基於高頻項目集結合近似樣式匹配之文件分群

楊燕珠
大同大學資訊經營研究所

陳志豐
大同大學資訊經營研究所

摘要

網際網路普及，越來越多使用者在網路上搜尋相關資料進行閱讀，本研究目標是將大量文件資料進行主題集群分析，方便使用者能很快瞭解文件集有哪些主題，迅速選擇所需主題的文件進行閱讀。本研究以關聯規則之高頻項目集結合近似樣式匹配，探勘出「近似高頻樣式」作為文件特徵；並將近似匹配的距離（相似度）納入特徵權重的衡量中。此外，本研究提出以「密度和相似度為基礎之二階段分群演算法」，此方法不需預先設定群集數目，適合於大量文件分群。經過實驗結果顯示，「近似高頻樣式」的特徵數量是彈性詞對的1.42倍，單一詞彙的0.84倍，透過此特徵分群，平均召回率、精確率和正確率皆較彈性詞對、相鄰詞對、單一詞彙等特徵的分群結果為高，證明以「近似高頻樣式」確實能抽取出更多有意義且具備區別力的特徵，搭配所提出的分群演算法，可以提昇分群速度，易於決定適當的群數，並提高文件分群的品質與正確性。

關鍵字：高頻項目集、樣式匹配、特徵抽取、文件分群

Document Clustering Based on Frequent Itemset Integrated with Approximate Pattern Matching

Yen-Ju Yang
Department of Information Management, Tatung University

Chih-Feng Chen
Department of Information Management, Tatung University

Abstract

Due to the popularization of the Internet, more and more users read desired data by directly searching from the Internet. This research aims to group a large number of texts by thematic document clustering for users rapidly realizing how many topics in those texts and picking up the interested topics to read. In order to extract more meaningful features, we propose an approach integrating frequent itemset with approximate pattern matching to mine the "Approximate Frequent Patterns". The distance (similarity) of approximate matching is adopted in measurement of feature weights, which is different from the traditional support count (frequency) of itemsets. In addition, the "Two-Phase Density and Similarity-Based Clustering Algorithm" is presented. This method doesn't need setting cluster number in advance, so as to be suitable for thematic document clustering. The experimental results show that the number of "Approximate Frequent Patterns" is 1.42 times of that of flexible word pairs and 0.84 times of that of single terms. Using this feature extraction, the clustering result in average recall, precision and accuracy are all higher than flexible word pairs, bigram and single word. This proves that "Approximate Frequent Patterns" can really extract more meaningful and discriminative features. Besides, our presented clustering algorithm can promote the speed, easily decide appropriate cluster number, and improve the quality and accuracy of document clustering.

Key words : Frequent Itemset, Pattern Matching, Feature Extraction, Document Clustering

壹、緒論

近年來由於網際網路的普及方便，幾乎所有的文件資料都已變成電子化格式，方便傳送、管理和閱讀，所以每天都有很多文件資料被產生出來，而人們的閱讀習慣也因此逐漸在改變。根據2007年1月Nielsen/NetRatings公佈的調查報告[1]，在2006年美國前10家線上報紙網站的流量在過去的一年中增長了9%，可以看到已經有越來越多的人轉向新聞網站或者雅虎(Yahoo)、Google等入口網站閱讀線上新聞。

網際網路上的文件繁多，暴增迅速，若沒有良好的整理，分門別類，使用者得花費大量的時間去瀏覽他並不需要的文件。例如，新聞網站雖然有政治、社會、國際、財經、科技、體育、娛樂等各類新聞，但除了Google News有相關新聞整理，大多數新聞網站不同主題的新聞混雜在一起，導致使用者必須花時間先將標題掃瞄一次，篩選想要閱讀的新聞，即使透過搜尋引擎檢索，使用者亦不容易將主題描述出來（使用者在閱讀前並不知道發生了哪些新聞事件），效果不彰。

本研究目標是將大量文件資料進行主題集群分析，方便使用者能很快瞭解文件集有哪些主題，迅速選擇所需主題的文件進行閱讀。而文件分群主要需解決兩個問題，即文件特徵抽取與有效的分群演算法。所以本研究擬提出能抽取出更多有意義且具備區別力特徵項的方法。此外，我們也研擬快速分群演算法於主題式文件分群，並能獲得適當的群集數目，以提昇文件分群的品質與正確性。

貳、文獻探討

一、文件特徵

「特徵」是在樣式辨認 (Pattern Recognition) 中擷取出具有區別力的項目，作為辨認相似程度之用。故本節將介紹幾種廣被使用的文件特徵。

（一）關鍵詞 (Keyword)

每份文件都是由詞彙組合成句子，再由句子組合成文章，所以詞彙就是組成文件的關鍵之一，如何決定哪些詞彙是文件重要的關鍵詞，可由詞彙頻率 (Term Frequency, TF) 來決定，詞彙頻率就是詞彙在文章中出現的次數，一篇文章中出現很多次的詞彙，必定有其重要性 (Salton and McGill 1983)。

（二）反向文件頻率 (Inverse Document Frequency)

Jones (1972) 提出Inverse Document Frequency，因為每篇文件中的詞彙在整篇文件的重要性，其實是不太相同的。Salton and Buckley (1988) 經過實驗驗證詞彙頻率和反向文

[1] http://www.emarketer.com/Article.aspx?id=1004479

件頻率 (term frequency-inverse document frequency, tf-idf)，當一個詞彙在文章中tf很高，且出現在文件集的少數文件中，代表這個詞彙越能區分文件，越具代表性，適合當成特徵；反之，如果出現在多數文件中，代表這個詞彙較不具備區別力，不適合作為特徵。

（三）詞對 (Word Pairs)

Baeza-Yates and Ribeiro-Neto (1999) 認為並非所有的詞彙都具有同等的重要性，大部分的語意都是由名詞帶出，因此建議可以將兩個或三個名詞聯合組成一個單元形成索引項。Al-Kofahi et al. (2001) 選擇了word pair當特徵來解決分類問題，實驗結果證明，名詞及名詞詞對 (noun-word pair) 比起一般的單詞或bi-gram更具特徵區別的能力。

（四）彈性詞對 (Flexible Word Pair)

