一般文件分類研究以一篇文章作爲分類的基本單位，本研究把論文的各個區段切出，分別是摘要、簡介、結論和參考文獻區段，藉由整合各區段的相似度比對結果來改善分類效果。

3.1 取出論文區段的方法

一篇論文的各個區段往往都有特別的詞作爲開頭，因此本研究利用每段區段的開頭詞來做分區段的依據。我們採取一列列讀取每篇論文文件的做法，以便找出各區段的開頭詞。由於從 PDF 檔轉爲文字檔的成功率約略爲七成多，轉檔時可能會有文字的錯誤，因此會有區段取出不完整的情形。不同的論文會有不同的區段開頭詞敘述法，因此我們建立一個區段開頭詞的相關用語表，如表 1 所示。

摘要區段通常位於文章前段，以「摘要相關用語」爲摘要區段的開頭詞，而摘要區段後面通常是接關鍵字段落，因此取「關鍵字相關用語」爲摘要區段結尾。本研究取以「摘要相關用語」作爲開頭的一行到以「關鍵字相關用語」作爲開頭的一行之間這段文字作爲摘要區段。

表 1、各區段開頭詞相關用語表

摘要	摘要
關鍵字	關鍵字、關鍵詞
簡介	緒論、概論、簡介與相關研究、前言及研究背景、前言、背景動機、序論、簡介、研究背景與動機、研究動機與目的、研究動機、引言、背景與理論基礎、研究背景、介紹、導論、背景、緒言、緣由與目的
結論	結論、結語、討論、啓示、建議、未來發展方向、未來發展、未來研究方向、未來研究、未來工作、未來展望、未來後續工作、後續研究建議、後續研究、研究成果
參考文獻	參考文獻、參考資料
系統架構	系統架構、系統運作流程、設計架構、系統架構與方法、系統架構與規劃
相關研究	相關研究、相關文獻、文獻探討、理論背景與文獻探討、研究目的、背景與相關研究、相關文獻探討、背景知識與相關研究、相關研究背景說明、相關工作、相關文獻研究
研究方法	研究方法

簡介區段多位於摘要和關鍵字區段後面，簡介區段便以「簡介相關用語」作爲開頭，簡介的結尾卻是難以認定，我們觀察數篇論文的簡介開頭詞，發現一般論文中簡介開頭詞的寫作方式可大致分爲兩類：

- 用數字、英文字或羅馬符號對開頭詞標號
- 無任何標號

對於有標號的簡介開頭詞，我們建立一個對應表去對應標號跟數字間的關係，如此可得知簡介區段是標號在第幾段落，再推出簡介區段的下一區段是標號在第幾段落，便可找到簡介區段的結尾詞，進而切出簡介區段。表 2 是標號跟數字的對應表。

表 2、標號數字對應表

阿拉伯數字標號	1	2	3	4	5	6	7	8	9
中文數字標號	一	二	三	四	五	六	七	八	九
中文國字大寫標號	壹	貳	叁	肆	伍	陸	柒	捌	玖
羅馬標號	Ⅰ	Ⅱ	Ⅲ	Ⅳ	Ⅴ	Ⅵ	Ⅶ	Ⅷ	Ⅸ

若今天簡介開頭詞標號是Ⅰ，那麼下一段落的開頭詞標號就會是Ⅱ，可由此開頭詞標號切出簡介段落。其他開頭詞標號作法亦同。若簡介開頭詞無標號，本研究觀察簡介區段的下一區段通常是系統架構、相關研究或研究方法區段，因此藉由這三個區段的開頭相關用語來找出簡介區段的結尾，進而取出簡介區段。

結論區段多位於論文的後段，後面通常是接參考文獻區段，因此取以「結論相關用語」爲開頭詞的一行到以「參考文獻相關用語」爲開頭詞的一行之間的段落作爲結論區段。參考文獻區段則是取以「參考文獻相關用語」爲開頭詞的一行到文章結尾的段落。

3.2 摘要、簡介及結論區段處理

在論文切出的區段之中，由於參考文獻區段可細分出參考文獻作者及參考文獻標題，因此參考文獻區段我們額外處理，其他區段則一致使用向量空間模型做相似度比對。

要將文章轉成向量，首先要將所有的文章去做分詞的動作，我們便可以得知各個詞彙在每篇文章中出現的次數，再利用資料檢索的 tf-idf(term frequency - inverted document

frequency)[16]技術，計算出每個詞彙的 tf-idf 值。tf-idf 的計算法是資訊檢索以及文件探勘等相關領域中相當重要的公式，是由每個詞彙的 tf 值(term frequency)和 idf 值(inverse document frequency）所相乘所得出的一個常數。其中 tf 爲詞彙在單一文件中的出現頻率，可視爲在該文件內部的分布特性；idf 則是用來量測詞彙在所有文件中的重要程度，可視爲全域資料的分布特性。

$$idf_i = \log(N / n_i) \tag{1}$$

其中 N 爲論文訓練資料的總篇數，i 代表詞彙，n_i 則是包含詞彙 i 的論文總數，由公式(1)得知當一個詞彙 idf 值越小時，表示該詞彙在絕大部分的文件都有出現，因此鑑別度就會很低。一個詞彙 i 的 tf-idf 算法如公式(2)所示。

