網際網路學習式購物代理人之設計與建構

黃 國 倫
元智大學資訊管理學系碩士班
s887741@mail.yzu.edu.tw

李 錫 捷
元智大學資訊管理學系副教授
imhlee@saturn.yzu.edu.tw

摘 要

由於網際網路的便利性，使用者只要透過搜尋引擎就可以找到許多相關的商品、服務或內容資訊。然而對於搜尋引擎所查詢到的資料廣度及深度都已經有非常不錯的結果，但是面對網頁內容不一，而且又不是以標準的格式來呈現，也造成資訊蒐集及整合工作上會有相當大的負擔。

本研究希望運用代理人技術，並透過學習式機制，其中主要結合智慧搜尋、資訊過濾、資訊擷取、規則學習的方法，以提升資訊擷取的品質，達到資訊蒐集目的。最後建置成購物代理人系統，並以網路上的購物資訊作為資料來源，讓使用者能多樣化的查詢特定商品資訊並提供個人化資訊整合服務。同時以評估系統對於資料的收集能力和擷取的正確率，來作為驗證本研究可行性和實用性之所在。

關鍵字：搜尋引擎、學習式、資訊過濾、資訊擷取、購物代理人

一、緒論

自從全球資訊網成為常見的資訊提供來源後，在網頁的資料呈現方式就一直被研究及改進。從早期只能有文字、圖形的網頁內容，接著有了JAVA、ASP、PHP 等網頁程式開發語言，使得網頁內容可以輕易的與資料庫連結，這也讓網頁呈現上能有變化及彈性。然而當網際網路上充斥了許多資料，使用者當然需要透過網路尋找特定資料，因此如何在廣大網路中搜尋到真正需要資料，也就非常重要[3]。早在全球資訊網發展的初期，就有一些網站提供如何快速尋找網址的搜尋引擎服務，相信到目前為止大部份的人都知道如何使用搜尋引擎。現今網路提供的搜尋引擎或入口網站也不少於二、三十個，由此可知資訊搜尋一直是網路使用者所需要的。

然而搜尋引擎提供的是較廣泛搜尋方式，所以往往得到的也包含許多不正確資料。這樣的結果對於資訊蒐集或是特定資訊擷取應用上就會造成很大的問題[6]，其中有幾項因素造成了處理的障礙：

- 網頁更新快速。由於資訊的變化非常迅速，而網際網路正好可以配合上這種特性，使得內容提供網站(ICP)必須不斷的更新與強化網頁內容，才能把最新資訊提供給使用者。在資訊蒐集上，同時也具有時效性，如何在資訊過時前充分利用，並且能夠適時更新資訊，以創造出符合使用者需求的價值，是一項極大的挑戰。
- 資訊過量現象。網際網路是開放的、流通的，所以資訊可能分散或存在網路各地。當使用者面對如此廣泛且龐大的資訊來源時，必須以最快、最符合個人需求的搜尋功能找到相關資訊。目前許多研究針對這個問題提出解決方法，如搜尋引擎(Search Engine)、智慧型代理程式(Intelligent Agents)等[10]。
- 資訊格式不一致。這使得使用者針對資訊加以過濾或整理時，沒有一個準則或是固定的方

法，造成資訊蒐集成本增加，更何況是將處理過的資訊加以利用。以目前搜尋引擎所查詢的結果，只是把相似資料回傳給使用者，並無法針對資料內容加以篩選、整合。因此如要有效解決該問題必定需要有一套良好的資訊擷取工具，透過內容的分析過濾取出真正資訊[5]。

過去有許多學者投入了資訊擷取或資訊蒐集相關研究，突破了網頁格式不統一的問題，使得能從內容中粹取出有用的部份，這也是資訊的價值所在。然而為了提高資訊擷取精確度，學者試圖在擷取過程中找尋方法，這些方法就像是樣板式擷取[8]所比對出來的樣板(Template)或是規則式擷取所找到的文脈規則(Contextual Rule) [1]。因此這些像是經驗一樣，而且是可以透過訓練得到的經驗。有了這個概念，在本研究的主要目的有三：

- 自動化資訊蒐集與擷取工作

希望能把原本花費人工成本的資訊蒐集及擷取工作交由電腦負責，因此在系統建置上，是以網路網路購物代理人為目標，透過系統運作到網路上蒐集許多購物網站上所提供特定的商品資訊，把真正有用的資訊抓取出來，如：品牌、價格或規格等，並且整合成統一的資訊呈現介面，讓使用者可以輕鬆查詢商品資訊。

- 提供資訊擷取上的學習機制[13]