每個人對同一件事情的表達方式不盡相同，在寫文章時也是一樣的，雖然所描述的是同一件事情，但因個人有個人的表達方式，使用詞彙順序也可能不同。當相同意義但前後排列不同的兩個詞彙，在word pair會被視為不同的特徵項，使得特徵的權重被分散，所以楊和王 (2007) 提出「彈性詞對」允許同一個特徵項的詞對之間可以有更多的詞彙間隔和相反的順序，增加特徵項的使用頻率。

（五）高頻項目集 (Frequent Itemset)

大文件集因為詞彙非常多而產生大量的特徵項，造成文件向量空間維度變的龐大，可能會造成分群效果不佳，所以Beil et al. (2002) 提出利用關聯規則演算法產生frequent term set的概念，把itemset和交易資料庫的觀念應用在term和文件上，所找到的term set不但是frequent，而且也可以找到由一個詞彙以上組成的詞彙集合，在分群上有很好的效果。Fung et al. 等人 (2003) 也是利用frequent itemset的概念去做分群來增加分群的精確度。

二、向量空間模型 (Vector Space Model)

向量空間模型 (Salton and Buckley 1988)，是目前資訊檢索領域常用的模型，每篇文件我們可以視為空間中的一個向量，而其維度則由文件集中抽取出有意義的關鍵詞彙之數目而定。

例如一篇文件有 t 個索引特徵項，可以表示成向量d（$w_1, w_2, w_3, \cdots\cdots, w_t$），$w_i$ 為第 i 個特徵項的權重，由 tf-idf 求得。在向量空間模型中，常被用來計算兩文件之間相似程度的為夾角之餘弦(cosine)值，介於 [0, 1] 之間，夾角越小值越大，表示兩向量之間相似度越高。

$$sim(d_i, d_j) = cosine(d_i, d_j) = \frac{d_i \bullet d_j}{|d_i| \times |d_j|} = \frac{\sum_{k=1}^{t} w_{k,i} \times w_{k,j}}{\sqrt{\sum_{k=1}^{t} w_{k,i}^{\,2}} \times \sqrt{\sum_{k=1}^{t} w_{k,j}^{\,2}}} \quad (2\text{-}1)$$

三、資料分群 (Data Clustering)

資料分群主要是藉由辨識和量化資料項目間的相似度或相異度將資料聚集。在各種

分群演算法中，大致可區分為已知群數與未知群數兩大類。

已知群數的分群演算法，必須事先給定群集數目K，將資料分成K個子集合，求出品質最好的K個子集合。最常見的方法為分割法 (Partitioning Methods)，其中最有名的是K-means，但K-means容易落入區域最佳解，故有學者導入演化式演算法來分群。演化式演算法通常用來求解組合最佳化問題，如遺傳演算法 (Genetic Algorithm, GA)，螞蟻族群最佳化 (Ant Colony Optimization, ACO)，粒子群最佳化 (Particle Swarm Optimization, PSO)等，此類方法應用在資料分群上，求解各種分群組合中最佳的結果，以找出群內距離最小（相似度最大）為目標。

然而在實際的應用中，事先設定最適合的群數並不容易，故另外一支未知群數的分群演算法應運而生。最常見的方法為階層法 (Hierarchical Methods) 和密度為基礎法 (Density-Based Methods)。階層法又分為由下而上 (bottom-up) 凝聚式(agglomerative) 或由上而下 (top-down) 分裂式 (divisive) 兩種，前者漸漸將相似度高的較小群合併成較大的群集，後者漸漸將較大的群集進行分離，經過重複迭代直到停止條件。由於階層法，需將任兩群之間的距離或相似度保留，需要大量的儲存空間與計算時間，當資料數量龐大時可能無法在電腦記憶體中執行。此外，在前面的階層中若分群錯誤，後面的階層也無法修正，錯誤會繼續蔓延 (error propagation)。於是有學者提出較快速的密度為基礎法 (Ester et al. 1996)，利用相鄰區域的觀念來產生群集，適用於任何形狀的群集，且容易處理雜訊或偏離資料 (outlier)。

基於密度分群的方法利用相鄰區域的觀念來發現群聚，一開始每個資料點代表一個集群，接著對於每個集群內的資料點，根據鄰近區域半徑 ε 及資料點數量門檻值(Minpts)，找出其半徑所含鄰近區域內的資料點。如果資料點大於門檻值，將這些鄰近區域內的點全部歸為同一集群，以此類推，慢慢地合併擴大集群的範圍。目前較常見的密度式群聚演算法有DBSCAN、OPTICS、DENCLUE。

DBSCAN之相關定義(Ester et al. 1996)：

1. 距離資料點半徑ε以內的鄰近區域，則為該資料點的ε-鄰近區域。
2. 資料點的ε-鄰近區域中包含了至少Minpts個資料點，則該資料點為核心物件。
3. 資料點p的位置是在某核心物件q的ε-鄰近區域內，則資料點p被稱為由q直接密度可達(directly density-reachable)的物件。
4. 假如資料點p由q_1直接密度可達、而q_1由q_2直接密度可達、……、而q_{i-1}由q_i直接密度可達，則資料點p被稱為由q_i密度可達 (density-reachable)的物件，如圖1，p由q密度可達。
5. 假如資料點p和q都可由o密度可達，則p和q可以被稱為密度連接(density-connected)，如圖2。

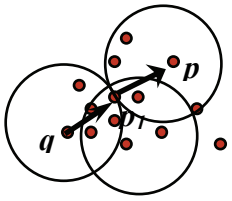

圖1：密度可達 (density-reachable)

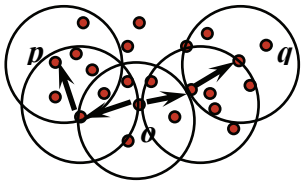

圖2：密度連接(density-connected)

四、文件分群 (Document Clustering)

將大量文件以主題分群時，並無法事先預知整個文件集包含幾個主題，故一些發展成熟的分群演算法架構在已知群數的先決條件無法適用於文件分群。在前人的研究中，大多數都是以凝聚式階層分群演算法為基礎，再整合其他方法，進行文件分群。Dubes and Jain (1988) 以凝聚式階層法應用於文件分群，稱為UPGMA (Unweighted Pair Group Method with Arithmatic Mean)。以凝聚式階層法作資料分群時，必須計算任兩群之間的距離（相似度），以便決定合併哪兩個最接近的群集，衡量兩群距離一般常用單一連結(single link)、完全連結(complete link)、平均連結(averge link)、中心距離(centroid distance)、華德法(Ward's method)等。UPGMA提出新的計算兩個文件群相似度的公式如(2-2)

$$similarity(cluster_1, cluster_2) = \frac{\sum_{\substack{d_1 \in cluster_1 \\ d_2 \in cluster_2}} cosine(d_1, d_2)}{size(cluster_1) * size(cluster_2)} \quad (2\text{-}2)$$

