自動擷取研討會資訊之研究

魏世杰
淡江大學 資訊管理所
sekewei@gmail.com

陳康毅
淡江大學 資訊管理所
kangpond@gmail.com

摘要

目前許多學校與機構都會舉辦研討會，邀請在該領域有興趣的人一同來共襄盛舉。研討會資訊的公佈又以網路公佈為最主要，許多人常是藉由網路的搜尋來找尋自己想參與的研討會。

在網頁資料日漸豐富的現在，網頁資訊的擷取越來越成為一個重要的議題。當使用者想從網路中找尋自己想投稿的研討會時，常常需要一個一個網頁進行檢視，這相當的花費時間，因此本研究提出一個系統，經由系統自動整合各個不同的研討會網頁，將研討會網頁的各個投稿相關資訊擷取下來，讓使用者能更快且有更豐富的研討會資訊，幫助其投稿。

關鍵詞：自動擷取，研討會資訊

1. 緒論

1.1 研究背景與動機

在現今網際網路蓬勃發展的時代，網頁資料與日俱增，如何能從眾多的網頁中取得使用者想要的資訊逐漸變成一個重要的課題。雖然目前有不少如 Google、Bing、Yahoo 等搜尋引擎來幫助使用者查詢相關關鍵字的資料，但仍需使用者在為數眾多的結果中一個一個審視，以取得所需資訊。

目前在研討會的投稿過程中，尚未有一個自動整合的機制。若一個使用者想要投稿研討會，他必須自己瀏覽各個學校的網頁，查看是否有公佈徵稿的資訊，或是從各大搜尋引擎搜尋相關的關鍵字，再從結果清單中一個一個檢視，這都是相當費工的步驟。其實使用者只是想知道一些研討會的特定資訊，例如：名稱、時間、地點、主題等等，來幫助自己決定是否可不可以投稿此研討會。如果可以有系統把這些資訊整合起來讓使用者參考，就會變得方便許多。

目前相關文獻當中，雖然已經有研究針對一些特定網站做研討會資訊的擷取，但是仍然沒有一個研究是全針對搜尋引擎的研討會資料所做的整合。現在的人依賴搜尋引擎來查詢，因為搜尋引擎有豐富的網頁資料，但是卻需要使用者一個一個檢視網頁結果的缺點。所以本研究希望能發展一套系統，針對搜尋引擎查詢後的研討會資訊，做有效的分析，整合出使用者在投稿研討會時，所需要的資訊。

1.2 研究問題與目的

目前在從網路上擷取研討會資料的相關研究中，都會遇到每個網頁格式不一的問題，因為格式不一，也增加了資訊的擷取的困難。雖然有些網頁擷取的研究會針對原始碼的 HTML 標籤來幫助資訊的擷取，不過在研討會資訊上常常是直接整篇研討會徵稿內容貼在一個 HTML 的標籤裡面，或是有些網頁會把數字用一種標籤，中文字用另一種，目的是為了排版使用，這種情形以標籤當作識別資料的條件就難以確定是不是同一個欄位的資料。

因此本研究想以文字與文字之間的關係來進行分析，但考量到原本網頁內容中的文字順序也是將不同意義的文字分割開來的依據，所以使用 htmlUnit 套件，將網頁上的文字依原始碼的順序取下來當資料來源以作為研討會資訊擷取的分析資料來源。

本研究希望能透過簡單又通用的方式，提出一套研討會徵稿資訊擷取程式，讓使用者能夠省去經由搜尋引擎查詢後一筆一筆檢視的這個過程，並且藉著系統自動把每個網站的研討會資訊擷取下來，幫助使用者進行研討會的投稿。

2. 文獻探討

2.1 HTMLUNIT

htmlUnit[8]是一個以 Java 為基礎撰寫的套件，它模擬部分瀏覽器能做到的行為，例如表單的提交、JavaScript 的執行、基本的 Http 身份驗證、Cookie 的設置和自動頁面重新導向等等行為，並且允許 Java 測試程式來檢查網頁文本，也支援 XML Dom Tree 的結構，讓使用者可以使用程式設定網頁上的參數的一個測試網頁的輔助工具。

本研究主要使用 htmlUnit 模擬瀏覽器讀取網頁 URL 的連線功能，並取得網頁的表單，讓系統能進行設置網頁表單的參數以達到本研究的需求，得以擷取網頁原始順序的原始碼，讓系統執行後續的分析。

2.2 Regular Expressions

正規表示式[11]（Regular Expression、regex 或 regexp，縮寫為 RE），也譯為正規表示法、常規表示法，在電腦科學中，是指一個用來描述或者匹配一系列符合某個句法規則的字串的單個字串。在很多文字編輯器或其他工具裡，正則運算式通常被用來檢索和/或替換那些符合某個模式的文字內容。許多程式語言都支援利用正則運算式進行字串操作。例如，在 Perl 中就內建了一個功能強大的正則運算式引擎。正則運算式這個概念最初是由 Unix 中的工具軟體（例如 sed 和 grep）。

正規表示式用在很多文字內容的匹配上，例如當我們要找手機電話號碼的字串時，可以使用 09[0-9]{2}-[0-9]{6}這個正規表示式，表示我們要找的文字內容為 09 開頭需要加上任意兩個 0 到 9 的數字，再加上一個短線，接著為任意六個 0 到 9 的數字。