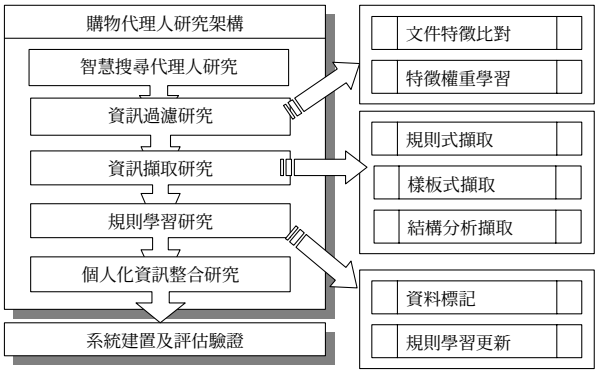
學習目的是希望能夠有更精確的擷取能力，因此每次擷取的過程就是針對資料在作訓練，使用者可以借由調整結果，使得系統學習出來更好的規則，另外系統內的規則是可以再利用並且互相共享的，這讓系統減少重覆學習上的額外成本。

- 提供客制化的資訊整合服務與知識管理

在資訊整合上，建立成為提供其他系統與資料源中間的包覆程式(Wrapper)[12]，讓蒐集的資訊可以再創造價值。在如果呈現上，則是以標準的結構化文件方式，提供使用者新穎、更多強而有力的解釋和更有效系呈現知識的理想環境，透過自訂介面得以篩選、整合系統所擷取之資訊與個人電腦上原有之資訊，或其他網際網路上的所有相關資料，以達到個人知識管理的功能。

二、研究內容與方法

本研究架構，如圖表 1 所示。



圖表 1：研究架構圖

(一) 智慧搜尋代理人

首先在使用者新增商品的查詢記錄後，必須根據查詢的條件尋找出可能的網頁，在這裏並不強烈要求得到網址準確度非常高，由於之後的處理上會根據網頁的內容加以過濾排除，所以目前網路上搜尋引擎所提供的搜尋服務已經相當成熟完備，因此本研究並非建立自己的網址搜尋功能，直接透過目前常用的搜尋引擎，如：Google, Openfind 尋找出可能相關網址。

然而往往一次查詢所找到的結果非常多，因此搜尋引擎在呈現上都會自動分頁，這也造成了擷取上必須能夠自動換頁才能把所有結果抓完。如此重覆直到將所有結果擷取完畢，最後去除相同網址後儲存。

(二) 資訊過濾

雖然透過搜尋引擎查詢到許多相關的網址，可是搜尋引擎是透過關鍵字的搜尋方式，亦即網頁

內容中有出現關鍵字的網址都會被搜尋出來,可是搜尋出來的網址雖然在關鍵字比對上有相當程度的相似,但也並非是購物網站或報價網站,尤其絕大部份搜尋出來的都是新聞網頁或文章,這會造成之後資訊擷取上誤判,所以必須能夠將它們過濾出來。

資訊過濾處理主要是為了解決所搜尋回來網址並非都是購物網站或報價網站。在這個部份,本研究採用「文件相似度比對」,來過濾不要網址。

首先,要找出購物或報價內容的文件特徵,一般常用的是以計算詞頻的關鍵詞所建立特徵向量,因此計算前必須先去除掉虛詞及 HTML Tag 符號,並且透過事先建立的詞庫斷字斷詞取得出現在文章內容中頻率高的關鍵詞來代表讓文件特徵。然而這種方式適合一般性的文章求取特徵,可是針對購物或報價網站大多為表格或依序排列方式的結構性較高文件,如果透過關鍵詞所計算出來的詞頻幾乎都是跟商品名稱、型號或規格相關的詞句出現頻率較高。這會造成處理不同類商品時,得到差異極大的特徵向量,因此並無法代表購物或報價內容的文件特徵。

為了解決詞頻計算的問題,本研究改由人工設定關鍵詞,即由人來決定該有那些關鍵詞當作系統預設文件特徵,如表格 1,並且每一個關鍵詞都有權重來代表出現的頻率。所以在資訊過濾處理上便將每一個搜尋到的網址內容,同樣地計算出該網址內容中出現相同文件特徵的次數,作為該網址的特徵向量,而比對時就以每個網址的特徵向量與設定的特徵向量求得向量間的距離,亦或稱相似度。

表格 1:關鍵詞的文件特徵及向量

關鍵詞	型號	售價	訂購	特價	產品	規格	搶購價	廠牌
權重	1	1	1	1	1	1	1	1

1. 權重學習訓練及更新法則

雖然由人工設定關鍵詞能夠找出代表文件的特徵,但是對於每個詞的重要度卻可能有所差異。如果沒有好的對應權重,當資訊過濾處理時,在文件相似度比對就會造成過濾上的不正確,也會影響接下來的擷取處理。

