國立交通大學

資訊學院　資訊學程

碩 士 論 文

以個人化標籤推薦系統探討網路標籤使用行為

Investigating user tagging behaviors in social bookmark system

研究生：鄧睿清

指導教授：孫春在 教授

中華民國九十七年六月

以個人化標籤推薦系統探討網路標籤使用行為

Investigating user tagging behaviors

in social bookmark system

研 究 生：鄧睿清　　Student: Jui-Ching Teng

指導教授：孫春在　　Advisor: Chuen-Tsat Sun

國 立 交 通 大 學
資訊學院　資訊學程
碩 士 論 文

A Thesis
Submitted to College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master of Science
in
Computer Science
June 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年六月

以個人化標籤推薦系統探討網路標籤使用行為

學生：鄧睿清　　指導教授：孫春在 博士

國 立 交 通 大 學　　資 訊 學 院　　資 訊 學 程 碩 士 班

摘　　要

隨著 web2.0 時代的來臨，網路應用掀起一股 tagging 風潮。資訊的消費者將資訊貼上標籤不只是管理個人的檔案，也管理個人的知識。同時藉由網路的分享，共享標籤更是讓知識傳遞及匯集的速度到達前所未有的境界。標籤推薦可以提供候選字以及標籤關聯性使得知識管理、資訊取回、搜尋排序更有效率。

在觀察社會標籤系統(social tagging system)中由使用者、標籤、文件所形成的三分關聯網路(tripartite network)之後認為，社會標籤系統中一個好的推薦系統應展現使用者們所形成的群體智慧，以眾人的智慧幫助個人，也以個人的力量幫助眾人。在過去的標籤推薦演算法在尋找相似人群以及相似標籤上的侷限，本論文提出一個基於社會網路分析理論的使用者標籤推薦的演算方法「派系篩檢法(Clique Fitering)」，演算法的架構主要是衍生於協同過濾法(Collaborative Filtering)但其中演算的精神來自於社會網路中的派系過濾法(Clique Percolation)，在使用者對文件貼標籤或使用者想利用標籤對文件進行過濾的情境(scenario)下提高標籤推薦的準確度。以這種演算法，可以直接應用在標籤推薦系統中，不需系統對於字詞有所認知，可以適用於現行以社會標籤為管理的應用系統中，並且也可以將結果應用在其他個人化的系統當中。除此之外，也以目前最多人使用的論文文獻檢索系統 CiteULike 作為範例，利用社會網路的分析方法，分析其中人際之間標籤使用的群聚行為，發現標籤的使用反映人的思考，標籤使用的習慣有「物以類群、人以群分」的小世界行為模式。

關鍵字：群眾分類、社會標籤、社會書籤、標籤系統、推薦系統

Investigating user tagging behaviors in social bookmark system

Student: Jui-Ching Teng Advisor: Dr. Chuen-Tsai Sun

Degree Program of Computer science
National Chiao Tung University

ABSTRACT

Folksonomy is a popular application of web2.0. Information consumers label resources with arbitrary words, so-called *tags*. Social tagging systems not only help people share resources but also share knowledge. Tag recommendations can help user in Knowledge Management by providing candidate tags, in Information Retrieval by discovering relations of tags, and in Search by providing personalized keywords reminding.

After observing social tagging system, we focus on the tripartite network that formed by users, tags, and items in the system. A good recommend system should present the co-active intelligence. In the past, tag recommendation algorithm is difficult to find similar people and similar tags. In order to improve tag recommendation, we propose a modified tag recommendation approach Clique-Filtering that based on social network theory. We evaluate and compare it with collaborative filtering on real-life dataset. We show the performance of modified approach is better than collaborative filtering in the sparse dataset. We can apply the result to other personalized recommendation system. After analyzing one of popular bibliography site CiteULike, we discover the personal tag clustering in the real-life system.

Keywords: folksonomy, social tagging, social bookmarking, tagging system, recommendation system, clique filtering

誌　　謝

首先誠摯地感謝指導教授 孫春在博士，老師的教導讓我能將社會網路與標籤系統兩者合而為一來做研究。老師的指點讓我發現複雜網路的背後存在單純的道理。同時也要感謝論文口試委員 張智星教授與 袁賢銘教授給予我寶貴的意見，讓此篇論文更加精確嚴謹。

感謝崇源學長在研究的過程中的指導及討論，以及吉隆學長給予的資料及幫助。使得我在論文的寫作期間，能夠抓住方向不致迷失。同實驗室的同學互相討論、互相打氣更是讓我在最後階段持續前進以完成論文，非常感謝學長、學姐、同學們所給予的支持與鼓勵。

由衷的感謝我的家人、朋友還有同事。特別是在身旁的家人，給予我最多的新的思考方向。沒有平常的閒聊討論，就沒有這麼簡單、直接、有創意的研究方法。也感謝家人、朋友、同事的諒解，讓我可以兼顧課業及工作。

最後，感謝我摯愛的雙親 鄧友強及 張小娟對於我的栽培及教導。

目　　錄：

中文摘要：.....ii

英文摘要：.....iii

誌　　謝：.....iv

目　　錄：.....v

表目錄：.....vii

圖目錄：.....viii

一、　緒論.....1

1.1　研究動機.....1

1.2　研究背景.....2

1.3　研究目標.....4

1.4　研究流程與論文架構.....4

二、　文獻探討.....6

2.1　社會性的使用者分享系統 Social sharing system.....6

2.2　分眾分類 Folksonomy.....7

2.3　協同過濾 Collaborative filtering.....10

2.4　小世界與分眾分類 Small world and Folksonomy.....11

2.5　派系過濾法 Clique percolation method.....12

三、　研究方法.....14

3.1　基本性質.....14

3.2　資料來源.....15

3.3　標籤網路 tag network 的形成與複雜網路特性.....15

3.4　派系篩檢 Clique Filtering 的標籤推薦清單.....19

3.5　協同過濾 Collaborative Filtering 推薦清單的形成方法.....21

3.6　評估 Evaluation.....22

四、 實驗結果 23

4.1 標籤網路 tag network 的基本性質 23

4.2 推薦標籤清單 31

4.3 討論 42

五、 結論 45

5.1 研究結論 45

5.2 未來方向 45

六、 參考文獻： 47

表目錄：

表格 1 CiteULike 資料集性質 15

表格 2 標籤網路性質 24

表格 3 不同 p 核心資料集性質 32

表格 4 實驗方法表 32

表格 5 “CF neighbor and Clique sorting” and “Clique neighbor and Clique sorting” 推薦評比 33

表格 6 “CF neighbor and CF sorting” and “Clique neighbor and CF sorting”推薦評比 33

表格 7“CF neighbor and Clique sorting” and “CF neighbor and CF sorting”推薦評比 34

表格 8 “Clique neighbor and Clique sorting” and “Clique neighbor and CF sorting” 推薦評比 34

表格 9 各種推薦方式的平均推薦數 34

圖目錄：

圖 1 貼標籤背後的認知過程[9] 6

圖 2 分類背後的認知過程[9] 7

圖 3 在「分眾分類」的使用者(user)、標籤(tag)、物件(item) 8

圖 4 人-物投影二分關聯圖 9

圖 5 二分關聯圖投影至單一關聯圖 10

圖 6 使用者次數、標籤次數、論文次數 25

圖 7 標籤網路(文件) v.s.分支度機率 26

圖 8 標籤網路(人) v.s.分支度機率 27

圖 9 標籤網路(文件) v.s.群聚度 30

圖 10 標籤網路(人) v.s.群聚度 31

圖 11 推薦數 V.S. 檢全率、檢準率(p-核心=3) (doc based) 35

圖 12 推薦數 V.S. 檢全率、檢準率(p-核心=3) (doc based) (tag based) 36

圖 13 p 核心 V.S. 檢全率 37

圖 14 p 核心 V.S. 檢準率 38

圖 15 檢全率與檢準率(p 核心= 3 及 p 核心= 8) (doc-based) 39

圖 16 檢全率與檢準率(p 核心= 3 及 p 核心= 8) (tag-based) 39

圖 17 推薦數與 F1(p 核心= 3 及 p 核心= 8) (doc-based) 40

圖 18 推薦數與 F1(p 核心= 3 及 p 核心= 8) (tag-based) 40

圖 19 推薦數與 F2(p 核心= 3 及 p 核心= 8) (doc-based) 41

圖 20 推薦數與 F2(p 核心= 3 及 p 核心= 8) (tag-based).....41

圖 21 Doc based V.S. Tag based (檢全率) 42

圖 22 Doc based V.S. Tag based (檢準率) 43

圖 23 檢全率與檢準率(doc-based and tag-based) 44

圖 24 推薦數與 F2 (doc-based and tag-based).....44

一、 緒論

1.1 研究動機

近來網路上出現一種新型態的網站，這類的網站被稱做「社會性的使用者資源分享系統(social resource sharing system)」，或是稱做「分眾分類(folksonomy)」[1]，可以讓使用者上傳「資源(resource)」以分享給其他人為主。其後慢慢演變為不只是分享個人的創作。分享的內容有來自自行創作的文字、圖片(Flickr)，也有很大部份是來自其他人的創作甚或是網路書籤也成為分享的一個內容(del.icio.us)。這些網站多數使用「標籤系統(tagging system)」[2](2006)，讓分享者可以管理自己的分享，也讓觀看者可以針對自己有興趣的部分做搜尋。

「分眾分類」的使用者可以依照使用網站的方式分成兩種，分享者(sharer)與觀看者(browser, reader)。分享者的使用方式，是將欲分享的資源，上傳之後，給予一個或數個的任意文字。這些字，即稱做「標籤(tag)」，將會成為分享者在將來找回該資源的「關鍵字(keyword)」，因此，這些關鍵字是分享者對於資源了解之後的簡單描述。當然，這些標籤，也是其他人要找尋資源時的關鍵字。若是分享者也有考慮到觀看者的需求時，分享者也會用其他人比較容易了解的文字做為該資源的標籤，讓該資源可以容易地被其他人找到。而觀看者，則是使用附屬在「分眾分類」的搜尋引擎，針對自己有興趣的主題範圍「搜尋(finding)」，標籤可以越加越多以縮小找尋的範圍，或是「瀏覽(browsing)」任何吸引觀看者的 tag 內所含的資源[3]。

在這類的網站上，當資源多到一定程度之後，使用者面臨的是要如何在網站給予標籤，才能找到想要的資源。不單單是觀看者對於該資源的描述應該準確，才能讓系統搜尋得到，亦需要分享者給予足夠的標籤，才能讓不同背景的觀看者有機會找到想要的資源。該如何給予標籤，一直是個問題。在網站初期的確是可以隨心所欲，也因此吸引了大量的人為整理分享自己的資源而加入。但到了後期，資源的數量多到使用者不知該如何開始「瀏覽」時，許多研究指出，因為人類語言的本質，使得在搜尋資源上反而被隨心所欲這個特性所拖累，因此「控制性詞彙(controlled vocabulary)」的方法被提出來想要解決標籤、關鍵字增加、描述不明確的問題。但是「控制性詞彙」的方法，會增加使

用者認知上的「負擔(cost)」，也需要使用者學習「控制性詞彙」的明確分類[3]，這明顯的是一個兩難的問題。

在隨心所欲的開放式使用者自訂標籤與控制性詞彙的使用上，各自有所缺陷，但是，若透過有效的使用者標籤的分享與推薦，可以有效地彌補這兩個缺陷。標籤推薦（Tag Recommendation）可發揮的時機有二，一個是分享者要分享資源時，主動提供相關聯的標籤給分享者做判斷。另一個是觀看者在「瀏覽」或是「搜尋(finding)」時，提供可能標籤給觀看者做選擇。如此看來，「標籤推薦」所造成的認知負擔與關鍵字的明確性，都可以有一定程度的兼顧，這是對於「標籤推薦」的基本期望。然而問題就在於目前「標籤推薦」的機制似乎不多，在應用上也多是採用「協同過濾(collaborative filtering)」的方式來進行，在「資訊擷取(information retrieval)」上的幫助是有的，但在尋找「有用的(useful)」及「驚喜的(serendipitous)」的資源上的幫助卻是比較小的[4][5]。本論文的目標，以複雜網路與概念認知的研究方向，提高「標籤推薦系統(tag recommendation system)」應用上的價值。

1.2 研究背景

「分眾分類」這類網站，流行的原因，主要是使用者不需要具備專業的分類知識，分享者的資源可以被觀看者輕易的找到。這是由於「標籤系統」的使用方式，使用者們在使用上有著比較小的認知負擔，因而廣為流行。「標籤系統」的使用簡單，鼓勵了人們整理資源，藉由分享使得整個網站成為一個資料的管理系統，甚至成為一個知識的系統。也由於這種多人參與的模式，以前由專家來做的資料分類(taxonomy、categorization)或是「後資料(metadata)」的建立，也可以參考到來自許多各種不同人的角度與看法[3]。在這些系統中，可以從巨觀地來看所有使用者對於資訊的共識，也可以微觀地從個別使用者對於資訊的觀點。分享者對於要分享的資料給予任意數量，任意內容的標籤。這個動作可以視為個人式的索引化，也就是加上關鍵字。而觀看者，使用自己的認知，將所認為的可能的標籤輸入該系統查詢以獲得資料。

對於觀看者而言，若是沒有「標籤推薦」，「分眾分類」網站亦不過是個配有內部搜尋功能的網站，因此「標籤推薦」的存在，使得「分眾分類」網站有別於搜尋網站，尤其在「瀏覽」這個動作上，觀看者亦不知道自己要的資源是什麼，必須仰賴推薦找到觀看者有興趣的資源，同樣是屬於「協同合作(collaborative)」的網站，例如 wikipedia。如

果沒有關鍵字，如果首頁沒有這些導引連結 (其實這些連結就屬於推薦)，觀看者要如何從瀏覽獲得資訊？因此推薦系統在「協同合作」(wiki, blog)的網站有它的重要性，可以維持人們不斷上去瀏覽、更新。在「分眾分類」的網站上亦是。

推薦系統不光是在「分眾分類」網站上出現，它很早就出現在市場的應用，舉個在網路上的應用，如購物網站或是網路書店的熱賣商品，就推薦銷售量大的商品、或是推薦新商品。比較進階一點的推薦，是網路書店(Amazon)的「買了這本書的人，也買了以下的書」。甚至是網路新聞(yahoo)的「對這則新聞有興趣的人，也對以下新聞有興趣」，這種推薦也是存在的。這些來自「資訊擷取」領域所採用的「協同過濾」的方法，也許使用了「明確評分(explicit rating)」的方式請使用者對該資源做評分，以獲得使用者的評價，或者使用「暗示評分(implicit rating)」的方式，計算使用者是否根據推薦點擊，或是計算使用者過去所選擇的資源來做推薦。在「分眾分類」這類網站上的推薦系統可以明顯看見的有，推薦最多人看過的資源，推薦最新加入的資源，但沒有如前所述推薦相關聯的，但我們可以預期推薦的方式會慢慢地多出來。這是可以預測的進程，先滿足「找得到」(找到的 resource 越多越好)，再滿足「找得好」(越滿足使用者需求的越要排在前面)。

比較「協同合作」與「分眾分類」網站的構成，可以得知「協同合作」的網站是一個「二分關聯網路(bipartite network)」。而「分眾分類」的網站上是一個「三分關聯網路(tripartite network)」[6]，而這個差異對於推薦系統的設計的影響到底為何？目前沒有人做過比較。

在「資訊擷取」的方法裡，將「二分關聯網路」轉換成每人的物-物網路，比對兩者的「相似度(similarity)」。「分眾分類」的三分關聯網路，由於加入了標籤，使得情況複雜多了。應該可以預期的是有了標籤的幫助，找尋相似度應該要更容易，但因為標籤是「自由格式(free-form)」的格式，有文字上本質的多字一義，一字多義的問題，也有使用者使用標籤的技巧的問題，甚至使用者對於關鍵字的認知問題，反而讓「分眾分類」的推薦系統的困難度加大。Hotho, Jäschke, Schmitz 等人[2] (2006)因應三分關聯網路的特性，借用使用結構分析的 PageRank 所設計出來的 FolkRank，則是給予了一個新的思考方向，在推薦系統應用上[1]，在「檢全率(recall)」及「檢準率(precision)」方面勝過了「最常用標籤(most popular tags)」以及「協同過濾」的方法。然而，接下來要面臨的是

在「有用的(useful)」及「驚喜的(serendipitous)」這個方向提供幫助，想從資訊擷取這個領域獲得幫助的機會卻很少[4]。於是，藉由其他領域對於「有用的(useful)」及「驚喜的(serendipitous)」的研究，希望能夠有所突破。

由「分眾分類」的三分關聯網路所能形成的三種網路，人-人網路(之後簡稱為人際網路)、標籤-標籤網路(之後簡稱為標籤網路)以及物-物網路(之後簡稱為物件網路)，每個網路的網路性質當然是令人好奇的[7]。正如複雜網路裡面，小世界網路所帶給人們的驚奇，我們也從近來的研究中注意到，「分眾分類」所形成的三種網路正巧皆符合小世界網路裡「無尺度網路(scale-free network)」的「冪次律(power law)」。從複雜網路性質的研究方向，研究「分眾分類」的「有用的(useful)」及「驚喜的(serendipitous)」的推薦，可以借用的是重疊社群結構的研究。在社群結構中，處在重疊位置的人，其重要性比社群內其他人還要重要[8]，因為處在重疊位置的人通常與其他人的連結屬於弱連結，而帶來意外幫助的，通常來自於關係為弱連結的人。在這顯而易見的關聯之下，我們將會試著找出重疊的標籤及物件，分析其可能的重要性為何，並且探討對於標籤推薦系統的幫助為何。

