

整合社群關係的 OLAP 操作推薦機制

陳信固

國立政治大學資訊科學系

Garychen.twn@gmail.com

李蔡彥

國立政治大學資訊科學系

li@nccu.edu.tw

摘要

近幾年在金融風暴及全球競爭等影響下，企業紛紛導入商業智慧平台，提供管理階層可簡易且快速的分析各種可量化管理的關鍵指標。但在後續的推廣上，經常會因商業智慧系統提供的資訊過於豐富，造成使用者在學習階段無法有效的取得所需資訊，導致商業智慧無法發揮預期效果。本論文以使用者在商業智慧平台上的操作相似度進行分析，建立相對於實體部門的凝聚子群，且用中心性計算各節點的關聯加權，整合至所設計的推薦機制，用以提升商業智慧平台成功導入的機率。經模擬實驗的證實，在推薦機制中考慮此因素會較原始的推薦機制擁有更高的精確度。

關鍵詞：社群網路分析、推薦機制、社群偵測、商業智慧、網絡中心性

整合社群關係的 OLAP 操作推薦機制

壹、 導論

根據 CHAOS Summary 2009 報告，企業在導入資訊系統時，由於資訊系統複雜的特性，無法成功完成專案目標的機率高達 76%。在商業智慧專案失敗的原因上，主要分為系統功能建置不良與平台推廣不良兩種類型。目前在商業智慧領域，大部分的研究都在探討倉儲系統建置、多維度系統開發方法論與資料採礦的運用及演算法的研究，也有人研究商業智慧系統需求訪談的分析模型，這些研究都是為了提升商業智慧專案成功的機率。本研究希望能從商業智慧系統推廣的方向著手，探討如何藉由輔助決策系統的方式提升商業智慧在企業內推廣的成效。本研究主要的研究目的如下：

- 即時提供操作建議，經由參考其他相關使用者的操作紀錄，降低生手在商業智慧系統上學習的時間。
- 讓使用者在操作的過程中能檢視同群體的人員在分析資料時的面向，提昇群體間知識分享與使用者資料檢索的效率。

本文概略分成以下幾節：第二節裡，我們將介紹本文相關研究與改善處。在第三節中，我們將利用社群網絡分析，說明如何對使用者進行分群，並且說明如何定義使用者系統老手的程度。第四節為推薦機制說明，將對操作相似度判斷與推薦機制分別作說明；最後在第五節與第六節中，我們將說明本次研究的實驗結果以及未來研究的目標。

貳、 相關研究

一、 線上分析處理輔助研究

線上分析處理 (On-Line Analytical Processing, OLAP) 是提供使用者存取資料倉儲的前端應用。經由分析模型的建置，提供決策者依據不同的主題與面向來對資料進行分析，OLAP 系統主要包含 Cube、維度與量值。資料的操作上，提供使用者經由維度向下探勘 (Drill-Down) 與向上彙總 (Roll-Up) 的方式，有效率的切換資料的面向且取得一致性的決策資訊，下列是本次研究兩個重要的參考研究。

第一、在 Carsten Sapia(1999) 的研究中，作者利用內容式資訊過濾 (Content-based Filtering) (Balabanovic & Shoham, 1997) 方式作使用者行為的預測，將 OLAP 上的操作 (MDX 查詢語句) 正規化為集合的表示式，做為此查詢的 Query Prototype。作者將正規化後的集合，經由『維度的層級 (Level) 差異』與『維度的功能 (Result 或 Selection Dimension)』兩個面向，計算兩個查詢語句的相似度。最後再以相似度的判斷結果，預測該使用者下一步操作。不過在此研究中，存在因忽略認同度而產生『最相似的查詢等同是最適合推薦的項目』的錯誤，單純以資料本身的相似度作為判斷，有可能此查詢本身是錯的，但只因它最相似而誤認為最適項目。因此在本研究中會先以相似度高低找出候選的項目，然後進行認同度進行投票後，來找出最適合的推薦項目來改善此問題。

第二、在 Chen & Hsu (2008) 的研究中，作者進行生手推薦機制、老手推薦機與分群推薦機制的比較。作者認為 OLAP 系統上的使用者應該有生手與老手間之分，生手在此研究中將考慮在所有的使用者的操作紀錄，計算出使用次數最高的三個項目進行推薦。但隨著使用者在系統操作經驗的累積，使用者在系統上的推薦需求反而是以往不曾使用過的項目，以提升使用者對於系統功能的使用率。作者利用 K-Means 分群演算法依

據使用者行為的相似性進行分群，提升推薦分析資料的精確度。不過此研究主要是推薦各個 Cube 使用次數最高的前三名項目，不過實際上 OLAP 系統的操作是一個序列的操作，需要經由多個相關的項目組合才能取得一個有效的結果。不過在同一個 Cube 上使用次數最高的三個項目組合起來，絕大部分不會是我們問題解決所需的資訊，因此在本研究中，我們依據使用者當下的操作內容的相似度判斷，並且整合社群關係的推薦機制來找出最適合的推薦項目。

二、使用者評分

使用者評分 (User Rating) 是一種經常見的推薦方式，以使用者的角度進行推薦。藉由使用者上傳對於各項目的評分後，經由公式的計算後統計評分的結果。由 Google 創始人 Brin 和 Page (1998) 所提出的 Page Rank 便是其中最為著名的一個評分演算法，此方法將查詢結果依據實用性作為排名的一個主要因素，用以表現網頁的相關性與重要性。Page Rank 演算法中具有兩個特性：

- 頁面連結越多代表此網頁越重要：利用網頁間連結的特性，每一個頁面的連結都代表使用者對於此頁面投一認同票，因此連結越多代表此頁面重要性越高。
- 如果此網頁被重要的網頁連結時，則此網頁也是很重要的網頁。

此研究利用投票方式取代傳統以資料出現次數判斷項目的好壞，以往在判斷推薦項目的優先權時，都只利用『點選的次數』作分析，會造成只以『點』的方式判斷推薦項目的好壞。不過在 OLAP 系統上的操作與網頁的瀏覽一樣，都存在以 Session 為導向的特性 (Newman, 2006)。因此透過鏈結關係投票時，可以將最頻繁操作的網頁視為最適合推薦的標的。以往在線上分析處理上的研究上，認同度的計算通常都是一人一票，但在本論文中我們將會依據老手的程度，對於每個人的票作加權的效果。