2.3 研討會網站資訊擷取之研究

學者胡姝涵等人[3]在研討會的網站研究上主要針對國際性會議（International Conference）公告網站，擷取來自不同佈告者公告的國際會議資訊，包括會議名稱、會議地點、會議日期和論文接受日期。國際會議內容以純文字為主，針對英文的研討會網頁擷取資訊，但是因為該網站有固定欄位給佈告者發佈，某些欄位的資訊範圍也相對較固定明確，胡姝涵等人以機器學習的方式，整合分析非結構化的純文字會議內容，並加入特徵值選取。在不具結構性的會議名稱中，由於沒有明顯會議名稱的起始和結束邊界，所以胡姝涵等人採用 Sliding Windows 的切割方式，在 Compress 和 Sparse 格式下，Naïve 、SVM 和 FOIL[9]的 F-Measure 值皆沒有很好的擷取結果，因為 Sliding Windows 的切割方式會產生太多的反例。

史嘉淋[1]提出以 rule-based 為主的擷取方法，並使用 VIPS[6]演算法切割網頁進行主題資訊的擷取，但是使用 VIPS 切割網頁之後，並不能保證一定可以找到對的主題資訊區塊。又史嘉淋等人是找出所有的日期，而非以特定資訊的日期來作擷取。而且在評估時，是以擷取資料是否符合正確欄位的區塊面積，而不是細分會議時間與截稿時間來評估，因此可能失去評估的準確性。

李信賢[2]認為在同一領域的網頁通常具有類似的網頁結構或共同的內容特徵。在擷取特定領域的網頁資訊時，需要針對該領域建立擷取規則，而針對不同領域的網頁擷取資訊時則需要另外一套擷取規則。因此使用 VIPS 切割與支援向量機（Support Vector Machine, SVM）技術擷取特定領域網頁資訊的方法，但使用者需要改變訓練時的特徵才能對不同領域的網頁

建立擷取規則以擷取資料，而且同樣會遇到切割出來的區塊未必是真正需要的資訊區塊的問題，實驗資料主要也只有來自 DBWorld[7]、IEEE[10]、ACM[4] 網站等網站。

3. 方法介紹

3.1 問題定義

在報名一個研討會時,通常需要一些資訊來幫助我們參加研討會，可能是時間或地點，而本研究的目標為使用程式從每個可能帶有研討會徵稿相關資訊的網站裡，自動擷取出使用者在投稿時，應該取得的六個重要資訊。

分別定義擷取對象為研討會名稱、研討會舉辦日期、研討會截稿日期、研討會地點、研討會主題與研討會原網站等六個欄位。

本系統處理的輸入資料為具完整研討會資訊的純文字檔案。系統會就六個欄位依事先訂定不同的規則做資料的擷取，以便後續的分析與評估。

3.2 系統架構

本研究的系統最主要是由幾個步驟所組成(如圖 1)，一開始先從 Google Search 的關鍵字查詢結果，依指定筆數得到符合查詢的網頁，接著使用 htmlUnit 取出每個網站的純文字內容，將每筆網站的純文字內容，以換行符號「\n」 進行字串的分割，目的將這些文字內容能依照原本網頁中的段落而分割成好幾個文字片段，每一個文字片段都儲存在陣列之中。再來取出一筆陣列中的文字片段給予程式分析，由系統的六個欄位之相關正則表示式分析欄位資訊，選取符合正則表示式的文字片段加入至該欄位的候選值之中，若當陣列中還有文字片段時，則重覆取出陣列中文字片段的步驟，直到陣列裡沒有任何文字片段為止，最後再從六個欄位候選值中篩選出最代表該欄位資訊的結果。

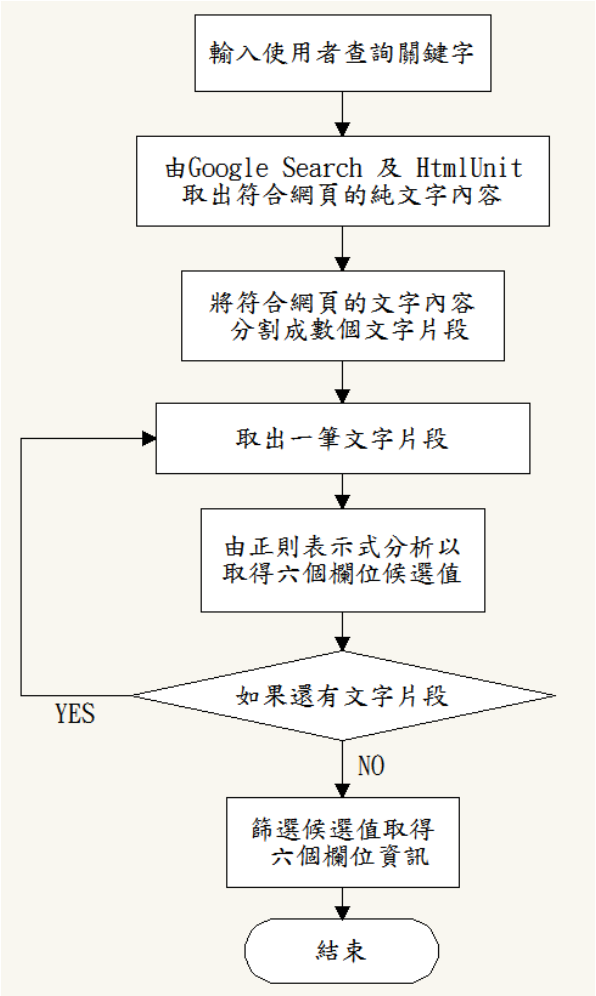

圖 1 系統架構

本系統在六個欄位候選值的選取上是將每一筆文字片段都輸入分析，其中因為文字片段有長有短，本研究因應文字片段長度不同的特性，定義了一個門檻值。若文字片段長度小於此門檻值，則定義為短文字片段，而若文字片段長度大於等於門檻值，則定義其為長文字片段。以下 3.3 節的方法即是分析這些兩種文字片段資料有沒有符合六個欄位資訊的介紹。

