探勘高頻瀏覽路徑與高效益項目集之相關性型樣

黃仁鵬
南臺科技大學資訊管理系
jehuang@stust.edu.tw

摘要

隨著網路購物平台密集的發展，購物的環境與行為已經由實體店面漸漸轉移到網路虛擬店面，當使用者在網路購物的同時也留下了大量的瀏覽資料，因此網頁探勘（Web Mining）的技術變得日益重要，這樣技術已經廣泛的應用在商業上的預測以及決策的支援。目前有許多資料探勘的研究，除了探討網頁瀏覽路徑之外，亦同時考慮與購買商品之間的關聯性，進而獲得更詳細的資訊。但在關聯規則裡提供項目間的關係裡項目的重要性都一樣，因此無法得知道項目間是否存在高效益，因此近年來新興起高效益探勘相關研究，高效益探勘考慮項目的數量和效益，因此可探勘出最高效益的商品組合，以達到提升效益。為了達到上述目的，故本研究提出具有連續路徑及高效益產品組合之相關性型樣的 CCU（Correlation Patterns of Consecutive Paths and High Utility Itemsets）演算法。CCU 演算法流程分為三個階段；在第一階段裡，首先找出瀏覽路徑的高頻連續瀏覽路徑。第二階段，再找出高效益產品組合項目集。最後，在第三階段，將第一階段與第二階段的結果進行交叉搭配產生相關性型樣後，再掃描資料庫一次，計算每個相關性型樣的支持度與實際效益值。在前二階段裡，皆會以一個過濾機制來避免大量的候選項目集產生，以提升整個執行效能。在實驗評估裡，從實驗結果可得知在不同參數下，CCU 演算法是具有不錯的執行效率。

關鍵詞：網頁探勘，資料探勘，高效益探勘，過濾機制

Mining Correlation Patterns of Frequent Traversal Paths and High Utility Itemsets

Jen-peng Huang
Department of Information Management, Southern Taiwan University of Science and Technology

Abstract

In recent years, with the development of the online shopping platforms, people's shopping behaviors have been changed from physical stores to virtual stores. Web mining technology has thus become an important issue. It has been widely applied to support of making decisions in business from web transactional databases and web traversal path databases. Since the association rules only provide the relationship among items and the importance of each item is the same, it cannot be found whether there are itemsets with high utility or low frequency from web transactional databases. On the other hand, some studies discussed not only the traversal paths of most users but also the relationship among items to obtain more detailed information, but they did not consider the quantity of items in transactions and their corresponding profits in the database. They could thus not provide users with high utility itemsets information from the web databases. In this study, we proposed a novel algorithm called CCU (Correlation Patterns of Consecutive Paths and High Utility Itemsets), to discover correlation patterns with consecutive and utility aspects from the web databases composing transactions and traversal paths. The mining

Received: Nov. 11, 2020; first revised: Dec. 7, 2020, accepted: Dec. 2020.
Corresponding author: J.-P. Huang, Department of Information Management, Southern Taiwan University of Science and Technology, Tainan 710031, Taiwan.

process can be divided into the three phases. In the first phase, the proposed algorithm can efficiently find large consecutive sequences via the filtration mechanism from the web traversal path database. Moreover, it can efficiently discover the high utility itemsets via the filtration mechanism from the web transactional database in the second phase. The candidate correlation patterns are generated by the large consecutive sequences discovered and the high utility itemsets discovered. The database scan is then executed again to find their support and utility values in the third phase. Finally, the correlation patterns, in which their supports and utility satisfy the support threshold and the utility threshold, respectively, are output as information. In the experimental evaluation, the experimental results show that the proposed CCU algorithm has a good performance under different parameters.
Keyword: Web Mining, Data Mining, Utility Mining, Filtration Mechanism

壹、緒論

近年來，網際網路的興起，網路購物平台密集的發展，購物的環境與購物的行為已經由實體店面漸漸轉移到網路虛擬店面；然而面對這快速增加且又龐大的購物行為資料庫其中，如果要探勘出隱含且未知的資訊，則勢必要透過較特殊的技術來取得，如資料探勘便是此類技術之一。

而網頁探勘也是資料探勘的一種，它是以序列樣式為基礎，主要是探討從網站日誌中，探勘出具有意義與順序性的瀏覽路徑。在過去的應用方面，大都以探勘使用者有興趣的網頁或是教學網站中的最佳學習路徑等，另外也可應用於網路行銷方面，除了探討網頁瀏覽路徑外，亦同時考慮購買商品之間的關聯性，例如：大部份使用者瀏覽路徑為<網頁 A->網頁 B->網頁 F>時會下訂單購買衣服和皮帶，而這類的資訊可以提供給網站設計者，以加強商品與網頁之間的關聯性，進而提昇企業的利潤。

然而，由於高利潤商品發生的購買次數相較一般商品的次數來得少的，例如：以運動品專賣店來說顧客購買襪子的次數與數量比球鞋來的頻繁，但球鞋產生的利潤優於襪子，因此，若使用關聯規則的方式來進行探勘，則會失去關於高利潤商品的規則資訊。

而在網路資料庫裡，若只以商品的相關性和瀏覽路徑做為考慮因素，則無法能有效地預測最高效益；因此，若能將商品的數量、效益並和瀏覽路徑同時考慮且透過資料探勘的技術，則將獲得精確性較高的資訊亦可提昇企業的利潤。

為了解決上述問題，本研究提出一個連續路徑及高效益產品組合之相關性型樣的演算法。所提的演算法可分為三階段；首先，第一階段先從瀏覽路徑資料找出高頻的連續瀏覽路徑。在第二階段裡，則從購買商品資料中找出商品的高效益項目集資訊。在最後第三階段裡，依據第一、二階段得到的結果產生候選相關性型樣，再掃描資料庫以獲得各個候選相關性型樣的支持度與實際效益值，最後，則找出滿足門檻值的相關性型樣規則。

本論文總共分為五個章節。第一章為描述本研究所欲解決問題的動機與目的。第二章則對相關文獻進行探討。CCU 演算法的問題描述、相關處理方法與實際範例將說明於第三章。在第四章的實驗評估則進行所提方法的效能分析與探討。最後，在第五章裡，本研究將給予一個總結。

貳、文獻探討

一、Apriori 演算法

在資料探勘的廣泛議題裡，關聯規則（association rule）是最常被探討的研究議題之一，關聯規則的主要概念是由 Agrawal 等人於 1993 年所提出的[1]，主要是從大量的資料中找出資料項目與項目之間的關聯性，並提出一個著名的 Apriori 演算法來達成探勘目的。爾後，許多研究皆以此概念為基礎延伸出許多有意義的探討議題[1, 6–7]；除此之外，為了能有效率地完成探勘目的，目前有許多研究仍針對演算法