三、社群網絡分析

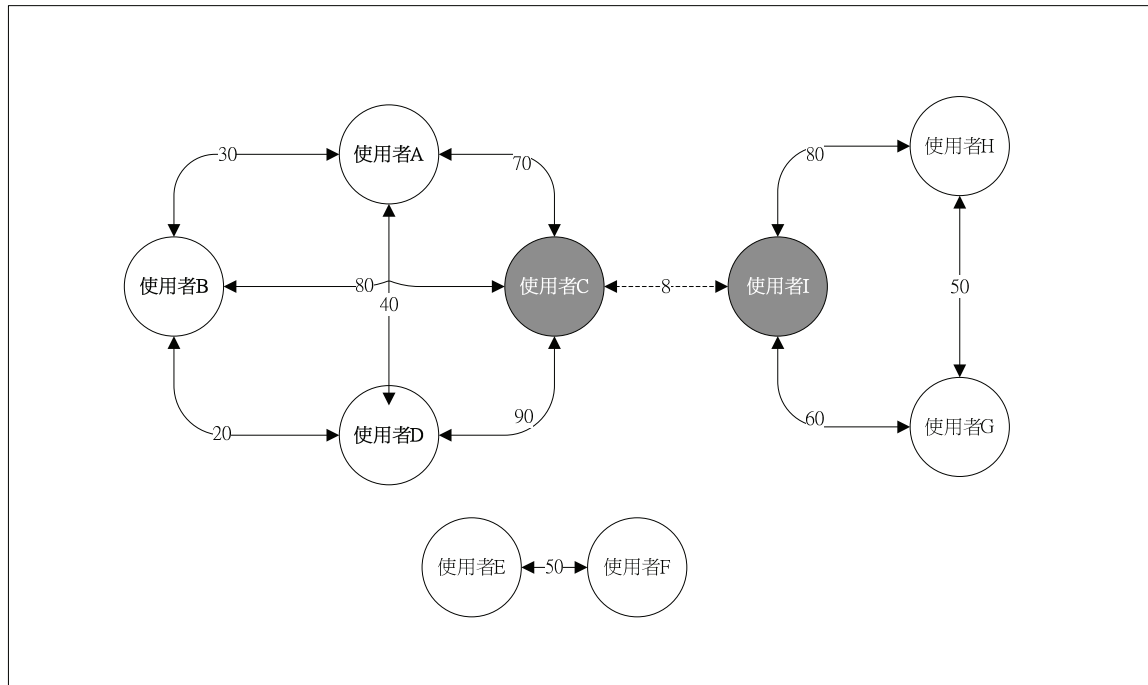
社會網絡分析 (Social Network Analysis, SNA) 的研究，主要是探討存在於社會網絡內節點間的互動關係 (Newman, 2006; Wasserman & Faust, 1994)，以及關聯間存在的『潛在結構』 (Latent Structure)。通常我們會用密度 (Density)，是用來描述網絡中節點彼此之間的互動狀況。在 W.D. Nooy (2005) 的凝聚子群分析研究中，密度越高的群體代表節點關係越密切，也表示著此子群內的節點因為某種的關聯而形成一個社群。對於社群的詮釋，可能會因不同研究目的而有所不同。參考 M. E. J. Newman (2006) 提出的研究，我們依據 OLAP 系統上操作行為相似的歸屬於同一社群，由於他們在實際的企業組織屬同一部門的機率相對高，相對其操作對於推薦上的幫助比較有效益。在大部分社群網絡的研究中，都會利用網絡中心性 (Brass & Burkhardt, 1992) 來判斷節點在『潛在結構』的網絡重要性，節點在群組中的中心性越高相對代表具有越高的影響力。Freeman (1979) 的研究則指出代表中心性概念的三種形式為程度中心性、中介中心性及接近中心性。在本研究中，我們將會利用社會網絡分析的中心性，來找出哪些人在多維度分析平台中屬於老手，並且在進行推薦的時候進行關聯加權。

參、研究方法

一、社群網絡分析

在商業智慧平台的社會網絡中，使用者就是此社會網絡中的節點，而網絡中各種關聯所建立出來的連結都可能影響到推薦的準確。在於商業智慧平台系統中，相同部門的使用者存在相同的背景與資訊需求，產生的推薦項目也相對更能符合期望。在本研

究中，為了增加此因素在於 OLAP 推薦機制的通用性，我們將會依據推薦反饋所建立的聯結作為網絡中的連結，建立取代原本組織關係的虛擬人際關係。圖一即是我們依據此連結關聯所產生的社會網絡圖形。在此圖中連結上的數字表示的是兩節點之間的關聯程度，程度越高的表示的是兩者之間的同質性越高，其屬於相同部門的可能性越高。



圖一、社會網絡圖形

(一)、凝聚子群

表一中，我們對每個使用者依據在集團中相對應的實際組織單位進行編碼，例如 EC 開頭的表示為電子商務部門；然後我們從錄製的資料中找出兩千筆資料，逐筆拿出來跟另外的一萬四千筆操作紀錄作比較，當有完全相似操作的人時，系統會自動將兩個使用間增加一條連結，而所有人之間建立的連結結果統計於關聯分佈矩陣表上，我們可發現在相同部門的使用者之間，於推薦上的採納程度會比不同部門間的推薦高。

表一、虛擬關聯分佈矩陣

加總 - RELAT	EC_PU_IV	EC_PU_V	EC_PU_VI	EC_PU_VIII	EC_PU_XIII	FA_PU_I	Merchant_PU_I	Merchant_PU_II	Merchant_PU_III	Merchant_PU_IV
操作者										
B_PD_PU_I							5	4	20	8
B_PD_PU_III										
EC_PU_A		11	17	13		9	3			
EC_PU_C1		179	43	83	70	34				
EC_PU_C2		22	16	10	8	1				
EC_PU_C3		65	9	38	33	5				
EC_PU_C4		60	26	53	85	8	1			
EC_PU_C5		30	10	3	27	10				
FA_PU_I							96			
Merchant_PU_I			1					14	12	8
Merchant_PU_I								1	31	1
Merchant_PU_I									1	
Merchant_PU_I										6

在本研究中，我們利用 UNCINET 對系統使用者進行虛擬組織的分群，並且在社群中找出相對的老手。本研究執行的步驟如下：

- 計算網絡中最適合的社群數：依據 Newsman and Girvan 的方法，計算在符合最小連結數(Ties)情況下各社群數的 Modularity Q 值，找出 Modularity Q 最大的做為最適合的分割社群數。

- **社群切割(Subgroups)**：我們依據 Girvan and Newsman 演算法，利用邊中性的結果區分出 N 個社群。
- **識別社群中的老手**：依據中心性的分析結果，我們採用分支度中心度〈In-degree〉來判斷各節點在於社群中相對的重要性與影響力。