本研究中採用了類神經網路的監督式學習訓練及權重更新法則,在學習前先任意設定初始的權重值,並且給定一個門檻值 ρ ,其中 $0 \leqq \rho \leqq 1$。當大於 ρ 代表了兩個向量相似,小於 ρ 兩個向量不相似。接下來由使用者輸入需要學習的網址,其中包括了正確的網址及不正確的網址。有了這些網址,便將每個網址內容代表的特徵向量計算出來,並且與設定特徵的權重向量比對。當比對出來的相似度大於 ρ 而且又是正確的網址時,那麼代表預設的權重正確,因此不需要更新,如果小於 ρ ,那麼表示權重不正確需要調整,其中調整的幅度ΔW是由兩個向量間每個元素的距離、訓練次數及更新比率 α 計算求得,見方程式(1,而調整方法見方程式(2。反之,比對出來的相似度大於 ρ 但卻是非正確的網址,因此需要增加彼此距離,並且反向調整ΔW使差異加大,見方程式(3。

$$\Delta W_{Ji} = \alpha^{(1+\frac{\beta+1}{r})}(I_i - W_{Ji}^{(old)}), \forall i$$

α 更新比率　$0 \leqq \alpha \leqq 1$

β 訓練次數

γ 為常數　$0 \leqq \gamma$

I,W 為欲比對之文件特徵向量與權重向量

方程式(1):文件特徵權重調整計算

$$W_{Ji} = W_{Ji}^{(old)} + \Delta W_{Ji}, \forall i$$

方程式(2):正確網址之權重更新方法

$$W_{Ji} = W_{Ji}^{(old)} - \Delta W_{Ji}, \forall i$$

方程式(3):非正確網址之權重更新方法

當經過多次訓練後，其中的權重將不再產生重大的改變，而這種狀態稱為收斂，亦即達到 $\lim_{\beta\to\infty}\Delta W = 0$。

2. 文件相似度比對

有了學習訓練得到的文件特徵權重，代表了內容已經被轉換成關鍵詞組成的數值資料，這些資料也就是用來比對的來源。如果希望找到的網頁能夠符合購物或報價資訊，那就必須與訓練出來的權重相似。在本研究中主要是採用 SimNet（方程式(4)）中之核心函數來計算各個網頁的文件特徵向量及學習訓練的特徵向量之Match Degrees 用來表示文件的相似程度。

透過學習訓練方法分析出網頁內容應該具備特徵及權重，並且有一個能夠衡量彼此相似度的數學函數，因此資訊過濾處理上便以此為篩選準則，當相似度大於門檻值 ρ 時，表示該網頁極有可能包含商品資訊，也就必須進一步執行擷取工作；相反的，相似度小於門檻值 ρ，那麼這些網址都必須被剔除掉。針對門檻值 ρ 的決定上並沒有一定的標準，而是根據使用者對於資料篩選出來的精確度與數量來評估。當希望得到正確率高的資料，那麼可以提高 ρ，相反的，如果不希望遺漏任何可能資料，則可以降低 ρ 使過濾上能更寬鬆。

$$MD(I,W)=\sqrt{\frac{(\sum_i \min(I_i,W_i))^2}{(\sum_i W_i)(\sum_i I_i)+\varepsilon}} \quad \exists\varepsilon\in R,\ 0<\varepsilon<<1$$

W,I 為欲比對之特徵參考向量之單位向量

MD(I,W)兩個向量間的相似度

方程式(4)：SimNet 相似度比對[7]

(三) 資訊擷取

本研究的主要部份，是綜合資訊擷取、蒐集與整合技術之研究，採用了三種不同的擷取方法，透過不同方法達到彼此互補而能夠有較好的擷取精確度及資料量，以下分別詳述之。

1. 資料分析擷取

網頁文件並沒有統一的格式，但在資料型態及特性上仍然有脈絡可尋，所以要從已知的資料屬性開始分析[9]。其中常見的部份包括名稱、價格和規格。而名稱和價格是必要的，如果缺少任何一個都會使購物資訊沒有意義，見表格 2。

所以使用資料分析擷取的方法上，便是從名稱和價格為主要優先搜尋對象，確定找到這兩個部份，才從附近位置判斷出其他可能資料。其中名稱部份是由文字所組成，而且沒有太明確的規則可以判斷，除非在系統中有限定擷取某一項商品名稱否則建議從價格著手。價格資料很清楚一定有數字，並且在數字的前後可能會出現特別的文字或符號，如＄、元等，因此透過這種文字樣式可以判斷出價格位置，其他資料就可利用長度來判斷內容。另外一個分析方法是從資料的先後順序來找出規則。有了這樣規則，在擷取過程中可以協助判斷出每個資料的可能性。

表格 2：購物資訊包含的內容及特性

內容	型態	順序	資料特性
名稱	文字	前面	中等長度文字
價格	數字	中間	配合上其他文字符號 如$,NT,元
規格	文字	中間	大多由較長的文字組成
來源	文字	後面	較短的文字(長度小於 5)
廠牌	文字	最前面	較短的文字，且為連結 (長度小於 5)
日期	數字	後面	擁有特定格式，如##/##/##