改良的議題在進行探討[6, 12]，這些研究所提出的方法大多以 Apriori 演算法為基礎而加以修改與延伸而來的。

二、EFI 演算法

EFI 演算法[12]是黃仁鵬等學者於 2007 年所提出的，此演算法的特色是二階段過濾的方式，所謂二階段過濾的方式如下：

第一階段： 利用長度 1 的頻繁項目集（large 1-itemsets）進行交易內容的修剪，藉以達到縮短交易的長度，進而加速探勘的效率。

第二階段： 利用長度 2 的頻繁項目集（large 2-itemsets）進行新項目集合的產生，在產生新項目集的過程裡，長度 2 的頻繁項目集可被使用來作為項目間的頻繁關係判斷，因此，所產生出來的項目集數量會接近實際頻繁項目集的數量。

由於 EFI 演算法所採用的過濾機制可大量減少非高頻項目集的數量，因此，其更能適用於探勘交易長度較長的資料庫，且僅需掃描資料庫四次，同時，亦不需要產生任何候選項目集，即可快速找出滿足最小支持度門檻值的頻繁項目集與最小信賴度門檻值的關聯規則。

三、TFA 演算法

TFA 演算法[11]是由黃仁鵬等學者於 2006 年所提出的演算法，其主要特色是藉由利用長度 2 的高頻連續子序列作為在產生連續子序列的過程時，可判斷項目之間是否存在頻繁關聯性的判斷資訊，可有效降低非必要子序列的產生。

TFA 演算法僅需掃描資料庫三次且不需要產生任何候選連續子序列，即可快速找出瀏覽路徑序列型樣；除此之外，由於 TFA 所採用的過濾機制僅會產生最有可能成為高頻的連續子序列，因此，TFA 演算法能大量降低探勘過程中所需耗用的記憶體空間，可有效提昇記憶體的使用率。

四、 Two-Phase 演算法

Two-Phase 演算法[8]，是由 Liu 等學者於 2005 年所提出的探勘演算法，其主要目的是為了改善 MEU 演算法需產生高效益項目集時未符合向下封閉性的問題。因此，Liu 等學者提出一個具有向下封閉性之特性，稱為 TWU（transaction-weighted utilization）模式，其中對不同商品定義其交易效益（transaction utility），可能是商品淨利潤或是其他可代表其商品之價值，主要概念是藉由交易效益作為在該筆交易記錄裡的所有可能子項目集的高估效益值，如此將可避免任何一個可能是高效益項目集資訊的遺失。

參、研究方法

在此章節中，本研究提出高頻瀏覽路徑與高效益項目集之相關性型樣，所提的相關性型樣結合了兩種類型的資料，其中，一種為瀏覽路徑，另一種為商品的購買項目和數量，使得取出的資訊將能更豐富，資訊的準確度也能提高更多。

本研究主要是針對網路資料庫裡的瀏覽路徑和購買商品等資料，探勘出具有連續路徑與高效益產品組合之相關性型樣。在輸入資料部份，需要每位瀏覽者的瀏覽路徑與該瀏覽路徑所購買的交易記錄，每筆交易記錄皆記錄每個購買商品與其對應的購買數量，此外決策者亦需訂定最小支持度與最小效益度等兩個檻門值；而輸出資訊則為連續路徑與高效益項目集之相關性型樣規則。

一、 相關定義

定義 1：瀏覽路徑

瀏覽路徑是將瀏覽的路徑依照使用者瀏覽該網站裡相關網頁一系列的先後順序做排列，表示成<P_1-P_2-…-P_i-…-P_n>，其中 P_i 為一個網頁或項目，而該瀏覽路徑是具有連續性。假設有一個瀏覽路徑為<P_1-P_2-P_3-P_4-P_5>，故<P_1-P_2-P_3>、<P_2-P_3-P_4-P_5>這二條路徑稱為瀏覽子路徑；若有一路徑為<P_1-P_4-P_5>，則該瀏覽

路徑不為瀏覽子路徑，主要原因為在原本的瀏覽路徑裡< P_1-P_2-P_3-P_4-P_5>，P_1的下一個網頁是 P_2 而非 P_4，因此該瀏覽子路徑是不合乎本研究欲探討的瀏覽路徑定義。

定義 2：瀏覽路徑不具有重覆性

本研究所探討的瀏覽路徑是不具有重覆性的性質，因此，假設現有一瀏覽路徑為< P_1-P_2-P_3-P_1-P_4-P_5-P_6>，其中在此瀏覽路徑裡，因為第四個網頁（P_1）為之前已瀏覽過的網頁（如第一個網頁 P_1），針對此重覆的情況，本論文會先經過前置處理將該瀏覽路徑轉換成< P_1-P_4-P_5-P_6>。

定義 3：長度為 k 的瀏覽路徑

瀏覽路徑的長度指的是網頁的數量，因此若長度為 k 的瀏覽路徑，稱為連續 k-瀏覽路徑。例如，假設有一個瀏覽路徑為< P_1-P_2-P_3-P_4>，則稱此路徑為長度 4 的瀏覽路徑。

定義 4：頻繁關係與非頻繁關係

所謂的頻繁關係指的是當網頁 P_1 和網頁 P_2 組合出瀏覽子路徑< P_1-P_2>，當< P_1-P_2>是頻繁瀏覽子路徑時，則我們可以稱網頁 P_1 和網頁 P_2 之間的關係為頻繁關係，反之，當< P_1-P_2>為非頻繁瀏覽子路徑時，則網頁 P_1 和網頁 P_2 之間的關係稱為非頻繁關係。

定義 5：$o(i_p, T_q)$，表示在交易 T_q 裡項目 i_p 有多少數量。表 1 為一購買商品資料庫，其在交易編號 T_1 裡的 A 數量為 12，則表示為 $o(A, T_1)$=12。

表 1　購買商品資料庫

交易編號	購買商品
T_1	A(12)-B(6)-C(26)-E(2)
T_2	A(1)-C(12)-D(7)-E(3)
T_3	A(2)-B(11)-C(28)
T_4	C(3)

定義 6：$s(i_p)$ 表示在一個效益表（Utility Table）裡，每一個項目 i_p 的每一單位利潤值。例如，表 2 為一個效益表，在表裡的項目 A，其每一單位利潤值為 3，因此，可表示成 $s(A)$=3。

表 2　效益表

項目	效益值(每單位)
A	3
B	10
C	1
D	6
E	5

定義 7：$u(i_p, T_q)$ 表示在交易 T_q 裡項目 i_p 的效益，將 $o(i_p, T_q)$ × $s(i_p)$。圖 1 為購買商品和效益表，在交易 100 裡項目 A 的效益則表示成 $u(A, 100)$ = 12× 3 = 36。

交易編號	購買商品
T_1	A(12)-B(6)-C(26)-E(2)
T_2	A(1)-C(12)-D(7)-E(3)
T_3	A(2)-B(11)-C(28)
T_4	C(3)