因階層式演算法每次迭代僅增加或減少一群，必需花費相當多的時間，所以Steinbach et al. (2000) 提出二分K-means法 (Bisecting K-means)，是一種分裂式階層法，以二元分裂的方式讓分群速度加快。而 Beil et al. (2002) 提出階層式高頻詞彙為基礎分群法 (Hierarchical Frequent Term-based Clustering, HFTC)，以包含相同高頻詞彙組合的文件聚集在一起，解決文件大量且高維度的問題。

五、關聯規則(Association Rules)

關聯規則探勘的目的是從大量的交易項目資料庫中，發現商品之間的關聯性或是可以探勘出人類所不知道的關聯規則。在關聯規則演算法中，最常見的就是由Agrawal and Srikant (1994) 所提出的Apriori algorithm，還有加快速度的FP-growth Algorithm (Han et al. 2000)。

Apriori 演算法的概念是在大量的資料集中建立關聯規則候選項目的集合 (candidate itemset)，集合的項目組合從1, 2, …, k, 每個迴圈多1項。(k+1)-itemset是由兩個k-itemset具有共同(k-1)項目擴充而成，經過成分檢查修剪，若為合理的候選集合則掃瞄資料庫獲得頻率，當超過最小支持度時，此集合即成為高頻項目集。

FP-growth演算法，是利用FP tree資料結構為主的演算法，這種方法主要目的是改進因Apriori演算法產生候選項目集合，對於大量的資料無法快速且有效率處理。FP-growth

演算法不用產生候選項目集合，針對大量資料尋找高頻項目集，建構於樹狀結構中，只需掃描資料庫兩次，節省許多時間。

參、研究方法

一、研究流程

本研究的研究步驟如下，圖3為本研究之研究流程圖。

步驟一：文件集載入。

步驟二：文件前置處理：在中文文件方面，使用中研院CKIP中文詞知識庫小組所提供的斷詞與詞類標記系統Autag1.0處理後，保留名詞；英文文件集則先將停止詞移除，再經過Porter Stemming Algorithm還原詞根 (Porter 1980)。

步驟三：特徵抽取：修改Apriori algorithm將近似樣式匹配的距離融入，以產生超過最小支持度門檻值的「近似高頻樣式」，即為特徵項。

步驟四：特徵權重計算：計算每篇文件中每個特徵的pwf-idf，產生文件與特徵項的pwf-idf關聯矩陣。

步驟五：文件分群：最後進行本研究提出之密度和相似度為基礎之二階段分群。

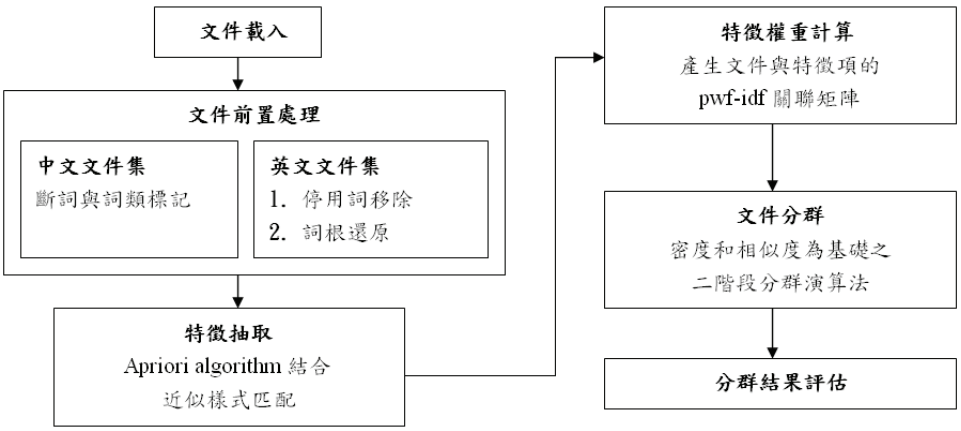

圖3： 研究流程圖

二、特徵抽取：近似高頻樣式(Approximate Frequent Pattern)

關聯規則中的高頻項目集探勘，是一種可以找出兩項以上，常見項目組合的方法，然而此方法只考慮每一次交易有哪些項目的組合，並沒有區別組合出現的頻率；以購物籃解釋，就是只看籃子裡有哪些商品，而不去計算各種商品的購買數量。本研究認為如果以詞彙代表商品項目，再考慮詞彙加權頻率，也就是每篇文件出現的詞彙數量可以區別文件相似程度，故本研究修改Apriori algorithm結合近似樣式匹配(Approximate Pattern Matching)，希望可以找出更多有意義且有區別力的特徵項。

舉例來說，文件d_1是「夏季賞花活動就在北海道開跑，民眾首選北海道夏季花之旅。」，文件d_2是「民眾日本賞花去，北海道夏季旅遊展活動開跑。」，經過中文斷詞，保留名詞後，如表1，<BOS>代表句首(Beginning of Sentence), <EOS>代表句尾 (End of Sentence)。

表1：中文文件斷詞結果

文件	詞彙
d_1	<BOS>夏季 花 活動 北海道<EOS><BOS>民眾 北海道 夏季 花 旅<EOS>
d_2	<BOS>民眾 日本 花<EOS><BOS>北海道 夏季 旅遊 展 活動<EOS>

由表1產生特徵項步驟如下：

步驟一：定義最小支持度。

步驟二：將資料庫中的詞彙整理出來，即C_1(Candidate 1-itemset)，並計算其TF(Term Frequency)和DF(Document Frequency)，若support大於或等於最小支持度，即為近似高頻樣式集合L_1(Frequent 1-itemset)，如圖4。

itemset	TF	文件
夏季	2	d_1
花	2	d_1
活動	1	d_1
北海道	2	d_1
民眾	1	d_1
旅	1	d_1
民眾	1	d_2
日本	1	d_2
花	1	d_2
北海道	1	d_2
夏季	1	d_2
旅遊	1	d_2
展	1	d_2
活動	1	d_2

