行政院國家科學委員會專題研究計畫　成果報告

在 STPN 網頁架構模型建立與應用網頁衡量基準

計畫類別： 個別型計畫
計畫編號： NSC94-2213-E-032-024-
執行期間： 94 年 08 月 01 日至 95 年 07 月 31 日
執行單位： 淡江大學資訊工程學系

計畫主持人： 陳伯榮

報告類型： 精簡報告

處理方式： 本計畫可公開查詢

中　華　民　國 95 年 10 月 30 日

行政院國家科學委員會專題研究計畫成果報告
在 STPN 網頁架構模型建立與應用網頁衡量基準
Construct and Apply the Web Metrics for STPN Web Structure Model

計畫編號：NSC 94-2213-E-032-024
執行期限：94 年 8 月 1 日至 95 年 7 月 31 日
主持人：陳伯榮　淡江大學資訊工程學系

1.中文摘要

我們延續九十三年國科會專案計畫“應用隨機過程時間派翠網路來強化網頁使用模式探勘”（NSC 93-2213-E-032-016）的心得, 提出並應用網頁結構特性(Web graph properties）中的向心性（Centrality）、整體衡量基準（Global Metrics）及局部衡量基準（Local Metrics）及網頁相似性（Web page similarity）中的使用習性相似性（Usage-Based Similarity）來增進網頁資訊的存取效能。
完成的主要重點包括：
一、加強網頁結構特性的分析。
二、藉由分析網頁使用記錄自動建立索引網頁。。

關鍵詞：隨機過程時間派翠網路、網頁結構特性、衡量網站的基準。

Abstract

Extending the previous project “Enhance web usage mining by using Stochastic Timed Petri Nets（NSC 93-2213-E-032-016）”, we identify three web graph properties：Centrality, global metrics and local metrics, and usage-based similarity in web page similarity can be applied for improving web information access.

In this report, we introduce:

[1] web structure properties and apply them to the STPN web model.

[2] the index page synthesis problem to adjust the web site structure adaptively.

Keywords: Stochastic Timed Petri Nets, Web graph properties, Web page similarity.

2.緣由與目的

我們在[1]中提出利用 STPN 來建構網站的網頁架構模型，以 STPN 的位置來代表網站中的網頁，以 STPN 的轉移來代表網頁中超連結所產生的轉移觸發動作。經由對網站的分析可以將網站的網頁架構轉換成 STPN 中的位置與轉移觸發動作的關聯矩陣 $[A_{ij}]_{nxm}$，並得到代表網頁檔案與 STPN 位置對應的「位置名稱與網頁名稱對應表」及代表網頁超連結與 STPN 轉移觸發對應的「轉移動作名稱與網頁標籤名稱對應表」。

並在[2]中進一步應用 STPN 的行為特性來協助資料前置處理中的使用者區段辨識及路徑填補。

在本計劃中我們應用 Dhyani 在[3]中談到的網頁結構特性（Web graph properties）及網頁相似性（Web page similarity）這兩種衡量網站的基準（Web metrics），來改善網頁資訊的存取及使用。

我們研究的主要目的包含：

- 加強網頁結構特性的分析。
- 藉由分析網頁使用記錄建立索引網頁。

在加強網頁結構特性的分析的部份，由於網站結構是以網頁內容範圍(Web Pageview)作為節點(Node)，超連結(Hyerlink)作為邊(Edge)，所組合成的圖形，如果網頁結構圖中有 N 個節點，則以距離矩陣（Distance Matrix）$[D_{ij}]_{nxn}$ 表示，其中 D_{ij} 的值代表網頁 i 瀏覽至網頁 j 的