效益表	
項目	效益值(每單位)
A	3
B	10
C	1
D	6
E	5

圖 1　購買商品和效益表

定義 8： *tu*（T_q），表示單筆交易（T_q）交易效益值，表示為 *tu*（T_q）= $\sum_{i_p \in T_q} u(i_p, T_q)$ 。以交易編號 T_1 為例，*tu*（T_1）=3 *12+10*6+1*26+5*2=132，以圖 2 所示。

交易編號	購買商品
T_1	A(12)-B(6)-C(26)-E(2)
T_2	A(1)-C(12)-D(7)-E(3)
T_3	A(2)-B(11)-C(28)
T_4	C(3)

效益表	
項目	效益值(每單位)
A	3
B	10
C	1
D	6
E	5

交易編號	交易效益
T_1	3*12+10*6+1*26+5*2
T_2	3*1+1*12+6*7+5*3
T_3	3*2+10*11+1*28
T_4	1*3

圖 2　計算交易效益之過程

定義 9： *twu*（*X*）是把所有包含 *X* 項目集的全部交易之交易效益值總和（$\sum tu(Tq)$, where X in Tq），例如：*twu*(A)= *tu*（T_1）*tu*（T_2）+ *tu*（T_3）=132+72+144=348，如圖 3 所示。

交易編號	購買商品
T_1	A(12)-B(6)-C(26)-E(2)
T_2	A(1)-C(12)-D(7)-E(3)
T_3	A(2)-B(11)-C(28)
T_4	C(3)

交易編號	交易利潤
T_1	132
T_2	72
T_3	144
T_4	3

圖 3　計算 TWU 之過程（以項目 A 為例）

定義 10：高交易加權效益項目集（HTWU）與高效益項目集（HTU）

所謂的高交易加權效益項目集是指所有包含 *E* 項目集的全部交易之交易效益值總和（TWU）高於最小效益門檻值，則我們稱 E 為高交易加權效益項目集，HTWU1 項目集為高交易加權效益 1-項目集；HTWU2 項目集為高交易加權效益 2-項目集。而項目 E 實際的效益值高於最小效益門檻值，則稱 E 為高效益項目集。

定義 11：高交易加權效益關係與非高交易加權效益關係

所謂的高交易加權效益（HTWU）關係指的是當項目 A 與項目 B 組合出項目集{AB}，當{AB}是高交易加權效益項目集時，則我們可以說項目 A 與項目 B 之間的關係為高交易加權效益關係，反之，當{AB}為非高交易加權效益項目集時，則項目 A 與項目 B 之間的關係稱為非高交易加權效益關係。

定義 12：頻繁瀏覽路徑與高效益項目集之關係

規則的型式有二種，第一種為瀏覽路徑推測可能發生的高效益事件，例如：當瀏覽路徑呈現為<P_1-P_3-P_5-P_6>時，此瀏覽者可能會購買 X 及 Y 的高效益商品組合；第二種則是由瀏覽路徑及高效益事件推估高效益的事件，例如：大部份的瀏覽者瀏覽路徑為<P_1-P_3-P_5-P_6>時，購買了 X，則此瀏覽者也有可能購買產品 Y。

定義 13：頻繁瀏覽路徑與高效益項目集之資料結構

S = (*Tid*, X_C, X_I)為相關性型樣的資料庫，*Tid* 為交易編號，其中 $X_C \subseteq C$ 和 $X_I \subseteq I$。*C* 為瀏覽路徑，我們表示成<P_1-P_2-…-P_i-…-P_n>，其中 P_i 為一個網頁或項目；I 為購買項目（Purchasing Items），我們表示成<$i_1(q_1)$, $i_2(q_2)$, …, $i_m(q_m)$, … $i_m(q_m)$>，其中 i_j 為一個商品或項目而 q_n 則是其數量，其資料庫如表 3 所示。

表 3　瀏覽路徑及購買商品資料庫

Tid	瀏覽路徑	購買商品及數量
100	1-2-3	W(1),W(6)
200	2-3	Y(2),W(3)

值得注意的是在相關性型樣裡的瀏覽路徑之網頁個數須大於或等於 2 以上，主要原因為二個網頁才會構成一個路徑，此外，在每個相關性型樣裡的頻繁瀏覽路徑或高效益項目集合部份，皆不可為空值狀態，亦即必須同時存在頻繁瀏覽路徑與高效益項目集之資訊。

二、CCU 演算法流程

CCU（Correlation Patterns of Consecutive Paths and High Utility Itemsets）演算法是本研究為了找出全部的高頻瀏覽路徑與高效益項目集之相關性型樣所提出的第一個演算法。CCU 演算法的整個流程可分為三個階段，在第一階段裡，先找出瀏覽路徑資料的高頻瀏覽路徑；接著，第二階段則是找出交易資料部份的高效益項目集；最後，在第三階段裡，將結合上述兩階段的探勘資訊來產生候選相關性型樣集合，當產生完候選項目集後，CCU 演算法會再執行一次額外的資料庫掃描，藉以獲得每個候選相關性型樣的支持度與效益值，當資料庫掃描完成後，即可找出同時滿足最小效益值門檻值與最小支持度的相關性型樣，並輸出作為決策者的參考資訊，其演算法流程圖與虛擬碼如圖 4 所示。