→

C1

itemset	support
夏季	3
花	3
活動	2
北海道	3
民眾	2
旅	1
日本	1
旅遊	1
展	1

→ >= min support →

L1

itemset	Support
夏季	3
花	3
活動	2
北海道	3
民眾	2

圖4：由C_1產生L_1

步驟三：利用L1近似高頻樣式集合產生候選項目C_2，如圖5，候選項目集合需要經過淘汰選擇，才是有效的候選項目，選擇規則如下：

1. 詞組要出現在同一句子裡，且詞組前後距離相距 r 個詞以內。由心理學家調查發現人類在短暫記憶中只能記得相鄰 7±2 個字，故本研究沿用詞與詞間的距離最大為5 (Chen et al. 2002)。
2. 詞組可以不用相鄰，也沒有順序限制。

根據表1， C_2 itemset <活動、名眾>兩個詞彙在兩篇文章中都不是在同一句子裡，所以刪除；距離方面C_2 itemset <夏季、活動>是2，C_2 itemset <夏季、北海道>是3，以此類推，皆在限制的距離r＝5內。

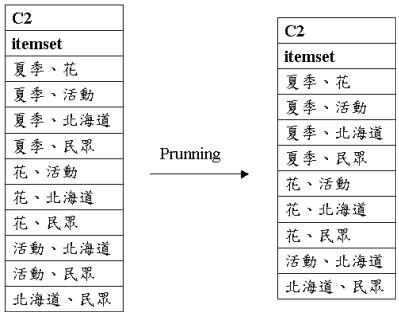

圖5：C2通過檢查前和檢查後的項目

步驟四：將圖5合格的C_2 itemset，掃描資料庫的文件資料，以近似樣式匹配找出該C_2 itemset，並計算其距離、TF和DF，若support大於或等於最小支持度，即為近似高頻樣式集合L_2。

在Apriori algorithm中，C_2 itemset <夏季、北海道>在文件資料庫中找到含有這兩個項目的文件，並未限定必須出現在同一個句子內，而且也不會考慮在文件中的頻率，也就是在購物籃中只會找到含有這些商品的交易，不考慮該次交易的購買數量。以一般的精確匹配(Exact Match)，C_2 itemset <夏季、北海道>在文件資料庫中只會找到相鄰且固定順序的兩個詞。故我們提出解決此兩種缺失的匹配方法。

根據近似樣式匹配，組成的詞彙只要在同一個句子裡，彼此距離不超過5，不限定順序，如C_2 itemset <夏季、北海道>，在文件d_1中間隔兩個詞，則位置距離為3：<夏季、北海道，3>，文件d_2中夏季在北海道相鄰之後，則位置距離為-1，取絕對值為1：<夏季、北海道，|-1|>，這兩個樣式在我們的研究中皆視為相同的樣式，經過整理和閥值過濾獲得L_2如圖6。

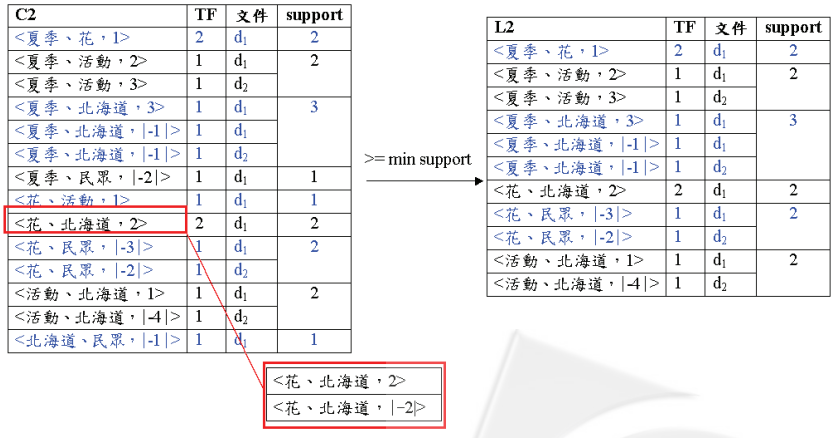

圖6：由C_2產生L_2

步驟五：利用L_{k-1}近似高頻樣式集合經過結合(Join)和修剪(Prune)產生候選項目C_k。而三個詞彙以上的樣式之距離計算如公式 (3-1)

$$dist_i = \frac{\sum_{t=1}^{l_i-1} \left| r(w_{i,t}, w_{i,t+1}) \right|}{l_i - 1} \quad (3\text{-}1)$$

l_i：特徵樣式p_i組成的詞數

$w_{i,t}$, $w_{i,t+1}$：樣式中第t與第t+1個詞

$\sum_{t=1}^{l_i-1} \left| r(w_{i,t}, w_{i,t+1}) \right|$：兩兩詞彙間的距離絕對值之和，其中r($w_{i,t}$, $w_{i,t+1}$) 為兩個詞間的距離，介於 {-5, 5}，並在同一個句子中。例如<夏季、活動、北海道> 在 d_1 的距離為 (2+1)/2 =1.5，在 d_2 的距離為 (3+|-4|)/2 =3.5。$dist_i$的倒數可以代表近似樣式的相似度，介於[0, 1]之間。

由C_k產生L_k的過程，以k=3為例，整理如圖7。

C3	TF	文件	support
<夏季、花、北海道，1.5>	2	d_1	2
<夏季、活動、北海道，1.5>	1	d_1	2
<夏季、活動、北海道，3.5>	1	d_2	

>= min support

L3	TF	文件	support
<夏季、花、北海道，1.5>	2	d_1	2
<夏季、活動、北海道，1.5>	1	d_1	2
<夏季、活動、北海道，3.5>	1	d_2	

圖7：由C_3產生L_3

步驟六：重複步驟五，直到無C_k候選項目產生後，即停止。最後會有近似高頻樣式集合 L_2, L_3, …, L_k，其TF、頻率和距離等資訊都會存下來，之後計算都會用到這些資料。

三、特徵項權重計算

（一）樣式加權頻率(Pattern Weighted Frequency, pwf)

在同一篇文件，有相同特徵樣式但距離不同的情形，所以要把所有出現的樣式頻率依據距離加權。樣式加權頻率可由公式(3-2)求得。

$$pwfreq_{i,j} = \sum_{dist_i} \frac{freq_j(p_i < w_{i,1}, w_{i,2}, \ldots w_{i,l_i}, dist_i >)}{dist} + \varepsilon \quad (3\text{-}2)$$

i：第i個特徵樣式

j: 第j篇文件

P_i: 特徵樣式p_i由l_i個詞組成，距離為$dist_i$, 如公式(3-1)