3.3 欄位候選值的選取

3.3.1 擷取研討會名稱

會議名稱的擷取主要藉由幾個正則表

示式的規則(表 1)，而規則又依據文字片段的長短有不一樣的擷取方式。其中表 1 裡的序號 1 到 5 主要使用在較短文字片段的擷取上。

在表 1 中，符號「.」代表任意字元，「研討會|會議」代表可允許字串包含研討會或會議，而「*」代表零到多個位數。中括號內的字元為可能出現的字元。若中括號內有「^」，則表示排除字元。而大括號內的數字為出現位數最小值與最大值。「\\w」為英文字元。「\\s」為空白字元。

在較短的文字片段方面首先最優先要考慮的是開頭有出現年份接著包含第幾屆，最後則是包含研討會或論文徵稿的字串，例如：「2011 第 22 屆國際資訊管理學術研討會」。其次可能有些名稱沒有年份但卻包含研討會或是論文徵稿等代表名稱的關鍵字。如果皆包含研討會與論文徵稿或是 Call for paper 等關鍵字也可以把這個文字片段加入進候選文字片段中。

一般來說網頁上呈現的研討會名稱大多是出現在前幾個標題的文字片段。由於研討會名稱可能是各種形式組成的名字，能確定的只有最後會出現研討會與會議，所以在較長的文字片段中，擷取的條件設置的比較嚴苛，目的主要是為了怕遺漏了研討會名稱是隱藏在文章段落中的情況。但若隱藏在文章段落中，則符合研討會名稱的正則表示式的候選文字片段皆會以逗點或一些標點符號來做區隔，因此不會整個文章段落都被正則表示式給篩選進去。

表 1 研討會名稱的篩選正則表示式

序號	正則表示式
1	“.*[0-9]{2,4}.*(屆).*(研討會|論文徵稿|會議|討會|論文發表會).*”
2	“[^，]{4,}(研討會|學術研討會|論文徵稿|徵稿通知|會議|討會|論文發表會)[\\s。]?”
3	“.{4,}(研討會|研究討論會|討會|論文發表會).*”
4	“[0-9]{2,4}.{6,}(研討會|論文徵稿|論文發表會).*”
5	“.*[0-9]{2,4}.*(屆|大學|學院).*”
6	“[^，投]{2,}(學術研討會|論文徵稿|國際研討會|會議|討會)”

3.3.2 擷取研討會舉辦日期與截稿日期

研討會舉辦日期與截稿日期的擷取使用相同的日期正則表示式(表 2)。在日期的正則表示式中，為了能夠擷取到阿拉伯數字及中文數字的各種樣式的日期，所以皆把這些字元列入考慮。又因為有些日期會在後面附上星期幾的描述故也將之列入正則式中。比較困難的情況是有些日期如：「2 月 17 日至 2 月 18 日」是連續的日期，則必須也將正則表示式的準則多加入這一個考量。截稿日期雖然與研討會舉辦日期使用相同的日期正則表示式，但不同的是很多網頁的截稿日期常常是描述在文章之中，也就是歸屬在較長文字段落裡，所以在擷取時要用比較口語化的正則表示式，才能將之擷取進來。

表 2 日期的正則表示式

日期正則表示式
“([0-9 元一二三四五六七八九十\\s]{1,6}(年|月|/)){1,2}([^，。截徵]{1,10}(日|號)|[0-9]{1,2})(([週(星期)元一二三四五六七八九十\\(（\\)）~～、]){0,7})?((至|到|、|~|～|及)([0-9 元一二三四五六七八九十\\s]{1,4}(年|月|/)){1,2}([^，。截徵]{1,10}(日|號)|[0-9]{1,2})(([週(星期)元一二三四五六七八九十\\(（\\)）~～]){0,8})?)?”