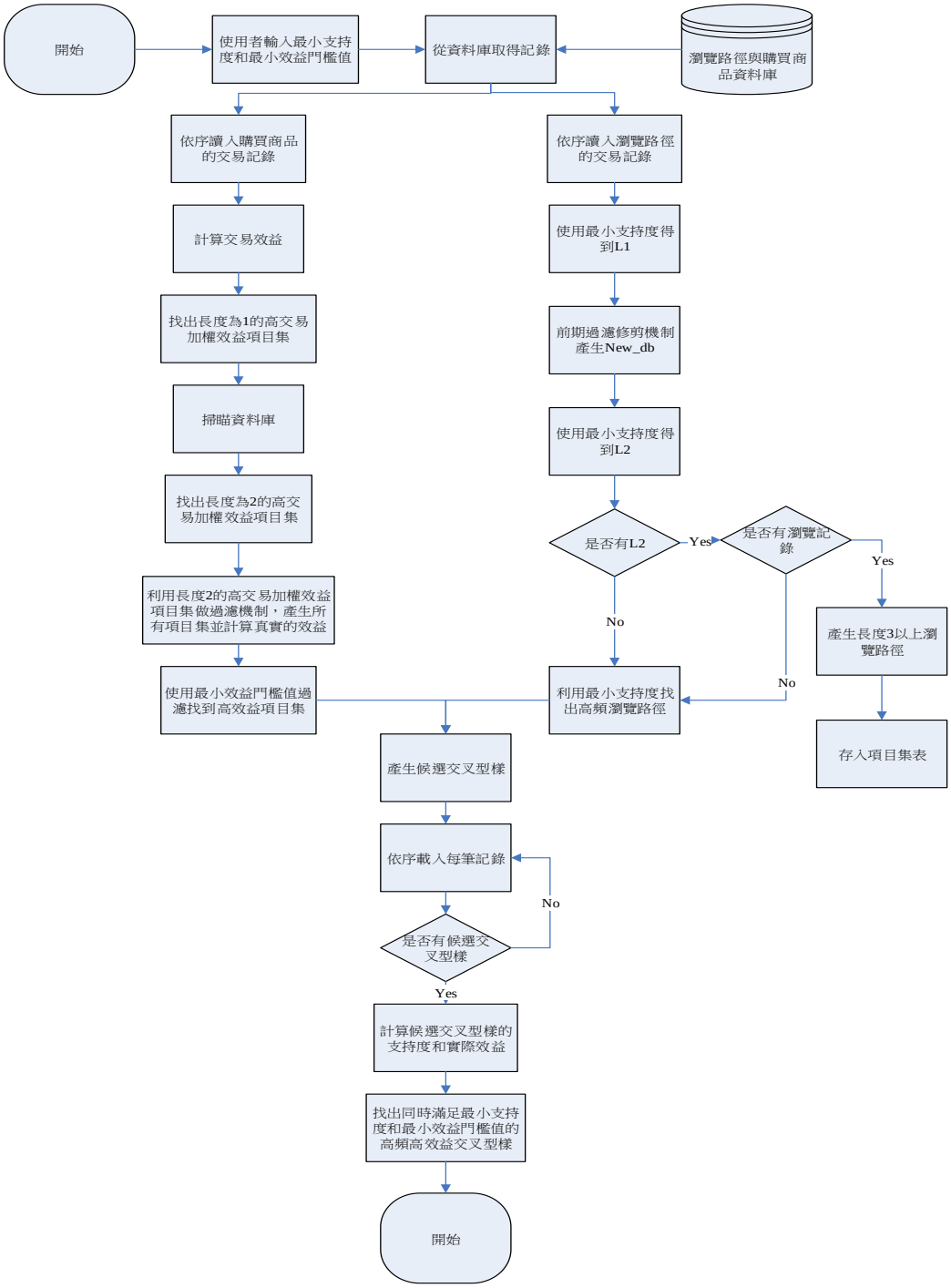

圖 4　演算法流程圖

二、瀏覽路徑產生候選瀏覽路徑過程之相關說明

本研究藉由遞移組合概念可快速地達到產生所有候選瀏覽路徑目的，此處理方式與 TFA 演算法類似。在項目配對的過程中，也將判斷每個相鄰項目之間是否具有非頻繁的關係，若成立，將不再產生以某項目為首的瀏覽子路徑組合，因為相鄰的兩項目所產生之瀏覽子路徑，也必定為非高頻瀏覽子路徑；反之，將繼續進行組合產生以項目為首與之後的項目產生候選瀏覽路徑，直到後面相鄰的項目具有非頻繁關係時才中止，且繼續遞移到下一個項目為首並重複上述的執行程序，直到該筆瀏覽序列到最後一個項目時，即可得到該筆瀏覽路徑所有可能為頻繁的瀏覽路徑組合，以<1-2-3-5>瀏覽路徑為例說明如圖 5 所示。由於使用這種減少非高頻瀏覽路徑方式後，使得整個產生候選瀏覽路徑之流程更加快速，也大幅提昇整個執行的效能。

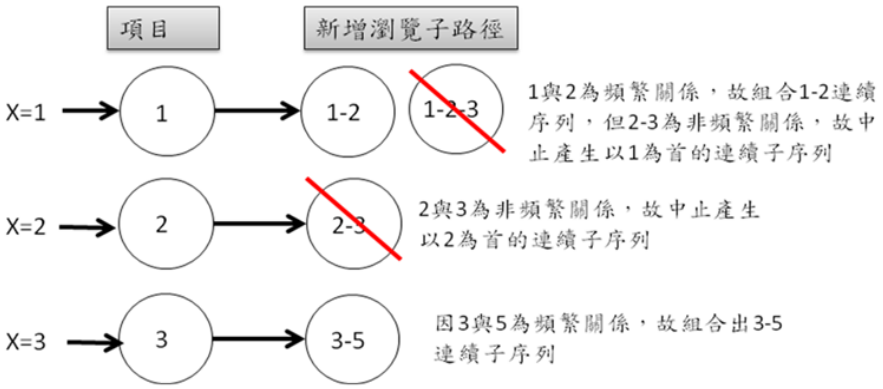

圖 5　減少非高頻瀏覽子路徑產生之過程圖

三、商品資料產生候選效益項目集過程之相關說明

本研究在商品資料產生項目集過程中，為了能完整的探勘出所有的高效益項目集，因此對於項目集的效益值計算是先採利用交易效益方式對項目集做效益值的高估以避免漏掉高效益項目集，找出高交易加權效益 1-項目集與 2-項目集，再藉由快速拆解方法來達到快速拆解的目的，其拆解方法是當讀取到一筆長度為 n 之交易時，只要將目前該項目與之前所產生之項目組合，直接進行搭配將可產生項目集並加上自身交易項目，即可產生所有該交易項目之項目組合，如圖 6。

在項目配對的過程中，將利用交易加權效益項目 2-項目集判斷該項目與之前每一項目之間是否具有高交易加權效益關係，若成立，將與之前所有項目集裡含該項目配對產生新項目集，因為兩項目所產生之所有項目集必定為高交易加權效益項目集，由於使用這種減少非高交易加權效益項目集方式後，使整個產生項目組之流程更加快速，在效能方面也大幅提昇，最後再使用實際效益值過濾出高效益的項目集。

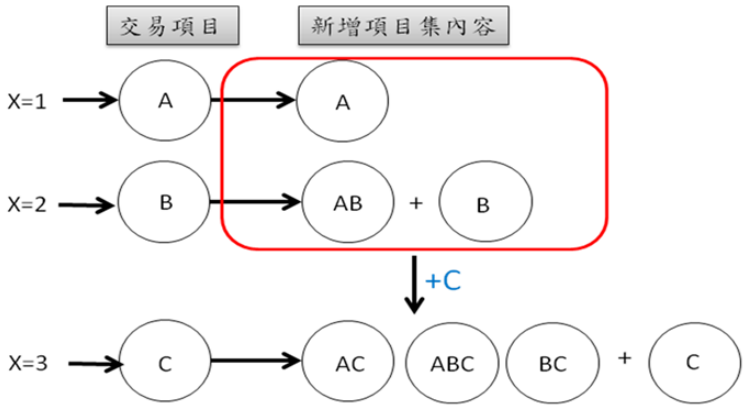

圖 6　商品項目集產生之過程圖

四、候選相關性型樣產生說明

經由上述方法可得到每筆交易內容裡高頻的連續瀏覽路徑與高效益項目集，接著，要將產生出來的高頻的連續路徑與高效益項目集搭配時，該筆交易必須同時皆有產生出高頻的連續路徑與高效益項目集，若是只有一方有高效益項目集的產生，而另一方並無高頻的連續路徑產生，則將會不進行互相交叉搭配的動作，因交叉搭配出來的組合並非是本研究所需要的特殊組合，因此，在交叉搭配的同時，必須稍加注意這點。將每筆交易的高頻的連續路徑與高效益項目集進行交叉相配，便可快速產生本研究所需要的組合，則高頻的連續路徑與高效益項目集交叉相配的過程，如下圖 7 所示：