連結數，若從網頁 i 無法透過連結瀏覽網頁 j，則 D_{ij} 的值以∞來表示。而在值的處理上，∞並不方便用來運算，因此，定義了改變後距離矩陣（ Converted Distance Matrix）$[C_{ij}]_{nxn}$，其中 Cij 的值代表網頁 i 瀏覽至網頁 j 的連結數，若從網頁 i 無法透過連結瀏覽網頁 j，則 C_{ij} 的值經常以節點的個數 N 來表示。

在[4] Botafogo 針對外部轉換距離（Converted out Distance，簡稱 COD）、內部轉換距離（Converted in Distance，簡稱 CID）、轉換距離（Converted Distance，簡稱 CD）做了如下的定義：

外部轉換距離(COD)是指一點到其它各點的總距離和 $COD_i = \sum_j C_{ij}$ 。

內部轉換距離(CID)是指其它各點至某一點的總距離和 $CID_i = \sum_j C_{ji}$

轉換距離(CD)是指 $[C_{ij}]_{nxn}$ 中所有元素的總和 $CD = \sum_i \sum_j C_{ij}$ 。

另外，Botafogo 在[4]中也介紹了三種網頁結構衡量基準：分別是向心度（Centrality），整體衡量基準（Global Metrics）及局部衡量基準（Local Metrics）。

向心度：它代表一個節點與圖形中其它節點的連結性，向心度可分為相對對外向心度（Relative Out Centrality，簡稱為 ROC）及相對對內向心度（Relative In Centrality，簡稱為 RIC）。

我們可以找出有較高 ROC 值（或較小 COD 值）的節點當作圖形架構中的根節點（Root），再依照廣度優先的搜尋順序來找出此圖的擴展樹（spanning tree），來當作我們的主要網頁架構，藉此提供一個方便的使用者瀏覽環境。

在整體衡量基準方面：最明顯的整體衡量基準是節點與連結的數量；雖然許多系統可以提供節點的數量，但是卻無法提供連結的相關數字。即使可以得到節點與連結的數量，我們仍然只能夠概略的知道網站上讀取資料的困難度。也因此，我們需要藉著緊密度（Compactness）與階層順序（Stratum）這兩個整體衡量基準來探討網站上讀取資料的複雜度。

局部衡量基準：局部衡量基準可分為深度（Depth）及不平衡度（imbalance）。一個節點的深度是指它到根節點的距離，經由深度衡量基準，網頁設計者可以找出並驗證無法到達或深度極深節點的存在是否恰當。

在藉由分析網頁使用記錄建立索引網頁的部份，Perkowitz and Etzioni[5][6]中提出這個問題，並介紹一個他們稱為 Page-Gather cluster mining algorithm 的演算法，這個演算法透過分析網站使用者使用網頁記錄，利用同時引用機率（co-occurrence frequencies）找出使用者可能會有興趣的相關網頁，再利用叢集（clusting）的方法，將在原來的網頁結構中並未連結的相關網頁加以分類。最後建立索引網頁，分別連結各個網頁叢集以便使用者瀏覽。

3.結果與討論

我們將在 3.1 中討論如何應用我們在[1]中以 STPN 所建構的網路架構模型進一步利用向心度來進行網頁結構衡量基準，以找出較佳的主要網頁結構，可以用來提供使用者最方便的瀏覽環境。在 3.2 中我們也討論整合 STPN 網頁架構模型及整體衡量基準來評估網站中網頁間的緊密度，以用來做為網站管理者是否進行網頁架構調整的參考依據。在 3.3 中則討論如何利用局部衡量基準來評估網站結構是否節點分佈過深或是分佈不平衡。最後並在 3.4 中說明藉由分析網頁使用記錄建立索引網頁的相關結果與討論。