圖 2 是研討會日期、截稿日期、研討會地點與研討會原網址四個欄位在較短文字片段的擷取上，判斷的一個流程。除了文字片段本身可能符合程式要讀取的資訊，也有可能此資訊是被分成上下兩句的形式，因此程式也有進行預看下一句的考量。

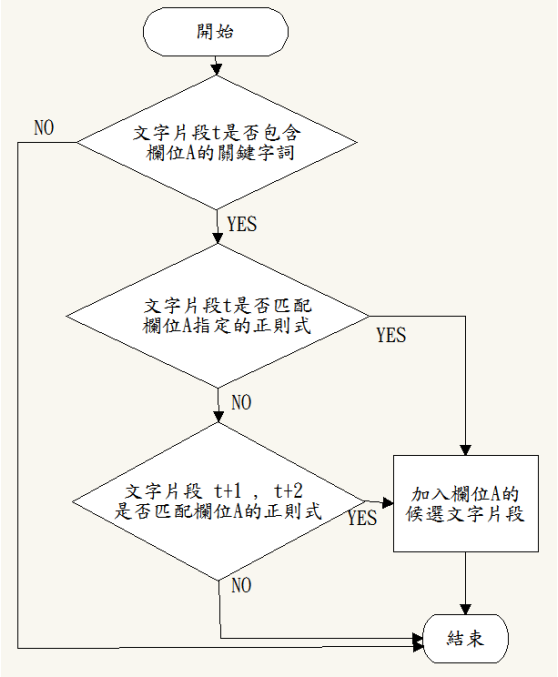

圖 2 短文字片段擷取流程

以研討會日期來說,首先判斷文字片段 t 是否包含「會議」、「開會」、「研討會」、「舉行」、「舉辦」、「發表」、「活動」等研討會日期的關鍵字。如果有，則判斷文字片段 t 是否也匹配日期的正則表示式。如果有，則將文字片段 t 加入至研討會日期的候選文字片段中。如果沒有，則判斷文字片段 t 的下一句，也就是文字片段 t+1 / t+2 是否有匹配日期正則表示式。如果有，則將文字片段 t+1 / t+2 加入至候選文字片段中，如果沒有，則程式結束。

截稿日期在短文字片段的擷取上也是採用圖 2 的流程。先找文字片段 t 是否包含「截止」、「截稿」、「交件」、「收件」、「定稿」、「載稿」等截稿日期資訊的關鍵字。如果有則判斷 t 是否匹配日期正則表示式。如果有，則將文字片段 t 加入至候選文字片段中。如果沒有，也考慮文字片段 t+1/ t+2 是否匹配日期正則表示式，若匹配就將 t+1/t+2 加入至截稿日期的候選文字片段中，沒有則程式結束。

因為有可能發生一種情況是這些日期出現在關鍵字後的下兩個文字片段之中，所以本系統在預看的判斷上有把下兩句囊括進來。但因網頁文字相同意義的資訊不會太分離的關係，所以本程式只檢查到關鍵字下兩句。

至於在長文字片段的研討會日期截取方面，大約會有兩種情形。一種是一個長文字片段中夾雜著如：「研討會日期：2011 年 02 月 01 日」的情況，這種情形雖然使用短文字片段的正則表示式可以擷取到此文字，但擷取時要排除文字片段中有包含逗點或是句點。因此將正則表示式修改成表 3 的形式，以免把其他欄位資訊的文字也擷取了進來。

表 3 研討會日期的長文字片段正則式(1)

正則表示式
“[^，。]{0,6}(會議|開會|研討會|舉辦|舉行|發表|活動)[^，。]{0,4}(日期|日|時間|日程){1}(:|：|為|是|－){1}”

另一種是長文字片段中出現口語化的日期描述，例如：「本研討會將於 2011 年 02 月 01 日舉辦」。所以要檢查「於」或「在」之後，是否有出現研討會的相關舉辦日期。因此使用新的正則式(如表 4)，此式代表的意義為開頭為「於」或「在」等字串，中間可以包含 0 到多個任意字元，但不能有逗點與句點，且後面應符合日期的正則表示式，又加入一些 0 或多個非逗號句號的字元，而最後則是有出現「舉行」、「舉辦」、「辦理」、「進行」或「召開」等字詞。如果符合，即將該文字片段加入到候選文字片段裡。

表 4 研討會日期的長文字片段正則式(2)

正則表示式
“(於|在)[^，。]*" + datePatten + "[^，。]*(舉行|舉辦|辦理|進行|召開)”

截稿日期在長文字片段中較口語化的描述中，會出現類似：「請於 2011 年 02 月 01 日前，將摘要寄至本研討會信箱」。這類型的文字就要先找出可能包覆日期的開頭字(例如：「在」、「於」)，和結尾字(例如：「前」)，如表 5 中的正則表示式：

表 5 截稿日期的長文字片段正則式(1)

正則表示式
“[^。]*(於|在)[^，。]*” + datePatten + “[^。]*”

另外也有符合表 6 的正則式所擷取出的案例，例如：「請盡速繳交摘要，至 2011 年 02 月 01 日止」。

表 6 截稿日期的長文字片段正則式(2)

正則表示式
“至[^，。]*” + datePatten + “[^，。]*止”

3.3.3 擷取研討會地點

研討會的地點也會有一般形式的資訊與出現在較長文字片段較口語描述的形式，所以擷取方式跟研討會日期與截稿日期使用類似的方法。但在地點的擷取正則表示式，有分兩種主要的基礎：一種是單純地址，如「研討會地點：新北市淡水區英專路 151 號」，第二種如「研討會地點：淡江大學驚聲樓國際會議廳」，所以在正則表示式上也分成地址正則表示式以及地點正則表示式兩種(表 7)。

表 7 地址與地點的正則表示式

名稱	正則表示式
地址	“(.{0,2}(縣|市|鄉|鎮|區))*.*路.*號”
地點	“.{2,22}(樓|館|所|廳|室|中心|堂|飯店|系|大學|學院|校區)”