$$tfidf_i = tf_i \times idf_i \tag{2}$$

我們將每篇文章分詞之後的詞彙分別去做各自的 tf-idf，將詞彙當作向量的一個屬性，tf-idf 的值則作爲屬性裡面的值，文件就可以向量的方式去表示。

3.3 參考文獻區段處理

本研究特別針對人工智慧與應用研討會，及全國計算機會議的論文集來處理參考文獻區段。發現這些論文的參考文獻格式，大多是用數字條列式標示各筆參考文獻，因此可以將參考文獻區段細分成一筆筆的參考文獻。一筆參考文獻，我們取出它的作者及論文標題。至於論文出處，由於期刊及會議數量眾多且規模大小不一；出版年份無法直接反映論文的領域，兩者皆不容易定義分類相關性，因此本研究目前先不去對其做處理。

3.3.1 論文標題的擷取

以人工智慧與應用研討會跟全國計算機會議論文集來說，一篇論文的參考文獻，論文名稱的寫法大致分爲兩種，一種是以引號框住論文標題，用來強調論文標題；另一種則是沒有引號標示出論文標題。第一種寫法的論文標題較易取出，只需取出每篇參考文獻裡引號內的文字。由於第二種寫法無法判定論文標題，我們利用了 Google 學術搜尋網頁[14]作爲找出論文標題的工具。

Google 學術搜尋網頁是 Google 的一個學術文獻資源搜尋引擎，Google 學術搜尋網頁會依關連性排序搜尋結果，也就是考量文章的內文、作者、文章所在出版物以及內容片段出現在其他學術文獻出現的頻率。雖然我們並不知道 Google 學術搜尋網頁如何去排序這些資料，但是以 Google 的強大功能，我們認爲他的技術可信度很高的。使用 Google 學術搜尋網頁搜尋論文時，有一個特別的地方，就是當作者跟論文標題一起查詢時，查詢結果會用論文標題回傳連結。我們撰寫了一個程式，能夠送字串給 Google 學術搜尋網頁，並抓回搜尋的結果，用連結來驗證是否是參考文獻的論文標題。一筆參考文獻的寫法，是由左至右依序先寫作者，再寫論文標題，因此我們將參考文獻切成數個參考文獻片段，用圖 2 所示的演算法得出參考文獻標題。

範例：利用 Google 學術搜尋網頁找出標題

J. Setubal and J. Meidanis, Introduction to Computational Molecular Biology, PWS, Boston, MA, 1997.

如圖 2 演算法所示，這一筆參考文獻會被切成數個參考文獻片段

J. Setubal and J. Meidanis

Introduction to Computational Molecular Biology

PWS

Boston

MA

1997

我們將第一個參考文獻片段送給 Google 學術搜尋網頁，其結果如圖 3。

輸入 ：一筆參考文獻 R
輸出 ：參考文獻的論文標題 T

步驟 1： 將 R 照逗號切開，成爲由數個參考文獻片段 k_j 組成的集合 K，K={k_1,k_2,…k_m} 爲參考文獻片段集合，m 爲參考文獻片段個數
步驟 2： 令 F 爲丟字串到 Google 學術搜尋網頁的程式，L 爲 Google 學術搜尋的前十名結果，L=F(k_j)，將切開後的參考文獻片段依序丟到 Google 學術搜尋網頁，並回傳搜尋到的前十筆資料 $\forall k_j \in K, L_j = F(k_j)$
步驟 3： 比對搜尋的結果
　步驟 3.1: 如果 L_j 是 k_j 字串的一部分，表示 L_j 爲論文標題，回傳 T=L_j
　步驟 3.2: 如果沒有相同比對的字串
　　若 j 不等於 m，則設字串 k_{j+1} 爲字串 k_j 跟字串 k_{j+1} 相連的結果，回到步驟 2
　　若 j 等於 m，表示找不到標題，結束

圖 2、利用 Google 學術搜尋網頁找出標題的演算法

圖 3、Google 學術搜尋網頁－字串搜尋結果 I

圖 4、Google 學術搜尋網頁－字串搜尋結果 II

搜尋結果沒有與 *J. Setubal and J. Meidanis* 相符的字串，於是將 *J. Setubal and J. Meidanis* 跟 *Introduction to Computational Molecular Biology* 兩字串相連，送給 Google 學術搜尋網頁，其結果如圖 4。發現搜尋結果其中之一爲 *J. Setubal and J. Meidanis*,

Introduction to Computational Molecular Biology 的子字串，因此回傳此搜尋結果爲參考文獻的標題，或是著作的書名。

3.3.2 論文作者的擷取

以一筆參考文獻來說，已經成功取出論文標題之後，在論文標題之前的部分就是作者群，之後的部份就是出處及年份。因此我們把參考文獻在論文標題之前的段落取出，作爲取出作者的根據。參考文獻的作者姓名可分中文和英文兩種。中文姓名的部分，李振昌[3]提出一套有效的人名識別規則，除了以中文姓氏來辨識外，還加入了性別常用字以及前後文變異性等來斷定是否爲人名。由於參考文獻爲簡短的文字段落，從文字能獲得的資訊很少，因此本研究只使用百家姓姓氏比對的方式去找出人名。演算法如圖 5。