$freq_j(p_i < w_{i,1}, w_{i,2}, \ldots w_{i,l_i}, dist_i >)$: 特徵樣式$p_i$在文件$d_j$，距離為$dist_i$的頻率。

$pwfreq_{i,j}$：特徵樣式p_i在文件d_j，各種距離的加權頻率(weighted frequency)。

即使在同一篇文件，$dist_i$也可能會有很多種，故必須把所有可能的$dist_i$相加。當l_i=2,

$dist_i$∈{1, 2, 3, 4, 5};當l_i=3, $dist_i$∈{(1+1)/2, (1+2)/2, (1+3)/2, (1+4)/2, (1+5)/2, (2+2)/2, (2+3)/2, (2+4)/2, (2+5)/2, (3+3)/2, (3+4)/2, (3+5)/2, (4+4)/2, (4+5)/2, (5+5)/2}，依此類推。

此外距離越遠相似度越小，故將頻率除以距離，形成以相似度加權。為了避免其值為0，以最簡單的平滑化(smoothing)處理，即加上一個很小的$\varepsilon(10^{-4})$。以d_1中p_i = <夏季、北海道> 為例：$pwfreq_{i,1}$ = (1 / 3 + 1 / 1) + 0.0001 = 1.33343…

（二）樣式加權頻率與反向文件頻率

根據Jones (1972) 提出的反向文件頻率(inverse document frequency, idf)，可以算出詞彙在文章的重要程度。整合idf的權重公式如 (3-3) 所示。

$$w_{i,j} = pwf_{i,j} \times idf_i$$

$$pwf_{i,j} = \frac{pwfreq_{i,j}}{\max_l pwfreq_{l,j}} \tag{3-3}$$

$$idf_i = \log \frac{N}{n_i}$$

$pwf_{i,j}$: 特徵樣式p_i在文件d_j的正規化加權頻率

max$_i$: $pwfreq_{i,j}$文件d_j中加權頻率最大的特徵樣式值

idf_i : 特徵樣式p_i的反向文件頻率

n_i : 包含特徵樣式p_i的文件數量

N : 文件集合的數量

四、密度和相似度為基礎之二階段分群演算法（Two-Phase Density and Similarity-Based Clustering Algorithm）

本研究提出兩階段的分群演算法，第一階段「密度為基礎」，將DBSCAN (Ester et al. 1996) 中以距離小於半徑為鄰居，改為相似度大於門檻θ為鄰居。θ若設定太小，會使很多文件匯聚到同一群裡；反之θ設定太大，會使文件分佈於眾多小群裡，θ的決定影響分群的結果。我們的作法是設定中等的θ，再進行第二階段「相似度為基礎」的小群聚合 (Zamir and Etzioni 1998)，將共同特徵超過門檻　的相似兩群合併。

在進行第一階段分群，不使用「距離」尋找鄰居，而是用相似度，因為相似的文件所擁有的特徵項具有類似的比例，而不是以多寡表示。兩篇類似的文件，但一篇精簡扼要，一篇是長篇大論，其特徵權重所形成的向量，以歐幾理德距離計算必然差距頗大；但是以兩個向量夾角的餘弦來計算 (Salton and Buckley 1988)，介於[0, 1]之間，夾角越小表示相似度越高，更為合理。

例如，特徵維度2, $\vec{d_1}$=(1, 2), $\vec{d_2}$=(100, 200)，兩篇文件的歐幾理德距離為$\sqrt{(100-1)^2+(200-2)^2} \cong 221$，距離遙遠，差異頗大，但兩個向量重疊只是長度不同，夾角0度，餘弦值為$\frac{1*100+2*200}{\sqrt{1^2+2^2}*\sqrt{100^2+200^2}}=1$，代表相似度最大。

第一階段密度為基礎：

1. 每個文件視為各自獨立的一群。
2. 設定文件鄰近相似度門檻值θ、鄰居數量門檻值Minpts。
3. 計算文件間的相似度，針對每份文件，找出相似度大於θ的鄰居。
4. 選擇鄰居數大於Minpts的d_i，即d_i為核心文件，且與其鄰居聚集成群。
5. 從群中的各個鄰居再延伸找出其鄰居文件（相似度可達），並將該鄰居包含的鄰居合併到此群。根據「相似度連接」的概念，重複此步驟，繼續擴充合併，直到找不到鄰居。
6. 選擇下一篇核心文件，重複步驟4和5。
7. 剩下未被合併或處理過的文件，若有文件擁有鄰居，但鄰居數未達 Minpts（非核心文件），仍予以合併成群；若只有單獨一篇，則為偏離文件(outlier document)，單獨成群。

第二階段相似度為基礎：

1. 計算任兩群相似度，以共同特徵（近似高頻樣式）所佔比例表示
2. 相似度大於門檻 τ 的兩群，予以合併。

$$similary(C_i, C_j) = \frac{|C_i \cap C_j|}{\max(|C_i|, |C_j|)} \tag{3-4}$$

$|C_i|$：第i群所有的近似高頻樣式數

$|C_j|$：第j群所有的近似高頻樣式數

$|C_i \cap C_j|$：第i群和第j群所共有相同的近似高頻樣式數

$\max(|C_i|, |C_j|)$：近似高頻樣式數較大者

肆、實驗評估及分析

一、文件集

（一）中文文件集

實驗中所採用的中文文件集是Yang & Yu (2006) 利用Spider軟體隨機收集2006年2月20日至2006年3月20日內的Google新聞台灣版的網路新聞，分別是財經(Business)、科技(Tech/Science)、體育(Sports)、綜合一 (Blend 1) 和綜合二 (Blend 2)等五個文件集，如表2；主題數是根據Google News的定義，Google News的每一則新聞下面都有「所有X則相關新聞」的連結，將這些相關新聞視為同一主題；所謂相關其實並不一定代表是同一主題，但此文件集是以此來當作標準答案。

表2：中文文件集

類別	篇數	主題數	只含一篇文件的主題數
財經Business	8	3	1
科技Tech/Science	17	3	0
體育Sports	20	6	3
綜合一Blend 1	145	43	18
綜合二Blend 2	251	56	20