2. 文件結構擷取

由於網頁的結構是由許多 HTML 標籤所構成，每個標籤也代表了資料的呈現方式與資料彼此

的關係。所以在資訊擷取上是利用這半結構特性，來輔助資訊取得。

在方法步驟上，先將網頁轉換成樹狀的標籤結構，這時候會發現相同的資料欄位上也會擁有相同的標籤結構。因此每個結構必須事先建立出結構樣本。結構樣本的目的，是為了能夠從樹狀的標籤中判斷出資料的位置，例如：表格結構中，必須能將所包含的資料分析出來，見表格 3。

當實際運用時，也會根據不同的結構樣板有不同的擷取方法，在本研究中，因為擷取的資料多為表格排列的形式，因此只採用了表格標籤結構樣本。在處理上只搜尋表格標籤，如果搜尋到再由結構樣本取出資料，因此這種方法對於大量且重覆性質資料擷取非常有用。

表格 3：表格標籤結構樣本

表格標籤結構樣本： 〔Tag:TABLE〕 〔Tag:TR〕 〔Tag:TD〕〔Data:TEXT1〕〔Tag:/TD〕 〔Tag:TD〕〔Data:TEXT2〕〔Tag:/TD〕 〔Tag:/TR〕 〔Tag:TR〕 〔Tag:TD〕〔Data:TEXT3〕〔Tag:/TD〕 〔Tag:TD〕〔Data:TEXT4〕〔Tag:/TD〕 〔Tag:/TR〕 〔Tag:/TABLE〕
擷取出來資料： Record1:〔TEXT1〕〔TEXT2〕 Record2:〔TEXT3〕〔TEXT4〕

3. 規則學習擷取

在這裏主要提出有效的規則學習機制及規則擷取方法針對已知的網站或曾經擷取過的網址，能學習得到文脈規則(Contextual Rule)[1][2][4]，這樣的規則會被保留下來，只要在未來需要擷取相同的網址或網站，都能夠利用這規則達成資訊擷取。接下來針對規則的學習及擷取進一步說明。

- 規則學習

在規則建立時，必須由人來告訴電腦什麼是需要的資料什麼不是，這稱為標記(Labeling)，然後交給電腦學習而得到規則，也就是監督式學習方法。在標記格式上必須指定出每一個資料屬性(Attribute)的範圍，在範圍內的文字代表該屬性內容。除了決定每個資料屬性外，還必須指定每一筆記錄(Record)的範圍，見表格 4。計算上是以網頁內所有標籤的順序來決定，即把網頁的內容透過分析器(Parser)轉換成標籤列表(Tag List)。而真正需要的資料也是存在於文字標籤之中，因此所設定屬性的範圍也必須在文字標籤的位置。

表格 4：記錄(Record)的標記內容

名稱	內容
G	標記第一筆記錄開始至最後一筆結尾
R	標記每筆記錄的開始至結尾

標記的方式則讓使用者指定每個屬性名稱及範圍，其中範圍表示開始至結束位置，因此每個名稱會出現兩個位置。而屬性排列順序必須按照每筆記錄中各個屬性出現的先後位置來決定，不可以前後顛倒，如表格 5。

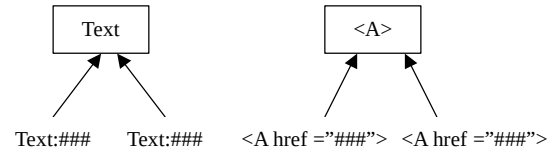
表格 5：YAHOO 購物網站資料標記

G 258 R 258 name 258 name 258 spec 275 spec 295 price 302 price 302 R 302 R 326 name 326 name 326 spec 343 spec 363 price 370 price 370 R 370 G 914

當完成了標記動作後，交由電腦學習過程中會將標記內的兩個屬性之間，轉換成邊(Edge)，邊

表示了前一個屬性到下一個屬性之間的標籤格式,而學習的主要工作也就是分析出存在於相同屬性邊之文脈規則。相同屬性的邊其實具備相似的結構,這是由於一般的網頁大多是查詢資料庫所呈現出來的結果,所以網頁撰寫上也會以迴圈來執行,除了資料會改變外,其標籤結構幾乎不會改變。

學習規則時,將相同屬性的邊排列在一起,有了對齊的標籤資料,就可以求得每個標籤在分類樹(Taxonomy Trees)(見圖表 2)中的分類項目並替代之,即計算出能替代相同欄位之分類標籤,最後合併每個邊,若有不同的標籤結構,則用「|」符號代表「OR」。



圖表 2：標籤分類樹(Taxonomy Trees)

然而對於邊界(Boundary)屬性的學習上,如G、R,因為只有單一個位置無法找到固定的邊,故不能直接採取上述的學習方法,必須根據該位置找到可能的邊,在這裏分為向左及向右對齊方式。向右對齊,包括:G-R、R-###,是以標記位置為主,然後向左找到相同結構之標籤成為邊;相反地,向左對齊包括:R-G、###-R,是向右找到相同結構之標籤。以 R 而言,所找到的標籤必須能夠表示一筆記錄的開始到第一個屬性及最後一個屬性到該筆記錄結束的結構,因此結構至少包括了一個文字標籤在之中;而 G 是為了能夠在整份文件之中定位出所有資料的範圍,所以在找到的邊必須是文件之中的唯一結構,即不能有其他重覆結構,否則會造成定位資料上的錯誤。