1.3 研究目標

本論文將會以線上既有之網站 CiteULike 做為分析的資料來源。本論文的研究目標列述如下：

1. 將網站資料做網路性質分析，針對小世界網路幾項特徵性質(分隔度、群聚度、分支度、度分布)做驗證。

2. 討論標籤在群聚度與分支度上所代表的意義。

3. 將推薦過程的兩個步驟，找鄰居與選標籤，分別應用「協同過濾」與我們提出的演算法，比較其推薦的結果，討論其適用之狀況。

1.4 研究流程與論文架構

接下來的章節安排如下：

第二章文獻探討，會回顧「社會性的使用者資源分享系統(social resource sharing

system)」的現況，介紹「分眾分類」的一般架構及其三分關聯網路的性質，「分眾分類」正式的數學模型[1]。這個數學模型會在接下來的各個章節中使用到。接下來介紹推薦系統中以「協同過濾」為基礎的演算法，「協同過濾」演算法會是本論文在做比較的標準。接下來介紹小世界的網路的結構特性，小世界網路在「分眾分類」的相關文獻，以及複雜網路中派系與重疊社群結構分析的演算法。本論文所提之演算法即由這些複雜網路的概念所延申。

第三章研究方法與實驗設計，會說明資料來源之網站提供之資料結構，資料處理的方法，對各個演算法評量的方法。

第四章研究結果與分析，先對資料所形成的標籤網路的複雜網路性質做討探，以驗證其複雜網路結構。接下來將各種演算法應用在資料來源，並將其所得之推薦結果，對其檢全率(recall)、檢準率(precision)做比較，以及各演算法之間相似度計算，再探討推薦結果的「有用的(useful)」及「驚喜的(serendipitous)」。

第五章結論，將整理實驗中的結果，說明其中的原因。提出重疊社群結構在「分眾分類」中的重要性及其限制之處，最後提出未來可應用或是發展的地方。

二、 文獻探討

2.1 社會性的使用者分享系統 Social sharing system

社會性的使用者資源分享系統(Social sharing system)，常見的有 CiteULike (http://www.citeulike.org)、Flickr (http://www.flickr.com)、del.icio.us (http://del.icio.us)。CiteULike 可以讓使用者管理、分享讀過的論文，Flickr 可以讓使用者管理、分享照片，del.icio.us 是讓使用者管理、分享網路書籤。雖然這些網站所分享的資源不同，但是他們的運作方式都是相同的：使用者登入之後，將要分享的資源上傳，並且貼上標籤。或是瀏覽別人的標籤與資源的關聯。讓使用者願意將資料放在網路上的原因，一是不用加裝其他程式，二能在有網路的狀態下，隨時隨地管理、瀏覽自己的標籤與資源的關聯，不受限於必須使用自己的電腦。

標籤系統成功的應用，從貼標籤行為來看，Rashmi Sinha [9](2005)認為，「貼標籤的動作減少了做決定的動作(決定對的分類)，對於大多數的人而言是免除了分析抉擇的過程」[1]。因此可以鼓勵使用者整理及管理自己所擁有的資源。

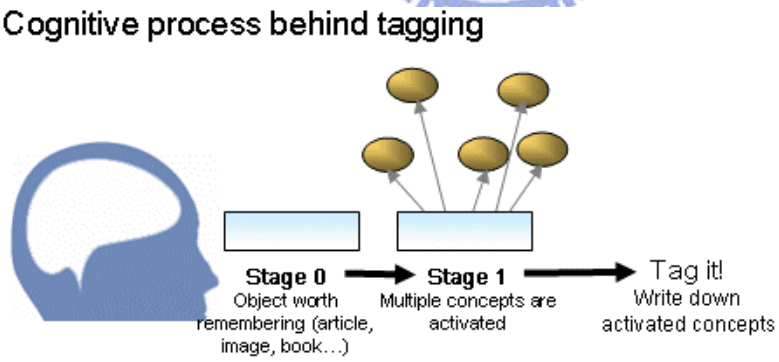


圖 1 貼標籤背後的認知過程[9]

[1] 原文：”tagging eliminates the decision - (choosing the right category), and takes away the analysis-paralysis stage for most people.”

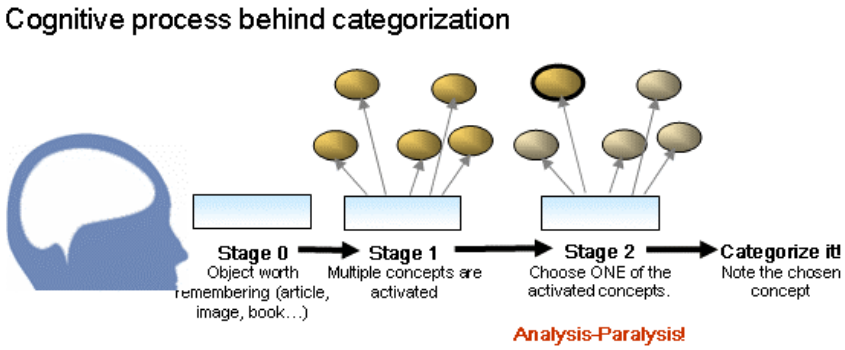


圖 2 分類背後的認知過程[9]

Hsieh et al {Hsieh, 2008 #48}(2008)認為下列四種資源適合標籤系統(1)非文字格式，例如圖片，(2)具有多重概念的檔案，例如論文，(3)需要經常搜尋重訪的資料，例如論文參考資料，(4)數量很大又缺乏適合檔名的檔案。因此，原本不易分類的資源，在標籤系統上可以經由標籤與搜尋(search)的使用，降低整理與資訊提取(retrieval)的困難度，提高使用者貼標籤的意願。

貼標籤且分享是一個鬆散社群互利的活動，使用者不只是從整理自己已有的資源獲得益處，也可以看其他使用者貼完標籤的資源而獲得益處。例如在 CiteULike 網站上，看有興趣的標籤之下，有哪些我沒看過的論文，或是我看過的論文，別的使用者是否有貼不一樣的標籤，該標籤是否為我未曾想過的概念。使用者能從「社會性的使用者資源分享系統」獲得重整個人知識、發現未知且有趣的事物(uncover the unknown interesting thing)，這是搜尋網站及入口網站做不到的。

2.2 分眾分類 Folksonomy

標籤系統是「非階層(non-hierarchical)」且「非互斥(non-exclusive)」的[6]，由於「社會標籤系統(social tagging system)」的本意含有分類(taxonomy)與眾人(folk)共同合作的意涵，因此「分眾分類」[10]的名詞出現，確立它與眾不同的特點。

Thomas Vander Wal 是被認為是「分眾分類」Folksonomy 這個字的創始者，在他個人的網頁中，給予了以下的定義[11](2004)：

「『分眾分類』是個人為了將來取用的需求，自由地對資訊及物件(任何可以用 URL 表示的東西)貼標籤的行為所形成的結果。貼標籤的動作是在一個社會環境中所完成的(social environment 對其他人而言是分享且開放的環境)，『分眾分類』是由資料消費者(或

叫做讀者)貼標籤的動作所創造出來的產物」[2]

Thomas Vander Wal 本身並沒有對「分眾分類」多加限制，但一般的研究認為「分眾分類」具有的以下幾個特性[6]，(1) 「非階層」(non-hierarchical)，(2) 「非互斥」(non-exclusive)，(3)「使用者創造」(user-created)，(4)「自由關鍵字」(free-keywords)。

Lambiotte and Ausloos [6](2005)的研究認為「分眾分類」可視為是一個三分關聯網路，整個系統中三個組成「使用者」user，「標籤」tag，「物件」item 的關係表達如下：

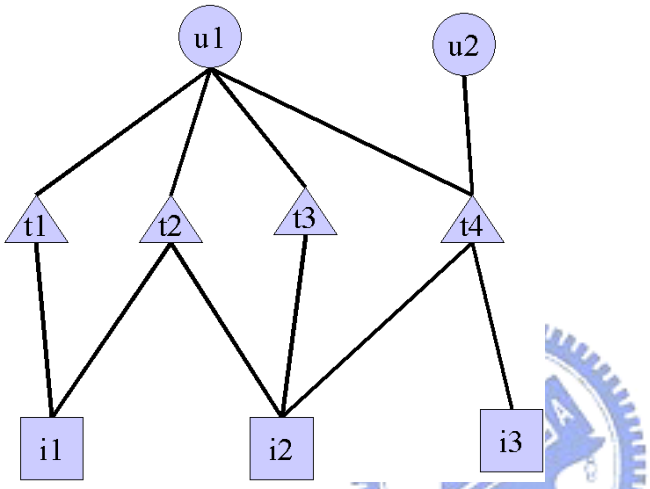


圖 3 在「分眾分類」的使用者(user)、標籤(tag)、物件(item)

u：user　t：tag　i：item

為了分析的方便，通常將「三分關聯網路(tripartite network)」投影成「二分關聯網路(bipartite network)」或「單一關聯網路(unipartite network)」。例如，要降階成「(人-物)二分關聯網路」，即針對所有的標籤計算且建立與使用者、物件的關聯，接著再把標籤拿掉。

[2]原文：“Folksonomy is the result of personal free tagging of information and objects (anything with a URL) for one's own retrieval. The tagging is done in a social environment (usually shared and open to others). Folksonomy is created from the act of tagging by the person consuming the information.”

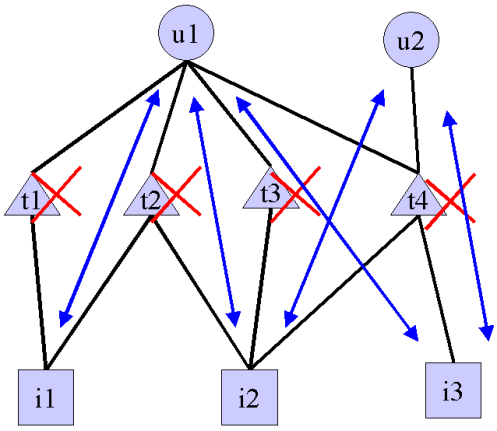


圖 4 人-物投影二分關聯圖

「分眾分類」的圖形化/視覺化分析即可由三分關聯 > 二分關聯 > 單一關聯做出不同層次及不同面向的結構分析。

在「分眾分類」的推薦系統的演算法，Hotho, Jäschke, Schmitz[2]提出的 Folksonomy-Adapted PageRank，是改編自 PageRank 的方法，屬於圖形為基礎(graph-based approach)。該演算法的精神，是這樣的：

「基本的精神是一個物件如果被重要的人貼上重要的標籤，那麼該物件就變得重要。這個觀念同時對於標籤、人也都成立」[3]

Folksonomy-Adapted PageRank 是全域的推薦，同時因為「分眾分類」的網路是「無向(undirected)」，作者認為這個因素造成「排行(ranking)」的結果接近節點的「度次數(degree count)」計算的排行(ranking)結果。於是，同篇論文 Hotho 另外提出一個同樣概念但不一樣的演算法，叫做 FolkRank，這兩者的精神一樣，但是 FolkRank 可利用「偏好向量(preference vector)」定義主題(topic)，這樣可應用在主題指定(Topic-specified)的時機，例如個人化的推薦。並且利用三分關聯網路的對稱性，該演算法可以推薦使用者、標籤及物件。

Jäschke et al [1] (2007)提到，在「分眾分類」裡的推薦系統，可以提供幾項服務，

[3] 原文：”The basic notion is that a resource which is tagged with important tags by important users becomes important itself. The same holds, symmetrically, for tags and users.”

增加物件被註釋的機會、提醒使用者物件的意義、讓使用者間用的字彙能夠得到統一[4]。綜合 Herlocker et al [4] (2004)對於推薦系統看法，「創新性(Novelty)」以及「驚喜性(Serendipity)是推薦系統比較需要努力的地方。我們感覺到尤其在可做為知識管理平台的「分眾分類」，使用者在這方面的需求，會比其他類型網站高。

2.3 協同過濾 Collaborative filtering

協同過濾 Collaborative filtering, (CF)的技術在推薦系統是最常被使用的。其基本的方法是收集每個使用者對於物件的評價，將之視為一個使用者與物件的「加權二分關聯網路(weighted bipartite network)」，其中權重是使用者對物件的評價。將二分關聯網路投影成人際網路(user-user network)與物件網路(item-item network)，系統經由相似度的比較，推薦物件給使用者。使用者被推薦的流程(scenario)為(1)使用者選擇物件(item)後，系統再推薦其他相似的物件，或系統主動推薦一些相似於使用者曾選擇過的物件。(2)系統主動推薦與使用者相似的其他使用者評價高的物件。

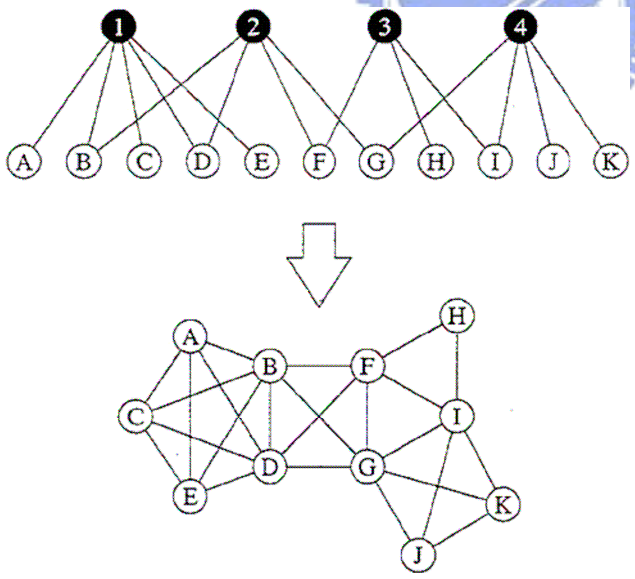


圖 5 二分關聯圖投影至單一關聯圖

[4] 原文：”Increasing the chances of getting a resource annotated, reminding a user what a resource is about and consolidating the vocabulary across the users.”

在「協同過濾」 的方法裡，把上述的概念表示成一個(人-物)的矩陣(user-item matrix)。Jäschke et al[1]使用以下的記號方式(notation)。對於一個系統中，有 m 個使用者(user)與 n 個物件(item)，其全體使用者記錄(user profile)的矩陣為 $\mathbf{X} \in \mathbb{R}^{m \times n}$，可以用「列向量(row vectors)」來表示：

$$\mathbf{X} := [\vec{x}_1, ..., \vec{x}_m]^\top \text{ with } \vec{x}_u := [x_{u,1}, ..., x_{u,n}], \text{ for } u := 1, \ldots, m,$$

$x_{u,o} \in \mathbb{R}$。$x_{u,o}$ 是指 user u 對於 item o 的評價。這種分解(decomposition)得到的是「使用者為基礎的協同過濾(user-based collaborative filtering)」。接下來用矩陣 $\mathbf{X}$ 來做計算。先決定一個 k 的值，k 的值是用來決定要選幾個相似的使用者。N_u^k，叫做「鄰居(neighbors)」，這是個集合，它就是系統中與 user u 最接近的 k 個 user。N_u^k 的數學表達式為 $N_u^k := \arg\max_{v \in U}^{k}(\mathrm{sim}(\vec{x}_u, \vec{x}_v))$，其中 arg_max 是個「函式(function)」，式子中的上標 k，就是指定回傳 k 個相以度最高的 user，sim 是相似度的函式(function)，一般是用 cosine similarity measure。接下來，再決定一個 n 的值，n 的值決定推薦系統的推薦清單裡幾個物件。推薦清單的排序依鄰居評價次數最多的放最前面，以此類推。

在應用「協同過濾」到「分眾分類」的環境時，因為三分關聯網路的特性需要修正，Jäschke 等人[1]修正的式子在第三章會一起列出。

2.4 小世界與分眾分類 Small world and Folksonomy

在 Golder 與 Huberman[8](2005)的研究裡發現，少數的人使用的標籤數很多，大多數的人使用的標籤數很少。其分佈符合冪次律(power law)。這個性質在複雜網路裡無尺度網路(scale-free network)的其中一個性質相近。Cattuto 等人[12](2007)研究「分眾分類」所形成的三分關聯網路並提出修正版的「特徵路徑長度(characteristic path length)」及「群聚係數(clustering coefficient)」，經過觀察後發現「分眾分類」具有小世界網路的特徵：低分隔度(low characteristic path length)及高群聚度(high clustering coefficient)，並且，隨著網路的成長，分隔度依然很低且群聚度依然很高。

Cattuto 等人[12](2007)也針對「分眾分類」的「語意性質(semantic property)」做了研究，將「分眾分類」轉化為「標籤共同出現網路(tag co-occurrence network)」。它的形

成方法，是對於每一個使用者，看標籤與物件兩種節點。如果 tag1 與 tag2 皆與 item1 有關，就將 tag1 與 tag2 建立連結，標籤之間連結的權重可用該連結建立的次數。這種加權網路可以視為整體使用者對於物件的認知投票。正式的「標籤共同出現網路」的定義如下：

$$W(t1,t2) := \{(u,r) \in U \times R \mid (t1,u,r) \in Y \wedge (t2,u,r) \in Y\}$$

連結權重是 $w(t1,t2) := \mid W(t1,t2) \mid$

而且，更進一步定義標籤的「強度(strength)」為：

$$s_t := \sum_{t1 \neq t2} w(t1,t2)$$

Cattuto 等人[12]研究發現，由「標籤共同出現網路」的「累積強度分布(cumulative strength distribution)」就可以偵測 spamming 的活動，而且發現不同網站的「累積強度分布」非常類似。對於推薦品質好又富創新性(Novelty)及驚喜性(Serendipity)的推薦系統，必須要過濾掉人為的操縱；具有推薦創新性的推薦系統對於新物件的敏感度要比一般的高，自然很容易受到 spam 的干擾，如果依照人類行為模式符合小世界網路特徵，並且可由此特性觀察意見成長動態及偵測 spam，Cattuto 等人[12]的發現讓推薦系統有一個修正的方式避免人為操縱。

2.5 派系過濾法 Clique percolation method

在複雜網路「分群(clustering)」的技術裡面，Palla 等人[8]提出可以找尋重疊(overlapping)結構的方法為「派系過濾法(Clique percolation method) (CP)」，標籤是概念的描述，通常一個物件是不會只屬於一個標籤，而是屬於一組標籤，每個人因為看法的不同，也會有不同組的標籤來描述。在不同人的標籤之中，重覆的標籤也就是標籤網路中重疊的標籤。把標籤網路當成人際網路，因此使用派系過濾法來快速找尋重疊標籤且決定重要的標籤。

派系過濾法的步驟如下：

1. 找尋 clique

(1) Let MAX_DEGREE = maximum degree of network.