上述步驟的進一步說明如下：

1. Modularity Q 的計算

Newman and Girvan(2004)提出來的模組性(Modularity)是社群網絡中用來表達的一種結構特徵，可用來衡量網路結構品質的一個指標。其核心概念為社群內的包含大量的關係連結，而在社群間存在少數的連結。因此假設一個社群模組性(Modularity)越高表示社群內的節點存在高關係強度，相對是好的分割。假設今日有一個 n 個節點的社會網絡， A_{vw} 表示的是此相鄰矩陣內的元素，

$$A_{vw} = \begin{cases} 1; & \text{如果節 v 點 vv 與節點 ww 之間有連結，且節 v 點 v 不為節點 w} \\ 0; & \text{其他} \end{cases}$$

在社群網路分群過程中，各個節點都有一個所屬的社群，我們以 C_v 、 C_w 分別表示節點 v 與節點 w 所屬的社群，因此兩個有連結的節點會落在同一個社群的機率如下：

$$\frac{\sum_{vw} A_{vw} \delta(C_v, C_w)}{\sum_{vw} A_{vw}} \quad \text{其中 } \delta(C_v, C_w) = \begin{cases} 1; & C_v = C_w \\ 0; & \text{其他} \end{cases}$$

假設我們以 m 表示網絡中所有關聯的個數，則 $m = \frac{1}{2} * \sum_{vw} A_{vw}$ 。因此我們可以將原本的公式調整如下：

$$\frac{\sum_{vw} A_{vw} \delta(C_v, C_w)}{\sum_{vw} A_{vw}} = \frac{1}{2m} * \sum_{vw} A_{vw} \delta(C_v, C_w)$$

假設 $K_v = \sum_u A_{vu}$ 表示為節點 v 的分支度(Degree)中心性，也就是說節點 v 有 K_v 個連結。假設網絡是隨機的情況下節點 v 與節點 w 的在同一個網絡的機率為 $\frac{K_v * K_w}{2m}$ ，因此在隨機網絡中的 Modularity 為：

$$\frac{1}{2m} * \sum_{vw} \frac{K_v * K_w}{2m} \delta(C_v, C_w)$$

通常我們會將分割社群的 Modularity 與隨機網絡的 Modularity 比較，判斷在此社群組合下結構特性的程度。因此我們將原本 Modularity Q 的公式調整後如下：

$$Q = \frac{1}{2m} * \sum_{vw} [A_{vw} - \frac{K_v * K_w}{2m}] \delta(C_v, C_w) \quad (1)$$

Q 為 0 表示社群內的關聯數與隨機網絡一樣，相對的其結構特性不強。根據下列的計算結果，我們可以發現在此網絡中最適合組織社群叢集數目為 3。

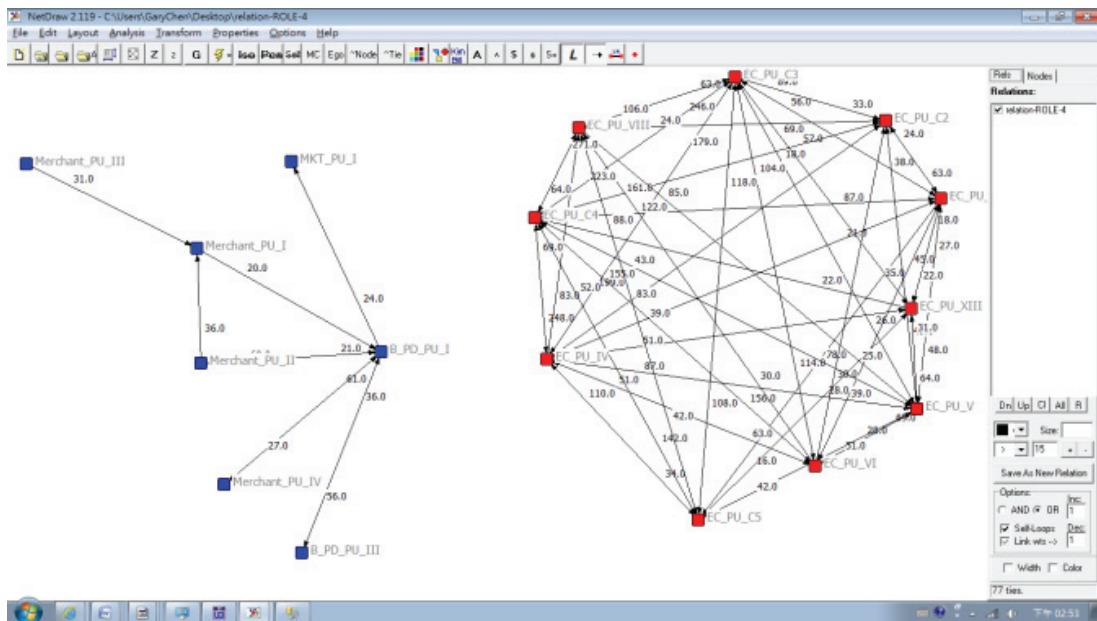
- Partition w/ 3 clusters: Q = 0.216
- Partition w/ 4 clusters: Q = 0.202
- Partition w/ 8 clusters: Q = 0.099
- Partition w/ 9 clusters: Q = 0.066

2. 社群切割(Subgroups)