地點的擷取可使用圖 2 中的流程，而差異在於地點的關鍵字為：「地點」、「地址」或「位置」。找到關鍵字之後一樣進行地址與地點正則表示式的比對，與下一個文字片段的比對。

而研討會地點資訊若隱藏在長文字片段之中，除了前面的擷取規則外，還會出現如：「本研討會即將在淡江大學驚聲樓國際會議廳舉辦。」這類型較口語的文字。因此需要使用表 8 中的正則式將其擷取之。即是先找出可能包覆地點的開頭字(例如「在」、「於」、「假」) 和結尾字(例如「舉辦」、「舉行」、「辦理」、「進行」、「召開」)，再檢查是否有符合地點或地址的正則表示式，其中「locPatten」為地點的正則表示式。

表 8 研討會地點較長文字段落正則表示式

正則表示式
“(於|在|假)[^，。]*” + locPatten + “[^，。]*(舉行|舉辦|辦理|進行|召開)”

3.3.4 擷取研討會主題

研討會主題資訊在純文字的擷取方面較困難，因為主題資訊包含各方面的資訊，沒有一些固定的文字可以確定主題包含哪些種類的文字。

在短文字片段的分析上，由較嚴謹到較寬鬆分成兩個篩選判斷(圖 3)，首先要找出可能的主題起始點，定義其為主題關鍵字，如：「主題」、「子題」、「議題」、「徵稿範圍」、「徵稿內容」、「主軸」、「研討題目」等。若文字片段 t 包含主題關鍵字則開始分析接下來的文字片段，直到文字片段 t + n 為止，t + n 為最後一個文字片段。

本研究將一些與主題無關的字，如其他五個欄位資訊的關鍵字與網頁最後出現的一些郵件、信箱等，定義成主題的停用字。因為主題資訊區塊較難找出其停止點，但可以確定若出現了其他欄位的關鍵字或是網頁結束的可能文字，即顯示主題資訊已終止。

之所以稱為較嚴謹，是因為除了要匹配主題關鍵字，也要匹配冒號關鍵字，冒號關鍵字定義為「：」、「是」、「為」、「有」等字，兩者皆包含，則開始分析接下來的文字片段。若接下來的文字片段沒有匹配主題的停用字，則判斷該文字片段是否有數字標題(圖 4)或相同符號開頭的字元，以圖 5 中的網頁畫面來看，每個主題前面有共同的項目圖案，在轉換為文字檔之後，這些項目圖案會變成某一個相同的字母(如圖 6)， 因此本系統在進行分析時，會先針對下面的字串擷取其開頭字，檢查是否一致。如果一致即加入主題候選字串之中，若無匹配數字標題或相同符號則繼續分析下一句文字片段。

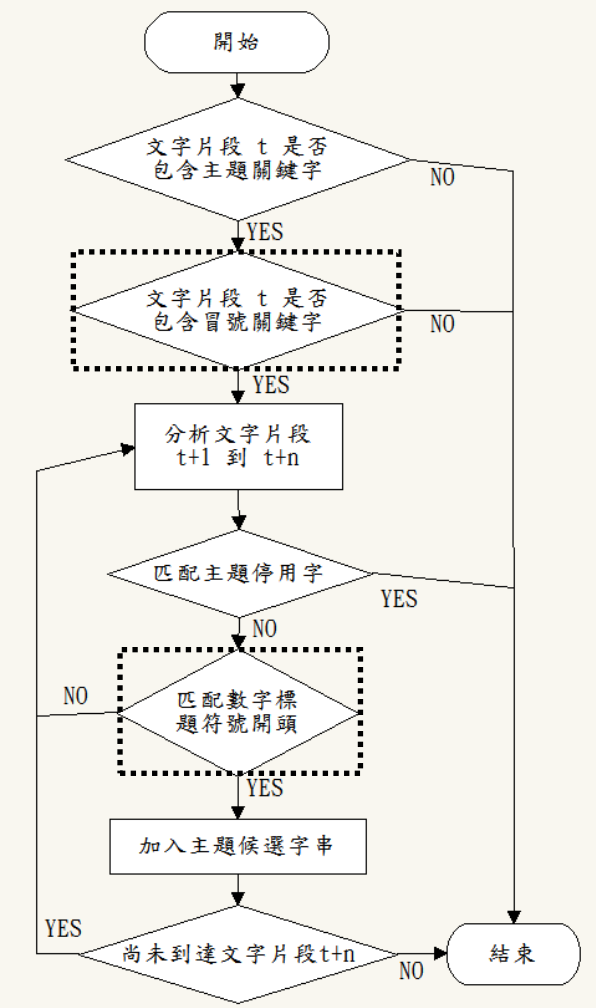

圖 3 主題欄位擷取流程(虛框為嚴謹模式使用)

二、研討會議題規劃

（一）迷群、名人、明星相關的文化與政治議題

（二）亞際間多地文化主體、組織的互動與參照

（三）文化參與、文化權利的民主政治

（四）流動的科技與性/別

（五）歡迎其他開放議題

圖 4 研討會主題網頁畫面(1)

研討會主題:

❖ 行銷與流通管理	❖ 網路行銷
❖ 財務管理	❖ 電子商務
❖ 生產與作業管理	❖ 兩岸管理議題
❖ 人力資源管理	❖ 非營利事業管理
❖ 企業電子化管理	❖ 服務業管理
❖ 全球運籌管理	❖ 中小企業管理
❖ 物流與供應鏈管理	❖ 管理教育
❖ 企業資源規劃	❖ 創業管理
❖ 顧客關係管理	❖ 社會責任及社會企業
❖ 科技管理	❖ 管理個案
❖ 知識管理	❖ 其他商管相關議題
❖ 資訊管理	