（二）英文文件集

本研究採用的文件集是Reuters-21578 中的子文件集 Reuters Transcribed Subset，是從Reuters-21578中的前10個大類別中，每個類別選出20篇出來，總共200篇文件。這些文件是由3個Indian演講者和Automatic Speech Recognition (ASR)系統所產生的文件檔。

目前，我們是將10個類別當成10個主題，以分成此10群為標準答案，但實際每個類別內的文件主題差異極大，這樣的文件集勢必大大影響分群結果評估的客觀性。雖然同類別內文件的共同特徵可能不是非常多，但期望不同類別間的文件差異更大，依然能夠利用分群演算法區別之。

二、分群品質評估方法

（一）召回率(Recall)、精確率(Precision)、F-measure、正確率(Accuracy)

在資訊檢索中，最常被用來衡量的方式為召回率(Recall)與精確率(Precision)。召回率越高，代表正確的文件都能被檢索出來；精確率越高，代表檢索正確率很高，不相干的誤判文件較少。由於精確率與召回率兩者之間存在著相反的相互依賴關係，為同時兼顧精確率和召回率，另外使用調和平均數F-measure，比較檢索的成效 (Baeza-Yates and Ribeiro-Neto 1999)。

一般評估分群品質乃延伸召回率、精確率、調和平均數的定義如公式(4-1)～(4-3)。由於文件集有很多主題，需要同時評估，所以把召回率、精確率和調和平均數皆採用Average方式來評估，公式中的K是指正確的主題數。$|R_i|$是每個主題包含的文件數，$|R_{ai}|$是所對應的群集內與R_i相同的文件數，$|A_i|$則為所對應的群集包含之文件數。

$$Average \quad Recall = \frac{1}{K}\sum_{i=1}^{K}\frac{|Ra_i|}{|R_i|} \tag{4-1}$$

$$Average \quad Precision = \frac{1}{K}\sum_{i=1}^{K}\frac{|Ra_i|}{|A_i|} \tag{4-2}$$

$$Average \quad F-measure = \frac{2\times AvgPrecision\times AvgRecall}{AvgPrecision + AvgRecall} \tag{4-3}$$

另外一種衡量整體調和平均數如公式(4-4)，每個類別找出在各群集中最類似的群，即F-measure最大的群，再按照每個主題的文件數量比例加權平均，F(C)即是整體分群的

品質。|D|是資料集全部文件數量，$|K_i|$是屬於第i類主題的文件數，K是資料集主題數，C是分群結果的群集數目。

$$F(C)=\sum_{i=1}^{K}\frac{|K_i|}{|D|}\max_{j\in\{1,2,\cdots,C\}}\{F(K_i,C_j)\} \tag{4-4}$$

$$F(K_i,C_j)=\frac{2\times Precision(K_i,C_j)\times Recall(K_i,C_j)}{Precision(K_i,C_j)+Recall(K_i,C_j)} \tag{4-5}$$

此外，為了瞭解文件分群的正確率，再計算Accuracy，即分配到正確主題的文件數除以總文件數 (4-6)。

$$Accuracy=\frac{\sum_{i=1}^{K}|Ra_i|}{N} \tag{4-6}$$

（二）均方誤差(Mean of Square Error)

集群分析主要是達成群內相似度最大，群間相似度最小，一般以均方誤差(Mean of Square Error, MSE)來作為分群品質的績效指標。

$$MSE=\frac{\sum_{C_j}\frac{\sum_{x_i\in c_j}(x_{i1}-\overline{x_{j1}})^2+(x_{i2}-\overline{x_{j2}})^2+\ldots+(x_{it}-\overline{x_{jt}})^2}{N_j}}{C} \tag{4-7}$$

C：群集數目

N_j：第C_j群文件數量

$(x_{i1},x_{i2},\cdots,x_{it})$：第$i$篇文件資料點$x_i$，共$t$個維度

$(\overline{x_{j1}},\overline{x_{j2}},\cdots,\overline{x_{jt}})$：第$C_j$群之群中心

（三）平均群內相似度 (Mean of Intra-Similarity)

由於文件分群並非以距離為聚集標準，以各維度平均當作群中心亦不合理，MSE似乎並不適合衡量文件分群的品質，故我們提出平均群內相似度(Mean of Intra-Similarity, MIS)來作為分群品質的績效指標，其值是將每一群內的文件兩兩之間的相似度相加求平均，介於[0, 1]之間，值越大越好，代表群內相似度越高。

$$MIS=\frac{\sum_{C_k}\frac{\sum_{d_i\in C_k}\sum_{d_j\in C_k}sim\left(\overrightarrow{d_i},\overrightarrow{d_j}\right)}{N_k*(N_k-1)*1/2}}{C} \tag{4-8}$$

C：群集數目

N_k：第C_k群文件數量

$sim(\overrightarrow{d_i},\overrightarrow{d_j})$：$C_k$群內任兩篇文件向量夾角的餘弦值

三、實驗結果

使用Apriori algorithm結合近似樣式匹配，所產生的近似高頻樣式(Approximate Frequent Patterns)，列於表3，並與單一詞彙、相鄰詞 (bigram)、彈性詞對在各文件集特徵抽取總數量作比較。

表3：各文件集特徵抽取總數

特徵 \ 文件集	Business	Tech/ Science	Sports	Blend1	Blend2	Reuters Transcribed Subset
名詞總數	300	589	604	3132	4136	8110
Bigram	28	253	214	1584	2956	n/a
彈性詞對	94	227	447	2618	5841	748
近似高頻樣式	98	693	217	3408	8437	1329

而在Tech/Science、Blend 1、Blend 2和Reuters Transcribed Subset文件集，最多可以找到五個詞彙所組成的近似高頻樣式，如表4。

表4：各文件集所找到近似高頻樣式的數目

文件集 \ 樣式長度	2	3	4	5
Business	74	21	2	0
Tech/Science	346	249	89	9
Sports	171	43	3	0
Blend 1	1869	1097	372	70
Blend 2	4359	2985	947	146
Reuters Transcribed Subset	833	353	120	23

〈實驗一〉 以中文文件集評估特徵抽取

我們以中文文件集來評估本研究提出之特徵抽取方法是否能提升分群品質，所以分群方法暫時使用和楊及王(2007) 一樣的「遞迴合併高相似度資料 (Recursive Merging High Similar Data)」，以便能夠單純評估特徵抽取的效果。首先在Business、Tech/Science和Sports皆是屬於較小的文件集，在Average Recall、Average Precision、Average F-measure和Accuracy上皆和「彈性詞對」一樣，分群效能都可以達到100%，在Business和Sports兩個文件集中的單獨文件也可各自成群。我們將以Blend 1和Blend 2文件集來詳細說明。