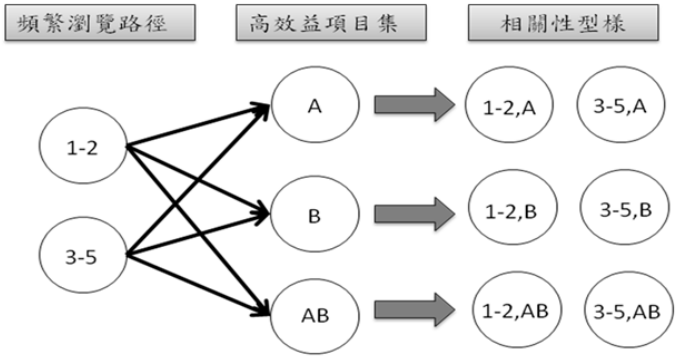

圖 7　候選相關性型樣產生之過程圖

五、 CCU 演算法實例說明

（一）第一階段:找出高頻瀏覽路徑

1. 步驟一：找出頻繁 1-瀏覽子路徑、2-瀏覽子路徑，如圖 8 所示。

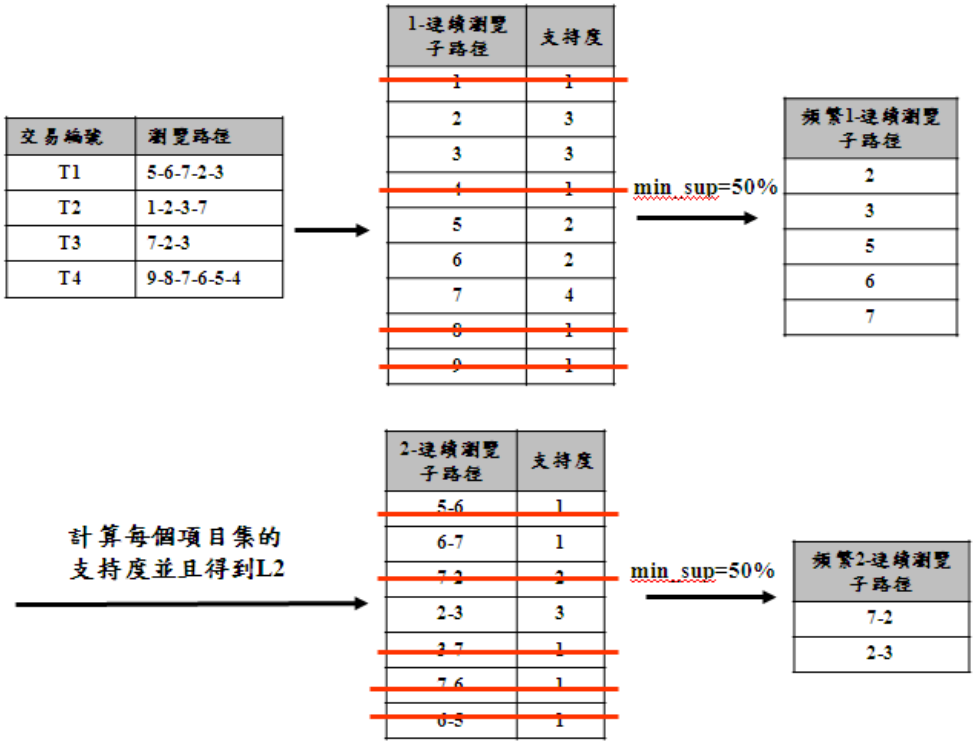

圖 8　產生頻繁 1-瀏覽子路徑、2-瀏覽子路徑之過程

2. 步驟二：依序讀入瀏覽路徑資料，並產生出長度 3 以上的候選瀏覽路徑，如圖 9 所示。

交易編號	瀏覽路徑	購買商品
100	5-6-7-2-3	A(12)-B(6)-C(26)-E(2)
200	1-2-3-7	A(1)-C(12)-D(7)-E(3)
300	7-2-3	A(2)-B(11)-C(28)
400	9-8-7-6-5-4	C(3)

→

候選瀏覽路徑	支持度
7-2	1
7-2-3	1
2-3	1

圖 9　產生所有候選瀏覽路徑（以交易編號 100 為例）

3. 步驟三：利用最小支持度門檻值（min_sup）過濾出頻繁的瀏覽路徑，如圖 10 所示。

候選瀏覽路徑	支持度
7-2	2
7-2-3	2
2-3	3

min_sup =50% →

頻繁瀏覽路徑
7-2
7-2-3
2-3

圖 10　產生高頻瀏覽路徑

（二）第二階段:找出高效益項目集

1.步驟四：計算出每筆交易的交易效益值，如圖 11 所示。效益值彙總值的計算公式為$\sum_{i=1}^{i=k} UI_i * Q_i$，$UI_i$為商品$i$ 的交易效益，Q_i為商品 i 的購買數量。

資料庫

交易編號	瀏覽路徑	購買商品
100	5-6-7-2-3	A(12)-B(6)-C(26)-E(2)
200	1-2-3-7	A(1)-C(12)-D(7)-E(3)
300	7-2-3	A(2)-B(11)-C(28)
400	9-8-7-6-5-4	C(3)

+

效益表

項目	效益
A	3
B	10
C	1
D	6
E	5

→

交易編號	交易效益(TU)
100	3*12+10*6+1*26+5*2=132
100	3*1+1*12+6*7+5*3=72
100	3*2+10*11+1*28=144
100	1*3=3

圖 11　計算每筆交易的交易效益值

2.步驟五：利用各交易的交易效益值計算各項目的交易加權效益 $twu（X）$，找出高交易加權效益 1-項目集（HTWU-1） 項目集和 2-項目集（HTWU-2），如圖 12 所示。

交易編號	瀏覽路徑	購買商品
100	5-6-7-2-3	A(12)-B(6)-C(26)-E(2)
200	1-2-3-7	A(1)-C(12)-D(7)-E(3)
300	7-2-3	A(2)-B(11)-C(28)
400	9-8-7-6-5-4	C(3)