聯絡方式

圖 5 研討會主題網頁畫面(2)

研討會主題:

- 行銷與流通管理
- 財務管理
- 生產與作業管理
- 人力資源管理
- 企業電子化管理
- 全球運籌管理
- 物流與供應鏈管理
- 企業資源規劃
- 顧客關係管理
- 科技管理
- 知識管理
- 資訊管理

- 網路行銷
- 電子商務
- 兩岸管理議題
- 非營利事業管理
- 服務業管理
- 中小企業管理
- 管理教育
- 創業管理
- 社會責任及社會企業
- 管理個案
- 其他商管相關議題

聯絡方式

圖 6 網頁原始碼案例

但是有些主題並沒有包含這些項目符號(如圖 7)，下面的主題沒有包含項目符號，屬於個別的文字，下面的這些文字片段仍可能是主題的一部分。

研討主題：
農業生產(農業行銷、農業發展、農業轉型...)
農村環境(農村規劃、農村地景、環境生態...)
鄉土文化(歷史建築、農村文化、農村營造...)

舉辦日期：100年6月中旬

圖 7 研討會主題網頁畫面(3)

所以當較嚴謹的流程無法找出主題資訊時，便使用較寬鬆的流程。較寬鬆的流程為圖 3 中，捨去虛線框內的判斷，即當遇到主題關鍵字之後便開始新增下面的字串。如圖 7 中，在主題後面是舉辦日期的文字字串，因為舉辦日期屬於研討會日期欄位的關鍵字資訊，是另一個資訊的開始，所以匹配主題停用字，程式也停止主題尋找。

研討會主題也可能存在於長文字片段之中，除了一般的：「主題：A、B 與 C」這類型的規則外，還有可能有偏口語化的文字片段包含主題資訊。所以使用了不同的正則表示式(表 9)來擷取這類型的文字片段。

表 9 研討會主題資訊長文字段落正則表示式

正則表示式
“[^，。]*(以|舉凡|諸如|訂定|為|如[^何]|包括)[^，。]*(、|及|與|和)+[^，。]*(主題|子題|議題|文章|研討會)”

這類型的文字片段通常會出現「舉凡」、「如」、「訂定」、「為」、「包括」等字詞，這些字詞後面可能帶有主題資訊。但這樣範圍太大，因此在找到這些資訊後，也加上一些輔助字串來做判斷，例如在找到「舉凡」等字之後，如果有出現「、」、「與」、「和」或是「及」這些字串，顯示在舉凡之後的文字可能是指同一方面的事。如果後面又出現「主題」、「議題」、「子題」等字詞，系統就會認定這個字串片段是代表研討會主題資訊。

3.3.5 擷取研討會原網址

研討會網頁只是別人轉貼或是轉述，並不是該研討會網站網頁，這時候該網頁會附上研討會原網址來讓使用者參考，稱為研討會原網址。

網址的擷取種類比較簡單，除了一定要能匹配超連結的正則表示式之外(表 10)，也可使用圖 2 裡的擷取流程，先尋找網址的關鍵字。因為超連結的正則表示式較獨特，一般的字串如果不是提到相關的訊息皆不會出現超連結樣式的文字，所以前面的關鍵字可以定義的廣泛一點。本系統以「網址」、「網站」、「連結」、「網頁」、「Site」、「Web」、「詳細」、「來源」，讓尋找的範圍變大，並加上關鍵字與超連結都符合的要求，也讓程式不易擷取到錯誤的文字片段。

表 10 超連結正則表示式

正則表示式
“(http(s)?(:|：)//)?([\\w-]+\\.)+[\\w-]+(/[\\w-./?%&=~]*)?”

3.4 篩選候選字串

在程式擷取六個欄位的過程中，一個網頁裡可能每一個欄位都會有一個以上的字串被正則表示式找出，這些字串都可能是真正屬於該欄位的正確值，因此需要進行第二次的篩選，而這些可能屬於該欄位的文字字串就定義為候選字串。在將一個網站的每個文字片段透過程式選取進候選字串之後，六個欄位可能不一定有文字被篩選進來，同樣的也可能會有多組候選字串，這時候要進行第二步的篩選。

這部分的篩選主要考慮該欄位常出現的字詞或符號，如果有出現，相對的優先考慮，如果都沒有則是以佇列裡的第一筆為優先。

4. 實驗結果

4.1 實驗設計

4.1.1 資料來源

本研究的資料來源為經由 Google Search 查詢後的多筆結果。為了不局限於單一領域，故以「研討會 徵稿」當作關鍵字來進行查詢。又因為需要大量的資料來測試，所以使用查詢結果會比較廣泛的查詢關鍵字，讓研討會的網頁能夠來自各種可能的探討主題。本系統暫將資料以中文網頁為基礎，故加入了繁體中文的參數。