3.1 向心度

ROC 及 RIC 的定義如下：

ROC 是 CD 以 COD 正規化後的值。

$$ROC_i = \frac{CD}{CODi} = \frac{\sum_i \sum_j C_{ij}}{\sum_j C_{ij}}$$

RIC 是 CD 以 CID 正規化後的值。

$$RIC_i = \frac{CD}{CIDi} = \frac{\sum_i \sum_j C_{ij}}{\sum_j C_{ji}}$$

為求得 ROC 及 RIC, 我們先利用關聯矩陣$[A_{ij}]_{nxm}$求得改變後距離矩陣$[C_{ij}]_{nxn}$, 其演算法複雜度為 O(n3)：

```
//先求得代表網頁結構的[Cij]nxn 初值
initially,  C[i,j] = n ,  if i != j
                    0 ,  otherwise
      for i=0 to n-1 do
        for j=0 to m-1 do
            if A[i,j] = -1
               for k=0 to n-1 do
                 if A[i,j] = 1
                 then C[i,k] = 1
               end;
        end;
      end;
```

```
//我們利用大家熟知的各頂點對之間最短路徑Floyd-Warshall演算法求出改變後距離矩陣[Cij]nxn
    for k=0 to N-1 do
       for i=0 to N-1 do
          for j=0 to N-1 do
             if C[i,k]+ C[k,j]< C[i,j]
             then   C[i,j]=   C[i,k]+
                    C[k,j]
          end;
       end;
    end;
```

以表一關聯矩陣$[A_{ij}]_{7x8}$為例可求得代表網頁結構的$[C_{ij}]_{7x7}$初值如表二，再求得改變後距離矩陣$[C_{ij}]_{7x7}$如表三：

$$[A_{ij}]_{7\times8}=\begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

表一：關聯矩陣$[A_{ij}]_{7x8}$

Nodes	a	b	c	d	e	f	g
a	0	7	7	7	7	7	7
b	7	0	1	7	7	1	7
c	7	7	0	7	7	7	7
d	1	7	7	0	7	7	7
e	1	7	1	1	0	7	7
f	7	7	7	7	7	0	1
g	1	7	7	7	7	7	0

表二：代表網頁結構的$[C_{ij}]_{7x7}$初值

Nodes	a	b	c	d	e	f	g	COD	ROC
a	0	7	7	7	7	7	7	42	5.5
b	3	0	1	7	7	1	2	21	11.0
c	7	7	0	7	7	7	7	42	5.5
d	1	7	7	0	7	7	7	36	6.4
e	1	7	1	1	0	7	7	24	9.7
f	2	7	7	7	7	0	1	31	7.5
g	1	7	7	7	7	7	0	36	6.4
CID	15	42	30	36	42	36	31	232	
RIC	15.5	5.5	7.7	6.4	5.5	6.4	7.5		

表三：改變後距離矩陣$[C_{ij}]_{7x7}$及 COD、CID、ROC、RIC

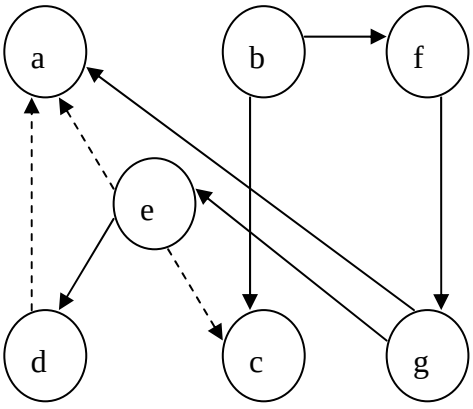
我們可以找出有較高 ROC 值（或較小 COD 值）的節點當作網頁結構中的根節點（Root），在表四中以節點 b 為根節點。由於 C[1,3]=7 和 C[1,4]=7 可得知無法由網頁 b 瀏覽到網頁 e 和 d, 假設網頁管理者決定從網頁 g 中加入一個連結到網頁 e 來調整網頁結構, 可藉由[1]找到調整後關聯矩陣$[A_{ij}]_{7x9}$如表四，進一步求得調整後的改變後距離矩陣$[C_{ij}]_{7x7}$如表五。我們找出有較高 ROC 值（或較小 COD 值）的節點 b 當作網頁結構中的根節點，在表三的第二列中發現可以由網頁 b 瀏覽到所有網頁，再依照廣度優先的擴展樹(spanning tree)方法，來求得交叉連結的樹狀結構(crosslinked tree structure)如圖一，當作我們的主要網頁結構，藉此提供一個方便的使用者瀏覽環境。圖中以實線代表階層連結(hierarchical link)，以虛線代表交叉連結(cross-reference link)，可進一步提供網頁設計者評估 e-->c、d-->a、e-->a 這些交叉連結是否適當。