(2) Let CLIQUE_MAX_SIZE= possible maximum size of complete subgraph in the network. This size can be guessed from MAX_DEGREE.

(3) Let s = MAX_DEGREE, do following:

(4) Select a node in the network do following

(5) Find all cliques of size s include the node you just selected. Remember the cliques found.

(6) Remove the node you just selected from the network and remove all links link the node from the network.

(7) Return to (4) there is a node in the network.

(8) if s > 1, restore the original network, then let s = s-1，go back to (4)。If s <= 1，all cliques are found。

2. 找尋 k-clique-communities

使用「派系重疊矩陣(clique-clique overlap matrix)」，每列(row)代表一個派系，每行(column)也代表一個派系，所以它是個沿對角線對稱的矩陣。非對角線上的元素，是兩個派系之間的共用節點數。在對角線上的元素，則填入該派系的大小(size)。

找尋「k 派系社群(k-clique-communities)」時，將對角線上小於 k 的元素設為 0，非對角線上小於 k-1 的元素設為 0，即可找到有重疊的 k 派系(k-clique)。

三、 研究方法

3.1 基本性質

接下來我們會使用 Hotho et al[2]提出「分眾分類」的正式模型的定義：

Definition 1. A folksonomy is a tuple $F := (U, T, R, Y, \prec)$ where

- U,T, and R are finite sets, whose elements are called user, tags and resources, resp.,
- Y is a ternary relation between them, i.e., $Y \subseteq Y \times T \times R$, called tag assignments (TAS for short), and
- $\prec$ is a user-specific subtag/supertag-relation, i.e., $\prec \subseteq U \times T \times T$, called subtag/supertag relation.

本論文並不研究使用者的 super-tag/sub-tag 的關係，因此 $\prec = \phi$。所以「分眾分類」的模型可記為 $\mathbb{F} = (U, T, R, Y)$。

以下說明各個符號的意義：

- user 的出現次數：將各 user 在 TAS 出現的次數加總。
- user 的總數：$|U| := |\{u \in U \mid (u, t, r) \in Y\}|$。
- tag 的出現次數：將各 tag 在 TAS 出現的次數加總。
- tag 的總數：$|T| := |\{t \in T \mid (u, t, r) \in Y\}|$。
- item 的出現次數：將各 item 在 TAS 出現的次數加總。
- item 的總數：$|R| := |\{r \in R \mid (u, t, r) \in Y\}|$。

3.2 資料來源

資料集(Dataset)為 CiteULike 網站所提供的資料庫[5]，是以文字檔形式，每列(row)提供的資訊為 article id，user(MD5 hashed)，TAS 的時間，標籤名。每一個資訊用分隔符號"|"分開，資料取的時間為 2008-02-12 04:44:18.648815+00。

```
42|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:25:05.373798+00|networks
42|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:25:05.373798+00|metabolism
42|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:25:05.373798+00|barabasi
42|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:25:05.373798+00|ecoli
…
2364841|469b74cc00a337639d76cb96aec58bda|2008-02-12 04:43:23.635387+00|buy
2364841|469b74cc00a337639d76cb96aec58bda|2008-02-12 04:43:23.635387+00|apple
2364841|469b74cc00a337639d76cb96aec58bda|2008-02-12 04:43:23.635387+00|macbook
2364841|469b74cc00a337639d76cb96aec58bda|2008-02-12 04:43:23.635387+00|ukbuy
2364845|c42f76025ff94b72c082195c7c79d65d|2008-02-12 04:44:18.648815+00|teacher
2364845|c42f76025ff94b72c082195c7c79d65d|2008-02-12 04:44:18.648815+00|education
```

其中，平均每一個使用者所使用的標籤數為 $\frac{151142}{22363} = 23.67$，平均每一個使用者所指定的文件數為 $\frac{715016}{22363} = 36.84$。我們在實驗結果的部份也會呈現出現次數分佈圖。表格 1 簡單列出 TAS、|U|、|T|、|R|各項的數量以說明資料集的大小及屬性。

表格 1 CiteULike 資料集性質

	TAS	|U|	|T|	|R|
總數	2,369,141	22,363	151,142	715,016

3.3 標籤網路 tag network 的形成與複雜網路特性

在「分眾分類」裡，標籤網路的形成可以有很多方式，這裡先討論兩個最直覺的方式，一個是經由標籤-物件(tag-item)，一個是經由標籤-人(tag-user)。

以標籤-物件所形成的二分關聯網路為基礎產生標籤-標籤的單一關聯網路(tag-tag

[5] http://www.citeulike.org/faq/data.adp

unipartite network)。與 Cattuto[12]提的標籤共同出現網路(tag co-occurrence network)不同的是，在這裡，user1 指定 tag1 給 item1，user2 指定 tag2 給 item1，tag1 與 tag2 之間便視為連結存在。

我們這裡定義標籤網路(物)(tag network (item))的定義為

$$TNr = \{(t1,t2) \mid t1,t2 \in T \times T, (t1,r) \in Etr, (t2,r) \in Etr\}$$

其中 $Etr = \{\{t,r\} \mid (u,t,r) \in Y\}$

標籤網路(物)的密度(Density)為 $\frac{|TNr|}{|T| \times (|T|-1)/2}$

另一個，以標籤-人所形成的二分關聯網路為基礎產生標籤-標籤的單一關聯網路。

標籤網路(人)(tag network (user))的定義為

$$TNu = \{(t1,t2) \mid t1,t2 \in T \times T, (t1,u) \in Eut, (t2,u) \in Eut\}$$

其中 $Eut = \{\{u,t\} \mid (u,t,r) \in Y\}$。

標籤網路(人)的密度(Density)為 $\frac{|TNu|}{|T| \times (|T|-1)/2}$

以上兩種，均是屬於整體「分眾分類」的標籤網路。

在我們所使用的符號中 Y 指的就是 TAS，也就是資料集中每一列(row)，假設我們的資料集有 4 個 TAS，

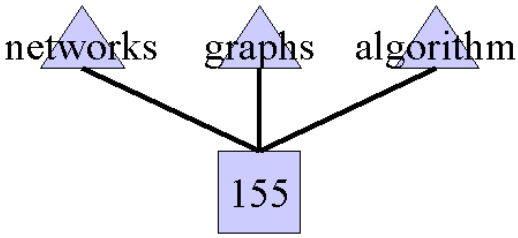
```
155|cb2e1f0222c692723674c4e679020f0b|2004-11-10 17:13:30.487332+00|networks
155|cb2e1f0222c692723674c4e679020f0b|2004-11-10 17:13:30.487332+00|graphs
155|a47d7aa28bdc3bef2ed4dcbcb2a8b5f2|2005-01-19 09:50:27.619901+00|networks
155|a47d7aa28bdc3bef2ed4dcbcb2a8b5f2|2005-01-19 09:50:27.619901+00|algorithm
```

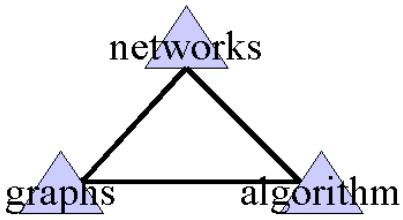
那麼 Etr 是一個集合，內容是(networks,155)、(graphs,155)、(algorithm,155)。標籤網路(物)TNr 也是一個集合，代表的是網路中有連結的標籤對(tag pair)，內容是(networks, graphs)、(networks, algorithm)、(graphs, algorithm)。標籤網路(物)的密度就等於

$$\frac{3}{3\times(3-1)\div 2}=1$$

Etr = {(networks,155), (graphs,155), (algorithm,155)}

TNr = {(networks,graphs), (networks, algorithm), (graphs,algorithm)}

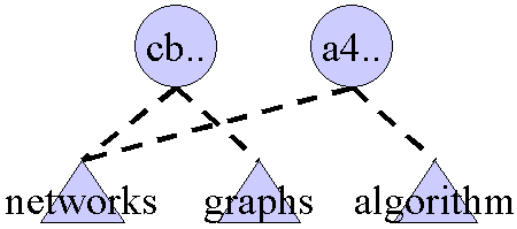


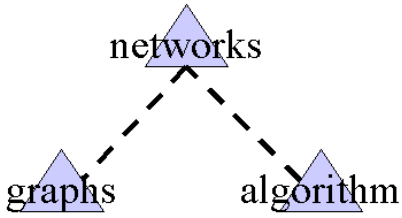


Eut 的內容是(cb2e1f0222c692723674c4e679020f0b,networks)、(cb2e1f0222c692723674c4e679020f0b, graphs)、(a47d7aa28bdc3bef2ed4dcbcb2a8b5f2, networks)、(a47d7aa28bdc3bef2ed4dcbcb2a8b5f2, algorithm)。標籤網路(人)TNu 的內容是(networks, graphs)、(networks, algorithm)。標籤網路(人)的密度的等於$\frac{2}{3\times(3-1)\div 2}=\frac{2}{3}$

Eut = {(cb.., networks), (cb.., graphs), (a4.., networks), (a4.., algorithm)}

TNu = {(networks,graphs), (networks, algorithm)}





接下來，我們定義個人的標籤網路，定義如下：

$$TNr(u_i) = \{(t1, t2) \mid t1, t2 \in T \times T, (t1, r) \in Etr(u_i), (t2, r) \in Etr(u_i)\}$$

其中 $Etr(u_i) = \{\{t, r\} \mid (u_i, t, r) \in Y\}$。

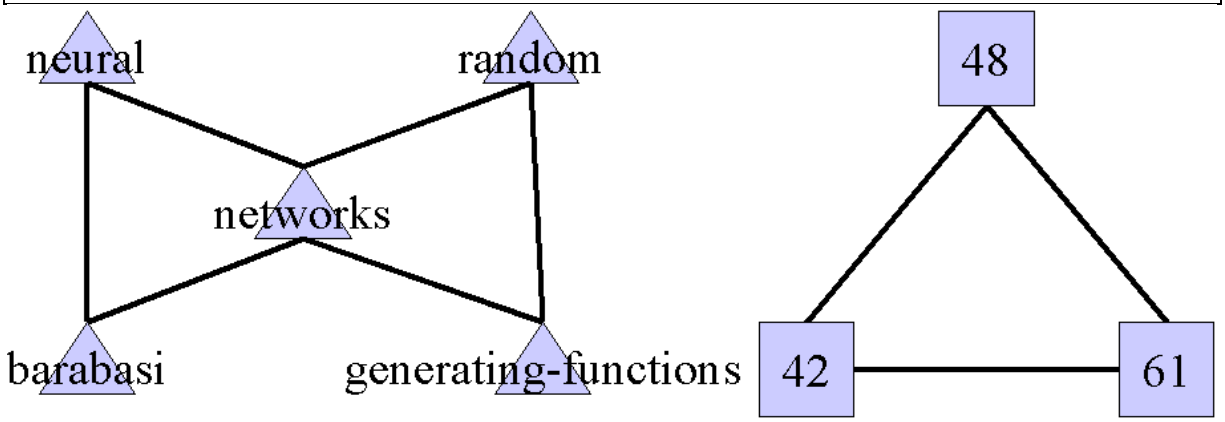
個人的物件網路，定義如下：

$$RNt(u_i) = \{(r1, r2) \mid r1, r2 \in R \times R, (r1, t) \in Etr(u_i), (r2, t) \in Etr(u_i)\}$$

其中 $TNr(u_i)$，可以視為個人的認知網路，也就是「個人對於物件的觀點」"personal point of view for items"； $RNt(u_i)$，則是個人有興趣的目標網路，也就是「個人的有興趣的物件」"personal interesting on items"。

我們以底下的 8 個 TAS 舉例子，使用者個人的標籤網路的內容是(networks, barabasi)、(networks, random)、(networks, generating-functions)、(random, generating-functions)、(neural, networks)、(neural, barabasi)。其個人的物件網路則是(42,48)、(42,61)、(48,61)

```
42|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:25:05.373798+00|networks
42|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:25:05.373798+00|barabasi
48|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:29:45.443118+00|networks
48|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:29:45.443118+00|random
48|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 02:29:45.443118+00|generating-functions
61|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 21:29:27.45403+00|neural
61|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 21:29:27.45403+00|networks
61|61baaeba8de136d9c1aa9c18ec3860e8|2004-11-04 21:29:27.45403+00|barabasi
```



3.4 派系篩檢 Clique Filtering 的標籤推薦清單

我們考慮推薦標籤的流程是使用者登入，輸入物件，準備要貼上標籤時，系統給予推薦清單。我們參照協同過濾(collaborative filtering)設計了派系篩檢(clique filtering)這個方法。

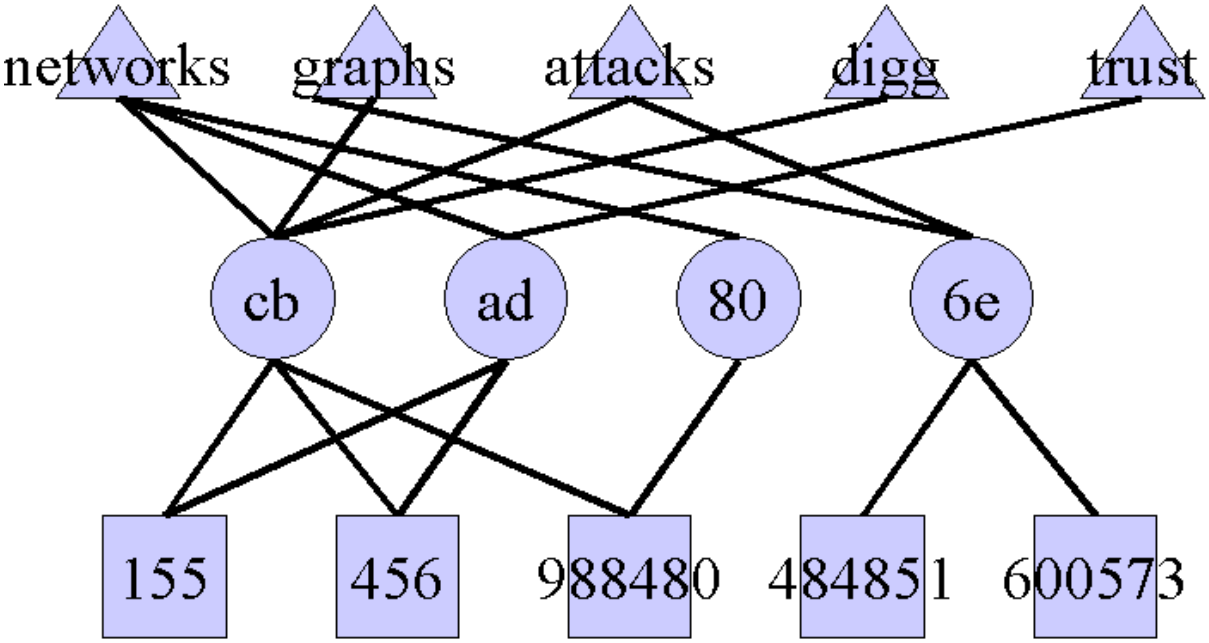
前人研究中的標籤推薦清單的產生，首先尋找鄰居，再從鄰居尋找標籤。使用者的鄰居的來源有兩個，一個是由標籤，一個由物件。如果經由標籤，則可以找到有相同看法的鄰居；如果經由物件，則可以找到有相同興趣的鄰居。因此，鄰居的產生方法有兩種，

$\text{Neighbor}\_R(u_i) = \{u_j \in U \mid \exists r : (u_i, r)(u_j, r) \in Eur, i \neq j\}$ ，其中 $Eur = \{\{u, r\} \mid (u, t, r) \in Y\}$。排序依 $RNt(u_i)$ 重疊數由高至低排列。

$\text{Neighbor}\_T(u_i) = \{u_j \in U \mid \exists t : (u_i, t)(u_j, t) \in Eut, i \neq j\}$ ，排序依 $TNr(u_i)$ 重疊數由高至低排列。

我們以底下的 9 個 TAS 舉例子，Neighbor_R(cb)是 ad, 80 而 Neighbor_T(cb..)是 6e, ad,80。

```
155|cb2e1f0222c692723674c4e679020f0b |2004-11-10 17:13:30.487332+00|networks
155|cb2e1f0222c692723674c4e679020f0b |2004-11-10 17:13:30.487332+00|graphs
456|cb2e1f0222c692723674c4e679020f0b |2004-11-15 10:31:41.245268+00|attacks
988480|cb2e1f0222c692723674c4e679020f0b|2007-02-15 10:28:33.943496+00|digg
155|ad6241687c8722ee0bafac52747ff8cf |2004-12-04 07:28:02.564785+00|networks
456|ad6241687c8722ee0bafac52747ff8cf |2005-02-11 02:43:36.568529+00|trust
484851|6edad60e9851a33cf5280cd2a4f5276b|2007-02-18 00:36:00.842007+00|graphs
600573|6edad60e9851a33cf5280cd2a4f5276b|2007-02-09 21:47:12.306949+00|attacks
988480|808eb76a3ebc6efe03feae67607af389|2006-12-11 11:22:41.529916+00|networks
```