為了能讓程式能動態的取得 Google Search 查詢後的最新結果，所以利用 Java 開發套件 htmlUnit 來撰寫程式，透過 htmlUnit 以進行 Google 與每個查詢網頁的連線，並取得網頁的表單，點擊、擷取結果網頁的超連結。使用者只需給定關鍵字，程式就能自動的擷取出想要的筆數，不過最大的筆數是根據 Google Search 結果的最大筆數而定，使用者可以依自己的需求輸入不同的關鍵字與筆數來用 Google Search 取得初步的查詢結果當作資料來源。

因為資料來源皆是從不同格式的網頁上取得的非結構資料，所以統一利用 htmlUnit 將每個網頁的文字內容擷取下來，該文字內容為原始網頁裡的每個 HTML Dom Node 之中的 Text Node，即每個文字片段。使用 htmlUnit 擷取下來的優點是這些文字片段的順序跟網頁原始碼中的順序是一樣的，其中空格或換行也是以網頁原本的結構為主。本系統利用這些特性，將每個網頁中的 Text Node 取下後供程式進一步分析使用。

4.1.2 網頁文字擷取

本研究的資料皆是轉換為純文字內容，來讓程式進行分析。這些文字內容在經過轉換後仍會保有網頁上的順序以及部分結構(圖 9)。我們可以發現到網頁上的呈現可能跟原始碼中有些差距，例如網頁的原始碼可能如圖 10 所示，因為字型需要或是符號、中文、數字的不同，所以在 HTML 標籤上可能分成好幾個片段，這樣即使將 HTML 標籤資訊拿來當作文字資訊的分析條件也會變得很困難。因此本研究使用 htmlUnit 套件來將文字擷取下來，這樣的好處是，取下的文字仍會保有網頁原本的順序與部分結構(圖 10)。

三、**承辦單位：**國立台北教育大學台灣文化研究所

四、會議時間：2009年8月29日(六)、30日(日)

圖 8 網頁所見樣式

```
<font size="3">
        <span style="font-family: 新細明體">
                三、承辦單位：
        </span>
        <span style="font-family: 新細明體">
                國立台北教育大學台灣文化研究所
        </span>
</font>
<font size="3">
        <span style="font-family: 新細明體">
                四、會議時間：
        </span>
        <span style="font-family: 新細明體">
                2009
        </span>
        <span style="font-family: 新細明體">
                年
        <span>8</span>月
        <span>29</span>日
        <span>(</span> 六 <span>)</span>
                、
        <span>30</span>日 <span>(</span>日<span>)
        </span></span>
</font>
```

圖 9 網頁原始碼案例

三、承辦單位：國立台北教育大學台灣文化研究所
四、會議時間：2009年8月29日(六)、30日(日)

圖 10 HtmlUnit 擷取文字案例

文字在經過 htmlUnit 從網站上擷取下來之後，這些網頁內容中的文字被區隔開，也就顯示同一組文字是具有相同欄位的資訊，但是有些網頁中的文字內容是一整篇的長段落而沒有被區隔，本研究以一個文字片段來當擷取的每一個單位，由於文字片段在長短不同時，代表的欄位資訊數量不同，因此本研究依文字個數定義了一個門檻值，來區隔短文字片段與長文字片段，而一個網頁中可能會有好幾個文字片段，當文字片段的字數小於門檻值時，定義為短文字片段，認定其只隱含一個意

義，例如圖 11 中的「三、承辦單位：國立台北教育大學台灣文化研究所」或是「四、會議時間:2009 年 8 月 29 日(六)、30 日(日)」皆屬於短文字片段，這類型的文字片段通常只會包含一個欄位的值，而且文字描述也沒那麼口語化。若一個文字片段的字數大於等於門檻值時，定義為長文字片段。這類型的文字片段通常包含多種欄位的資訊，且文字的描述也較口語，偏向自然語言，需要使用其他不同於較短字串的正則式來擷取出本研究想擷取的六個欄位資訊。

4.1.3 長短文字片段門檻值參數

本系統測試了 20~50 的文字長度之後，發現以 38 當作區別長短文字片段的門檻值，擷取欄位的精確率與召回率會有比較好的表現，因此取 38 作長短文字片段的門檻值。

4.1.4 條件限制

有些透過 Google Search 後的查詢結果屬於文件格式，例如 Word 檔案或是 PDF 檔案，雖然這些檔案也是有文字內容，但這些文件格式需要使用其他軟體來開啟，才能讀取其檔案中的文字，本研究暫不考慮這些型式的查詢結果，而有些結果為頁框型的網頁因為無法取得真正要的文字內容，故也將之排除

又因為一些網頁上的 javascript 程式或是 CSS 樣式，可能讓系統無法順利的進行網頁資料的擷取，故使用 HtmlUnit 抓取網頁時，關閉其解讀 javascript 和 CSS 功能，以利程式執行。

4.1.5 測試集的製作

本研究由上述資料來源找到 335 筆網頁，拿來製作六個欄位的答案集以供程式擷取後的答案比對用。其中在答案集裡取了研討會資訊較完整的 129 個網頁，這 129 個網頁在研討會名稱、研討會時間、截稿時間、研討會地點、研討會主題等五個欄位中皆有人工標記可匹配欄位的完整答案，因此取之當作實驗設計的評估答案集，而其對應的網頁的網址即為程式測試的資料集。