輸入 : 一筆參考文獻 R
輸出 : 作者名字集合 $A=\{A_1,A_2,\ldots A_\gamma\}$，$\gamma$ 爲一筆參考文獻的中文作者數

步驟 1 : 由前一節演算法找出的論文標題, 將參考文獻在論文標題 T 之前的文字段落 S 取出，S=R-T-(T 之後的段落)
步驟 2 : 將此文字段落 S 依標點符號切開，得出 $B=\{B_1,B_2,\ldots B_o\}$，B 爲可能是作者的人名集合，o 爲此參考文獻切成段落的段落總數
步驟 3 : $\forall B_j \in B$, 檢查 B_j 的第一個字元是否是百家姓
  步驟 3.1: 若是, 表示可能爲作者名稱，先檢查是否有「和」、「與」等中文連接詞，如果有則將 B_j 依該連接詞切開，並將 B_j 切開後的兩段落加入 A 中，若無連接詞，則直接加入 B_j 到 A 中
  步驟 3.2: 若否, 則可能抓錯，結束

圖 5、參考文獻找出中文作者的演算法

英文姓名部分，由於參考文獻的作者群寫法有一定的格式，單獨大寫字母往往表示英文名字縮寫，本研究觀察國內論文集常出現的參考文獻作者寫法，用正規表示法(Regular Expression)定義作者姓名的樣式(patterns)。表 3 是我們觀察幾個常見參考文獻作者的寫法。

表 3、常見英文姓名的寫法範例

T. Nishita 或 C. C. Liu	由至少一組一個英文字母加一個英文句號與一個空白的字串，加上英文字母字串組成。
Beckmann, N.或 Robinson, J. T	由英文字母字串，一個逗號加空白，一個英文字母，一個英文句號加空白，加上零個以上的英文字母組成。
John H. Holland 或 David Andre	由英文字母字串，空白，至少一組一個英文字母加一個英文句號的字串，加上英文字母字串組成。
C.-T. King	由一個大寫英文字母加英文句號，英文破折號加上一個大寫字母及英文句號，空白加上英文大寫加上至少一個英文字母組成

letter -> A|B|C…|Z|a|b|c…..|z
letters -> letter
dot -> .
comma -> ,
space ->
Name -> $(\text{letter dot})^+$ letters | letters comma letter dot letter^* |letters space $(\text{letter dot})^*$ letters

圖 6、英文姓名正規表示式

因此我們定義幾個非終止符號(non-terminal)以及單詞(token)，並建立幾個英文姓名的樣式。圖 6 是英文姓名樣式的正規表示式。這些樣式之中，樣式二有逗號存在，而逗號常用在分開作者姓名，因此優先權上樣式二要最後處理，樣式一跟樣式三並不衝突，因此先後做的順序並無差別。得知姓名的樣式之後，就可以取出一筆參考文獻作者或作者群。圖 7 是取出參考文獻作者的演算法。

輸入: 一筆參考文獻 R
輸出: 作者名字集合 $A=\{A_1,A_2,\ldots A_\Theta\}$，$\Theta$ 爲一筆參考文獻的英文作者數

步驟 1 : 由前一節演算法找出的論文名稱，將參考文獻在論文名稱 T 之前的文字段落 S 取出，S=R-T-(T 之後的段落)
步驟 2 : 令 W 爲符合表 3 樣式的字串，作者名字集合 A={}，先檢查 S 是否有對等連接詞 and，以查證 S 是否有一個以上的作者名字
步驟 2.1: 如果有，檢查對等連接詞前後是否有空白，避免誤認人名中間的字母爲連接詞
若有空白，表示確實爲對等連接詞，將 S 依照連接詞切開並將切開後的後面段落 C 加入 A 中，並將段落 C 從 S 中移除，S=S-C
若無空白，則表示誤認，跳至步驟 3
步驟 2.2: 如果沒有對等連接詞，跳至步驟 3
步驟 3 : 若 S 爲空，表示已經處理完畢，結束
若 S 有符合樣式的字串 W，則加入 W 到 A 中，並將字串 W 從 S 中移除，S=S-W
若 S 沒有符合樣式的字串 W，結束

圖 7、取出參考文獻英文作者的演算法

3.3.3 參考文獻的直接與間接引用

當兩篇論文引用到同一篇論文時，我們稱這兩篇論文直接引用同一篇論文。因此我們將論文的參考文獻區段細分成一筆筆的參考文獻，再找出兩篇論文的參考文獻中取出標題，最後對兩篇論文所取出的參考文獻標題進行字串比對，如果比對有相同的參考文獻標題，我們就認定兩篇論文有參考文獻的直接引用。