交易編號	交易效益
T1	132
T2	72
T3	144
T4	3

計算每個項目交易加權效益並得到HTWU1 →

項目	交易加權效益(TWU)
A	348
B	276
C	351
D	72
E	204

min_utility =110 →

高交易加權效益1-項目集(HTWU-1)
A
B
C
E

計算每個項目交易加權效益並得到HTWU2 →

項目	交易加權效益(TWU)
AB	276
AC	348
AE	204
BC	276
BE	132
CE	204

min_utility =110 →

高交易加權效益2-項目集(HTWU-2)
AB
AC
AE
BC
BE
CE

圖 12 產生高交易加權效益 1-項目集和 2-項目集之過程

3.步驟六：購買商品資料產生候選項目集並計算候選項目集實際效益，如圖 13 所示。

交易編號	瀏覽路徑	購買商品
100	5-6-7-2-3	A(12)-B(6)-C(26)-E(2)
200	1-2-3-7	A(1)-C(12)-D(7)-E(3)
300	7-2-3	A(2)-B(11)-C(28)
400	9-8-7-6-5-4	C(3)

A → A
B → AB、B
C → AC、ABC、BC、C
E → AE、ABE、BE、ACE、ABCE
BCE、CE、E

計算項目集實際的效益

項目	效益
A	3
B	10
C	1
D	6
E	5

候選項目集	實際效益	候選項目集	實際效益
A	36	ABE	106
AB	96	BE	70
B	60	ACE	72
AC	62	ABCE	132
ABC	122	BCE	96
BC	86	CE	36
C	26	E	10
AE	46		

圖 13　產生所有候選效益項目集之過程（以交易編號 100 為例）

4. 步驟七：利用最小效益門檻值（min_utility）過濾出高效益項目集，如圖 14 所示。

項目集	實際效益	項目集	實際效益
~~A~~	~~45~~	ABE	106
AB	212	~~BE~~	~~70~~
B	170	~~ACE~~	~~102~~
AC	111	ABCE	132
ABC	266	~~BCE~~	~~96~~
BC	224	~~CE~~	~~63~~
~~C~~	~~66~~	~~E~~	~~25~~
~~AE~~	~~64~~		

min_utility =110

高效益項目集
AB
B
AC
ABC
BC
ABE
ABCE

圖 14　產生高效益項目集之過程

（三）第三階段:產生候選相關性型樣

1. 步驟八：利用第一階段產生的高頻連續瀏覽路徑與第二階段產生的高效益項目集交叉配對出高頻瀏覽路徑與高效益項目集之候選相關性型樣，如圖 15 所示。

頻繁瀏覽路徑
7-2
7-2-3
2-3

高效益項目集
AB
B
AC
ABC
BC
ABE
ABCE

候選相關性型樣	候選相關性型樣	候選相關性型樣
7-2,AB	7-2-3,AB	2-3,AB
7-2,B	7-2-3,B	2-3,B
7-2,AC	7-2-3,AC	2-3,AC
7-2,ABC	7-2-3,ABC	2-3,ABC
7-2,BC	7-2-3,BC	2-3,BC
7-2,ABCE	7-2-3,ABCE	2-3,ABCE

圖 15 產生高頻瀏覽路徑與高效益項目集之候選相關性型樣

2. 步驟九：再次掃描資料庫計算每個候選相關性型樣之實際效益與支持度；當完成資料庫掃描後，將輸出滿足最小支持度門檻值與最小效益門檻值之全部相關性型樣，如圖 16 所示。

min_sup =50%

min_utility =110

交易編號	瀏覽路徑	購買商品
T1	5-6-7-2-3	A(12),B(6),C(26),E(2)
T2	1-2-3-7	A(1),C(12),D(7),E(3)
T3	7-2-3	A(2),B(11),C(28)
T4	9-8-7-6-5-4	C(3)

項目	效益
A	3
B	10
C	1
D	6
E	5

候選相關性型樣	支持度	實際效益	候選相關性型樣	支持度	實際效益
7-2,AB	2	212	7-2-3,ABC	2	266
7-2,B	2	170	7-2-3,BC	2	224
~~7-2,AC~~	~~2~~	~~96~~	~~7-2-3,ABCE~~	~~1~~	~~132~~
7-2,ABC	2	266	2-3,AB	2	212
7-2,BC	2	224	2-3,B	2	170
~~7-2,ABCE~~	~~1~~	~~132~~	2-3,AC	3	111
7-2-3,AB	2	212	2-3,ABC	2	266
7-2-3,B	2	170	2-3,BC	2	224
~~7-2-3,AC~~	~~2~~	~~96~~	~~2-3,ABCE~~	~~1~~	~~132~~

頻繁瀏覽路徑且高效益項目集之相關性型樣	
7-2,AB	7-2-3,BC
7-2,B	2-3,AB
7-2,ABC	2-3,B
7-2,BC	2-3,AC
7-2-3,AB	2-3,ABC
7-2-3-B	2-3,BC
7-2-3ABC	

圖 16　產生高頻瀏覽路徑且高效益項目集之相關性型樣的過程

肆、效能評估

一、實驗環境

CPU： Intel Core2 2.83GHZ

RAM： 2048MB DDR2 RAM

OS： Windows XP Professional SP3

資料庫來源：（1）IBM data generator （http://www.almaden.ibm.com/cs/quest/）

（2）取自 FIMI 的資料庫 （http://fimi.cs.helsinki.fi/data/）

開發工具： J2SDK 1.6.0

二、資料庫說明

本研究實驗為了求公正，在實驗裡所使用的測試資料庫皆由 IBM Data Generator （IBM Quest Data Mining Project, 1996） 所產生的；此外，本論文也使用由 FIMI 網站所提供的真實資料庫。在每個資料庫中，每筆記錄可分為瀏覽路徑和購買商品，其中，在瀏覽路徑的部份是以序列方式來表示，而購買商品資訊的部份，則是依據該篇論文（Liu et al., 2005）所描述的內容，進行交易資料的產生，也就是在購買數量方面，其數量範圍為 1~5，而每個商品所對應的效益值範圍為 0.01~10.00，如圖 17 所示。使用 IBM data generator 來產生虛擬的資料庫，其中所使用到的參數設定如表 4 所示。