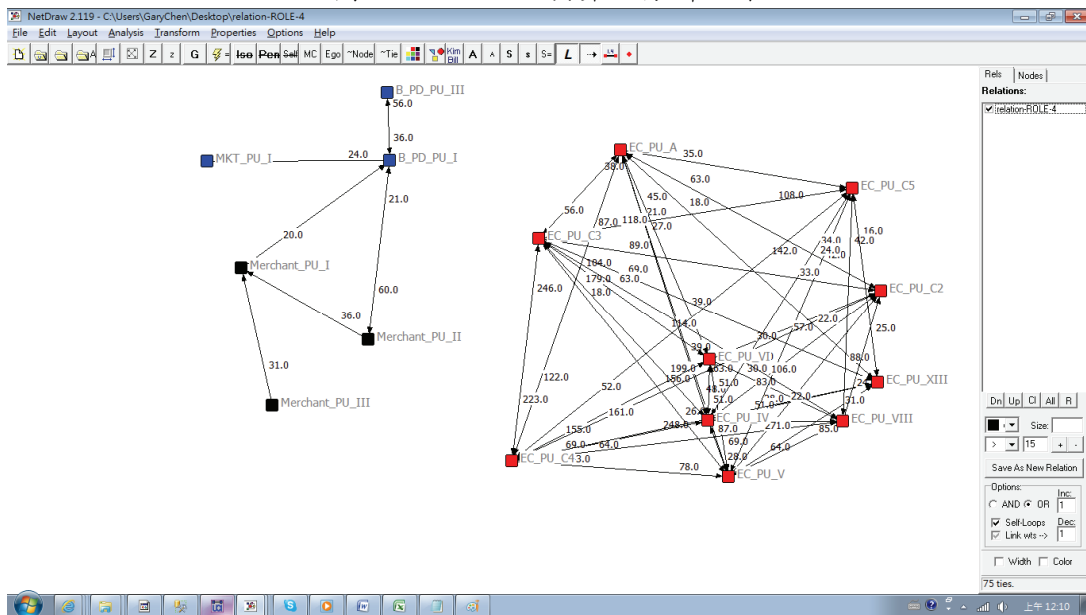
在此研究中，我們對 UCINET 提供的子群分割演算法 Faction 與

Girvan-Newman 演算法的作比較：

- **Factions 演算法**：是使用 Tabu 搜尋方法進行社群的辨識，所依循的原則是尋求集合內各個行動者之截面（profile）的最小「組內方差和」（Moody,2001），圖二即是 Factions 演算法的分群結果。
 - **Girvan-Newman 演算法**：利用邊中介性的特性，在一個網絡中假設節點之間的最短路徑都要經過某一個邊才能到達時，我們移除此邊後，即可很容易取得一個有效的分割結果，圖三即是 Girvan-Newman 演算法的分群結果。
- 由結果的比較，我們可以發現利用 Factions 作組織社群的分派上，只能夠有效的識別電子商務與實體通路的使用者，但是 Girvan-Newman 演算法中，我們在實體通路上，還能有效區分特約商管理部門與經營管理部門的使用者，因此在此次研究，我們將會利用 Girvan-Newman 演算法作為分群依據。



圖二、Factions 演算法分群結果



圖三、Girvan-Newman 演算法分群結果

下列是邊 e 中介性的計算公式：

$$C_b(e) = \sum_{s \neq t \in G} \frac{ost(e)}{ost} \quad ; e = \{x, y\} \quad (2)$$

ost(e)表示的網絡中所有節點 s 到節點 t 的最短路徑中，經過邊 e 的加總個數

$$ost(e) = \begin{cases} 0 & , \text{if } d_{st} < d_{sx} + 1 + d_{yt} \\ osx \times oyt & , \text{otherwise} \end{cases} \quad (3)$$

d_{st} ：表示節點 s 到節點 t 的最短路長度

d_{sx} ：表示節點 s 到節點 x 的最短路長度

d_{yt} ：表示節點 y 到節點 t 的最短路長度

osx ：表示節點 s 到節點 x 的最短路徑數

oyt ：表示節點 y 到節點 t 的最短路徑數

因此在分群的過程中，將依據下列的步驟進行：

- 計算網絡 G 中所有邊的中介性
- 移除中介性最高的邊，將網絡 G 拆分為 G^+ 與 G^-
- 重複步驟 1 與步驟 2，直到滿足 Modularity Q 計算出的最適合社群數

二、老手程度的判斷

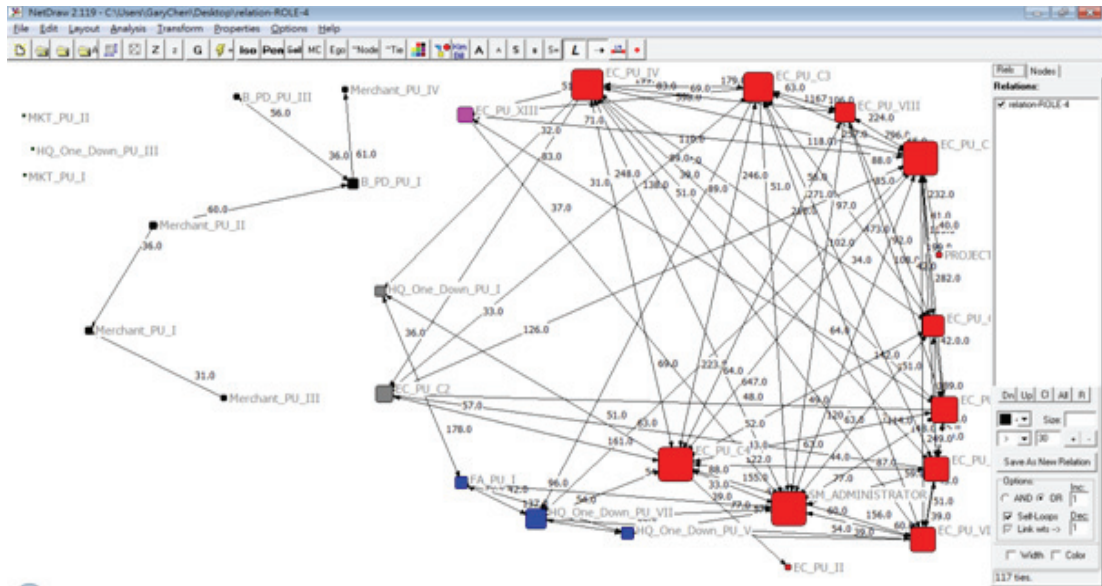
在商業智慧平台的使用上，有一些人在部門中相對重要且具影響力，通常這樣的人都是部門中對於系統使用比較久且比較了解系統功能的人。因此這樣的人在於推薦的判斷上，其操作相對於部門內新人應該來的更重要。而在網絡中通常我們都利用分支度 (Degree) 中心性、緊密度 (Closeness) 中心性、中介 (Betweenness) 中心性三種指標來評估一個節點的重要性。

- **分支度 (Degree) 中心性**：主要是利用每個節點的連結數目作為中心性的判斷，假設有一個節點與其他節點的連結越多的時候，表示該節點在網絡中相對於其他節點更為活絡，相對其擁有較高的中心性。假設一個社會網絡的節點個數為 N 時，我們計算節點 v 分支中心性的公式如下：

$$C_d(v_i) = \frac{\sum_{j=1}^N \alpha(v_i, v_j)}{(N-1)} \quad \forall i, 1 \leq i \leq N \quad (4)$$

$$\alpha(v_i, v_j) = \begin{cases} 1, & \text{節點 } i \text{ 與節點 } j \text{ 之間有連結} \\ 0, & \text{節點 } i \text{ 與節點 } j \text{ 之間無連結} \end{cases}$$

Degree 又可依據連結的方向分為 In-Degree 與 Out-Degree，In-Degree 代表的是推薦時節點 A 被其他節點參考到的連結數，而 Out-Degree 表示的是推薦時節點 A 參考其他節點的連結數。In-Degree 越高的節點所表示該節點在社群中的認可度相對高，該節點可能屬於部門中的老手，因此在系統進行推薦時經常會被其他人採納。而 Out-Degree 越高且分散在許多節點上時，表示的是該節點可能是系統的生手，由於對系統功能的不了解，因此常常會嘗試去參考不特定的人操作作為參考，圖四是我們在 UCINET 上以 Degree 找出來的中心性結果。