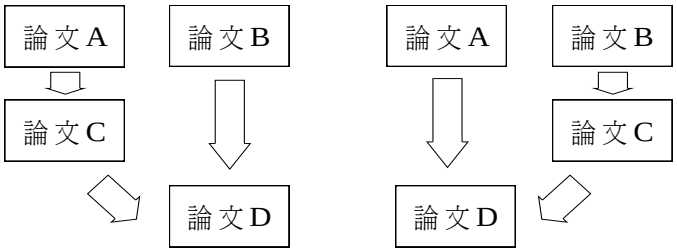

圖 8、論文間接引用的兩種情形

有時候論文會有間接引用的情形，也就是兩篇論文並不是直接引用同一篇，而是在引用到參考文獻的原文中引用到共同的論文，使得兩篇表面上沒有引用到共同參考文獻的論文卻有著極高的關連性。間接引用又可分成兩種情形，如圖 8 所示，圖 8 左邊論文 A 跟論文 B 並沒有直接引用到同一篇論文，但是論文 A 引用的論文 C 卻跟論文 B 共同引用論文 D；同樣的圖 8 右邊論文 B 跟論文 A 並沒有直接引用到同一篇論文，但論文 B 引用的論文 C 卻跟論文 A 共同引用了論文 D，我們稱論文 A 跟論文 B 有著間接引用。因此在處理參考文獻的引用上，就必須處理間接引用，其演算法如圖 9。

輸入: 兩筆參考文獻 P 和 Q
輸出: 是否間接引用

步驟 1 : 到資料庫找出 P 和 Q 的論文本文 U_p 和 U_q，分別取出 U_p 和 U_q 的參考文獻段落，集合 $R_p=\{p_1,p_2\cdots,p_g\}$，$R_q=\{q_1,q_2\cdots,q_h\}$

步驟 2 : $\forall p_j \in R_p$，取出論文的名稱，並檢查是否跟 Q 有直接引用

$\forall q_j \in R_q$，取出論文的名稱，並檢查是否跟 P 有直接引用

步驟 3 : 若有，回傳是。反之回傳否。

圖 9、檢查兩篇參考文獻是否間接引用的演算法

3.3.4 摘要、簡介與結論區段相似度計算

摘要、簡介與結論區段部分，由於採用向量空間模型的方法處理，本研究用餘弦函數[9]來計算評審教授發表的論文與待審論文相似度。在幾何學中，兩個向量若是越相近，所夾的夾角 θ 也會越小；而利用餘弦函數的特性，θ 夾角越小所得出的餘弦函數值也會越大。因此可以得到一個結論：兩個純文字向量的餘弦函數值越大，代表兩篇文章的文字向量所形成的夾角越小，則此兩篇文章內容越相似；反之，就代表兩篇文章越不相關。

3.3.5 參考文獻相似度計算

由參考文獻標題的直接引用、間接引用以及參考文獻共同作者，可計算出兩篇論文的近似程度，進而由最相似論文來建議評審委員。在論文間接引用處理上，由於並不是直接引用到同一篇論文，因此其關連性弱於直接引用。如果直接引用一筆參考文獻，我們定義配分是 1 分，若是像圖 8 那樣間接引用一筆參考文獻則定義 0.7 分。本研究中參考文獻共同引用作者也是給予 0.7 分的權重，一篇參考文獻可能引用同一作者的數篇論文，在相關性上也較直接引用爲弱。公式(3)是兩篇論文的參考文獻區段相似度計算公式。

$$Simularity_{reference} = Title_{direct} \times 1 + Title_{indirect} \times 0.7 + Authors \times 0.7 \quad (3)$$

其中 $Title_{direct}$ 表示參考文獻標題直接引用的數量，$Title_{indirect}$ 表示參考文獻標題間接引用的數量，Authors 表示參考文獻共同引用作者數量。

4. 實驗結果

本研究從 2007 年人工智慧與應用研討會的評審委員名單中，挑選九十八位作爲評審委員的候選人，並收集這些評審委員的相關論文，平均一位收集十篇到十五篇左右，總共約收集一千零八十九篇來做爲訓練資料。訓練資料來源除了從網路上下載，還包含了 2002 年、2003 年、2004 年及 2005 年人工智慧與應用研討會論文集的論文以及 2001 年、2003 年及 2005 年全國計算機會議 (National Computer Symposium)論文集的論文。測試資料則是選自 2007 年人工智慧與應用研討會共 74 篇論文。

4.1 取出區段的精確度

計算取出區段的精確度，我們以人工方式取出原始論文的區段文字作爲標準答案，分別對系統取出的區段及標準答案區段進行中文分詞處理。中文分詞處理我們是透過對詞庫

裡詞彙的比對搜尋，依照「長詞優先」法則來對中文語句作分割，分詞處理上能得出較正確的詞彙。對論文做完分詞工作後，可以得出論文出現了哪些詞彙，以及詞彙在論文中出現的頻率，因此可計算系統取出區段的總詞彙數，標準答案區段詞彙數以及系統取出區段詞彙與標準答案區段詞彙的交集詞彙數，我們以這些資訊計算出摘要、簡介與結論區段的 precision 與 recall。公式(4)和(5)中 $\text{precision}_{\text{part}}$ 表示區段的 precision 值，$\text{recall}_{\text{part}}$ 表示區段的 recall 值，y 代表字典檔的詞彙總數，i 代表詞彙，V_i 表示詞彙 i 在取出區段的出現次數，W_i 表示詞彙 i 在標準答案區段出現的次數，TP_i 表示詞彙 i 在取出區段跟標準答案區段共同出現次數。

$$precision_{part} = \frac{\sum_{i=0}^{y} TP_i}{\sum_{i=0}^{m} V_i} \quad (4) \qquad recall_{part} = \frac{\sum_{i=0}^{y} TP_i}{\sum_{i=0}^{m} W_i} \quad (5)$$