本實驗流程為將此 129 筆網頁的網址給予程式執行，經由程式擷取出六個欄位的資訊，再將此結果與答案集進行比對，評估六個欄位的精確率與召回率。

4.2 實驗環境

本研究的實驗環境如表 11：

表 11 實驗環境

參數名稱	設定值
作業系統	Windows XP Professional
RAM	2.00GB
CPU	Q8200 2.33GHz
開發環境	Java: 1.6.0_21

4.3 評估方式

由於程式與答案集皆屬於文字資料，本系統以每一筆網頁的每一個欄位當成一個單位，進行程式結果與答案集文字的文字面積比對，來算出程式結果的精確率與召回率。

在精確率的部分，以程式結果的文字與答案集的文字交集的面積當成分子，以程式結果的文字面積當成分母。

在召回率的部分，以程式結果的文字與答案集的文字交集的面積當成分子，以答案集的文字面積當成分母。

本系統為了評估每一筆網頁的六個欄位的文字資料，將該網頁所有文字存在一個字串 S 之中。藉由答案集的任一欄位 A 中的文字在字串 S 裡的下標數字集合，當成欄位 A 的正確解答範圍。因為答案集之中的每個欄位的答案值皆是從網頁中的文字裡找出，所以欄位 A 的答案文字必定會在字串 S 裡找到下標值的範圍。同樣的程式是自網頁中所有文字也就是字串 S 之

中，嘗試找出符合要求的欄位值，因此經由程式找出的欄位值一定也在字串 S 中存在下標值。將程式擷取與答案集同樣的欄位進行下標值的比對，即可求算出交集面積，故可計算成比對結果。

4.4 實驗結果

實驗結果如表 12，實驗資料：129 筆網頁，六個欄位的精確率與召回率如下：

表 12 欄位評估結果(單位：%)

欄位名稱	精確率	召回率
研討會名稱	93.18	91.31
研討會日期	86.49	85.44
截稿日期	93.99	93.79
研討會地點	92.59	89.65
研討會主題	88.96	90.23
研討會原網址	93.15	91.97

結果顯示，本研究在研討會名稱、截稿日期、研討會地點及研討會原網址皆有 90% 以上的精確率。研討會日期與研討會主題的擷取效果上皆有 85%以上的精確率與召回率的成效。

5. 結論與未來發展

現今網路資訊多是透過搜尋引擎來找尋相關資料，在搜尋引擎廣泛使用的情況下，本研究只用搜尋引擎的第一次查詢結果，當成資料來源進行後續處理。本研究利用 Google Search 廣泛的網頁結果，來做研討會資訊的擷取。搜尋引擎的結果網頁裡，往往夾雜各種形式的網頁，本研究利用 HtmlUnit 套件將網頁保留原本的結構與順序，以純文字的形式取出，利用相關文字、正則表示式的交錯配合，包含加入找出特定字串，進行預看文字片段等機制，擷取出「研討會名稱」、「研討會日期」、「截稿日期」、「研討會地點」、「研討會主題」、「研討會原網址」等六個研討會資訊。

在精確率的表現上，除了研討會主題資訊為 88.96%和研討會日期為 86.49%之外，其他資訊皆有 90%以上的成效，而在召回率上除了研討會日期為 85.44%與研討會地點 89.65%之外，其他資訊也有 90%以上的成效。顯示在純文字的分析上，仍然可以有不錯的效果，幫助欲投稿的使用者，可以省去一筆一筆查看搜尋網站結果的時間，加速查詢投稿研討會的效率。

研討會資訊一直是每年許多欲參與研討會的人需要查詢的資訊。由於現在研討會資訊擷取的研究上以英文網頁為主，所以本文針對中文先發展一套中文的研討會資訊擷取系統，有效的進行欄位的擷取。將來會將資料範圍擴大至英文網頁並且結合了其他讀取軟體分析更多檔案類型，諸如 PDF 或 word 檔案等搜尋引擎查詢結果。目前本研究只針對 Google Search 的結果做處理，在未來也希望能把其他搜尋引擎，如 Bing、Yahoo 等的查詢結果都囊括進資料來源以做分析。最後希望能自動找出各個研討會真正原始的研討會網頁，以讓研討會投稿者有更正確研討會資訊。

參考文獻

[1]史嘉淋，***應用資訊擷取技術實作研討會資訊檢索系統***，國立嘉義大學資訊工程學系，2007。

[2]李信賢，***使用網頁切割與支援向量機技術擷取特定領域網頁資料***，國立嘉義大學資訊工程學系，2010。

[3]胡姝涵，張嘉惠，***會議公告網站資訊擷取之研究***，第十一屆人工智慧與應用研討會，2006。

[4]ACM , http://portal.acm.org/conferences.cfm?CFID=10820168&CFTOKEN=91350223

[5]C. D. Manning, P. Raghavan and H. Schütze, *Introduction to Information Retrieval* , Cambridge University Press , 2008

[6]D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma, *VIPS: A Vision Based Page Segmentation Algorithm*, Microsoft Technical Report,

MSR-TR-2003-79, 2003.

[7]DB World , http://www.cs.wisc.edu/dbworld/browse.html

[8]HtmlUnit, http://htmlunit.sourceforge.net/

[9]I. H Witten, E. Frank. *Data Mining* , Morgan Kaufmann, 2005

[10] IEEE, http://ieeexplore.ieee.org/Xplore/dynhome.jsp

[11] Regular expression (regex or regexp), wiki , http://en.wikipedia.org/wiki/Regular_expression