推薦清單 $\tilde{T}(u,r)$ 產生的方法，主要是依照「派系重疊(clique overlapping)」的次數排列，我們簡稱做「派系排序(Clique Sorting)」。

(1) 在使用者 u_i 的 $Neighbor\_R(u_i)$ 或 $Neighbor\_T(u_i)$ 中，找出鄰居們對該物件所使用的標籤們。

$$Neighbor\_Recommends(u_i, r_k) := \{t \in T \mid (t, u_j, r_k) \in Y, u_j \in Neighbor\_R(u_i)\}$$

(2)先將 $Neighbor\_Recommends(u_i, r_k)$ 與 $TNr(u_i)$ 內的標籤派系分成以下三個集合

$Class1 = Neighbor\_Recommends(u_i, r_k) \cap TNr(u_i)$ 、

$Class2 = Neighbor\_Recommends(u_i, r_k) \setminus TNr(u_i)$ 、

$Class3 = TNr(u_i) \setminus Neighbor\_Recommends(u_i, r_k)$ 。

其中，每個集合中標籤的順序皆以出現次數多寡排列。推薦清單候選

$$Candidate\tilde{T}(u,r) := \{Class1, Class2, Class3\}$$

(3) $$\tilde{T}(u,r) := \arg\max_{t \in T}^{n}(Candidate\tilde{T}(u,r))$$

以 c5846a653b59b10a9a9fd77c8e950bdc 看過的論文 320258 舉例，我們請推薦系統找出他的鄰居 $Neighbor\_R(u_i)$ 有兩個人 a92cf14d4e8997cae4ac64d49f0c6d6e,

b5936fd11f395abfe96b780bca1e127d。他們分別使用以下標籤['usability', 'mentalhealth'], ['user', 'communities', 'sociology', 'motivation', 'online']，推薦系統將這些標籤與使用者的標籤網路 $TNr(u_i)$ 的派系做重疊計算之後的結果是('user', 1)('usability', 1)('motivation', 7)('sociology', 7)('mentalhealth', 7)('online', 7)('communities', 7)('user', 6)('usability', 6)('medical', 6)，在去掉重覆推薦的標籤之後，最後結果是'user', 'usability', 'motivation', 'sociology', 'mentalhealth', 'online', 'communities', 'medical'

3.5 協同過濾 Collaborative Filtering 推薦清單的形成方法

在三分關聯網路裡，要應用協同過濾，Jäschke[1]等人改編了傳統的方法以適應「分眾分類」。為了要找尋使用者的鄰居們，需要知道使用者間的相似度。所以需要將「分眾分類」投影成 2D 矩陣。可以有兩種方向，一是由人-物(user-item)的關係，另一是由人-標籤(user-tag)的關係。

以人-物角度來看，

$$X := \pi_{UR} Y \in \{0,1\}^{|U|\times|R|} \text{ 其中 } (\pi_{UR} Y)_{u,r} = \begin{cases} 1 & t \in T, (u,t,r) \in Y \\ 0 & \text{else} \end{cases}$$

以人-標籤角度來看，

$$X := \pi_{UT} Y \in \{0,1\}^{|U|\times|T|} \text{ 其中 } (\pi_{UT} Y)_{u,t} = \begin{cases} 1 & r \in R, (u,t,r) \in Y \\ 0 & \text{else} \end{cases}$$

推薦清單 $\tilde{T}(u,r)$ 的產生方式是給定一個使用者及一個物件，找出最接近使用者的 k 個鄰居們，在那些鄰居們的物件中，找出最適合的 n 個標籤。

$$\tilde{T}(u,r) := \arg\_\max_{t \in T}^{n} \left(\sum_{v \in N_u^k} \mathrm{sim}(\vec{x}_u, \vec{x}_v) \delta(u,t,r) \right)$$

$$\delta(u,t,r) \begin{cases} 1 & (u,t,r) \in Y \\ 0 & \text{else} \end{cases}$$

arg_max 是一個函式，類似於最大值函式(max function)，上標是指要回傳前幾個。

例如 $\arg\_\max^{3}_{t\in T}(|\{t\in T \mid (u,t,r)\in Y\}|)$ 會回傳出現最多次的前三個標籤的次數。

$\mathrm{sim}(\vec{x}_u,\vec{x}_v)$ 是兩個使用者的相似度，在這裡使用 cosine similarity measure。

3.6 評估 Evaluation

為了要測量推薦清單的好壞，我們採用 Jäschke 在[1]使用的方法 LeavePostOut。每一個在「分眾分類」的使用者，將其所張貼的文章(英文稱為 post，中文稱為「帖」)中任意挑選一篇移出原資料集。而推薦方法的好壞，就是以接近被移出的帖(post)的程度來測量。計算有多接近的方法，則是使用如下的方程式，其意義與資訊頡取(information retrieval)的檢全率(recall)、檢準率(precision), F-measure 一樣。

$$\mathrm{recall}(\tilde{T}(u,r)) = \frac{1}{|U|}\sum_{u\in U}\frac{|\,\mathrm{tags}(u,r)\cap\tilde{T}(u,r)\,|}{|\,\mathrm{tags}(u,r)\,|}$$

$$\mathrm{precision}(\tilde{T}(u,r)) = \frac{1}{|U|}\sum_{u\in U}\frac{|\,\mathrm{tags}(u,r)\cap\tilde{T}(u,r)\,|}{|\,\tilde{T}(u,r)\,|}$$

$\mathrm{F-measure}(\tilde{T}(u,r))$ 我們使用 F_1 與 F_2。

$$F_1(\tilde{T}(u,r)) = \frac{1}{|U|}\sum_{u\in U}\frac{2\times\mathrm{recall}(u,r)\times\mathrm{precision}(u,r)}{\mathrm{recall}(u,r)+\mathrm{precision}(u,r)}$$

$$F_2(\tilde{T}(u,r)) = \frac{1}{|U|}\sum_{u\in U}\frac{(1+\beta^2)\times\mathrm{recall}(u,r)\times\mathrm{precision}(u,r)}{\mathrm{recall}(u,r)+\beta^2\times\mathrm{precision}(u,r)},\quad \beta=2$$

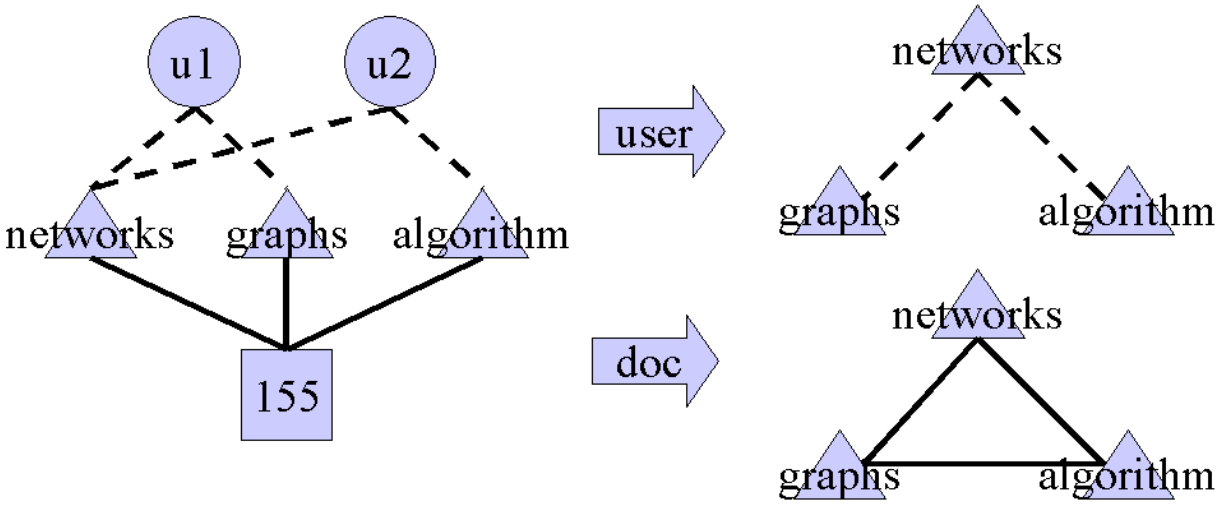
四、 實驗結果

4.1 標籤網路 tag network 的基本性質

從 CiteULike 網站下載回來的資料集，其資料格式為文字檔，每筆資料之間是用分行符號"\n"分開，每筆資料內含四個資料，分隔符號是"|"，依序為論文序號，使用者代號，張貼時間，標籤。如果一篇論文被同一個使用者貼上不同標籤，資料集內會有多筆資料。例如論文 155 被使用者 cb2e1f0222c692723674c4e679020f0b 貼上 networks 與 graphs 兩個標籤，因為張貼的時間都是 2004-11-10 17:13:30.487332+00。

```
155|cb2e1f0222c692723674c4e679020f0b|2004-11-10 17:13:30.487332+00|networks
155|cb2e1f0222c692723674c4e679020f0b|2004-11-10 17:13:30.487332+00|graphs
155|a47d7aa28bdc3bef2ed4dcbcb2a8b5f2|2005-01-19 09:50:27.619901+00|networks
155|a47d7aa28bdc3bef2ed4dcbcb2a8b5f2|2005-01-19 09:50:27.619901+00|algorithm
```

從這樣的資料集中，我們首先想了解的是標籤網路的性質。標籤網路(文件)的建立方法如同 3.3 節介紹，以上面四筆資料來舉例，因為論文 155 的關係，networks、graphs 與 algorithm 三者之間互有連結。而標籤網路(人)，networks 與 graphs 因為使用者 cb2e1f0222c692723674c4e679020f0b 有連結；在同樣四筆資料中，networks 與 algorithm 因為沒有共同使用者所以沒有連結。我們統計整個資料集中，總共出現了 151,142 個不同的標籤。標籤網路(文件)中的連結數量是 4,756,376，而在標籤網路(人)中的連結數有 100,279,514。



為了瞭解群聚度是高還是低，所以計算了網路密度(Density)與群聚度做比較。網路密度的算法是$\frac{\text{實際連結數}}{\text{可能連結數}}$，可能連結數是由標籤網路的節點數算出來，在此也就是由標籤的數量算出來：$\frac{(151142)\times(151142-1)}{2}=11421876511$。因此，標籤網路(文件)的網路密度是$\frac{4756376}{11421876511}=0.00041...$，標籤網路(人)是$\frac{100279514}{11421876511}=0.00877...$。後者的網路密度約是前者的 20 倍高。網路密度在複雜網路中的意義可以視為：「若在標籤網路(文件)任選兩個標籤之中會有直接連結的機率」，所以在標籤網路(文件)任兩點有直接連結的機率是 0.041%，在標籤網路(人)任兩點有直接連結的機率是 0.8%。下一段舉例說明分支度(degree)與群聚度(clustering coefficient)的算法。

以前面四筆資料為例，在標籤網路(文件)中，graphs 的分支度為 2，因為它與其他兩個標籤 networks 及 algorithm 有連結。而在標籤網路(人)，graphs 的分支度則為 1。將整個標籤網路(文件)，每一個標籤的分支度計算下來，平均的分支度 62.93；標籤網路(人)的平均分支度是 1326.95。我們比較一下網路密度不到 1%的標籤網路(人)，若該標籤網路是隨機網路的話，平均分支度是$\frac{100279514}{151142}=663.478...$。而在標籤網路(文件)的群聚度方面，networks 這個標籤與兩個相連的標籤，由這三個標籤所形成的完全圖的連結數是 3，而實際的連結數也是 3，所以群聚度是$3/3=1$；標籤網路(人)的部份，實際的連結數只有 2，所以群聚度是$2/3$。將每個標籤的群聚度平均之後得到標籤網路(文件)的平均群聚度是 0.38 左右而標籤網路(人)是 0.9 左右。要判別一個網路的群聚度，不只是接近 1 就叫高群聚度，例如標籤網路(文件)的群聚度是 0.38 也就是 38%，但是其網路密度是 0.041%，兩相比較就可以知道，連結必定非常集中在某些節點上才使得群聚度能達到 38%。所以標籤網路(文件)及標籤網路(人)的群聚度都是屬於高的。

表格 2 整理出兩個標籤網路的數據互相比較。

表格 2 標籤網路性質

	Tag network (doc)	Tag network (user)
Tag count	151,142	151,142
Link count	4,756,376	100,279,514
Density	0.000416426845048	0.00877960061146

Average degree	62.9391697874	1326.95761602
Average clustering coefficient	0.383398450884	0.904992656084

接下來，我們將論文、使用者與標籤的出現次數分佈圖畫出來如圖 6。次數分佈圖符合冪次律，出現次數多的少，出現次數少的多。越是熱門的文件越多人看，越是熱門的標籤越多人用。

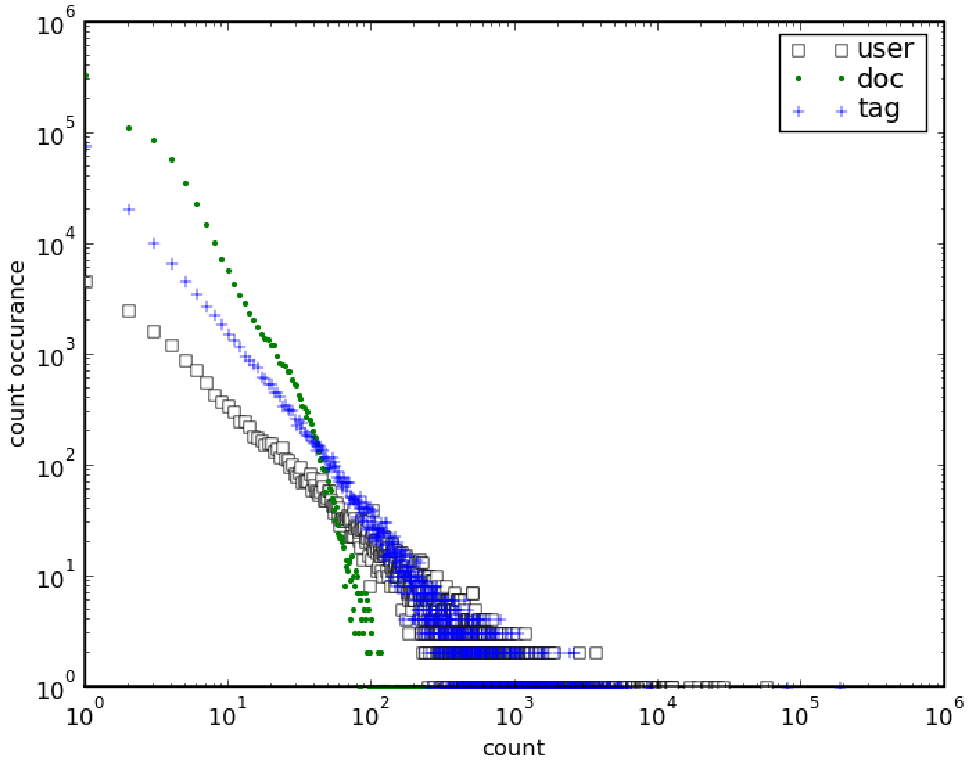


圖 6 使用者次數、標籤次數、論文次數

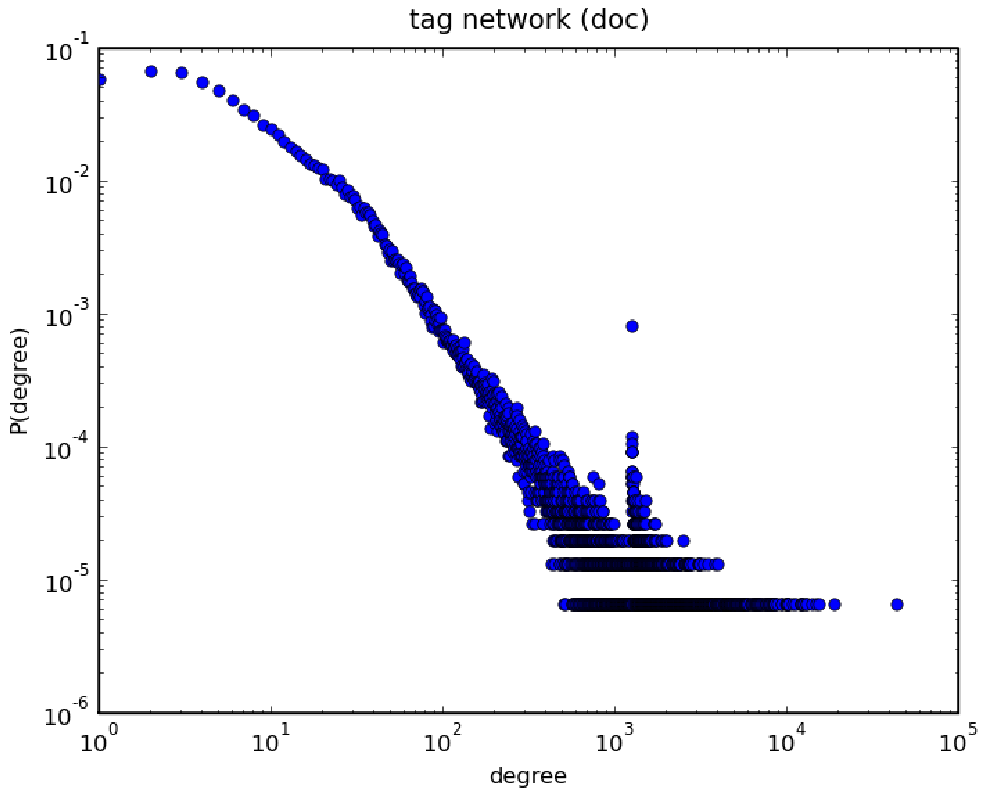


圖 7 標籤網路(文件) v.s.分支度機率

我們將標籤網路(文件)(tag network(doc))的分支度機率分佈圖繪製在圖 7，裡面的節點(node)(也就是標籤)，其分支度(degree)來自於(1) 標籤能描述多少種意思，(2)每一份文件有包含多少種看法。分支度約在 1000~2000 之間有些標籤與眾不同。分支度的次數分配(degree distribution)，符合冪次律，也就是說標籤網路(文件)是一個無尺度(scale-free)的網路。在標籤推薦的時候，雖然推薦分支度大的標籤可以提高準確度，但是有可能這些字是一些過於模糊概念的字，所以我們必須在推薦的時候注意這件事情。以下列出前十名的分支度排行：bibtex-import|43807，no-tag|18999，review|11182，evolution|10072，research|14837，support|15341，learning|7636，govt|14063，network|7604。像前 2 個是系統自動產生給匯入功能的標籤，對正常使用者是無意義的，review、research、support 這些就過於模糊或一般化了。