$$\left[A_{ij}\right]_{7\times9}=\begin{bmatrix}0&0&1&1&0&0&0&1&0\\-1&-1&0&0&0&0&0&0&0\\1&0&0&0&1&0&0&0&0\\0&0&-1&0&0&1&0&0&0\\0&0&0&-1&-1&-1&0&0&1\\0&1&0&0&0&0&-1&0&0\\0&0&0&0&0&0&1&-1&-1\end{bmatrix}$$

表四：加入 g->e 後的關聯矩陣$[A_{ij}]_{7x9}$

Nodes	a	b	c	d	e	f	g	COD	ROC
a	0	7	7	7	7	7	7	42	4.7
b	3	0	1	4	3	1	2	14	14.0
c	7	7	0	7	7	7	7	42	4.7
d	1	7	7	0	7	7	7	36	5.4
e	1	7	1	1	0	7	7	24	8.2
f	2	7	3	3	2	0	1	18	10.9
g	1	7	2	2	1	7	0	20	9.8
CID	15	42	21	24	27	36	31	196	
RIC	13.1	4.7	9.3	8.2	7.3	5.4	6.3		

表五：調整後的改變後距離矩陣$[C_{ij}]_{7x7}$及 COD、CID、ROC、RIC



圖一：調整後的交叉連結樹狀結構

3.2 整體衡量基準

在整體衡量基準方面：最明顯的整體衡量基準是節點與連結的數量；雖然許多系統可以提供節點的數量，但是卻無法提供連結的相關數字。即使可以得到節點與連結的數量，我們仍然只能夠概略的知道網站上讀取資料的困難度。也因此，我們需要藉著緊密度（Compactness）與階層順序（Stratum）這兩個整體衡量基準來探討網站上讀取資料的複雜度。

緊密度的值定義為 0~1 之間，值愈大表示點與點之間連結性愈高，當一個圖形其所有點都彼此相連時，其緊密度為 1，反之，所有點都不相連時，其緊密度為 0。就網頁的設計規劃而言，太高的緊密度，表示頁面間有太多交叉連結，使用者不見得可以容易的找到所需要的頁面，將增加整個網站的瀏覽時的困難度。緊密度的計算公式如下：

$$C_p=\frac{Max-CD}{Max-Min}=\frac{K\times(N^2-N)-CD}{(N^2-N)\times(K-1)}$$

，其中 Max，Min 分別為，轉換距離 CD 的最大值與最小值，Max=$\left(N^2-N\right)\times K$ ，Min= $\left(N^2-N\right)$

K 經常被設為節點個數 N。

若點與點不相連，則所有的 Cij（i≠j）值被設為 K，此時 CD=$\left(N^2-N\right)\times K$ ，因此 Cp 為 0。若所有點彼此相連則所有的 Cij（i≠j）值為 1，此時 CD=$\left(N^2-N\right)$ ，因此 Cp 為 1。

階層順序性的值定義為 0-1 之間，被用來計量網頁讀取的順序性，其值愈高，表示頁面間的線性連結愈多，必須依序讀取的頁面愈多；值愈低，表示頁面間的環狀連結愈多，從哪一個頁面開始讀取並沒有顯著的差別。Prestige 被定義為由「外部距離」(status) 與「內部距離」(contrastatus) 間的差；LAP(Linear absolute prestige)被定義為依線性連結的 N 個網頁的絕對 prestige 值。階層順序性的計算公式如下：