為了避免 G、R 可能標記在相同位置,所以順序上是先學習出所有 R 的規則,接著 G 再從 R 的位置開始學習,如此可以避免 G 和 R 學習出重疊規則情形。而規則的產生方法,則跟上述的相。

- 規則擷取

經由學習產生的規則,其實就是代表每個屬性與每筆記錄內的標籤結構,因此也就能夠再從規則中找到每個屬性與記錄的位置。在擷取方法是以 G-R 第一個搜尋結構,因為任何的擷取都必須定位資料範圍。只要搜尋到 G-R 的位置,接下來搜尋以 R 為開頭的邊(R-###),只要有搜尋到位置,那麼表示該屬性的開始位置,如果再搜尋以該屬性為開頭到下個屬性的邊,那麼屬性的結束位置也就找到,這時候就能夠擷取出開始位置及結束位置內的文字資料,即是該屬性的資訊。如此一直循環下去,當搜尋到 R 結尾的邊(###-R)時,表示該筆記錄的結束,如果搜尋到 R-G,那麼整個擷取工作完成。如上述例子,最後擷取出來的結果為表格 6。

表格 6：YAHOO 購物網站規則擷取結果

name	spec	price
Palm Vx	Palm Palm OS 8 MB 160 x 160 No	$235.00 - $433.00
Palm m505	Palm Palm OS 8 MB 240 x 320 Yes	$382.00 - $469.00
Palm VIIx	Palm Palm OS 8 MB 160 x 160 No	$85.00 - $454.74
Palm IIIxe	Palm Palm OS 8 MB 160 x 160 No	$194.95 - $275.00
Palm IIIc	Palm Palm OS 8 MB 160 x 160 Yes	$237.00 - $448.95
Palm m100	Palm Palm OS 2 MB 160 x 160 No	$95.00 - $199.00
Palm m500	Palm Palm OS 8 MB No	$338.98 - $429.00
Palm m105	Palm Palm OS 3.5 8 MB No	$166.00 - $210.00
Palm VII	Palm Palm OS 2 MB 160 x 160 No	$178.99 - $489.00
Palm IIIx	Palm Palm OS 4 MB 160 x 160 No	$175.99 - $314.99

- 規則更新

其實學習得到的規則，會影響擷取資訊的正確度，然而往往在標記的過程中若不適當或是網頁結構上有部份更動，都會造成資訊擷取的錯誤。當規則擷取無法正確處理時，只要針對擷取不正確的記錄標記，再學習並能更新之前的學習規則，使得規則能夠成長完整，達到資訊擷取的正確性，這也是本研究中學習的主要目的。

所以在更新方法上，首先將後來學習的規則取出每一個邊，再把相同邊的標籤與原有規則比對，比對上採取之前的方法，即對齊後以分類樹的分類項目取代，最後再合併每個邊的標籤即完成了規則更新。

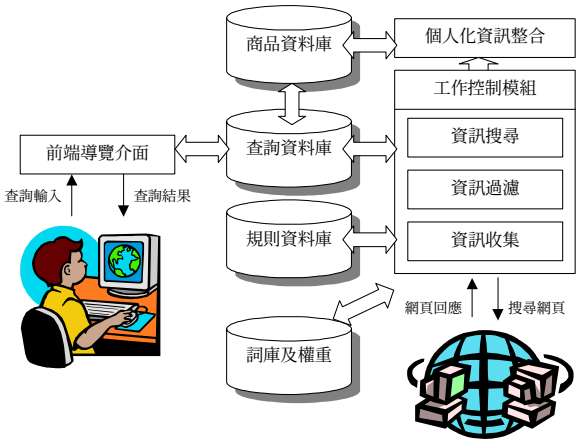
(四) 個人化資訊整合

在本研究中，提出搜尋、過濾、擷取的方法，是為了解決網路中資料過量問題，並能夠從網頁中擷取出有用資訊，只是對於資訊的定義上，每個人往往不同。

因此在資訊的整合上，也必須能夠呈現出每個使用者的差異性，亦即每個人可以擁有自己對於網頁內資訊的定義，可以擁有屬於自己的規則，並且在資訊呈現上也必須能夠根據每個人的屬性設定不同，而呈現出不同的結果。這樣的差異不只是人與人之間，連網站之間所提供的屬性項目並不相同，所以資料上，也會有所不同。這也是本研究中要達到的個人化資訊整合。