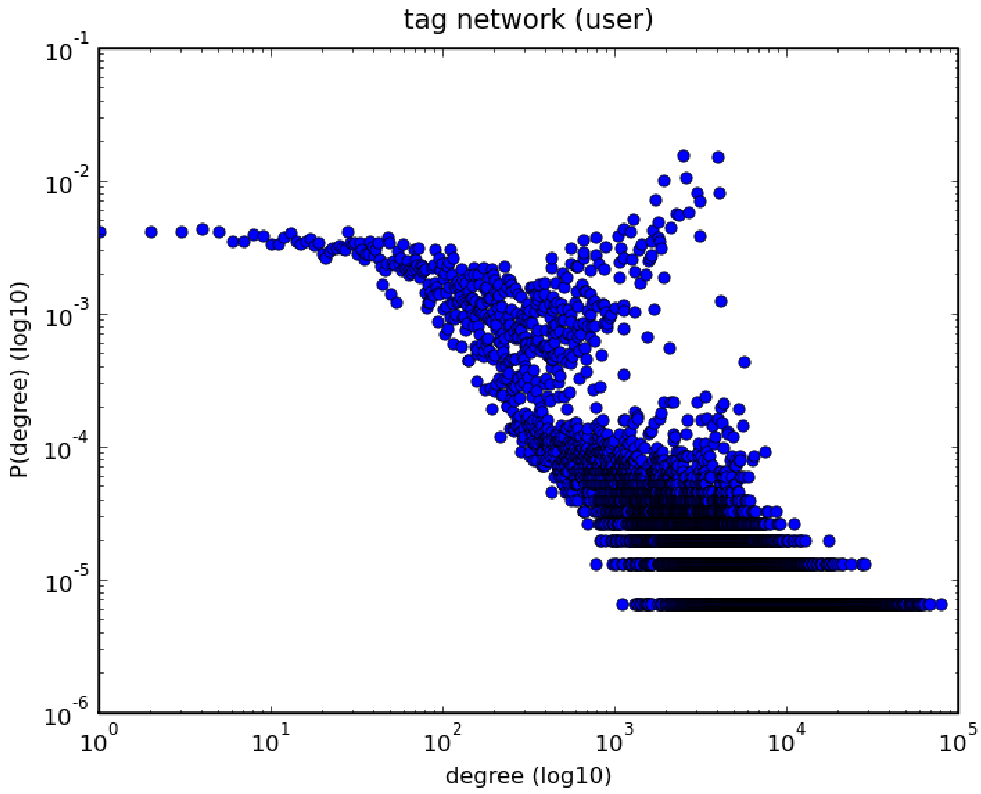


圖 8 標籤網路(人) v.s.分支度機率

在標籤網路(人)(tag network(user))的分支度機率分佈圖(圖 8)，標籤的分支度來自(1)標籤能描述多少種意思，(2)使用者會看多少領域的文件。在分支度 200 到 1000 的區間內，我們也看到了一群與眾不同的標籤。

在標籤網路(人)，分支度前 10 名是：no-tag|78575，analysis|68677，review|67544，statistics|62979，bibtex-import|62347，history|61353，development|59963，model|59575，data|59408，theory|58749。

這兩個結果相互比較，可以看出(1)文件包含多重概念的程度，比使用者擁有多重概念的程度小。這可以從約略從網路的密度(density)感覺到，這裡則是由分支度直接看出來。(2)文件與使用者跨領域的分佈，從一開始比較平緩，接下來以接近冪次的趨勢下降，說明包含少量概念的文件及僅擁有少量概念的使用者是比較少的。例如像 yeast(酵母)這種比較專門的名詞所連結到的論文[6]，這些論文被貼上的標籤數量就比較少。而雖然

[6] 可上網觀察 http://www.citeulike.org/search/all?q=yeast

像 gene(基因)、neural(神經)這種常被用在其他領域的詞所連結到的論文，因為含有跨領域及多種概念，這些論文所連結到的標籤數量就很多[7]。

由於標籤分支度的來源除了上述所說的原因之外，多字一義的同義字、打錯的字、複合字、組合字、縮寫字、不同語言的字…等也都是分支度的來源，因此分支度的數量非常的大。一字多義如 networks 這個字，會與 social 一起使用，成為社會網路[8]，與 computer 一起使用則成電腦網路[9]，與 neural 一起使用，則成神經網路[10]。多字一義如，在研究分眾分類的人會使用 node、vertex 都是指節點；link、edge 都是指連結；network、graph 都是指由節點與連結所形成的東西[11]。

以節點分支度來決定節點的重要度，是複雜網路中常用的方法之一，但是標籤分支度的來源也包含了前段所述的一些來源，那些並不是我們想要的。我們分析群聚度與分支度的可能因素，在一般情況下，多重概念、一字多義的標籤，分支度與群聚度是負相關；單一概念的標籤，分支度與群聚度是正相關。

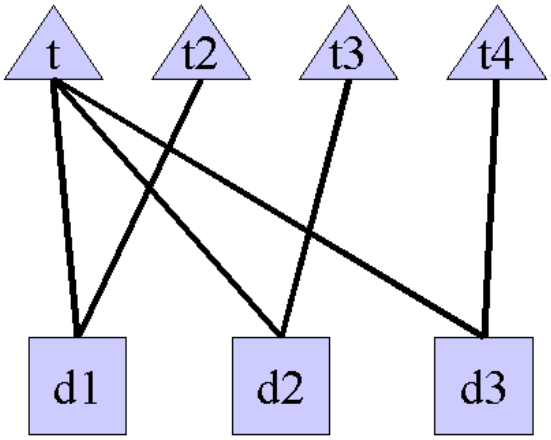
[7] 可上網觀察 http://www.citeulike.org/search/all?q=gene、http://www.citeulike.org/search/all?q=neural

[8] 可在 CiteULike 找尋以下論文的全體使用者的標籤 J. Kleinberg, "The small-world phenomenon: An algorithmic perspective," in Proceedings of the 32nd ACM Symposium on Theory of Computing, # 2000

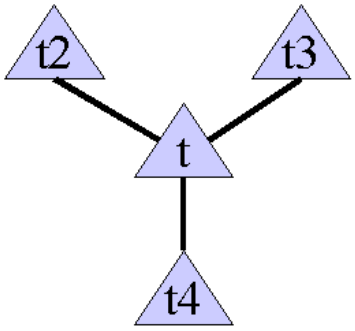
[9] 可在 CiteULike 找尋以下論文的全體使用者的標籤 A. Clauset and C. Moore, "Why mapping the internet is hard," July 2004.

[10] 可在 CiteULike 找尋以下論文的全體使用者的標籤 O. Sporns, D. R. Chialvo, M. Kaiser, and C. C. Hilgetag, "Organization, development and function of complex brain networks." Trends Cogn Sci, vol. 8, no. 9, pp. 418-425, September 2004.

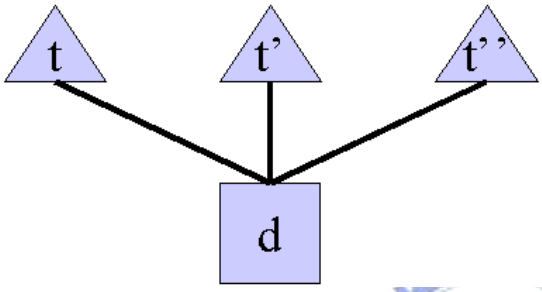
[11] M. E. Newman, "Analysis of weighted networks." Phys Rev E Stat Nonlin Soft Matter Phys, vol. 70, no. 5 Pt 2, November 2004.

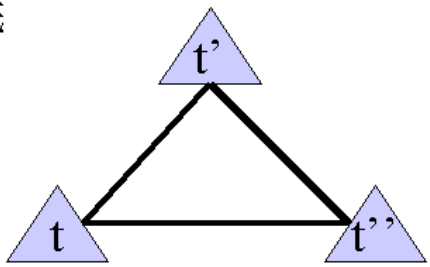


一字多義

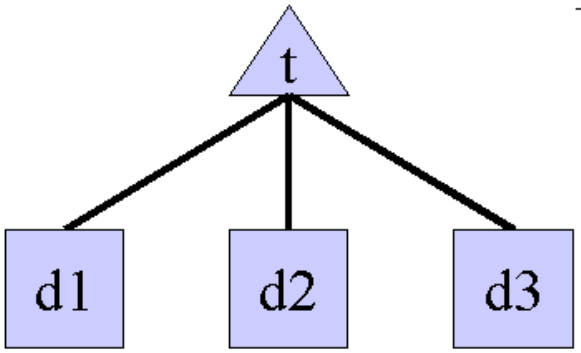


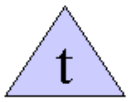
多字一義



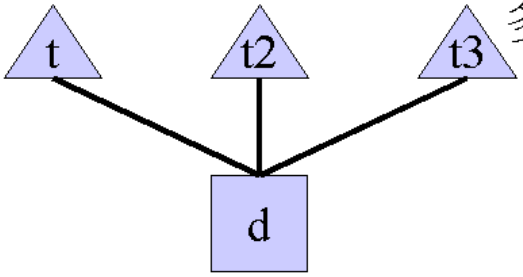


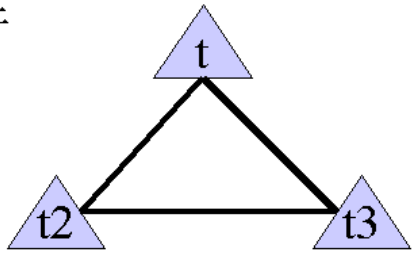
單一概念文件





多重概念文件





我們從標籤網路(文件)圖 9 分支度與群聚度的分佈發現，分支度與群聚度不是隨意分佈，而是(1)基本上整個網路的分支度與群聚度是呈負相關。(2)在分支度 1000~10000 有明顯的一群標籤，群聚度比同等分支度的標籤高。

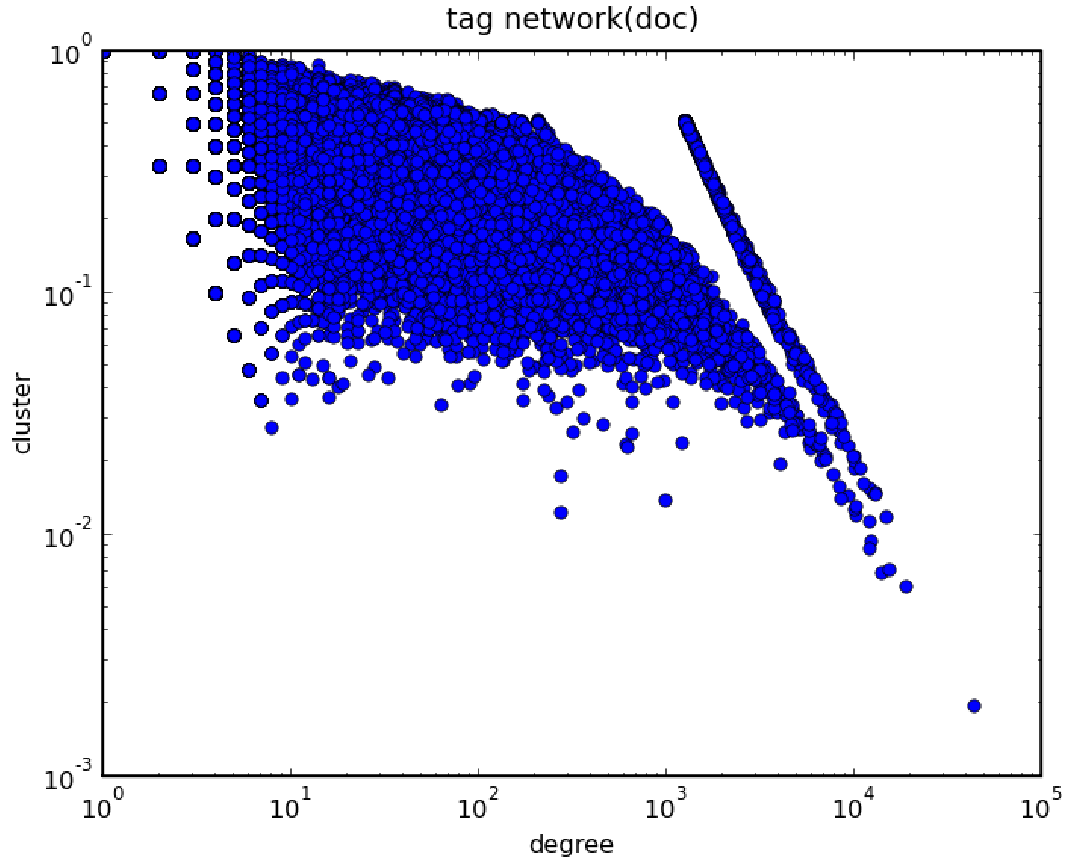


圖 9 標籤網路(文件) v.s.群聚度

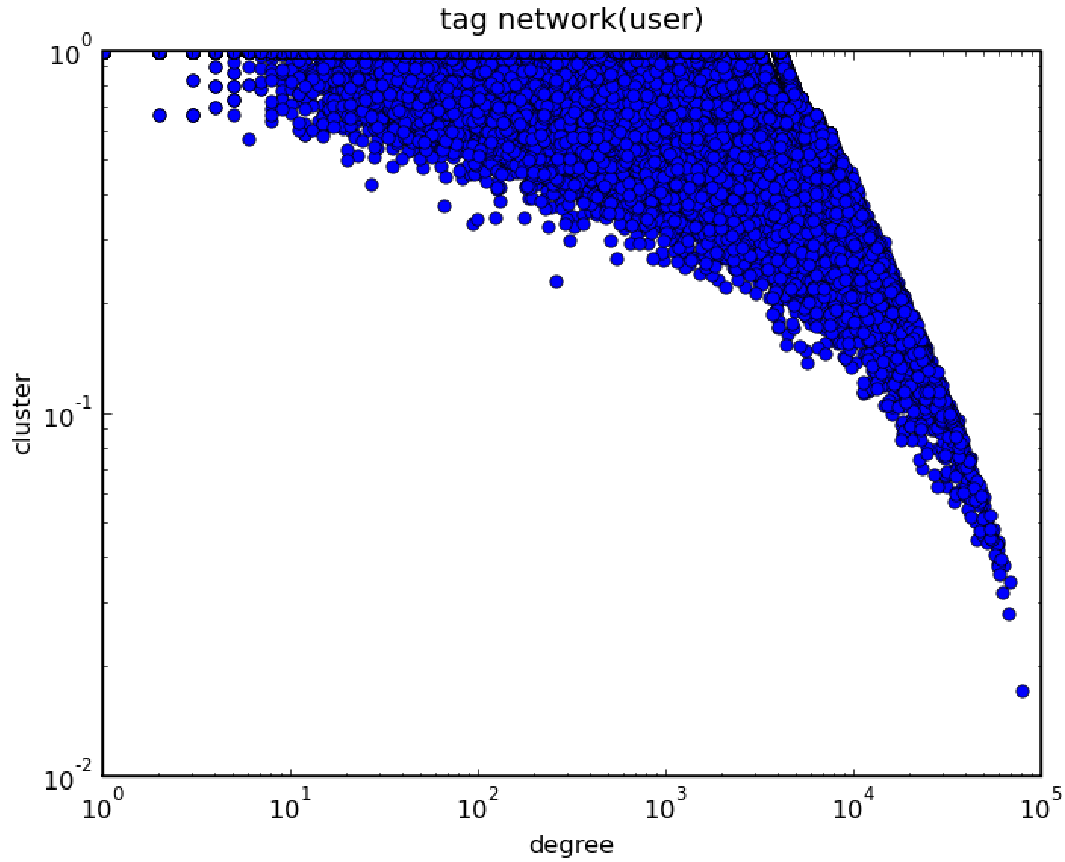


圖 10 標籤網路(人) v.s.群聚度

而在標籤網路(人)圖 10 上，整個網路的分支度與群聚度是呈負相關，而且更明顯的聚在一起。

4.2 推薦標籤清單

資料的前處理：為了比較不同程度稀疏(sparse)的資料集(dataset)，對於推薦系統的影響，我們產生從 2 到 8[12]等不同程度的「p 核心(p-core[13])」的資料，之後再從每一個使用者隨機抽取一份文件出來做為測試。推薦系統的工作就是以剩下的文件為基礎，推薦標籤給使用者，我們依此計算抽出來的文件的標籤與推薦結果之間的檢全率(recall)與檢準率(precision)。

[12] p-core level=9 時，整個 dataset 內已沒有資料

[13] This notion is used in [1]

表格 3 列出，不同 p 核心在抽取一份文件之後，TAS、|U|、|T|、|R|的個數。

表格 3 不同 p 核心資料集性質

p-core	TAS	| U |	| T |	| R |
處理前	2,369,141	22,363	151,142	715,016
2	372510	7742	21790	54957
3	161762	3961	7432	15659
4	80593	2181	3115	5886
5	37644	1085	1319	2237
6	15095	487	500	743
7	5225	192	169	209
8	862	45	27	39