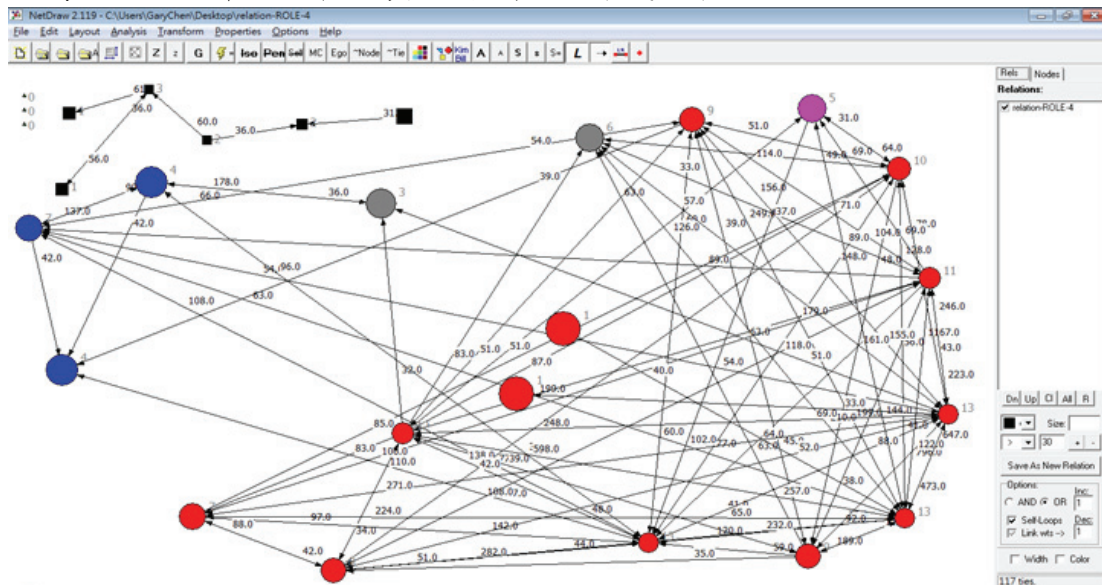
圖四、Degree 的 Centrality measures 結果

- **緊密度(Closeness)中心性**：Closeness 著重的是該節點與其他節點的靠近程度，值介於 0 到 1 之間，值越高表示該節點與越容易到達其他節點，相對表示此節點與其他節點之間的密度高。我們在計算節點 v 的緊密度中心性公式如下：

$$C_c(v_i) = \frac{(N-1)}{\sum_{j=1}^N d(v_i, v_j)} \quad \forall i, 1 \leq i \leq N \quad (5)$$

$d(v_i, v_j)$ 為節點 i 與節點 j 最短路徑之連結數

通常 Closeness 值高的節點通常是資訊傳播者，圖五是我們在 UCINET 上以 Closeness 找出來的中心性結果，在結果中發現此指標雖然能區分出哪些人可能會是老手，但是在老手之間的程度卻無法有效的區分出來。尤其當子群內的節點數比較少的時候，彼此間互動所建立的連結數是很平均的，因此會容易將整個子群的人都視為程度相當的老手或生手。



圖五、Closeness 的 Centrality measures 結果

- **中介(Betweenness)中心性**：在網絡中任兩個節點間一定存在一個最短路徑，假

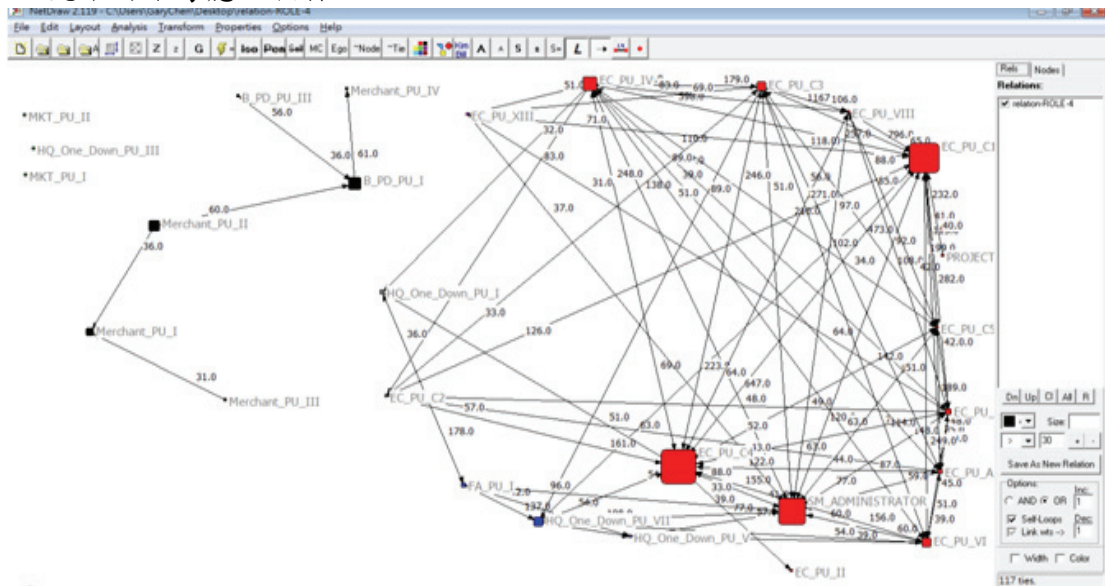
設一個節點經常在最短路徑上面表示越多節點需要經由他才能到達其他節點。通常這樣的節點扮演的是橋樑的角色，其 In-Degree 與 Out-Degree 的值會很平均。計算節點 v 的中介心性公式如下：

$$C_b(v_j) = \frac{\sum \beta(v_i, v_k) / \delta_{ik}}{(N-1)(N-2)} \quad (6)$$

$\beta(v_i, v_k)$ ：表示在兩個節點間的最短路徑中經過 j 的次數

δ_{ik} ：表示在兩個節點的最短路徑的數量

圖六是在 UCINET 上以 Betweenness 找出來的中心性結果。在商業智慧平台上，老手通常對於系統的操作與功能上都很了解，相對 Out-Degree 的數值通常都會偏低。如果採用此指標通常會讓老手的中心性降低，因此在此研究中將不考慮此指標。