三、系統實作

為了建置研究的概念，在系統中結合了智慧搜尋代理人技術、資訊過濾及資訊擷取等研究方法並且提供了個人化資訊整合服務，如圖表 3 所列示為本系統的架構圖。



圖表 3：系統架構圖

(一) 系統開發環境

在開發上，是採用研究中主要方法去建立一套完整功能的系統，而且在未來系統運作時，能夠具備跨平台的執行及分散式處理能力。因此開發工具上主要選擇 JAVA 語言為建置核心，並以元件模組的開發方式。在後端系統部份，是以 JAVA 的多執行緒(Multi-Thread)元件所組成的應用程式系統；前端則是以網頁為主的呈現方式，在設計上是使用 JSP(JavaServer Pages)，並將每個模組寫成 JavaBean 元件，共同組織而成的線上程式系統。

(二) 使用者查詢介面

操作介面是採取網頁瀏覽方式，在系統中為了能夠區別出每個使用者的差異及個人化資料管理目的，都是以 E-mail 來代表每個使用者的身份。最主要的服務是希望提供網路上特定商品資訊的擷取蒐集工作。在查詢內容部份，區分為商品類別及特定商品名稱查詢，可依照使用者需要設定適合關鍵字到查詢項目之中。每一項商品查詢也可以設定額外條件限制，例如：價格範圍是介於多少或是擷取的過程是否使用規則以及限制查詢總筆數。這些設定會影響查詢得到的結果，所以可以依照每個人偏好來調整，見圖表 4。

圖表 4：新增使用者查詢

當查詢新增完成後，後端的應用程式系統會根據查詢的內容及設定進行查詢處理。系統能夠根據使用者所定義的擷取內容，而呈現不同的資料屬性，亦表示不同的使用者對於擷取內容可以不同，所以便能達到個人化的資訊整合功能，如圖表 5。

圖表 5：商品查詢結果

(三) 線上規則學習

規則學習的目的，是希望能夠將已知網頁的擷取方法記錄下來，當下一次須要擷取相關網頁時，就能使用這些規則。這不但能夠提升系統擷取的精確度也可以增加擷取出來的資料量。因此在系統中必須由使用者輸入學習網頁，並且考慮到每個人對於資訊的需要差異，決定出什麼才是擷取資料，這樣的動作就[illegible]資料標記。

線上規則學習，就是要提供每個使用者可以自行定義擷取的個人化內容，並且透過系統中的規則學習元件，取出擷取規則，經由測試擷取即能夠知道規則的完整與否，如圖表 6。

圖表 6：線上規則學習

(四) 工作控制模組

在後端系統中，開發上是由許多模組元件所組成，然而每個獨自模組負責其中處理工作。因此在執行中如何避免每個模組間的衝突及系統資源的有效分配，所以必須在所有模組之上建置工作控制模組。當系統啟動之後，系統的控制權會交由該模組運作，該模組負責檢查所有系統資源是否完備，如資料庫連接，系統設定等，之後便啟動系統中的其他子模組，並且給予每個模組優先權(Priority)設定，一直到子模組執行完畢後，再將模組暫停且釋放出系統資源。

(五) 資訊搜尋代理人(Query Agent)

資訊搜尋代理人主要將使用者的查詢轉換成搜尋引擎的查詢關鍵字，並從搜尋引擎中找出可能的網址。在系統之中，是採用 Google 為搜尋引擎，

這裏考慮到 Google 有不錯的查詢結果以及支援多語系搜尋方式,所以在系統資料來源選擇上或未來擴充都會有很好的彈性。

(六) 資訊過濾代理人(Filter Agent)

從 Google 搜尋出來的網址,雖然在內容中包含了查詢關鍵字,但是並不能夠確定所有的網址皆可能存在使用者需要的商品購物資訊,因此系統中必須能夠過濾出真正的有用的網址。

主要工作必須判斷網址內容中是否擁有購物或報價資訊,為了能夠以數值來衡量,所以必須將文件量化。本系統是採取使用者設定關鍵詞作為文件特徵,並以類神經網路學習訓練所得到的權重作為比對之特徵向量。文件衡量方法,系統中採用了 Dice Coefficient、Consine Coefficient、Jaccard Coefficient 及 SimNet 中的 Matching Degree 之全部平均值,如果大於設定的門檻值,則網頁符合擷取條件,反之,則過濾掉。

(七) 資訊收集代理人(Gather Agent)

系統中重要的功能,就是能夠將網頁中的資訊擷取出來,同時希望達到最多且最完整的資訊收集工作。在該模組中的擷取工作主要區分為兩種方式,一種是以規則擷取方法,即透過規則學習的步驟將使用者已知的網站預先學習出規則,等到未來使用者查詢到相關網址時,就能利用該規則正確的擷取出資料。

在學習的功能上,是由使用者標記出網頁中需要擷取的部份,而且在屬性設定上並不限制資料屬性,基本上只要有商品名稱和價格,其他部份皆可由使用者自行定義,當在擷取時,也會根據每個人學習出來的擷取出資料。每個人的學習規則可以透過網頁管理,並直接在線上測試或更新。