本研究從測試資料中選出 25 篇論文來評估取區段的精確度，將 25 篇論文取各區段的 precision 與 recall 平均值，可得出取出各區段的平均 precision 與 recall。參考文獻區段由於要取出作者跟論文名稱，因此不同於其他段落要進行分詞，取出參考文獻精確率的算法，我們細分成參考文獻作者取出精確率，以及參考文獻標題取出精確率。公式(6)和(7)分別是取出參考文獻作者的 precision 與 recall。

$$precision_{Author} = \frac{NAC}{NAR} \quad (6) \qquad recall_{Author} = \frac{NAC}{NAA} \quad (7)$$

其中 *NAR* 表示系統取出的參考文獻作者數，*NAC* 表示系統取出且合乎標準答案的作者數，*NAA* 表示標準答案的作者數。參考文獻標題取出精確度算法與參考文獻作者取出精確率算法相同，只需將 *NAR* 代換成系統取出的參考文獻標題數，*NAC* 表示系統取出且合乎標準答案的參考文獻標題數，*NAA* 表示標準答案的參考文獻標題數。我們以人工去對 25 篇論文取出一筆筆參考文獻的標題，以及參考文獻作者作爲標準答案，計算取出參考文獻標題以及作者的 precision 跟 recall。表 4 是取出摘要區段、簡介區段、結論區段、參考文獻標題及參考文獻作者的 precision 與 recall 對應表。

表 4、取出摘要、簡介及結論區段、參考文獻標題與作者的 precision 和 recall

	摘要	簡介	結論	參考文獻作者	參考文獻標題
precision	94.32%	99.67%	91.47%	44.23%	60.74%
recall	94.50%	77.83%	72.93%	52.11%	61.68%

4.2 推薦單一評審委員

由之前所介紹的方法，一篇論文可轉換成數個由不同區段而成的向量以及數條參考文獻。經由相似度的計算，可得出一篇論文各區段的各自最相似論文，再由找出論文作者得出建議評審委員的名字。因此一篇待審論文會有各個區段的建議評審委員，本研究採取各區段權重相同的做法，也就是以區段評審委員名字出現最多次的作爲該篇論文的建議評審委員。爲了顯示的方便，本節實驗列表僅列出測試資料中選出六篇論文的實驗結果，建議評審委員則是從 2007 人工智慧與應用研討會的評審委員名單中選出。

藉由相似度的計算，我們可以找出論文區段的建議評審委員。以摘要區段爲例，取一篇待審論文的摘要區段，對所有訓練資料論文的摘要區段做餘弦函數值，同時記錄訓練資料論文的作者名稱。再找出最大的餘弦函數值的論文作者，即是建議的評審委員。同樣的，簡介及結論區段也可找出個別的建議評審委員。參考文獻區段則是利用公式(3)，計算待審論文跟所有訓練資料論文參考文獻區段的相似度，進而找出建議評審委員。圖 10 是由參考文獻找出建議評審委員的演算法。

輸入： 一篇論文 E 與資料庫 F
輸出： 建議的評審委員

Step1： 取出論文 E 的參考文獻集合 R={r_1,r_2…r_Δ}，Δ 爲論文 E 的參考文獻總數量
Step2： $\forall r_j \in R$，取出論文的名稱，找出資料庫 F 的論文是否有共同引用此論文，並找出參考文獻跟資料庫 F 內論文的共同作者
Step3： 計算 E 與 F 內論文其參考文獻標題直接引用、間接引用以及參考文獻共同作者的數量，並依照公式(3)算出總值。
Step4： 回傳最大總值論文作者作爲建議評審委員。若總值爲零，則回傳“找不到建議的委員”

圖 10、由參考文獻建議評審委員的演算法

表 5、參考文獻標題與作者共同引用數量範例表

檔名	建議評審委員	標題直接引用篇數	標題間接引用篇數	共同引用作者數
基於 SVM 與 LDA 演算法之人臉辨識	曾守正	1	0	2
模組化線性鑑別式分析應用於人臉辨識	劉吉軒	0	0	1
基於紋理特性之移動物體偵測法則	劉吉軒	0	0	1
應用於 BDI Agent 之案例式推理系統開發工具	李宗南	0	0	1
使用小腦模型類神經網路控制冷氣空調機馬達	找不到建議的委員	0	0	0
可拓基因演算法	吳志宏	1	0	0

表 5 是參考文獻建議委員的列表，欄位從左到右依序是檔名、建議評審委員、標題直接引用篇數、標題間接引用篇數以及共同引用作者數。由表 5 可看出參考文獻的共同引用機會其實很低，以「使用小腦模型類神經網路控制冷氣空調機馬達」這篇論文來說，沒有任何直接引用、間接引用及共同作者，這樣的情況下我們會在建議評審委員欄位填上「找不到建議的委員」。