表5：Blend 1和Blend 2文件集分群評估表

Blend 1						
特徵抽取	特徵數	Recall	Precision	F-measure	Accuracy	群數
彈性詞對	2618	94.12%	99.22%	96.60%	95.17%	48
近似高頻樣式	3408	96.18%	99.61%	97.87%	96.55%	47
Blend 2						
特徵抽取	特徵數	Recall	Precision	F-measure	Accuracy	群數
彈性詞對	5841	92.42%	94.75%	93.57%	87.25%	59
近似高頻樣式	8437	93.38%	95.34%	94.35%	88.05%	57

表5為兩個較大的實驗集，可以看到實驗結果在Average Recall、 Precision、F-measure和Accuracy上皆比「彈性詞對」好，所獲得最佳群數也較接近Google News所定義的相關主題數43與56。

●分析一：正確率提昇

例如有一篇文件主題是「黃大仙靈驗：泰來否極」，在以彈性詞對為特徵時單獨成群，因為該篇文章較短，找出來的特徵項有限，在我們的研究方法中，可以找出由兩個詞彙以上組成的特徵項，所以經由特徵項增加，找到了與其相似的文件分配到同一個主題群，使得所分出的群數也較接近標準群數。

另外有兩個主題群，第一個主題群相關文件內容較偏向韓劇，如「抵制韓劇？韓國根本不在意」和「姚文智訓媒體：沒說禁韓劇 多讀點書」，第二個主題群則為「標下國華產險　台壽保：將成立龍平安產險公司」。以彈性詞對為特徵時，第一個主題群其中有一篇文件主題是「黃金時段七成自製？有困難」，被分到第二主題群，藉由本研究方法找出更多詞彙組成的特徵項，使得該篇文件分到正確的第一群。故本研究所提出的特徵抽取方式，可使文件被錯分的比率變低，進一步使得所分出的群數較接近標準主題數，提昇正確性。

●分析二：錯誤原因

實驗最佳分群數較資料集主題數略多，觀察發現有一些文件單獨成群，即在分群時找不到其他相似的文件，這些非單獨文件而被單獨分成一群的文件，文件主題雖被Google News認定是與其他群相關，其內容大致上卻是不同的。例如有一篇文件主題是「換車牌招標規範放寬 ETC翻版」，其文章主要是寫換車牌相關的內容，但被Google News認為是和ETC相關的主題。在我們的特徵抽取中，該文件與ETC主題類似的特徵，除了詞彙“ETC”，其他較少，故無法聚集在同一群內。根據文件的實際內容判別，我們的分群結果並不能算是錯誤。

〈實驗二〉以英文文件集評估特徵抽取與分群方法

在實驗二，除了以英文文件來評估本研究提出的特徵抽取方法，也要驗證本研究提出的「密度和相似度為基礎之二階段分群演算法」。

表6：Reuters Transcribed Subset文件集分群評估表

特徵抽取	特徵數	Recall	Precision	F-measure	Accuracy	群數
彈性詞對	748	57.97%	78.51%	66.70%	56.79%	17
近似高頻樣式(一階段分群)	1329	73.96%	61.23%	67.00%	80.25%	12
近似高頻樣式(二階段分群)	1329	74.41%	61.62%	67.42%	81.48%	10

本研究第一階段分群是以密度大的核心文件主導文件的聚集，為了避免快速聚集在少數群內，寧可成為分散的群集，如表6第2列；接著再以第二階段衡量群與群的相似度，合併相似度高的群，降低群數，如表6第3列。從結果可知，經由二階段分群，使得分群結果達到文件集正確的10個分類。

表6也可看出「近似高頻樣式」明顯比「彈性詞對」優秀。其中平均精確率較低的原因是，第1列中資料分散在17群，計算精確率時的分母較小(4-2)所致，故當群數不同而要比較結果的優劣時，以F-measure和Accuracy來衡量，較為公平。

根據研讀相關研究發現，很多學者將F-measure進一步以文件數量比例加權(4-4)，使得分群品質評估較不會受到群集大小不一的影響，可以更公平比較，故接下來我們也將採用加權F(C)與其他研究比較。

表7列出其他學者使用常見的特徵及分群方法的加權F(C)，由表可以發現實驗資料皆不盡相同，也都是再經過自行整理，文件數和類別數也不同，像Fung et al. (2003) 和Beil et al. (2002) 的實驗集都是整理出只含單一類別的文件，但在類別數，可以看到Fung et al. (2003) 的主題數65比Beil et al. (2002) 52多，主題數分的較細，可以使標準答案較正確，所以目前本研究所使用的Reuters Transcribed Subset分成10大類，應該可以再整理細分，使文件主題更為合理。

此外，大部分的學者採用階層式分群法，搭配二分K-means 或高頻項目集合樹狀結構，但是分群數都是已知的，即迴圈會停止在設定的主題數。這在實際的應用中其實是不可能的，我們無法預先得知一堆文件到底含有多少主題，所以我們所提出的分群方法，不用事先設定群數，由演算法決定適合的群數，更適合於文件分群。

根據研究整理，可以發現在不同實驗集，在預先設定停止群數下，其F(C)大多介在0.4～0.6 之間，而本研究無須預先設定群數，F(C)亦可達到0.5029，代表本研究方法可以有相當的分群品質。

表7：其他特徵與分群方法之比較

Authors	Dataset Source	Number of Documents	Number of Classes	F-Measure
Steinbach et al. (2000)	Reuters-21578	1504	13	單一詞彙 + UPGMA：F(C) = 0.5859 單一詞彙 + Bisecting K-means：F(C) = 0.5863
	Reuters-21578	1657	25	單一詞彙 + UPGMA： F(C) = 0.6855 單一詞彙 + Bisecting K-means： F(C) = 0.7067
Beil et al. (2002)	Reuters-21578	8654	52	Frequent Term + HFTC：F(C) = 0.49 單一詞彙 + Bisecting K-means：F(C) = 0.57 單一詞彙 + 9-secting K-means：F(C) = 0.43
Fung et al. (2003)	Reuters-21578	8649	65	Frequent Itemset + FIHC：Average F(C) = 0.6
	Reuters-21578	1504	13	Frequent Itemset + FIHC：Average F(C) = 0.45
Liu et al. (2005)	Reuters-21578	8654	52	Frequent Term + FTSC：F(C) = 0.46 Frequent Term + FTSHC：F(C) = 0.49 單一詞彙 + K-Means：F(C) = 0.57
本研究	Reuters-21578	200	10	Approximate Frequent Pattern + Two-Phase Density and Similarity -Based Clustering + Cluster Number Decision: F(C) = 0.5029