St=$\sum$ |prestige|/LAP

prestige = statusi - contrastatus i

$$status_i=\sum_j D_{ij}$$

$$\text{contrastatus}_i=\sum_j D_{ji}$$

Dij = - ，if Cij = N ；

Cij，otherwise

LAP= $N^3/4$，if N is even；

$(N^3-N)/4$，otherwise

$$St=\frac{\sum_i\left(\left|status_i-\text{contrastatus}_i\right|\right)}{LAP}$$

$$=\frac{\sum_i\left(\left|\sum_j D_{ij}-\sum_j D_{ji}\right|\right)}{LAP}$$

其距離矩陣與階層順序相關計量表格如表六：

Nodes	a	b	c	d	e	f	g	status	Absolute prestige
a	0	-	-	-	-	-	-	0	8
b	3	0	1	4	3	1	2	14	14
c	-	-	0	-	-	-	-	0	7
d	1	-	-	0	-	-	-	1	9
e	1	-	1	1	0	-	-	3	3
f	2	-	3	3	2	0	1	11	10
g	1	-	2	2	1	-	0	6	3
Contrastatus	8	0	7	10	6	1	3	35	54

表六：調整後的距離矩陣$[D_{ij}]_{7x7}$及 status、contrastatus、absolute prestige

在此例中，Cp 值為 98/252＝0.39；LAP 值為 ＝84，所以 St 值為 54/84＝0.64。這兩個數值可提供網頁管理者作為調整網頁結構的參考依據。譬如說，St = 0.64 是否隱含著依照網頁結構所呈現的擷取資料的順序對使用者而言，並不容易找到他要的資料，也就是說是否該考慮加入適當的交叉連結。

3.3 局部衡量基準

局部衡量基準可分為深度（Depth）及不平衡度（imbalance）。一個節點的深度是指它到根節點的距離，經由深度衡量基準，網頁設計者可以找出並驗證無法到達或深度極深節點的存在是否恰當。：

不平衡度衡量基準是用來讓網頁設計者了解整個網站結構是否平衡，有深度不平衡（depth imbalance）及子節點個數不平衡（child imbalance），網頁管理者可針對較不平衡的子樹加以分析並視需要做進一步的調整，兩種不平衡度衡量基準定義如下：

一個節點 a 的絕對深度不平衡(absolute depth imbalance)是深度向量（depth vector）D(a)中所有元素的標準差。

深度向量的定義如下：

若 a1,a2,…,an 是 a 的子節點，則 a 的深度向量，深度不平衡（D（a））為：

$$D(a)=\begin{cases}[1+Max(D(a_1)),1+Max(D(a_2)),.....,1+Max(D(a_n))], & if\ a\ has\ no\ child\ (n=0)\\ [0]\end{cases}$$

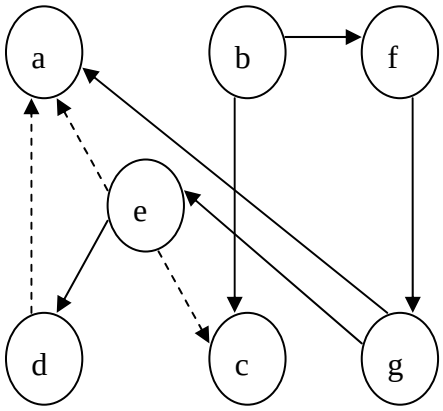
其中 Max(D(a))是指 D(a)中最大的值

一個節點 a 的絕對子節點個數不平衡(absolute child imbalance)是子節點個數向量（child vector）C(a)中所有元素的標準差。

子節點向量的定義如下：

$$C(a)=\begin{cases}\{1+\sum C(a_1),1+\sum C(a_2),.....,1+\sum C(a_n)\}, & if\ a\ has\ no\ child\ (n=0)\\ \{0\}\end{cases}$$