表 6、各區段建議評審委員

	摘要	簡介	結論	參考文獻	評審委員
基於 SVM 與 LDA 演算法之人臉辨識	黃有評	張嘉惠	張嘉惠	曾守正	張嘉惠
模組化線性鑑別式分析應用於人臉辨識	蔡正發	方國定	張智星	劉吉軒	找不到建議的委員
基於紋理特性之移動物體偵測法則	范欽雄	范欽雄	范欽雄	劉吉軒	范欽雄
應用於 BDI Agent 之案例式推理系統開發工具	林豐澤	許永真	劉吉軒	李宗南	找不到建議的委員
使用小腦模型類神經網路控制冷氣空調機馬達	古鴻炎	王學亮	楊正宏	找不到建議的委員	找不到建議的委員
可拓基因演算法	許永真	張嘉惠	陳慶瀚	吳志宏	找不到建議的委員

藉由整合一篇論文各區段的建議評審委員，我們可以找出該篇論文的建議評審委員。由表 6 所示，第一行欄位爲論文檔案名稱，第二行欄位爲這些論文摘要區段的建議評審委員，第三行欄位爲這些論文簡介區段的建議評審委員。同理，第四行、第五行欄位分別代表這些論文結論與參考文獻區段的建議評審委員。最後一行評審委員欄位則是各區段投票後的結果，以最多區段建議的評審委員作爲該篇論文的建議評審委員。

以「基於紋理特性之移動物體偵測法則」這篇論文爲例，該篇的摘要、簡介及結論區段建議評審委員皆是「范欽雄」，而參考文獻區段則是「劉吉軒」，因此投票數最高的評審委員爲「范欽雄」，投票數爲三票。由於有些論文有跨領域的情形，各區段建議評審委員可能會是不同的人而導致沒有共同的建議評審，這時就無法建議評審委員。以「模組化線性鑑別式分析應用於人臉辨識」這篇論文爲例，該篇各區段的建議評審委員皆是不同的人，這樣的情況我們就無法建議評審委員，因此在評審委員欄位填上「找不到建議的委員」。像這樣區段無法建議評審委員的情形我們視同無投票能力，通常參考文獻區段欄位會有這樣的情形。對一般論文來說，由於領域眾多因此論文數量相當可觀，只有領域很相近的論文才有可能引用到同篇參考文獻。兩篇論文引用到同一篇論文的機率可說是微乎其微，因此參考文獻推薦評審教授的論文數量很少是可以預見的。

4.3 推薦多重評審委員

現實的會議論文指派情形，一篇論文不會只指派給一位評審委員，而是會分給三位評審委員左右，議程主席再整合評審委員們的意見決定該論文是否被會議接受。因此我們考慮取出前十名近似的論文，再以這些論文的作者選出三位來作爲建議的評審委員。在建議評審委員的選擇上，論文最近似無疑是最佳選擇，但是如果有多篇近似的論文則也暗示著作者對該領域有興趣，可以作爲建議評審委員的考量。

在各區段的建議評審委員上，不再是以取最相似的論文作者而是改以取前十名。本研究將前十名的十篇論文作者由近似度高到低排序，並分別給予分數第一名 10 分到第十名給予 1 分，接著觀察這些前十名的作者是否有重複出現，若有，則表示該作者有數篇近似的論文，於是將該作者出現在這前十名的這幾篇論文分數加總，作爲待審論文建議給該作者的分數。若無，則以該作者現有分數作爲分數。再由分數最高的作者依序排序選出最高的前三名作爲評審委員。本節列表只僅列測試資料中選出六篇論文的實驗結果，建議評審委員則是從九十八位 2007 人工智慧與應用研討會的評審委員中選出。

表 7、摘要區段前十名近似論文的作者

基於 SVM 與 LDA 演算法之人臉辨識	黃有評	林豐澤	林豐澤	呂永和	陳士杰	方國定	廖純中	陳慶瀚	曾憲雄	呂永和
模組化線性鑑別式分析應用於人臉辨識	蔡正發	李昇暾	陳慶瀚	陳士杰	方國定	廖純中	孫光天	林豐澤	呂永和	曾憲雄
基於紋理特性之移動物體偵測法則	范欽雄	林正堅	曾新穆	呂永和	廖純中	方國定	陳士杰	陳慶瀚	曾憲雄	林豐澤
應用於 BDI Agent 之案例式推理系統開發工具	林豐澤	方國定	陳士杰	廖純中	呂永和	楊東麟	黃有評	曾守正	吳志宏	曾憲雄
使用小腦模型類神經網路控制冷氣空調機馬達	古鴻炎	楊正宏	陳慶瀚	林豐澤	方國定	陳士杰	廖純中	呂永和	曾憲雄	孫光天
可拓基因演算法	許永真	呂永和	陳士杰	廖純中	方國定	呂永和	林豐澤	曾新穆	曾憲雄	林豐澤