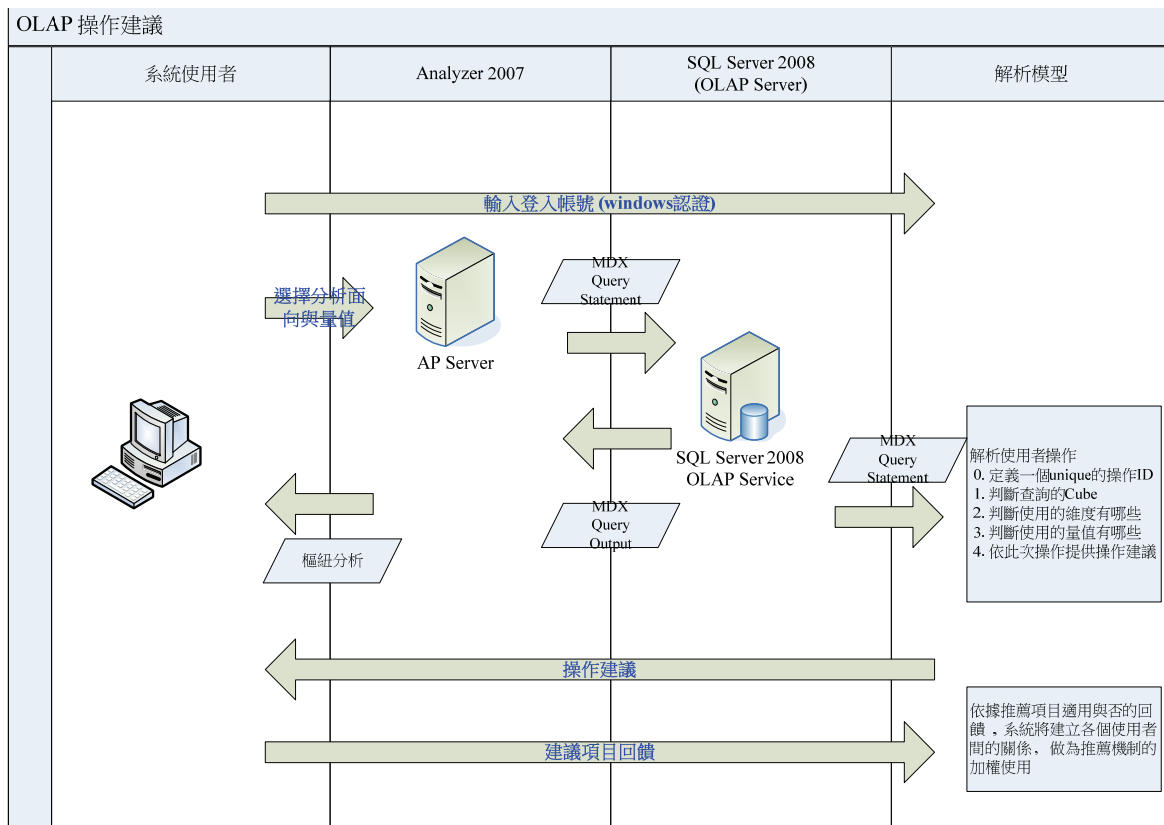


圖六、 Betweenness 的 Centrality measures

最後我們將三種中心性的結果與使用者訪談的結果比對後發現，In-degree 的結果是最符合我們在商業智慧平台上，做為老手程度判斷的指標。因此在後續的推薦機制上我們將會以此方式作為投票加權的判斷。

肆、 線上分析處理推薦機制

本研究對於商業智慧平台上的使用者操作行為，提出一套基於查詢相似度判斷與使用者參考關聯的推薦機制。圖七為本次研究的個人化多維度分析平台推薦的模型架構，此一模型可視為使用者在前端操作介面與資料庫資料中的一個中間層 (Middleware)。系統在使用者在進行操作的過程中，主動記錄使用者的所有查詢紀錄，然後解析個別的查詢描述式進行正規化，並且將結果保存於操作紀錄的資料庫中；系統依據記錄下來的資料，經由相似性與使用者關聯度的判斷，最後將推薦的項目以清單的方式提供使用者作為參考。使用者在取得推薦項目清單後，將會對推薦的項目採納的結果進行系統反饋，用以提供系統進行人員關聯強度的學習，用以提升推薦準確率。



圖七、系統架構圖

一、候選項目的判斷

表二是經由 Profiler 所記錄下來的操作都是一連串複雜的 MDX 語法，這樣的資料無法直接運用在研究中使用。因此在此研究中，我們參考 Carsten Sapia [R3]，將每筆操作記錄的正規化成一個可比較用的有限操作項目集合。

表二、商業智慧平台的操作紀錄

TextData	ConnectionID	NTUserName	StartTime
SELECT SUBSET([EC_商品資訊].[提報日期].[All].ALLMEMBERS, 0, 1) ...	32	GaryChen	2010-04-05 11:54:28.000
SELECT SUBSET([EC_商品資訊].[提報日期].[提報日期_年].ALLMEMBE...	32	GaryChen	2010-04-05 11:54:28.000
SELECT {[EC_產品銷售館別].[銷售館別].[全公司].&[000]}. {[EC_產品銷...	34	GaryChen	2010-04-05 11:54:28.000
SELECT NON EMPTY { { { { AddCalculatedMembers([EC_產品銷售館...	27	GaryChen	2010-04-05 11:54:07.000
SELECT SUBSET([EC_產品銷售館別].[銷售館別].[All].ALLMEMBERS, 0...	18	GaryChen	2010-04-05 11:53:54.000
SELECT SUBSET([EC_產品銷售館別].[銷售館別].[全公司].ALLMEMBER...	18	GaryChen	2010-04-05 11:53:54.000
WITH MEMBER [Measures].[_PseudoMember] AS '0' SELECT NON EMP...	20	GaryChen	2010-04-05 11:53:54.000
WITH MEMBER [Measures].[業績_訂單數 Pct] AS '[EC_供應商.供應商]...	2526	amiutseng	2009-06-01 21:48:14.000
WITH MEMBER [Measures].[業績_訂單數 Pct] AS '[EC_供應商.供應商]...	2522	amiutseng	2009-06-01 21:48:11.000

實作上，我們依據資料庫內的資料屬性，開發了一個解析程式來剖析每筆的操作結果，將每次的操作紀錄拆分為四個部分，使用者、Cube 名稱、維度-報表、維度-過濾條件與量值，最後再經由簡化的方式將操作紀錄正規化為下列集合：

$$P = (User , Cube , Ds = \{s_1, \dots, s_i\} , Dr = \{r_1, \dots, r_j\} , Mm = \{m_1, \dots, m_k\})$$

操作紀錄正規化後，我們依據下列相似度判斷公式計算查詢之間的相似度，找出與此次操作相似度為百分之百的資料，做為產生候選項目使用。

$$|p_2 - p_1| = \Delta C_t * (\Delta D_s + \Delta D_r + \Delta M_m) \quad (7)$$

$$\Delta C_t = \begin{cases} 1, p_1 \text{ 與 } p_2 \text{ 操作相同的 Cube} \\ 0, p_1 \text{ 與 } p_2 \text{ 操作不同的 Cube} \end{cases}$$

$$\Delta D_s = |(D_{s1} \cap D_{s2})| / (|D_{s1}| + |D_{r1}| + |M_{m1}|)$$

$$\Delta D_r = |(D_{r1} \cap D_{r2})| / (|D_{s1}| + |D_{r1}| + |M_{m1}|)$$

$$\Delta M_m = |(M_{m1} \cap M_{m2})| / (|D_{s1}| + |D_{r1}| + |M_{m1}|)$$

ΔD_s 表示的是兩個查詢在過濾維度(Ds)間的不同個數， ΔD_r 表示的是兩個查詢在報表維度(Ds)間的不同個數， ΔM_m 表示的是兩個查詢在量值的不同個數。