其中的$\sum C(a_t)$ 是代表 C(a) 所有元素的總和。每個節點的深度向量及子節點向量計算後如圖二所示：



圖二：調整後的交叉連結樹狀結構及深度向量、子節點向量

這兩個向量數值可提供網頁管理者作為調整網頁結構的參考依據。譬如說，b 節點的子節點向量{1,5}差異過大，可提醒網頁管理者網頁結構中相對資訊量的不平衡。

3.4 藉由分析網頁使用記錄建立索引網頁

Perkowitz and Etzioni[5][6]所提出的 Page-Gather cluster mining algorithm 的演算法的步驟如下：

1. 將所有的使用者記錄藉由資料前置處理的方法，整理成以使用者區段為單位的使用記錄。

2. 計算在使用者區段中任兩個網頁被同時引用機率（co-occurrence frequencies），即對任兩個網頁 P1、P2，計算 P(P1|P2)—使用者瀏覽過 P2 後再瀏覽 P1 的機率，以及 P(P2|P1)— 使用者瀏覽過 P1 後再瀏覽 P2 的機率，並取這二個條件機率之間的最小值，接著針對 co-occurrence frequencies 大於一個門檻值(threshold)但沒有連結在一起的網頁建立一個相似度矩陣（similarity Matrix）。此相似度矩陣也代表一個圖。

3. 應用圖學的方法從相似度矩陣來找出叢集（connected components），相似度矩陣的每一個索引即代表一個節點，每一個非 0 的元素即代表一個邊。

4. 將找到的叢集依叢集中的平均 co-occurrence frequencies 大小加排序。

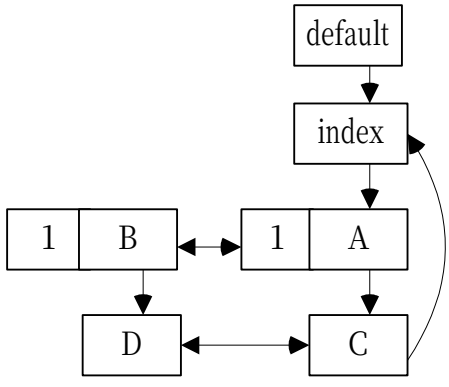
5. 依照上述排序，針對每一個叢集，建立一個索引網頁，這個索引網頁包含可以指向叢集中所有網頁的超連結。

若以使用者記錄如下的網站紀錄為例：

```
127.0.0.1 - - [08/Mar/2005:23:29:15 +0800] "GET /default.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:29:18 +0800] "GET /index.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:29:18 +0800] "GET /1.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:29:18 +0800] "GET /A.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:30:02 +0800] "GET /C.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:33:23 +0800] "GET /index.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:33:23 +0800] "GET /1.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:33:23 +0800] "GET /A.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:34:07 +0800] "GET /B.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:34:17 +0800] "GET /A.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:43:03 +0800] "GET /C.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:43:04 +0800] "GET /index.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:43:04 +0800] "GET /1.html HTTP/1.1" 304 –
127.0.0.1 - - [08/Mar/2005:23:43:xx +0800] "GET /B.html HTTP/1.1" 304 -
127.0.0.1 - - [08/Mar/2005:23:43:06 +0800] "GET /D.html HTTP/1.1" 304 –
```

表七：網頁使用者記錄

當網站架構如以下圖：



圖三：網站架構圖

透過執行步驟二：計算任二個網頁之間被同時引用的機率，可得以下矩陣

	Default	Index	1	A	B	C	D
Default	0	0	1/15	1/15	1/15	1/15	1/15
Index	0	0	0	0	2/15	0	1/15
1	1/15	0	0	0	0	0	0
A	1/15	0	0	0	0	0	1/15
B	1/15	2/15	0	0	0	2/15	0
C	1/15	0	0	0	2/15	0	0
D	1/15	1/15	0	1/15	0	0	0