接下來，以 p 核心= 3 為例，說明推薦清單的比較過程。當 p 核心=3 時，資料集內的性質為每個使用者至少張貼三篇論文，每篇論文至少被三個使用者張貼過，每個標篇至少被貼到三篇論文。

我們可以將推薦清單產生的步驟，分為兩個步驟，一是尋找鄰居們，一是排序標籤。產生鄰居們的方式選擇兩種，一是協同過濾(CF)的 $\pi_{UR}Y$，$\pi_{UT}Y$，採取 cosine similarity。另一是採用 Neighbor_Item(u_i)， Neighbor_Tag(u_i)。在這裡取鄰居數為 10，推薦數為 1 到 10 來觀察不同推薦數對於準確度的影響。排序標籤的方法也有兩種，一是協同過濾的鄰居的熱門標籤排列法，另一個是使用者的標籤派系網路與鄰居推薦的標籤重疊數目排列法。所以推薦方法總共會有四種不同的組合，我們將它列在表格 4。

表格 4 實驗方法表

實驗編號	推薦名稱	尋找鄰居的方法	標籤排列法
1	CF neighbor, Clique sorting	Cosine similarity of doc and tag	Overlapping of clique of tag between user and neighbor
2	Clique neighbor, Clique sorting	Overlapping of cliques of doc and tag	Overlapping of clique of tag between user and neighbor
3	Clique neighbor, CF sorting	Overlapping of cliques of doc and tag	Most popular

4	CF neighbor, CF sorting	Cosine similarity of doc and tag	Most popular

由於想知道，尋找鄰居的效果如何，在 CF neighbor 部份，我們計算前十名相似度平均做為參考。Doc based 的鄰居相似度平均為 0.168，而 tag based 的鄰居相似度平均為 0.352。在 Clique neighbor 部份，使用者鄰居的數目是不固定的，有的人的分支度高，有的人分支度少。為了與 CF 方法做比較，鄰居仍為 10。若分支度高於 10 的人則只取前 10 個鄰居；若分支度低於 10 的人則全取。我們計算能找到的鄰居的數目。Doc based 部份的平均是 9.366，tag based 部份的平均為 9.814。

在表格 5，我們列出實驗 1 及 2 在推薦標籤數為 10 的結果，藉以比較不同尋找鄰居的方法的差異(以 clique sorting 為基礎)。在這個比較之下，Clique neighbor 的 recall 與 precision 皆比 CF neighbor 的高。

表格 5 “CF neighbor and Clique sorting” and “Clique neighbor and Clique sorting”推薦評比

實驗 1/實驗 2	Doc based	Tag based
Avg. recall	0.176 / 0.218	0.099 / 0.171
Avg. precision	0.023 / 0.028	0.014 / 0.022

在表格 6 ，我們列出實驗 3 與實驗 4 在推薦標籤數為 10 的結果，藉以比較在 CF sorting 的基礎下，不同尋找鄰居的方法的相異。為了方便觀察方便，故意把實驗 4 的資料放在左邊，與表格 5 對齊。由數據來看，Clique neighbor 的檢全率與檢準率比較高。

表格 6 “CF neighbor and CF sorting” and “Clique neighbor and CF sorting”推薦評比

實驗 4/實驗 3	Doc based	Tag based
Avg. recall	0.140 / 0.178	0.095 / 0.157
Avg. precision	0.052 / 0.060	0.044 / 0.052

比較完尋找鄰居的效果後，接下來比較標籤排列方法的效果。在表格 7，我們將實驗 1 及實驗 4 的結果放在一起；表格 8 則是將實驗 2 及實驗 3 的結果放在一起。

表格 7“CF neighbor and Clique sorting” and “CF neighbor and CF sorting”推薦評比

實驗 1/實驗 4	Doc based	Tag based
Avg. recall	0.176 / 0.140	0.099 / 0.095
Avg. precision	0.023 / 0.052	0.014 / 0.044

表格 8 “Clique neighbor and Clique sorting” and “Clique neighbor and CF sorting”推薦評比

實驗 2/實驗 3	Doc based	Tag based
Avg. recall	0.218 / 0.178	0.171 / 0.157
Avg. precision	0.028 / 0.060	0.022 / 0.052

比較表格 7 及表格 8 的數據，Clique sorting 在檢全率方面比較好，而 CF sorting 在檢準率方面比較好。

我們將 Clique sorting 與 CF sorting 這兩種方法的平均推薦數列出來，在推薦數限定 10 個時：

表格 9 各種推薦方式的平均推薦數

	Doc based	Tag based
實驗 1(CF neighbor and Clique sorting)	2.509	1.324
實驗 2(Clique neighbor and Clique sorting)	3.411	2.589
實驗 3(Clique neighbor and CF sorting)	2.232	1.551
實驗 4(CF neighbor and CF sorting)	1.552	0.597

由表格 9 我們可以知道，在這幾種方法之內，平均能找到的推薦標籤數都不足夠 10 個。由於 Clique neighbor 及 Clique sorting 的平均推薦數比較多，所以在檢準率 precision 方面比較低，是可以理解的。因為在檢準率的分母部份，即有著不同的基礎。如同 Sean, John, and Joseph 等人[5](2006)的研究，認為在推薦系統中，太過講求準確(accuracy)反而導致實用性降低。因為人們除了要找尋已知的東西之外，未知的東西是更感興趣的，然而不論是檢全率或是檢準率都無法將這個部分完整評估出來。其中兩者來比較，檢全率比較符合我們的需求，但是檢準率這個指標仍然可以做為輔助驗證實驗結果，因此我們

還是將這兩個指標放在一起做討論。

接下來，我們將推薦數由 1 到 10 來觀察推薦清單的檢全率及檢準率。

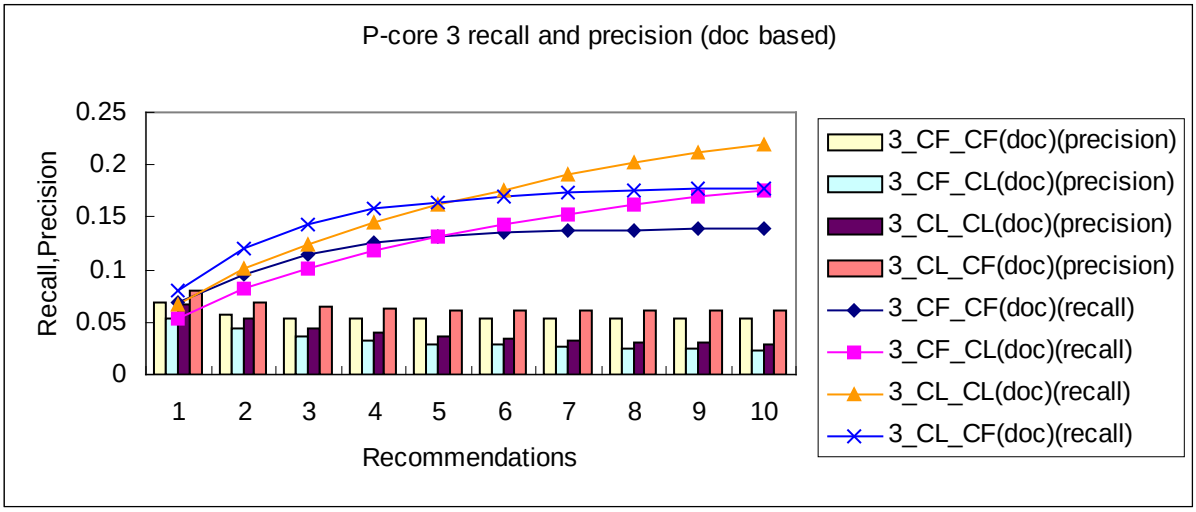


圖 11 推薦數 V.S. 檢全率、檢準率(p-核心=3) (doc based)

由圖 11 來看，推薦數越來越多的時候，檢全率越來越高，檢準率越來越低。比較值得注意的是，Clique-Clique 與 CF-Clique 的檢全率上升幅度相近，Clique-CF 與 CF-CF 的檢全率上升幅度相近。在檢準率部份，四個方法的下降幅度則是相近，CF-CF 的檢準率比較高。我們從圖表中的檢全率值裡面，CL_CF 與 CL_CL 的交叉點以及 CF_CF 與 CF_CL 的交叉點可以看到，在同一批鄰居中，推薦方法的不同所產生的不同，來自於標籤排列法。CF sorting 是使用「最熱門標籤」(most popular tag)而 Clique sorting 是依照重疊的遠近及數量來排列。所以我們可以看到在 CiteULike 的網站中，鄰居間都用(熱門)的標籤數目約是 5 個左右。

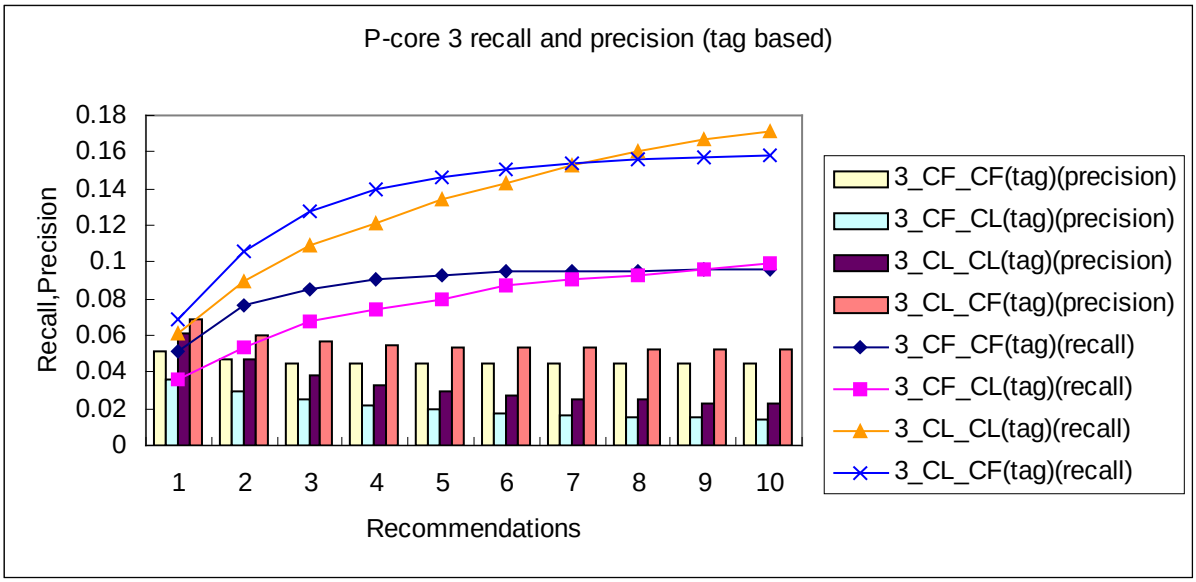


圖 12 推薦數 V.S. 檢全率、檢準率(p-核心=3) (doc based) (tag based)

圖 12 呈現以 tag based 的檢全率及檢準率。Clique-Clique 與 Clique-CF 的檢全率比較高。檢準率則變成 Clique-CF 最高。這裡同樣有 Clique-Clique 與 CF-Clique 的檢全率上升幅度相近，Clique-CF 與 CF-CF 的檢全率上升幅度相近的情形。Clique neighbor 的鄰居間的熱門標籤數目約是 7 個左右，CF neighbor 的鄰居間的熱門標籤數目約是 9 個左右。

接下來我們看不同 p-core level 對於檢全率與檢準率的影響。

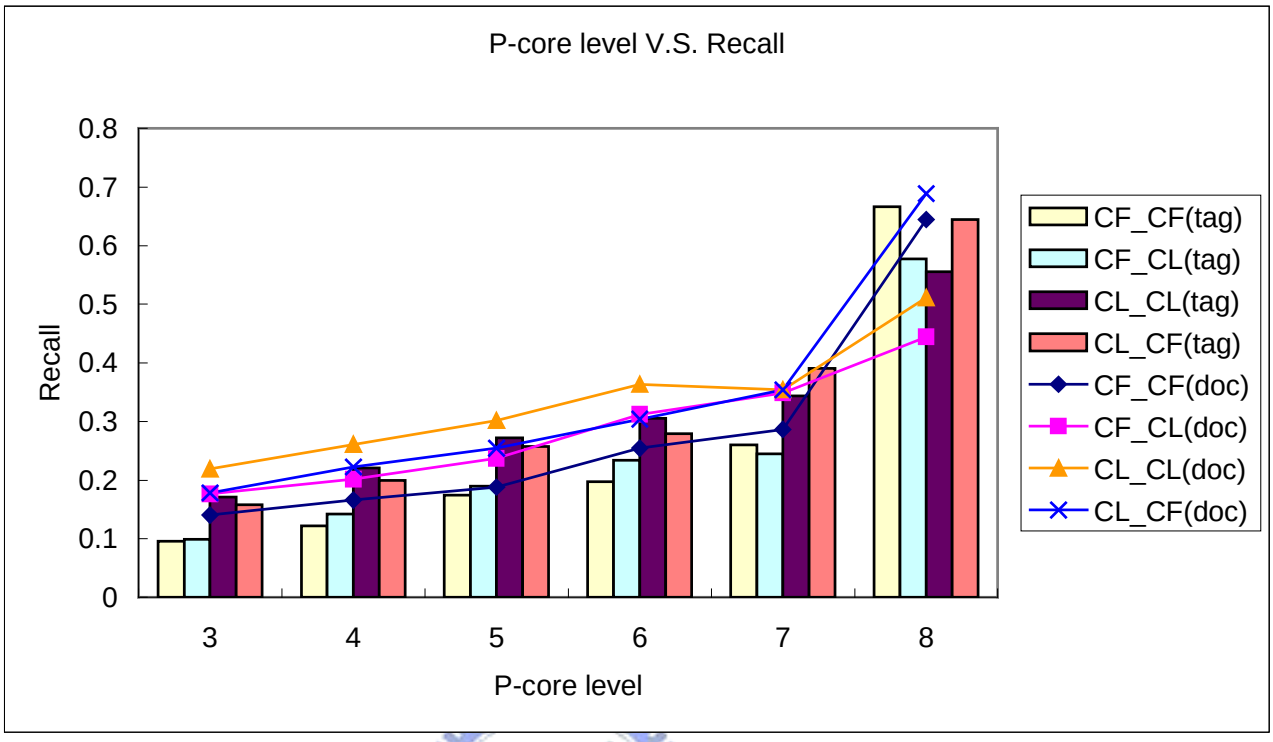


圖 13 p 核心 V.S. 檢全率

從圖 13，我們可以看到 CF-CF 的方法，在 p 核心從 7 到 8 的時候，檢全率變化很大，從表格 3 中可以知道其中的不同在於使用者的數量多於文件的數量，在一般 CF 的使用場合，皆是這種情況。而在 p 核心程度下降時，使用者數量少於文件，也就是 CF 所謂的稀疏資料集(sparse dataset)裡，CF 的方法就越來越差。反過來說，在面對稀疏資料集時，Clique-Clique，Clique-CF，CF-Clique 的適應性比 CF-CF 好。另外，也可以看到一個現象，在 p 核心程度高的時候，tag based 的方法多數比較好，在 p 核心程度低的時候，是 doc based 的方法比較好。

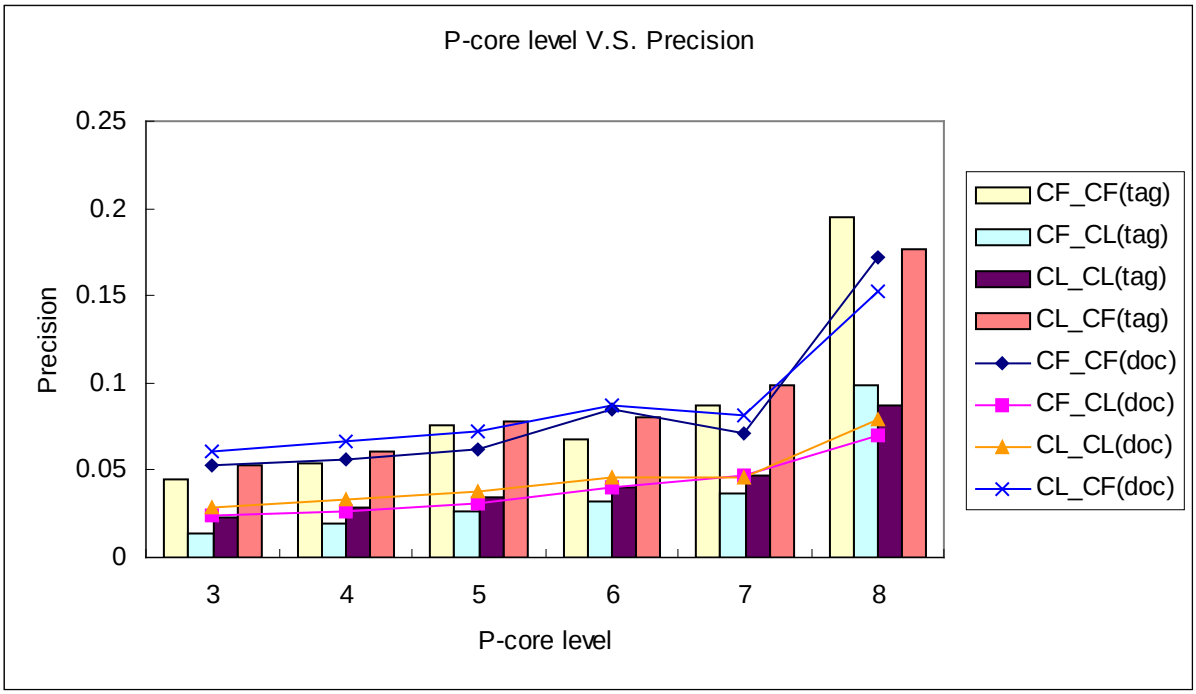


圖 14 p 核心 V.S. 檢準率

在圖 14 裡，明顯地看到 Clique sorting 與 CF sorting 在檢準率方面，CF sorting 比較好。Clique-neighbor 在檢準率的影響不大。我們發現比較低的四個值，都是使用 CL sorting 的方法。比較高的四個值，都是使用 CF sorting 的方法。使用相同標籤排列法的推薦方法，結果都很相近，也就是在檢準率這個部份，找鄰居的方法影響比較小，找標籤的方法，影響比較大。