以摘要區段爲例，表 7 是跟待審論文最近似的前十名論文作者表，在這裡取六篇論文來觀察，最左邊的欄位是論文檔案名稱，接著欄位由左到右是最近似論文的作者到較不近似論文的作者，觀察「基於 SVM 與 LDA 演算法之人臉辨識」這篇論文，前三名近似部分有兩名是林豐澤，表示這位作者有近似且數量不少的著作，在建議委員的選擇上可能更勝於最近似的黃有評；「可拓基因演算法」這篇論文也是類似的情況，「可拓基因演算法」這篇論文呂永和佔了兩篇名額，一篇較爲近似另一篇則大約處於中等近似的程度，但由於篇數加成的關係在選擇上優先權要高過第一名的許永真。表 8 是這六篇論文的摘要區段前三名建議評審委員的列表。

表 8、摘要區段前三名建議評審委員

基於 SVM 與 LDA 演算法之人臉辨識	林豐澤(17 分)	黃有評(10 分)	呂永和(8 分)
模組化線性鑑別式分析應用於人臉辨識	蔡正發(10 分)	李昇暾(9 分)	陳慶瀚(8 分)
基於紋理特性之移動物體偵測法則	范欽雄(10 分)	林正堅(9 分)	曾新穆(8 分)
應用於 BDI Agent 之案例式推理系統開發工具	林豐澤(10 分)	方國定(9 分)	陳士杰(8 分)
使用小腦模型類神經網路控制冷氣空調機馬達	古鴻炎(10 分)	楊正宏(9 分)	陳慶瀚(8 分)
可拓基因演算法	呂永和(14 分)	許永真(10 分)	陳士杰(8 分)

表 9、利用多重委員建議評審結果表

	摘要	簡介	結論	參考文獻	評審委員
基於 SVM 與 LDA 演算法之人臉辨識	林豐澤(17 分) 黃有評(10 分) 呂永和(8 分)	張嘉惠(10 分) 林正堅(9 分) 陳慶瀚(8 分)	張嘉惠(10 分) 陳慶瀚(9 分) 陳士杰(8 分)	曾守正(10 分)	張嘉惠 陳慶瀚 林豐澤
模組化線性鑑別式分析應用於人臉辨識	蔡正發(10 分) 李昇暾(9 分) 陳慶瀚(8 分)	陳士杰(16 分) 方國定(10 分) 廖純中(8 分)	張智星(10 分) 林正堅(9 分) 陳慶瀚(8 分)	劉吉軒(10 分)	陳士杰 陳慶瀚 蔡正發
基於紋理特性之移動物體偵測法則	范欽雄(10 分) 林正堅(9 分) 曾新穆(8 分)	范欽雄(10 分) 林正堅(9 分) 陳慶瀚(8 分)	范欽雄(10 分) 鄭炳強(9 分) 林正堅(8 分)	劉吉軒(10 分)	范欽雄 林正堅 劉吉軒
應用於 BDI Agent 之案例式推理系統開發工具	林豐澤(10 分) 方國定(9 分) 陳士杰(8 分)	林豐澤(17 分) 許永真(10 分) 廖純中(7 分)	劉吉軒(10 分) 曾憲雄(9 分) 廖純中(8 分)	李宗南(10 分)	林豐澤 廖純中 許永真
使用小腦模型類神經網路控制冷氣空調機馬達	古鴻炎(10 分) 楊正宏(9 分) 陳慶瀚(8 分)	王學亮(10 分) 林正堅(9 分) 曾守正(8 分)	楊正宏(10 分) 林豐澤(9 分) 陳士杰(8 分)	找不到建議的委員(0 分)	楊正宏 古鴻炎 王學亮
可拓基因演算法	呂永和(14 分) 許永真(10 分) 陳士杰(8 分)	呂永和(11 分) 張嘉惠(10 分) 林正堅(9 分)	呂永和(11 分) 陳慶瀚(10 分) 陳士杰(9 分)	吳志宏(10 分)	呂永和 吳志宏 許永真

簡介區段與結論區段作法跟摘要區段相同。如果有建議評審同分的情況，我們保留兩位教授都可作爲評審委員的候選人，將來指派教授當其中一位教授被指派到過多的論文時，就由另一位教授補上。參考文獻的建議評審委員方面，由於一般論文不容易有共同引用的現象，取前三名建議評審似乎沒有實質的幫助，因此在選擇上仍然是以取一名委員來做處理。參考文獻的建議評審委員如表 5。由於參考文獻的重要性，我們給予參考文獻段落建議委員 10 分。表 9 是利用多重委員建議評審結果的表。我們將各段落前三名的作者及其分數加總，選出總分最高的前三名作者作爲建議評審委員。