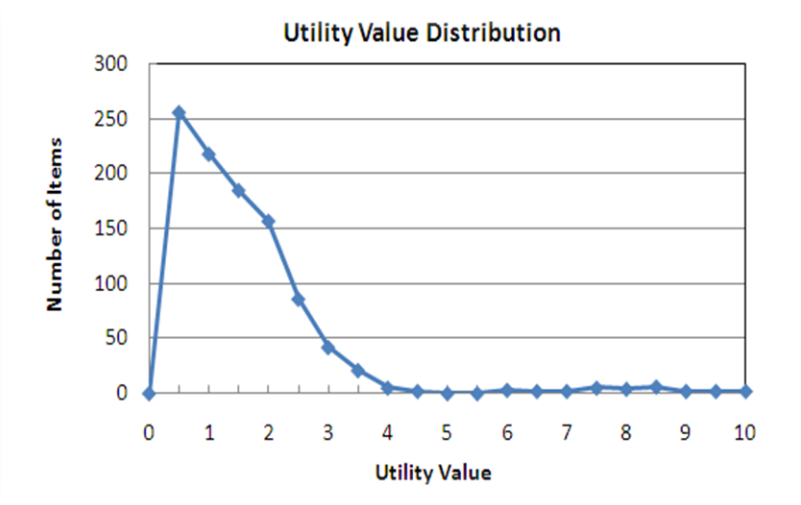

圖 17　測試資料庫裡全部單一商品的個別效益值之分佈圖

表 4　實驗參數描述

參數	參數定義
C	每筆瀏覽路徑的平均長度。
T	每筆交易購買商品的平均長度。
cN	不同網頁種類的數量。
nN	不同交易項目種類的數量。
D	交易資料的筆數。(每筆記錄包含瀏覽路徑資料與交易資料)
Min_sup	最小支持度門檻值
Min_utility	最小效益值門檻值

三、實驗設計

本研究共設計三項實驗，首先根據 IBM 所產生的資料庫分別測試各種不同的參數設定對演算法的效能影響，參數包含最小支持度門檻值、最小效益門檻值、不同的資料筆數、不同平均路徑長度、不同平均交易長度、不同網頁總數、不同商品總類，上述實驗設計如下：

（一）實驗 1：演算法在不同最小支持度門檻值下的影響

首先，在此實驗裡，將評估演算法在此實驗資料庫 C10cN1KT10nN2KD100K 與不同最小支持度門檻值（min_sup）下的效能表現，其中，在瀏覽路徑平均長度（C）部份設定為 10、交易平均長度則設定為（T）10、資料筆數（D）設定為 100K、網頁總數（cN）設定為 1K、商品總類設定為（nN）2K 與最小效益門檻值（min_utility）設定為 3%。此外，設定最小支持度門檻值（min_sup）為 1%至 10%，則演算法在上述的參數設定下的效能評估圖 18。

在圖 18 裡，可得知我們所提的演算法在不同最小支持度門檻值下，所需花費的執行時間仍可保持穩定的執行效能，其主要原因為 CCU 演算法使用了有效的過濾機制來避免大量非必要連續路徑的產生，因此在不同門檻值下，CCU 演算法能有效地探勘出所提的具有高頻瀏覽路徑與高效益的相關性型樣。

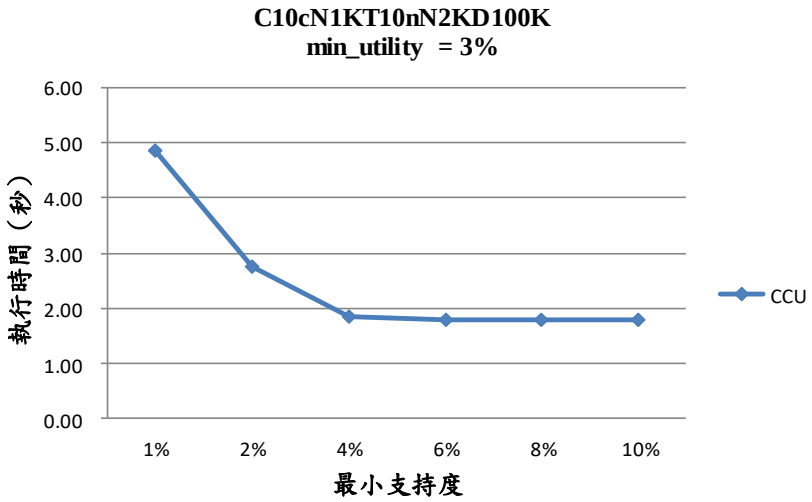

圖 18　在不同最小支持度門檻下，演算法的執行效能評估

（二）實驗 2：在不同最小效益值門檻下，演算法的執行效能評估

在此實驗裡，將評估 CCU 演算法在此實驗資料庫 C10cN1KT10nN2KD100K 與不同最小效益門檻值（min_utility）下的效能表現，其中，在瀏覽路徑平均長度（C）部份設定為 10、交易平均長度則設定為（T）10、資料筆數（D）設定為 100K、網頁總數（cN）設定為 1K、商品總類設定為（nN）2K 與最小支持度門檻值（min_sup）設定為 3%。此外，設定最小效益門檻值（min_utility）為 1%至 10%，則演算法在上述的參數設定下的效能評估示意圖如圖 19 所示。

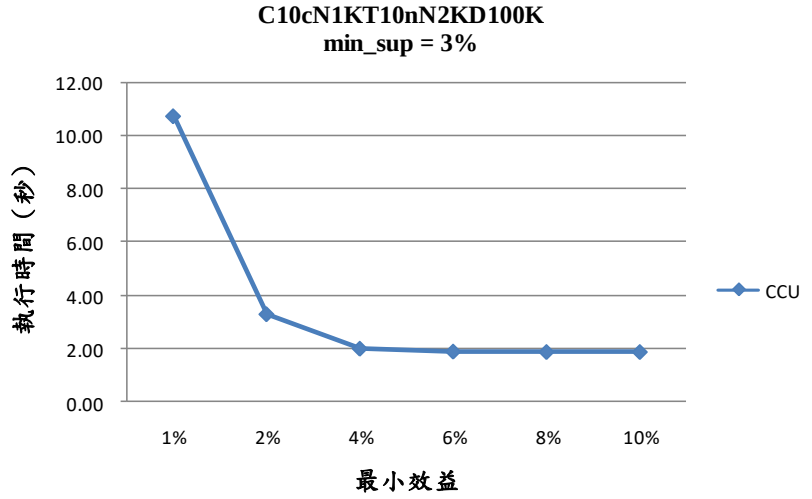

圖 19　不同交易的平均長度對演算法效能的影響圖

在圖 19 裡，可得知我們所提的演算法在不同最小效益值門檻值下，所需花費的執行時間仍可保持穩定的執行效能，其主要原因為 CCU 演算法使用了有效的過濾機制來避免大量非必要項目集的產生，因此在不同門檻值下，CCU 演算法能有效地探勘出所提的具有高頻瀏覽路徑與高效益的相關性型樣。

（三）實驗 3：在不同資料筆數下，對演算法執行效能的影響

在此實驗裡，將評估 CCU 演算法在此實驗資料庫 C10cN1KT10nN2K 與不同資料量（D）下的效能表現，其中，在瀏覽路徑平均長度（C）部份設定為 10、交易平均長度則設定為（T）10、網頁總數（cN）設定為 1K、商品總類設定為（nN）2K、最小支持度門檻值（min_sup）設定為 3%與最小效益門檻值（min_utility）設定為 3%。此外，設定資料量（D）為 20K 至 800K，則演算法在上述的參數設定下的效能評估示意圖如圖 20 所示。