表八：網頁之間被同時引用機率

根據網頁相連的定義，二個網頁被視為相連是指從其中一個 Page 存在一個 LINK 可連至另一個 Page 或有一個 Page 存在二個 LINK 連至此二個 Pages，因此矩陣中為 0 者皆視為相連，其餘非 0 項則計算其兩兩條件機率後取最小值得出結果。如：P(1|Default)=3/15，而 P(Default|1)=1/15，則取 1/15（在此 15 計算是以每個 Page 出現次數相加總數得出，再假設每個 Page 之間皆為互斥，因此條件機率計算公式為 P(A|B)=P(A)P(B)/P(B)），最後我們取 1/15 這個門檻值，即低於或等於 1/15 皆去除，以大於 1/15 項目值建立一個相似度矩陣。

執行步驟三：應用圖學方法可以找出此相似度矩陣的叢集為{(index)，(B)，(C)}。

步驟四：叢集要依機率大小加以排序，此例剛好三個 Page 皆為同值，可自由排序。

步驟五：針對每一個叢集(此例只有一個叢集)建立一個索引網頁，這個索引網頁包含可以指向叢集中所有網頁的超連結，此索引網頁將包含指向 (index)，(B)及(C)等三個網頁的超連結。

由於此例只針為單一使用者區段作分析，而此使用者區段所分析出的資料只得出一個叢集可方便供使用者瀏覽。

四、計劃成果自評

在本計劃中我們應用 Dhyani 在[3]中談到的網頁結構特性（Web graph properties）及網頁相似性（Web page similarity）這兩種衡量網站的基準（Web metrics），來改善網頁資訊的存取及使用。

我們整合了我們在[1]中提出的 STPN 網頁結構模型，並應用了向心度（Centrality），整體衡量基準（Global Metrics)及局部衡量基準(Local Metrics)這三種網頁結構衡量基準針對網頁結構進行網頁結構分析。此外，我們也應用 Perkowitz and Etzioni 所提出的 Page-Gather cluster mining algorithm 來藉由分析網頁使用記錄建立索引網頁。

我們並將計畫的成果發表於2006數位生活科技研討會[7]。

五、參考文獻

[1] 陳伯榮、楊士央、何仁中，”應用隨機過程時間派翠網路來強化網頁使用者習性探勘”，二００四數位生活與網際網路科技研討會。Session7C-3， June 24-26， 2004， NSC 92-2213-E-032-024.

[2] 陳伯榮、楊士央、季振忠、陳清祥、孫初豪，”應用一般隨機過程派翠網路來協助網頁使用者習性探勘中的前置處理”， 二００五數位生活與網際網路科技研討會。Session9D-1， June 2-3， 2005， NSC 93-2213-E-032-016.

[3] Devanshu Dhyani, Wee Keong Ng, and Sourav S. Bhowmick, “A Survey of Web Metrics”, ACM Computing Surveys, Vol. 34, No. 4, pp. 469-503, December 2002

[4] Botafogo R., Rivlin E., and Shneiderman B., “Structural analysis of hypertexts: Identifying hierarchies and useful metrics.”, ACM Transaction on Information System, 10, Apr., pp.142-180, 1992.

[5] Peterkowitz M., and Etzioni O., “Adaptive web sites: Automatically synthesizing web pages.” In Proceedings of the 15th National Conference on Artificial Intelligence, 1998.

[6] Peterkowitz M., and Etzioni O., “Towards adaptive web sites: Conceptual framework and case study.” In Proceedings of the 8th World Wide Web Conference, 1999.

[7] 陳伯榮、楊士央、孫初豪、王秉弘，” 在 STPN 網頁架構模型建立與應用網頁衡量基準”，二００六數位生活科技研討會。Session3B-6， June 1-2， 2006， NSC 94-2213-E-032-024.