在找出所有相似的操作集合後，我們會將所有符合的操作項目集合作聯集，並扣除此次的操作項目後，產生推薦的項目的候選清單。在實際的運作上，找出來的候選項目可能是幾百幾千個，這樣反而會造成使用者因自行過濾有效項目，而使得提供的建議會是無效的。為了解決此問題，後續我們將對過濾出來的項目依據最大信心度與 Link Rank 兩種推薦機制，對於每個項目計算一個分數，依據分數的高低決定哪些項目最適合作為推薦。

二、推薦機制：最大信心度選擇

此推薦機制參考關聯法則中的最大信心度選擇(Most-Confident Selection)觀念而來，主要是以”相同情境下各候選項目被操作過的次數”作為判斷依據，主要是找出使用頻率最高的操作項目作為推薦標的。在此機制中，系統會統計符合下列兩個限制的資料筆數作為分母：

- 操作內容與本次操作內容百分之百相似
- 候選項目符合最小支持度(Minimum Support)

接下來我們會逐一計算在此次的操作內容前提下，各個候選項目出現的次數(Support)作為分母，計算每個候選項目的適合推薦的分數。因此，當一個項目信心水準越高的情況，表示此項目的使用頻率高，推薦成功的機率相對也會比較高的。下列為 Support 機制在計算候選項目 b 在操作項目集合為 A 時的分數公式：

$$\begin{aligned} \text{Score}(A, b) &= \text{Confidence}(A \rightarrow b) \\ &= P(b|A) = \frac{\text{Support}(A \cup b)}{\text{Support}(A)} \end{aligned} \quad (8)$$

A：表示在此次操作項目的集合

b：表示候選推薦項目

Support(A ∪ b)：表示在同時操作項目集合 A 與項目 b 的操作次數

Support(A)：表示在包含 A 操作項目集合的操作次數

三、推薦機制：Link Rank

此機制主要是以”相同情境下各候選各候選項目操作過的人次”作為判斷依據，找出最多人使用的操作項目作為推薦標的。在此推薦方式中，系統會先統計所有候選項目曾經操作過的人數作為分數計算時的分子，接下來系統會將之前所列出來的候選項目逐一去判斷，計算每個候選鍵曾經使用過的人數作為分子。因此，當一個項目使用過的人數越多的情況，表示此項目是越多人關注的，相對推薦成功機

率會比較高的。下列為 Link Rank 機制在計算各候選項目分數時的公式：

$$\text{Score}(A, b) = \frac{\alpha(A \cup b)}{\alpha(A)} \quad (9)$$

A：在此次操作項目的集合

b：候選推薦項目

k：依據相似度找出的候選項目集合

$\alpha(A \cup b)$ ：操作過集合 A 與項目 b 的人次

$\alpha(A)$ ：操作過集合 A 的人次

四、推薦機制與參考關聯加權

根據上述狀況，我們認為每個人操作所佔的重要性是一樣的；但實際上，是否相同部門中每個人對於系統的熟悉程度都應該會影響其在候選項目重要性的判斷。舉例來說，有四個人甲、乙、丙、丁，甲是採購部門的新人，而丁是甲在採購部門的同事，乙與丙是另外一個行銷單位的同事，採購單位會比較著重於銷售商品的成本，但是行銷單位會著重於產品銷售的數量與金額。但由於行銷單位的同仁數相對於採購單位來的多，因此資料庫的操作紀錄也會相對的多；而這樣的情況很容易造成推薦的結果錯誤。而相同的在同部門中，通常會有部分的人對於系統的功能是比較了解的，因此該人員的操作紀錄也應該提高其加權，讓他的操作能有效的反應至推薦結果上。因此，我們將會把凝聚子群的社群結果與參考關係加權加入原有推薦機制，並且作了下列的改善：

- 相同社群內的使用者，其操作紀錄在於候選鍵判斷上才會有效。
- 使用者在該社群中分支度(In-Degree)越高，候選鍵判斷上的加權越高。

下列為各使用者加權係數的計算公式：

$$W_i = \begin{cases} 0; \text{使用者 } i \text{ 與推薦對象不屬於同一社群} \\ C_d(v_i) = \frac{\sum_{j=1}^{n-1} \alpha(v_i, v_j)}{(n-1)} ; \text{其他} \end{cases} \quad (10)$$

因此，我們將兩個推薦機制做下列的調整：

- 推薦機制 最大信心度選擇調整後公式如下：

$$\begin{aligned} \text{Score}'_c(A, b) &= \text{Confidence}(A, b, W_i) \\ &= \frac{\sum_{i=1}^n \text{Support}(A \cup b, i) \cdot W_i}{\sum_{i=1}^n \text{Support}(A, i) \cdot W_i} \end{aligned} \quad (11)$$

- 推薦機制 Link Rank 調整後公式如下：

$$\text{Score}'_r(A, b) = \frac{\sum_{i=1}^n \alpha'(A \cup b, W_i)}{\sum_{i=1}^n \alpha'(A, W_i)} \quad (12)$$

$\alpha'(A, W_i)$ ：使用者 i 操作過項目集合 A 的加權人次

$\alpha'(A \cup b, W_i)$ ：使用者 i 同時操作過項目集合 A 與候選項目 b 的加權人次

W_i ：使用者 i 的老手加權係數

五、評估方式

在本次研究中，推薦機制的優劣我們是以 Recall 值來做評估。我們將會把『推薦結果符合期望』的筆數除以『使用推薦系統』的筆數，得到的值就是推薦機制的正確率。

伍、 系統模擬實驗

在本次實驗中，我們利用使用者在商業智慧平台上一個月的操作紀錄作為實驗資料。在實驗的過程中，我們把資料依據資料量分為十組，每次會選擇其中一組作為實驗組，其他的九組做為學習組，每一組實驗時我們會隨機取兩百筆資料作兩種推薦機制在於一般與加權後的實驗。實驗過程中我們會依據操作的順序移除最後一個操作的項目，並且傳入推薦機制中取得計算後推薦項目。當該操作的最後一個項目包含於推薦的五個項目中時，我們會認為此次的推薦正確。實驗的過程中，我們分為兩階段進行驗證。首先，我們會模擬在不加權時，兩推薦機制的比較；在第二階段中，我們對兩個推薦機制加入關係加權後的比較。表三為兩推薦機制未加權時的驗證結果，而表四為兩推薦機制加權後的驗證結果。