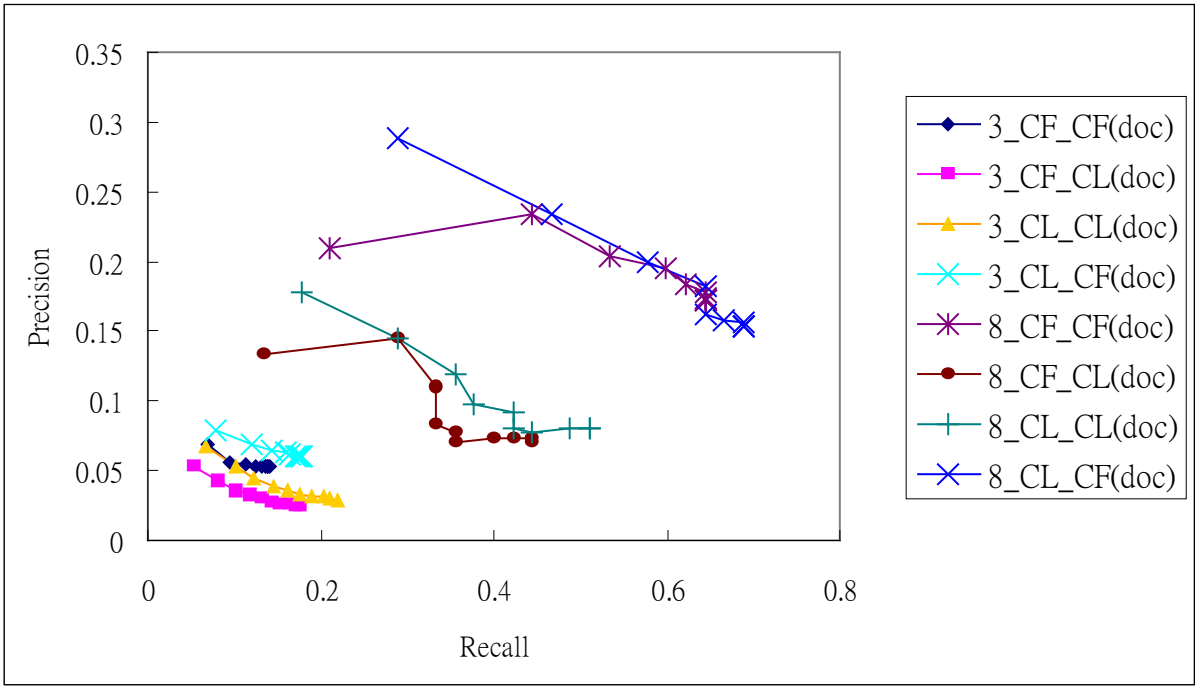


圖 15 檢全率與檢準率(p 核心= 3 及 p 核心= 8) (doc-based)

在圖 15 與圖 16 這裡將檢全率與檢準率畫在一起，用以比較兩者之間的關係。在 p-core 低的時候，檢全率越高則檢準率越低。

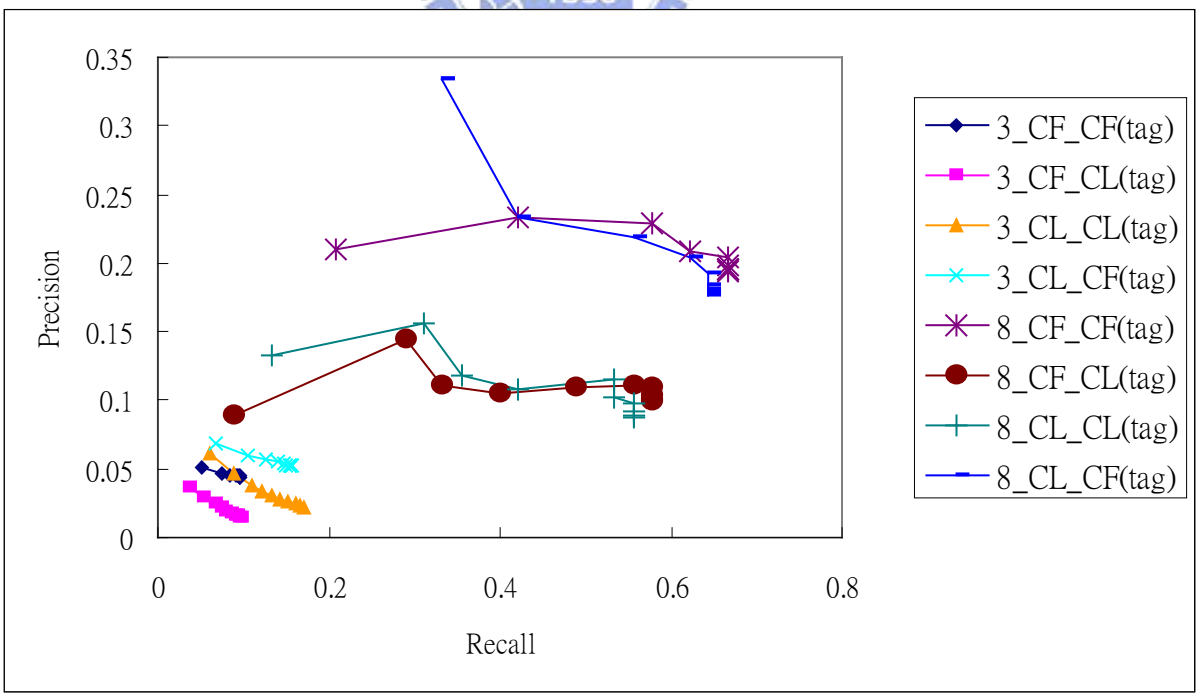


圖 16 檢全率與檢準率(p 核心= 3 及 p 核心= 8) (tag-based)

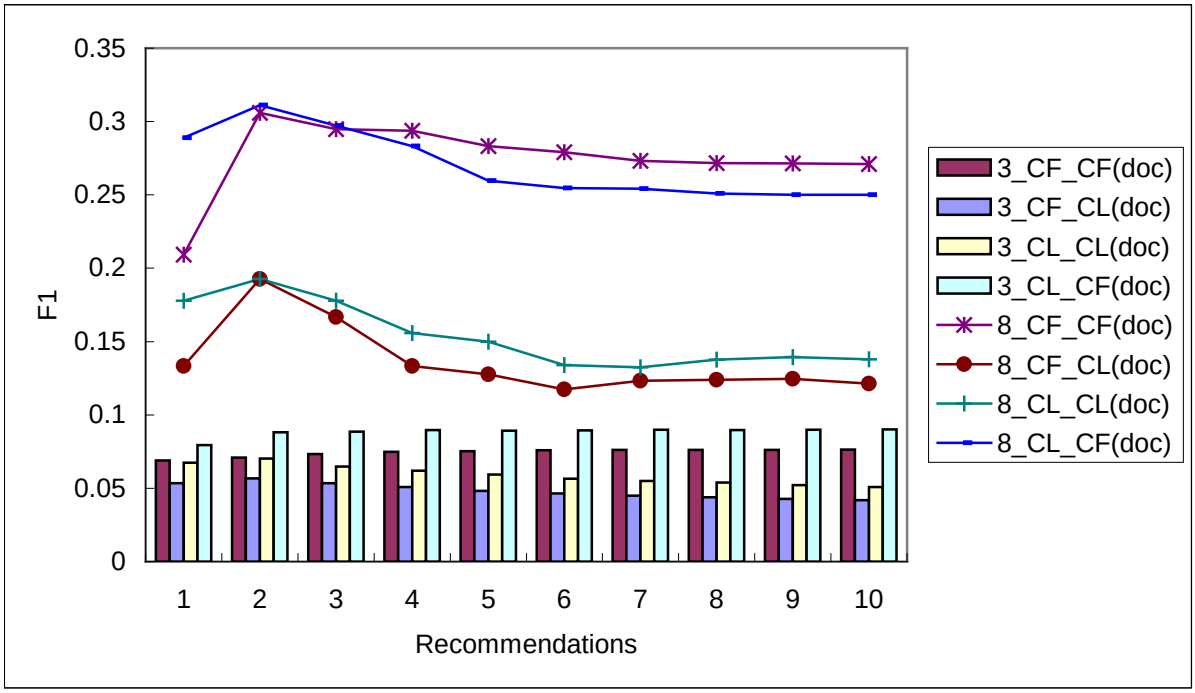


圖 17 推薦數與 F1(p 核心= 3 及 p 核心= 8) (doc-based)

在圖 17 與圖 18 這裡我們則是將 F-measure 中的 F_1 呈現出來，它是檢全率與檢準率的調合平均(harmonic mean)。在將檢全率與檢準率一同考量之後，CF sorting 的方法的 F_1 值都比較高。在相同 sorting 方法之中，Clique neighbor 的方法的 F_1 值比較高。

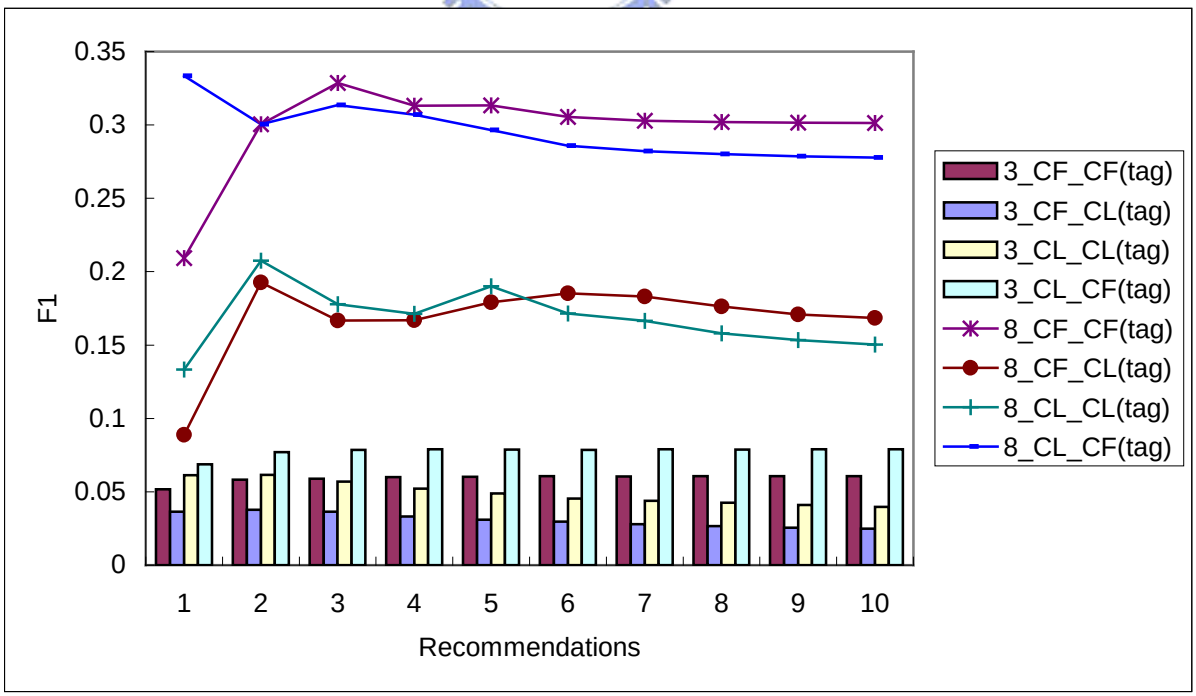


圖 18 推薦數與 F1(p 核心= 3 及 p 核心= 8) (tag-based)

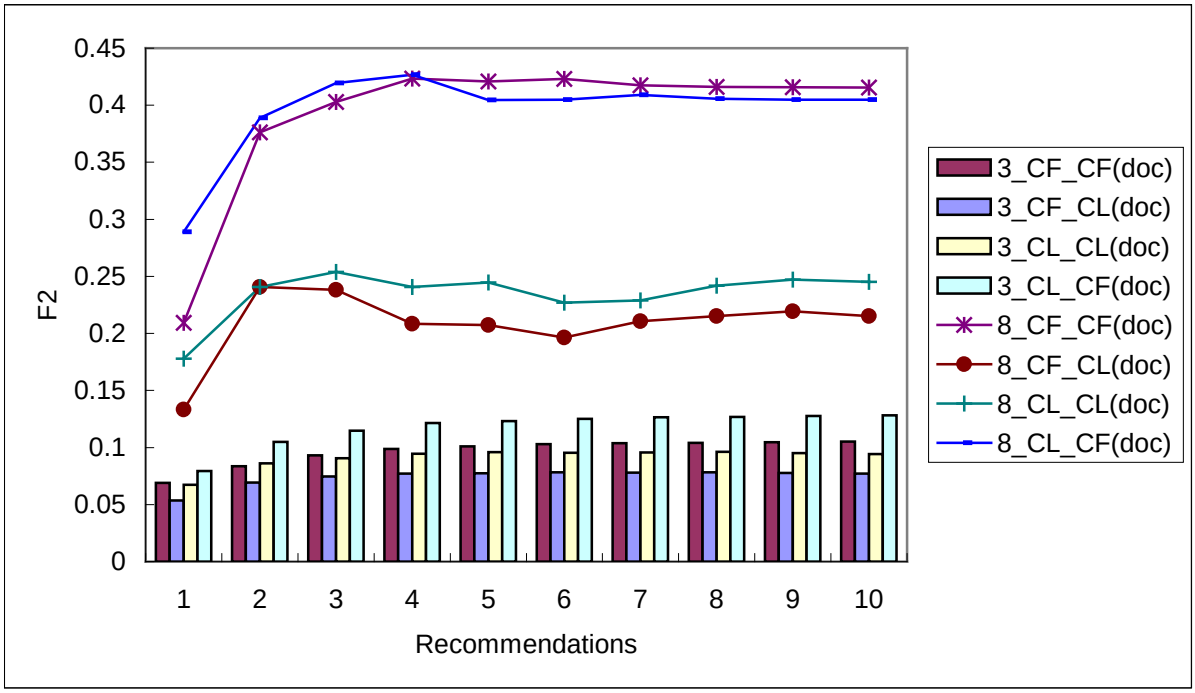


圖 19 推薦數與 F2(p 核心= 3 及 p 核心= 8) (doc-based)

在圖 19 與圖 20 這裡我們則是將 F-measure 中的 F_2 呈現出來。如同之前所述，由於我們比較關心檢全率值，所以使用 F_2 值，它是加重檢全率的權數為檢準率的兩倍。我們可以用另一個角度看到檢準率與檢全率在評估上的意義是如何相互影響。

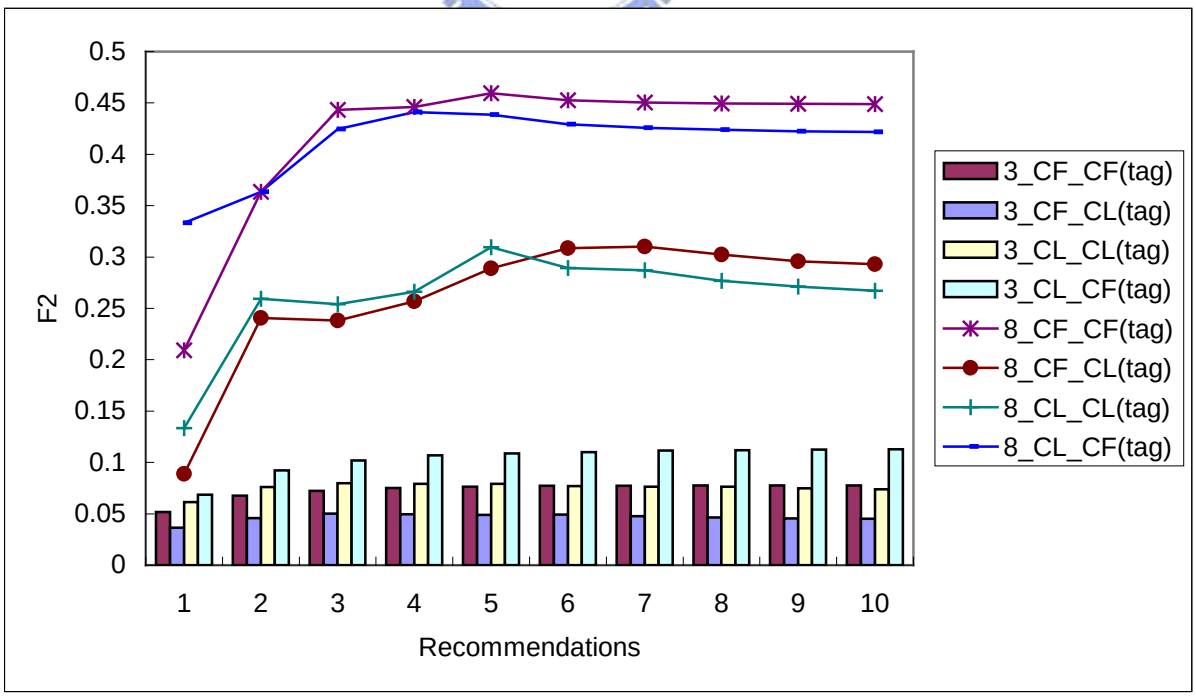


圖 20 推薦數與 F2(p 核心= 3 及 p 核心= 8) (tag-based)