〈實驗三〉：分類 vs. 分群

由於按照主題分門別類的文件集取得困難，文件分群研究者經常使用新聞分類文件集來當作分群實驗的資料，新聞文件是依專家定義幾個大類別，同類別的文件並不代表為同一主題，如財經類文件，雖然都是與財經相關，但所描述的事件主題可以非常廣泛。然而集群分析主要是達成群內相似度最大，群間相似度最小，依據大類別作為標準答案，可能有失公允，故本實驗再輔以均方誤差(MSE)和平均群內相似度(MIS)來驗證分群的品質。

表8：文件集與分群結果之MSE & MIS

	MSE	MIS
Reuters Transcribed Subset	4.4126	0.1032
本研究分群結果	1.3776	0.1875

表8是本研究和Reuter Transcribed Subset 正確答案之MSE和MIS比較，可以看到本研究的平均群內均方誤差MSE較正確答案低，平均群內相似度MIS也較正確答案高，證明本研究分群結果之群內文件間有較高相似度。所以對照之下，雖然實驗二的F(C), F-measure 和 Accuracy 分別為0.5029, 67.42%, and 81.48%，並不是都非常高，但其實我們的分群品質是較好的。

伍、結論與未來方向

本研究的主要目的在於使大量文件如新聞等能以主題的方式聚集成群，研究方法主要分成特徵抽取與分群演算法兩部分。在特徵抽取方面，本研究提出「近似高頻樣式(Approximate Frequent Pattern)」，以關聯規則中高頻項目集結合近似樣式匹配，並將近似匹配的距離（相似度）納入特徵權重的衡量中，能有效抽取出由兩個詞彙以上所組成更多有意義且有區別力的特徵項。在分群演算法方面，本研究提出無須預先設定群數的「密度和相似度為基礎之二階段分群演算法 (Two-Phase Density and Similarity-Based Clustering Algorithm)，避免某些接近的文件匯聚成一大群，可先將文件分散成小群，再以相似度合併，以達到理想的群數。經過實驗驗證，本研究所提出的方法能夠提高文件分群的品質與正確性。

未來我們也將採用「未知群數的模糊分群演算法」（楊、邱 2007）利用模糊係數研究一篇文件同時屬於多個群的可能性。此外，改良適用於已知群數的演化式分群最佳化演算法，使其能夠應用在未知群數的文件分群。

參考文獻

1. 楊燕珠、王千豪，2007，『基於近似詞彙樣式匹配之主題式文件分群 Thematic Document Clustering Based on Approximate Word Pattern Matching』，第13屆海峽兩岸資訊管理發展與策略學術研討會，pp. 388-393。
2. 楊燕珠、邱瑞民，2007，『未知群數的模糊分群之研究 Fuzzy Clustering with Unknown Cluster Number』，ICIM 2007 第十八屆國際資訊管理學術研討會。
3. Agrawal, R. and Srikant, R.,“Fast Algorithms for Mining Association Rules,”*in Proceedings of International Conference on Very Large Data Bases*, Santiago, Chile, 1994, pp.487-499.
4. Al-Kofahi, K., Tyrrell, A., Vachher, A., Travers, T. and Jackson, P.,“Combining Multiple Classifiers for Text Categorization,”*in Proceedings of the Tenth International Conference on Information and Knowledge Management*, Atlanta, Georgia, USA, 2001, pp.97-104.
5. Baeza-Yates, R. and Ribeiro-Neto, B., *Modern Information Retrieval*, Addison Wesley, 1999.
6. Beil, F., Ester, M. and Xu, X.,“Frequent Term-Based Text Clustering.”*in Proceedings of International Conference on Knowledge Discovery and Data Mining*, 2002, pp.436-442.
7. Chen, F., Han, K. and Chen, G.,“An Approach to Sentence-Selection-Based Text Summarization,”*IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering*, (TENCON '02), 2002, pp.489- 493.
8. Dubes, Richard C. and Jain, Anil K., *Algorithms for Clustering Data*, Prentice Hall, 1988.
9. Ester, M., Kriegel, H.-P., Sander, J., and Xu, X.,“A Density-Based Algorithm for

Discovering Clusters in Large Spatial Databases with Noise," *in Proceedings of International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226-231.

10. Fung, B. C. M., Wang, K. and Ester, M., "Herarchical Document Clustering Using Frequent Itemsets," *In SIAM International Conference on Data Mining*, 2003.
11. Han, J., Pei, J., Yin, Y., "Mining Frequent Patterns without Candidate Generation," *in Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2000, pp. 1-12.
12. Jones, K.S., "A Statistical Interpretation of Terms Specificity and its Application in Retrieval," *Journal of Documentation*, Vol. 28, No. 5, 1972, pp.111-121.
13. Liu, X.-W., He, P.-L. and Wang, H.-Y., "The Research of Text Clustering Algorithms Based on Frequent Term Sets," *in Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou, 2005, pp.18-21.
14. Porter, M., "An Algorithm for Suffix Stripping," *Program*, Vol. 14, No. 1, 1980, pp.130-137.
15. Salton, G. and Buckley, C., "Term-weighting Approaches in Automatic Text Retrieval," *Information Processing & Management*, Vol. 24, No. 5, 1988, pp.513-523.
16. Salton, G. and McGill, M., *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.
17. Steinbach, M., Karypis, G., and Kumor, V., "A Comparison of Document Clustering Techniques," *in Proceedings of International Conference on Knowledge Discovery and Data Mining Workshop on Text Mining*, 2000.
18. Yang, Y.-J. and Yu, S.-H., "Chinese Text Clustering for Topic Detection Based on Word Pattern Relation," *AI-2006 The Twenty-sixth SGAI International Conference on Artificial Intelligence*, 2006, pp. 408-412.
19. Zamir, O. and Etzioni, O., "Web Document Clustering: A Feasibility Demonstration," *in Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, Vol. 6. Melbourne, Australia, 1998, pp. 46-54.