表三、兩推薦機制未加權時的驗證結果

無加權測試結果					
GROUP	測試筆數	機制Support 正確筆數		機制Link Rank 正確筆數	
1	200	93	46.50%	101	50.50%
2	200	116	58.00%	137	68.50%
3	200	146	73.00%	137	68.50%
4	200	86	43.00%	92	46.00%
5	200	133	66.50%	117	58.50%
6	200	140	70.00%	138	69.00%
7	200	123	61.50%	115	57.50%
8	200	68	34.00%	75	37.50%
9	200	100	50.00%	96	48.00%
10	200	142	71.00%	142	71.00%
	2000	1147	57.35%	1150	57.50%

表四、兩推薦機制加權後的驗證結果

GROUP	測試筆數	機制Support 正確筆數		機制Link Rank 正確筆數	
1	200	98	49.00%	119	59.50%
2	200	126	63.00%	143	71.50%
3	200	152	76.00%	157	78.50%
4	200	96	48.00%	115	57.50%
5	200	158	79.00%	161	80.50%
6	200	163	81.50%	163	81.50%
7	200	126	63.00%	129	64.50%
8	200	110	55.00%	130	65.00%
9	200	119	59.50%	125	62.50%
10	200	149	74.50%	151	75.50%
	2000	1297	64.85%	1393	69.65%

在實驗結果上，我們發現在不考慮關係加權的情況下，兩種推薦機制的正確率幾乎差不多是 57 個百分比。但是在將分群結果與參考關係加權的加入推薦機制時，我們發現兩推薦機制在精確度都提升至少 5 個百分比。而 Link Rank 推薦機制的又比最大信心度推薦機制提升多 5 個百分比。另外，在未加權時，Link Rank 推薦機制在某些測試組的推薦結果不如最大信心度推薦機制，但在加入分群結果與關係加權後，幾乎每一組的推薦結果都比最大信心度推薦機制好。

陸、 結論及未來研究

企業在導入商業智慧平台的過程中，通常導入失敗的最大原因絕對不會是在工具與技術面上，而是商業智慧系統提供的資訊過於豐富，造成使用者在剛接觸戒斷容易面臨無法有效取得所需資訊，導致商業智慧無法發揮預期效果。傳統的 OLAP 系統的推薦機制，通常都只考慮到資料的相似度，在使用者的分群上也需要經由背景資訊與喜好的收集後才有辦法進行，但這樣的研究結果通常只能適用於特定的情境。因此為了讓企業在商業智慧平台的推廣上能不需要額外作背景資料收集與分析的作業，此研究將依據使用者在 OLAP 上面的互動關係與操作相似度，提供一個整合社群關係的 OLAP 操作推薦機制。在設計的推薦機制中，我們以同質的凝聚子群來取代實體的部門關係，此外在認同度上我們增加依據 In-degree 中心性計算的關聯加權係數，對於系統老手的操作進行加權的效果，用以提升推薦機制的正確率。在模擬實驗上我們發現，在加入此兩因素後，兩種推薦機制的正確率都有明顯的提升。

在後續的研究上，建議在候選項目演算法與關聯加權係數兩個方面進行改善。首先在候選項目演算法上，在 Link Rank 推薦機制的候選項目評斷上，我們需要逐一取出各候選項目，然後計算各別的連結人次並且作加權效果，但是此方式在候選項目過多時會非常耗時，因此建議可嘗試加入新的演算法改善判斷上的速率。在關聯加權係數上，此次研究我們在判斷老手程度的單純是以同社群中被參考的次數作為依據，但是有的專家可能是在以往的工作上培養對於 OLAP 系統的能力，但這樣的資訊並無法在此次的推薦中展現成效，因此建議可嘗試在社群中再找出新的指標來改善此不足的部分。

參考文獻

1. Balabanovic, M., Shoham, Y.: Fab Content-based, collaborative recommendation. *Communications of the ACM* (40:3), 1997, pp.66 – 72.
2. Brass, D.J. and Burkhardt, M. E., “Centrality and Power in Organization,” In Noria, N. & Eccles, R. G. (Eds.) *Networks and Organizations: Structure, Form and Action*, Boston, Massachusetts: Harvard Business School Press, 1992, pp.191-215.
3. Chen, Y.-T. and Hsu, P.-Y., “Supporting Tools to Query Data in Business Intelligence Systems,” *Proc. of International Conference on Business and Information, (BAI2008)*, 2008.
4. Freeman, L. C., Centrality in Social Networks: Conceptual Clarification Social Networks, *Social Networks* (1:3), 1979, pp.215-239.
5. Newman, M. E. J., “Modularity and community structure in networks,” *PNAS* (103:23), 2006, pp.8577-8582.
6. Newman, M., “The Structure and Function of complex Networkss,” *SIAM Review* (45:2), 2003.
7. Nooy, W.D., *Exploratory Network Analysis with Pajek*, New York: Cambridge University Press, 2005.
8. Page, L., Brin, S., Motwani, R., and Winograd, T. *The PageRank Citation Ranking: Bringing Order to the Web*, 1999.
9. Sapia, C., “On Modeling and Predicting Query Behavior in OLAP Systems,” in *Proc. of the International Workshop on Design and Management of Data Warehouses (DMDW’ 99)*, 1999.
10. Wasserman, S. and Faust, K., *Social Network Analysis: Methods and Applications*, New York, Cambridge University Press, 1994

A Recommendation Mechanism on OLAP Operations Based on Social Network

Hsin-Ku Chen

Computer Science Department, National Chengchi University

Garychen.twn@gmail.com

Tsai-Yen Li

Computer Science Department, National Chengchi University

li@nccu.edu.tw

Abstract

In recent years, enterprises are facing financial turmoil, global competition, and shortened business cycle. Under these influences, enterprises usually implement the Business Intelligence platform to help managers get the key indicators of business management quickly and easily. In the promotion stage of such Business Intelligence platforms, users usually give up using the system due to hung amount of information provided by the BI platform. They cannot intuitively obtain the required information in the early stage when they use the system. In this study, we analyze the similarity of users' operations on the BI platform and try to establish cohesive subgroups in the corresponding organization. In addition, we also integrate the associated weighting factor calculated from the centrality measures into the recommendation mechanism to increase the probability of successful uses of BI platform. From our simulation experiments, we find that the recommendation accuracies are higher when we add the clustering result and the associated weighting factor into the recommendation mechanism.

Keywords : Social Network Analysis, Recommendation Mechanism, Community Detection, Business Intelligence, Network Centrality