如果我們為了準確度 accuracy 很高，可以只推薦少數的標籤，檢準率是可以很高，但對於知識發現卻是沒有什麼幫助。但也不能亂槍打鳥所有的標籤都推薦，使得檢全率很高，但是反而帶來更多的「資訊超載(information overloading)」。因此，檢全率與檢準率兩者在推薦系統的評量上，並不能只看其中一個，但也不能以同等份量來看待。

4.3 討論

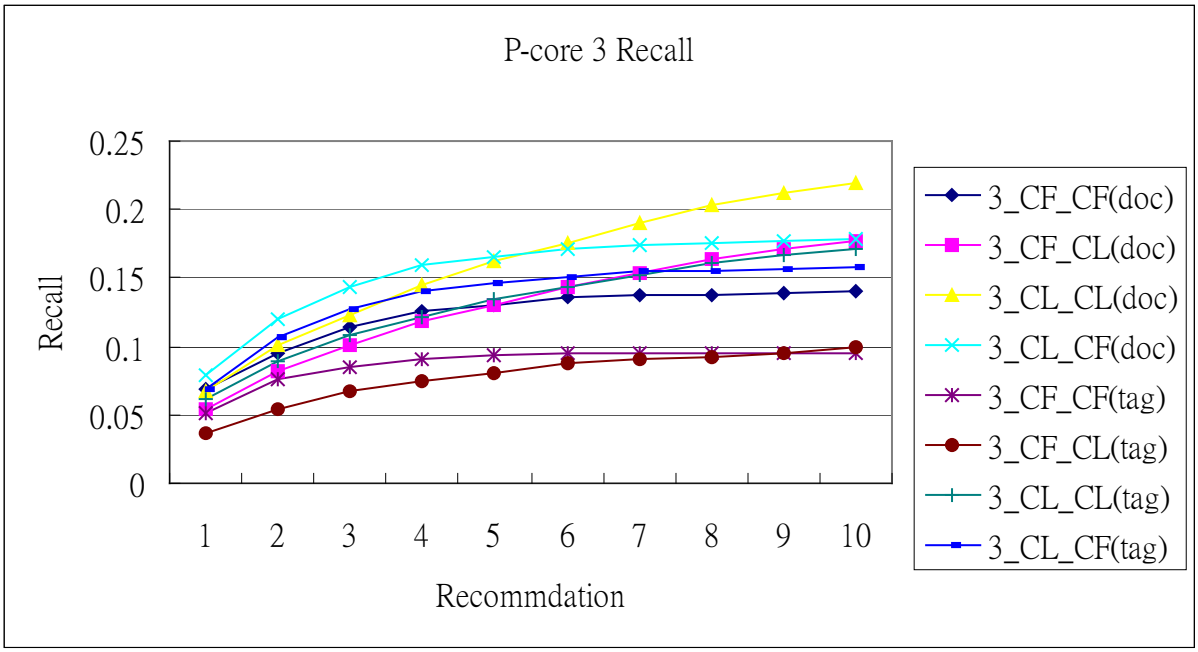


圖 21 Doc based V.S. Tag based (檢全率)

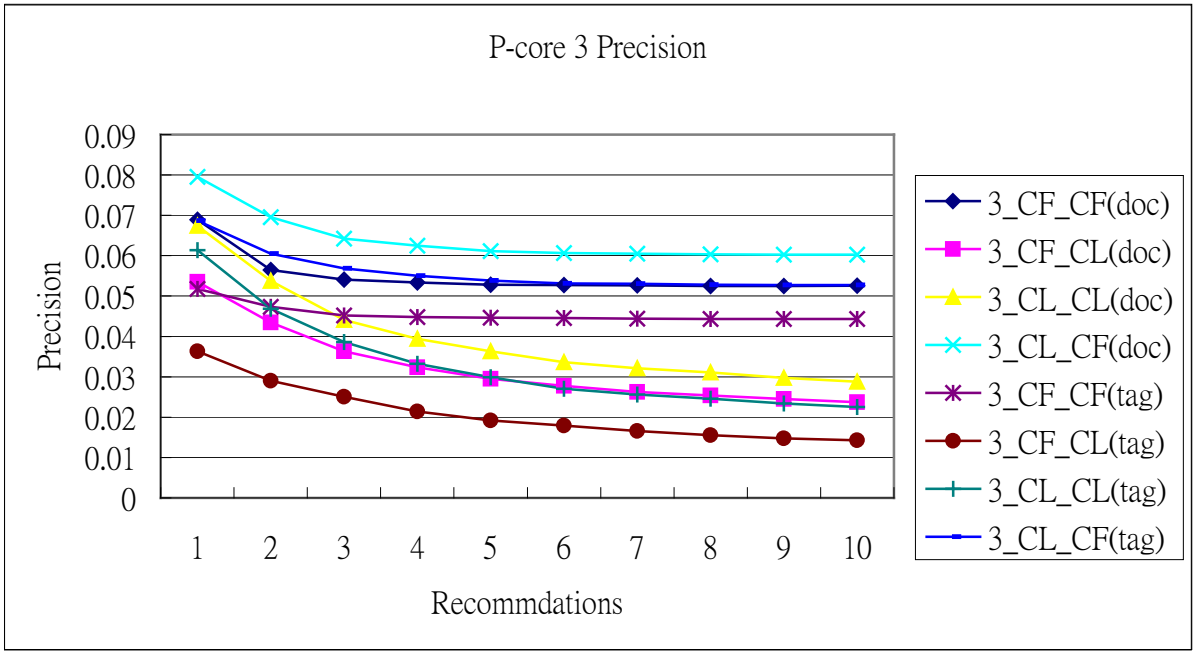


圖 22 Doc based V.S. Tag based (檢準率)

doc based 尋找鄰居，相似度比較低，由 tag based 尋找鄰居則相似度比較高。但是平均推薦數卻是由 doc based 的鄰居比較多，由文件的標籤網路的分支度來看，是有著無尺度 scale-free 網路的現象，假設標籤 1 被用在資源 1、資源 2 上，若標籤 2 被用在資源 1 上的話，標籤 2 也被用在資源 2 的機會也比較高。由「平均檢全率」與「平均檢準率」來看，有相同興趣的人所給的標籤比有相同看法的人所給的標籤準確，原因是在於 doc based 的鄰居看過同一份文件的機會比較高，自然在檢準率上是比 tag based 的高。但是因為 tag based 的鄰居使用的標籤是比較相近的，所以在檢全率上不致於差太多。在與 CF 方法比較過後，CF 方法採取的是從眾性質，在稀疏資料集 sparse dataset 的時候，就不容易找到好的鄰居、得到好的推薦。因為每個人的相似度都很低，所得到的資訊密度也很低，所以，標籤的使用上比較難以相同，但是標籤的曝光機會高的時候，常被使用的標籤就更容易被使用了。在資訊密度低的時候，同儕用語互相影響的關係，使用的標籤容易屬於是小團體的，也許描述的是同一個文件，也會因不同的團體而使用不同的詞語。所以從「分眾分類」的三分關聯網路尋找文件關聯及(或)標籤關聯的鄰居，(與 CF 方法相比之下)可以比較準確地找到使用者會使用的標籤。所以在個人化推薦上，推薦眾所皆知的標籤，並不能給予太大的幫助。推薦不同小團體使用的不同詞語，可以提醒使用者有另外一群人用另外一種認知來看待這份新的文件。甚至，使用者看過的文件，也可以經由重新檢視其他使用者的觀點來重新認識。

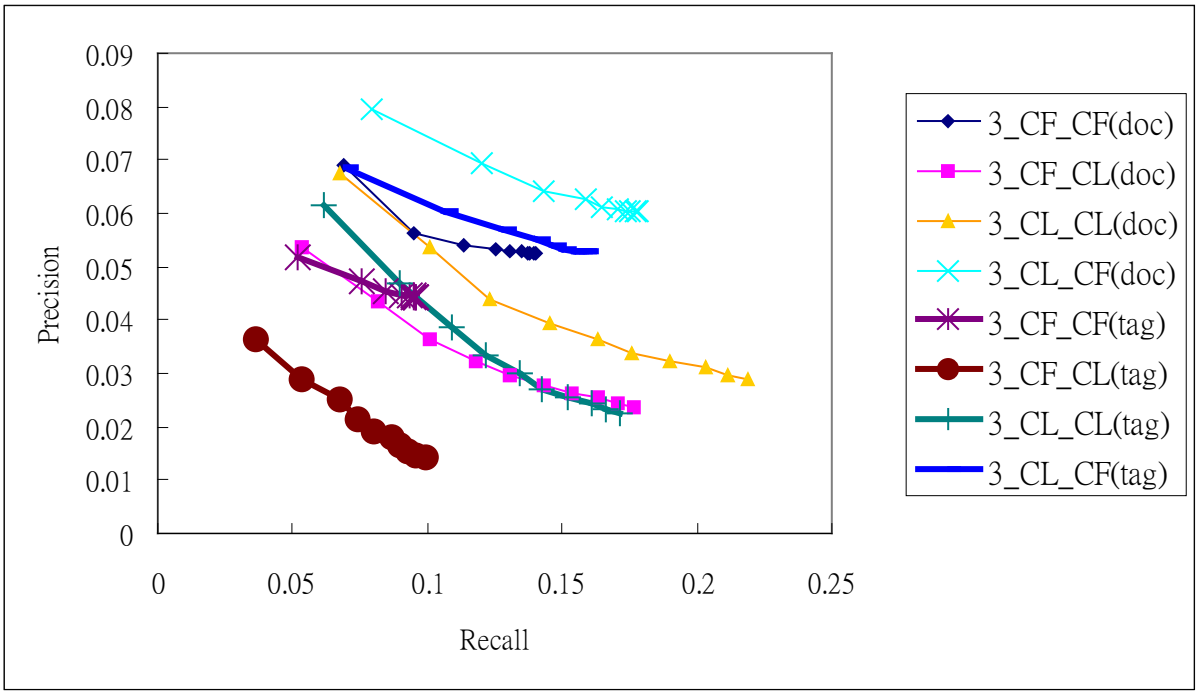


圖 23 檢全率與檢準率(doc-based and tag-based)

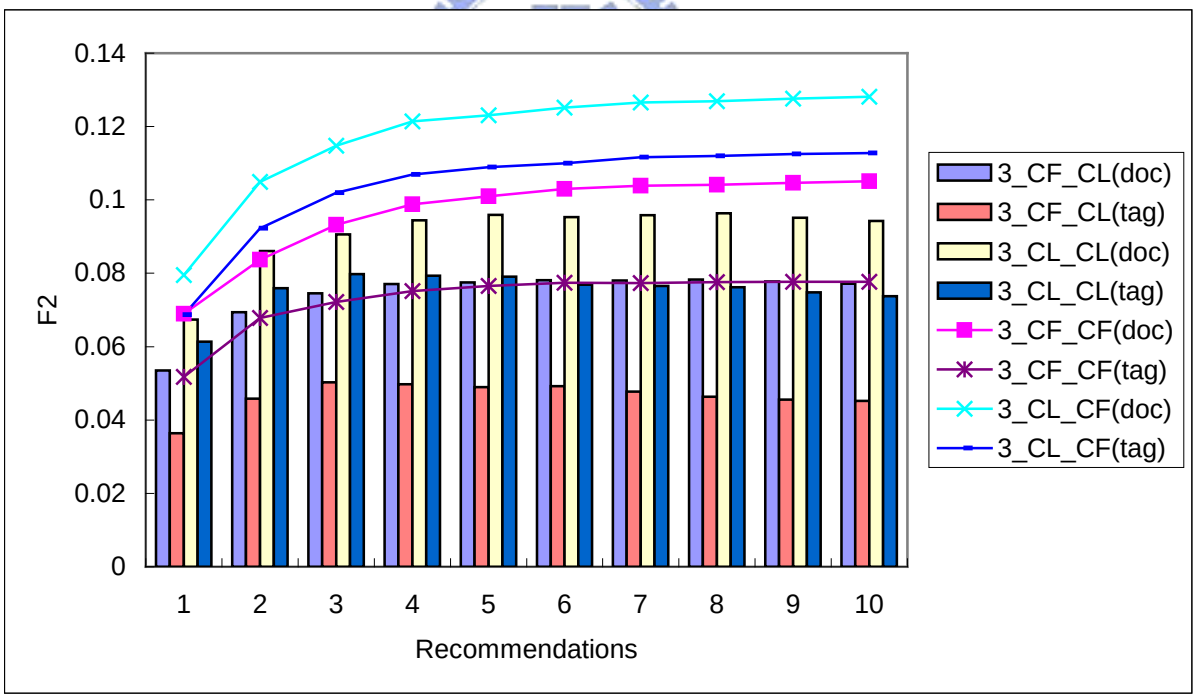


圖 24 推薦數與 F2 (doc-based and tag-based)

五、 結論

5.1 研究結論

本論文設計了一個新的推薦方法派系篩檢 Clique Filtering。將個人推薦清單的基礎分成兩個，一個從個人標籤網路，另一個從個人文件網路。過程分解成兩個步驟，一個從找鄰居階段來比較，一個從推薦清單排序做比較。由此來與協同過濾 Collaborative Filtering 比較探討，發現同樣興趣的人看同份文件的機會比較高，但那些人給的推薦清單準確度比較低；同樣看法的人看同份文件的機會比較低，但那些人給的推薦清單準確度比較高。結果兩份推薦清單的準確度非常接近。

若單獨看找鄰居的部份，我們發現用派系篩檢 Clique Filtering 方法找出來的鄰居所能提供的推薦準確度都比較高。因此，利用這種方式為使用者建立群聚關係，更接近於真實網站上使用者的群聚現象。

若單獨看推薦標籤的部份，CF 方法的檢準率 precision 是比較高，但是其原因來自可推薦數目較少。若將重點放在檢全率 recall 上，則派系篩檢 Clique Filtering 方法可推薦的標籤比較多，且又符合使用者的習慣，在實際應用上，可提供回想的準確度高，也有較高的機會給予使用者發現新知識的機會，也就是創新性(Novelty)及驚喜性(Serendipity)。

本論文示範了在社會標籤系統中，標籤推薦系統如何利用派系篩檢 Clique Filtering 做標籤推薦。而利用本篇實驗的結果，可以應用在三個地方，(1)在推薦系統部份，可以針對目前標籤系統中的密度，採取不用的標籤推薦清單順序。選擇鄰居部份，則是選擇派系篩檢 Clique Filtering 的方法。(2)在個人化搜尋系統部份，可以利用派系篩檢 Clique Filtering 找鄰居的方法，找出習慣相近的使用者，為搜尋結果做排序。(3)在線上社群研究部份，可以利用協同過濾 CF 及派系篩檢 Clique filtering 的差值，辨別標籤使用習慣。

5.2 未來方向

文字特性在複雜網路上會出現的特徵，可以再多加研究，可以語意網路的方法來研

究。在尋找鄰居上面，除了找人的鄰居，還可以朝向找文件的鄰居方向來著手。推薦清單的排序，是影響準確度結果的因素。若是創新性(Novelty)及驚喜性(Serendipity)有明確定義的評量標準，推薦清單排序的研究在知識發掘的部份還是有繼續研究價值。很可惜的是現在還沒有一種標準的方法，可評量創新性(Novelty)及驚喜性(Serendipity)。

六、 參考文獻：

[1] R. Jäschke et al., “Tag Recommendations in Folksonomies,” *Knowledge Discovery in Databases: PKDD 2007*, 2007, pp. 506-514; http://dx.doi.org/10.1007/978-3-540-74976-9_52.

[2] A. Hotho et al., “Information Retrieval in Folksonomies: Search and Ranking,” *The Semantic Web: Research and Applications*, 2006, pp. 411-426; http://dx.doi.org/10.1007/11762256_31.

[3] A. Mathes, “Folksonomies - Cooperative Classification and Communication Through Shared Metadata,” Sep. 2007; http://tc.eserver.org/29575.html.

[4] J.L. Herlocker et al., “Evaluating collaborative filtering recommender systems,” *ACM Trans. Inf. Syst.*, vol. 22, 2004, pp. 5-53.

[5] S.M. McNee, J. Riedl, and J.A. Konstan, “Being accurate is not enough: how accuracy metrics have hurt recommender systems,” *CHI '06 extended abstracts on Human factors in computing systems*, Montréal, Québec, Canada: ACM, 2006, pp. 1097-1101; http://portal.acm.org/citation.cfm?doid=1125451.1125659.

[6] R. Lambiotte and M. Ausloos, “Collaborative Tagging as a Tripartite Network,” *Computational Science – ICCS 2006*, 2006, pp. 1114-1117; http://dx.doi.org/10.1007/11758532_152.

[7] S.A. Golder and B.A. Huberman, “Usage patterns of collaborative tagging systems,” *Journal of Information Science*, vol. 32, Apr. 2006, pp. 198-208.

[8] G. Palla et al., “Uncovering the overlapping community structure of complex networks in nature and society,” *Nature*, vol. 435, Jun. 2005, pp. 814-818.

[9] R. Sinha, “A cognitive analysis of tagging « Rashmi’s blog,” *A cognitive analysis*

of tagging; http://rashmisinha.com/2005/09/27/a-cognitive-analysis-of-tagging/.

[10] “Folksonomy - Wikipedia, the free encyclopedia”; http://en.wikipedia.org/wiki/Folksonomy.

[11] T. Vander Wal, “Folksonomy :: vanderwal.net”; http://www.vanderwal.net/folksonomy.html.

[12] C. Cattuto et al., “Network properties of folksonomies,” *AI Commun.*, vol. 20, 2007, pp. 245-262.