另一種方法,結合研究中所提出的資料分析擷取、文件結構擷取,主要針對未知的網頁內容中,試圖找尋出可能的資料位置。首先在文件結構部份是以網頁中的表格標籤為擷取結構,只要網頁中包含有表格標籤則,擷取出表格內的每筆記錄。然而如何判斷每筆資料中的每個屬性呢?因此必須根據資料內容加以分析判斷,這裏所判斷的資料屬性包括:商品名稱、價格、規格。其中的價格是較容易判斷的屬性,格式上是由數字加上文字組合而成,所以比對上是以特定格式來判別,而名稱及規格的判斷則是以長度及價格的位置作為分類方法,當位於價格之後的文字則歸入規格,當位於價格之前,可能是名稱或規格屬性,故再以長度小於 8 的文字才歸入商品名稱。

(八) 類神經網路文件特徵權重學習

由於網站內的網頁格式並不相同,故學習的方式,是由使用者輸入多個網址,而該網址必須區分為正確及不正確。正確網址代表學習出來的權重必須相似於這些網頁,而不正確網址則必須能學習出與其有所差異的權重。

每次學習訓練時,就會比對網址的特徵向量與特徵權重間的距離是否符合門檻值並調整權重,如果是正確網址則調整為更接近,反之不正確網址則調整為更遠。這樣的步驟一直重覆,直到達到收斂為止。其中的過程都可以透過網頁來查看,如圖表 7。

圖表 7:權重學習狀態

四、系統測試

在系統建構完成後，必須有一套針對系統加以評估驗證的基準。因此這裏定義了幾個檢測變數，常用的正確率估算有兩種，一是正確率(Precision)、一是回收率(Recall)[11]。高正確率代表擷取出的資訊大部份是正確的，高的回收率則代表大部分的資訊已被擷取出來。

1. 正確率的定義：使用者每一次查詢之後，系統提供檢索結果中正確的資訊量占檢索結果出來的資訊總數比率，用來評估系統擷取的精準度。
2. 回收率的定義：使用者每一次查詢之後，系統回傳擷取結果裏正確的資訊量佔符合查詢要求的資訊總數比率，用來評估系統擷取的廣泛程度。

(一) 測試方法

本實驗主要為了評估系統中的兩個部份，第一，系統查詢出來的網站內容正確度及數量；第二，系統的主要目的就是能夠將網路上的特定資訊擷取出來，評估上也須針對資料結果加以測試。雖然在系統擁有規則與無規則兩種擷取方法，然而其中的規則擷取必須先將已知網址利用線上學習規則後才能使用，因此只要標記的步驟正確，幾乎能夠百分之百取出所有屬性的資料，所以無法驗證出系統對於未知網頁內容的分析能力。測試中皆不使用規則，僅以網頁結構及資料內容處理作為系統對於半結構文件下的擷取參考，最後根據不同的過濾門檻值所擷取的資料筆數及其中各個屬性正確度作為評估標準。在實驗中，共有四項不同的商品查詢，並且將網站搜尋的筆數上限都設為 500 筆。

(二) 測試結果

當在不同門檻值設定之中，可以發現當門檻值從沒有往上增加時，會使得過濾及擷取的正確度提升，這是合理的現象，因為系統是採用文件特徵比對，因此只要符合部份特徵就可能具有商品資訊。只是這裏必須注意一點，當要求高正確率時，往往可能忽略了潛在可能的資料，故使得回收率大幅降低。然而如果設定過高的門檻值，可能會造成篩選掉太多的網頁，如果恰好篩選掉的大多是正確的網頁，則會反而使正確度降低，見表格 7。

表格 7：系統整體測試結果

門檻值(大於) / 項目	無	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7
網站 Precision	54%	72%	71%	74%	77%	66%	76%	79%	100%
網站 Recall	86%	64%	57%	50%	39%	19%	18%	7%	4%
資料 Precision	68%	86%	86%	84%	85%	77%	79%	76%	100%

當設定門檻值時，得到的測試結果如下：

系統平均網站過濾 Precision≒77%

系統平均資料擷取 Precision≒84%

若從總體 Precision 及 Recall 比率來看，兩者彼此結果是互斥的。當存在較好的 Precision 時，相對的 Recall 也會較差，而為了得到較佳的 Recall 也會使 Precision 變差。

針對單筆資料來看，每個屬性的擷取能力比較，見表格 8。很明顯看到價格屬性有很高的正確率，因為價格是由數字所組成，而且有一定格式，所以較容易找到位置。同樣的商品名稱也有相同的正確度，這是由於商品名稱與價格存在相對位置，如果價格可以正確找到，那商品名稱也能被正確擷取出來。最後是規格屬性，由於規格並不是絕對存在於網頁之中，而且位置又不一定，所以往往較難正確擷取。