得出系統建議評審委員之後，我們評估建議評審的準確率，評估的標準包含 precision，recall 及 F-measure。本研究用人工建立建議評審委員的答案表，由「本研究的指導老師」協助建立，以計算出系統推薦論文評審委員的準確率。在本研究前面的段落，我們展示了整合各區段推薦委員的方法，並推薦出三位評審委員，同樣的可將推薦委員的人數增加，不一定只推薦三位。表 10 是多重委員推薦三位評審及五位評審的準確率對應表。系統推薦五名評審委員的 precision 稍高過兩成五，也就是說系統推薦五名的評審委員，至少有一名是人工推薦在答案表上的。另外由表 10 可看出系統的 recall 相當的低，可能是因爲是我們在建立建議評審委員的答案表時，並未限定一篇論文的評審委員人數，使得每篇論文可以指派給多位評審委員，因而造成 recall 值不高。此外，評審委員的著作論文篇數不同，也可能造成論文分派到著作多而非最適合的評審。

表 10、多重委員建議三位評審及五位評審的對應答案準確率

	precision	recall	F-measure
三名評審委員	28.16%	5.72%	9.13%
五名評審委員	26.20%	8.56%	12.11%

5. 結論

藉由計算待審論文區段與資料庫論文區段之間的相似度，我們得出各區段的建議評審委員。本研究整合不同區段的建議評審，來找出待審論文的建議評審委員。目前本研究是採取不同區段相等比重的方法，然而一篇論文中各個區段重要性不一定相同，因此在整合區段評審委員時，各區段應有著各自的權重。未來本研究可能會應用機器學習的技術，調整各區段最適當的權重，使得指派效果得以提昇。同時，如果待審論文是新的領域或技術，也會造成找不到適當的建議評審。也許可以採取設立門檻值的作法，找出跟每位評審相似度都不高的論文，並回報讓議程主席得知這些論文不容易被分配到建議評審。

由於論文領域的不同，論文評審的建議變得困難，也因此論文關鍵詞擷取相對來的重要。本研究目前採用以 HowNet 爲主的字典檔，未必能包含論文的重要關鍵詞。將來可能會找尋含有更多資訊技術關鍵字的字典檔，使得論文的建議評審結果更加準確。同時不同詞彙也存在不同程度的關連性，單純的使用 tf-idf 無法完全反映評審委員對該論文主題專長程度。另外，本研究並未進行適當篇數論文指派給評審，以及論文指派給適當數量評審的處理[12]，這些都是需要加強改善的項目。

致謝

本研究承蒙國科會研究計畫 NSC-95-2221-E-004-013-MY2 的部分補助謹此致謝。我們感謝匿名評審對於本文初稿的各項指正與指導。

參考文獻

[1] 人工智慧與應用研討會，http://www.taai.org.tw/ [Access: Jun. 28, 2008]

[2] 九二八電腦股份有限公司，http://www.928n.com.tw/928index.asp [Access: Jun. 28, 2008]

[3] 李振昌，李御璽，陳信希，“中文文本人名辨識問題之研究”，*第七屆自然語言與*

語音處理研討會，1994

[4] 李駿翔，*應用資料探勘分類技術於專利分析之研究*，碩士論文，中原大學，台灣，桃園，2003。

[5] 林蘭綺，*專利文件之自動分類研究*，碩士論文，國立交通大學，台灣，新竹，2006。

[6] 駱思安、李中彥及徐俊傑，“以 MMB 演算法改良中文網站自動分類系統的效能”，*全國計算機會議論文集*，論文光碟，2005。

[7] 錢炳全及廖雙德，“中文試題自動分類方法”，*第七屆人工智慧與應用研討會論文集*，論文光碟，2002。

[8] 顧皓光及莊裕澤，“網路文件自動分類”，*全國計算機會議論文集*，論文光碟，1997。

[9] Amit Bagga and Breck Baldwin, “Entity-Based Cross-Document Coreferencing Using the Vector Space Model”, *Proceedings of the 36th Annual Meeting on Association for Computational Linguistics*, 1998, Volume 1, Pages 79–85.

[10] Chyi-Ren Dow, Khong-Ho Chen, Shu-Yi Lin, Yan-Ling Liu, Chih-Chieh Peng, and Sheng-Jie Guan, “Design and Implementation of a DSPACE-based Recommender System for Digital Literature Retrieval”, *Proceedings of the 12th Conference on Artificial Intelligence and Applications*, CD-ROM, 2007.

[11] DSpace, http://www.dspace.org/ [Access: Jun. 28, 2008]

[12] David Hartvigsen, Jerry C. Wei and Richard Czuchlewski, “The Conference Paper-Reviewer Assignment Problem”, *Decision Sciences*, 1999, Volume 30, Issue 3, Page 865-876.

[13] HowNet, http://www.keenage.com. [Access: Jun. 28, 2008]

[14] Google Scholar, http://scholar.google.com [Access: Jun. 28, 2008]

[15] Ann P. Bishop, “Document Structure and Digital Libraries: How Researchers Mobilize Information in Journal Articles”, *Information Processing and Management*, 1999, Pages 255-279.

[16] Gerard Salton and Michael J. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill, 1986.

[17] Gerard Salton, A. Wong and C. S. Yang, “A Vector Space Model for Automatic Indexing”, *Communications of the ACM*, 1975, Volume 18, Issue 11, Pages 613–620.

[18] Leah S. Larkey, “A Patent Search and Classification System”, *Proceedings of the 4th ACM Conference on Digital Libraries*, 1999, Pages 179–187.