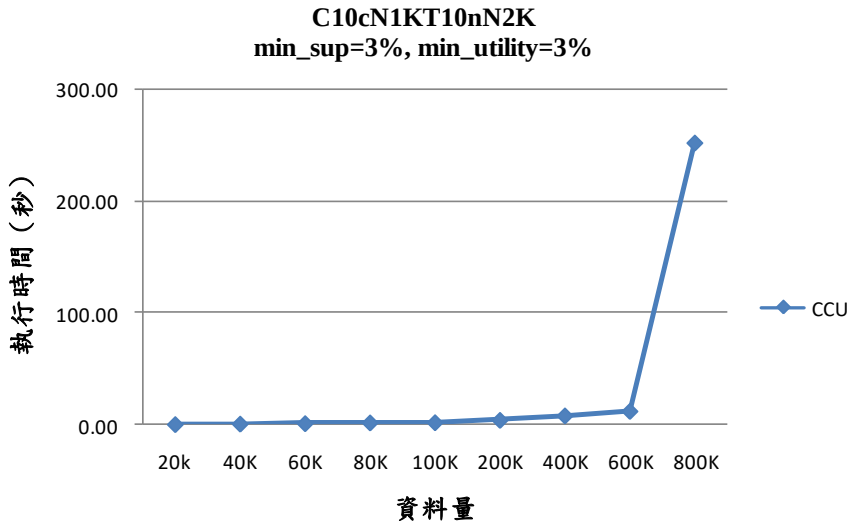

圖 20　不同資料筆數對演算法效能評估示意圖

在圖 20 裡，資料量為 800K 時，其執行時間呈現急速上升，主要原因是因為實驗設計的主機記憶體僅 2GB，造成當資料量變大時，演算法因高頻連續瀏覽路徑與高效益項目集交叉配對出高頻瀏覽路徑與高效益項目集之候選相關性型樣也相對也會變更大，而造成記憶體不足的情況，因此系統就會不斷利用硬碟進行資料的 swapping 動作，這是造成執行時間花費較多的原因。不過，CCU 在記憶體足夠的情況下，當資料筆數增加時，CCU 演算法仍可保持穩定的執行效能，其主要原因為 CCU 演算法使用有效的修剪機制避免非必要項目集的產生和資料庫的縮減，所以當資料筆數增加時，仍可維持穩定較佳的執行效能。

伍、結論與未來研究

本研究為了能從網路的瀏覽路徑和購買商品的資料裡，探勘出連續瀏覽路徑型樣及高效益產品組合之相關性型樣，則提出了一個 CCU 演算法，來解決上述的問題。而在探勘的過程中 CCU 演算法在連續瀏覽路徑和高效益項目集的探勘是採用相關拆解方式與過濾機制，因此能大量減少非高頻連續瀏覽路經和非高效益項目集的數量，能較有效率探勘出高頻的連續路徑且高效益項目集的相關性型樣。

在實驗評結果可得知 CCU 在記憶體足夠的情況下，在不同最小支持度門檻值、最小效益門檻值和當在不同資料筆數下，所需花費的執行時間仍可保持穩定的執行效能，其主要原因為 CCU 演算法使用了有效的過濾機制來避免大量非必要瀏覽路徑和項目集的產生，因此在執行時間可保持穩定的執行效能。

本研究所提出的演算法雖有使用過濾機制來避免產生大量非必要瀏覽路徑和項目集，但是由於本實驗的主機記憶體僅 2GB，當演算法因資料量變大，造成配對所產生的高頻瀏覽路徑與高效益項目集之候選相關性型樣超過記憶體的負荷而造成記憶體不足的情況時，系統因而會不斷利用硬碟進行資料的 swapping 動作，會造成執行時間急速增加。因此如何能利用更佳的資料結構方式，以更有效的方法儲存所產生的候選相關性型樣，以達到更有效的記憶體利用率，可作為未來可延伸的研究方向。

在應用方面，本研究所提出的相關性型樣可應用於網路行銷上，網路的決策者可藉由相關性型樣資訊，以決定在哪些網頁裡，可進行高效益商品的推薦，以達到企業收益的增加，進而達到企業與消費者兩方雙贏的局面。

參考文獻

[1] R. Agrawal, T. Imielinski, and A. Swami (1993, June). *Mining association rules between sets of items in large databases*. 1993 ACM SIGMOD international conference on Management of data (207–216). https://doi.org/10.1145/170036.170072

[2] R. Agrawal and R. Srikant (1994, September). *Fast algorithms of mining association rules*. Proceedings of 1994 International Conference on Very Large Data Bases (489–499).

[3] R. Agrawal and R. Srikant (1995, March, 6–10). *Mining sequential patterns.* Proceedings of the 7th International Conference on Data Engineering (3–14). Taipei, Taiwan. https://doi.org/10.1109/ICDE.1995.380415

[4] R. Chan, Q. Yang, and Y.-D. Shen (2003, Nov. 22). *Mining high utility itemsets*. Proceedings of the Third IEEE International Conference on Data Mining (19). Melbourne, FL, United States. https://doi.org/10.1109/ICDM.2003.1250893

[5] M.-S. Chen, J.S. Park, and P.S. Yu (1998). Efficient data mining for path traversal patterns. *IEEE Trans Knowl Data Eng, 10*(2), 209–221.

[6] J. Han, J. Pei, and Y. Yin (2000). Mining frequent patterns without candidate generation. *ACM SIGMOD Record, 29*(2), 1–12. https://doi.org/10.1145/335191.335372

[7] J. Liu, Y. Pan, K. Wang, and J. Han (2002, July). *Mining frequent item sets by opportunistic projection*. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (229–238). https://doi.org/10.1145/775047.775081

[8] Y. Liu, W.-K. Liao, and A. Choudhary (2005). *A two-phase algorithm for fast discovery of high utility itemsets*. In T.B. Ho, D. Cheung, H. Liu. (eds) Advances in Knowledge Discovery and Data Mining. PAKDD 2005. Lecture Notes in Computer Science, 3518 (689-695). Springer. https://doi.org/10.1007/11430919_79.

[9] B. Mobasher, R. Cooley, and J. Srivastava (2000). Automatic personalization based on Web usage mining. *Communications of the ACM,* 142–151. https://doi.org/10.1145/345124.345169

[10] 林曉薇 (2005)。**連續事件及無序性三值組式關聯規則演算法與其應用**（碩士論文）。南臺科技大學，台南市。

[11] 黃仁鵬與藍國誠 （2006）。高效率探勘瀏覽路徑序列型樣之研究。**科技管理學報，11**（4），1–36。

[12] 黃仁鵬與藍國誠 （2007）。高效率探勘關聯規則之演算法-EFI。**資訊管理學報，14**（2），139–168。