表格 8：每個屬性的擷取正確率

	商品名稱	價格	規格
擷取的正確率	92%	92%	66%

為了比較不同的相似度衡量方法對於系統的正確率影響，分別針對 Dice Coefficient、Consine Coefficient、Jaccard Coefficient 及 SimNet 中的 Matching Degree 與系統所採用的全部平均值測試每一個的差異。由結果得知，Dice Coefficient、Consine Coefficient、Jaccard Coefficient 計算出來的相似度數值差異較大，這是因為它們都是基於 Inner Product（內積法）為主的衡量公式，而 Matching Degree 得到的相似度較為集中，同樣的也擁有較佳的正確率，見表格 9。

表格 9：不同相似度衡量公式之比較

相似度衡量 / 項目	Dice	Consine	Jaccard	Matching Degree	全部平均
網站 Precision	60%	74%	75%	83%	75%
資料 Precision	91%	87%	74%	84%	85%

五、結論與未來工作

針對半結構文件自動化擷取功能上，採用了規則與非規則的擷取方式，在非規則中會根據資料的格式與網頁的結構尋找出可能的資料位置。當為了能夠提高擷取精確度及資料量，可以讓使用者將已知的網頁預先學習文件規則，當未來需要擷取資料時，就能夠正確取出文件資料，這也就是規則擷取方式。因此在未來也可將本系統用在不同領域之中，如網路新聞、財經資訊或人員名單等資訊收集工作上，同樣輕輕鬆鬆能夠達到資訊收集自動化。

在系統中為了能夠過濾出具有商品資訊的網頁，因此提供讓使用者將已知的網頁位址輸入，並透過線上類神經網路學習訓練，計算出該種網頁的特徵權重，即可篩選出許多傳統搜尋引擎查詢出但並不相似的網頁，所以可以很容易的處理不同種類資料的查詢應用。

對於資訊整合上，能夠呈現出每個使用者的差異性，每個人可以擁有對於資訊的定義及屬於自己的學習規則，而且資訊呈現上也根據每個人的屬性設定不同，會不同的結果。這樣的差異讓使用者在資訊蒐集及資訊運用上能夠有更佳的彈性。這也是研究中要達到的個人化資訊整合目的。

對於未來的研究上，可以有下列幾個方向：

1. 不同領域的資訊檢索及擷取
2. 非文字資料處理之研究
3. 完整個人化資訊整合服務

六、參考文獻

1. Chun-Nan Hsu, “Initial results on wrapping semistructured web pages with finite-state transducers and contextual rules”, In Proceedings of AAAI-98 Workshop on AI and Information Integration, Technical Report WS-98-01, Menlo Park, CA, 1998.
2. Chun-Nan Hsu and Ming-Tzung Dung, “Generating finite-state transducers for semistructured data extraction from the web”, *Journal of Information Systems*, Special Issue on Semistructured Data, Volume 23, Number 8, 1998.
3. Cowie, J. and.Lehnert W., “Information Extraction”, Communication of ACM, Vol 39, No1, pp.80-91, January 1996.
4. Dietrich H. Schuschel and Chun-Nan Hsu, “A weight analysis-based wrapper approach to neural nets feature selection”, In Proceedings of the 10th IEEE International Conference on Tools with AI, Taipei, Taiwan, 1998.
5. Faloutsos, C. and Oard D., “A survey of Information Retrieval and Filtering Methods”, University of Maryland College Park,

CS-TR-3514, 1995. Also available at http://www.enee.umd.edu/medlab/filter/papers/survey.ps

6. Kushmerik, N, “Gleaning the Web”, IEEE Intelligent Systems, Volume: 14 2 , Page(s): 20 –22, 1999.
7. Lee H. C, Dagli C. H., Ercal F., and Ozbayoglu A. M., “SimNet: A Parallel Neuro-Fuzzy Paradigm for Data Clustering”, OAI Neural Networks Symposium and Workshop(OAINN ’95), Athens, Ohio, USA, 1995.
8. Nie J., Briscbois M., and X. Ren, “Template-Based Information Mining from HTML.” In: *Proceedings of AAAI-97, Providence*, USA, 1996.
9. S. Aggarwal, F. Hung, Weiyi Meng, “WIRE-a WWW-based information retrieval and extraction system”, IEEE Database and Expert Systems Applications, Page(s): 887 -892, 1998.
10. San, M., “Intelligent agents on the Internet and Web”, TENCON ’98. IEEE Region 10 International Conference on Global Connectivity in Engery, Computer, Communication and Control, Vol.1, pp.97-102.
11. Robertson E. S., “The Parametric Description of Retrieval Tests”, Journal of Documentation 25:1, pp.1-27.
12. Wolfgang May, “An integrated Architecture for Exploring, Wrapping, Mediating and Restructuring Information from the Web”, Database Conference, 2000. ADC 2000. Proceedings. 11th Australasian, Page(s): 82 –89, 2000.
13. Yanlei Diao, Hongjun Lu, Songting Chen, Zengping Tain, “Toward Learning Based Web Query Processing”, VLDB 2000, 317-328, 2000.