行政院國家科學委員會專題研究計畫成果報告
電子報系統中個人化技術的設計與實作
Design and Implementation of Personalization Technology in E-News Systems

計畫編號：NSC 89-2218-E-004-007
執行期限：89 年 8 月 1 日至 90 年 7 月 31 日
主持人：沈錳坤　政治大學資訊科學系

一、中文摘要

隨著全球資訊網技術的發展，電子報系統越來越普遍。讀者將面對資訊氾濫的問題。為了減輕讀者負擔，根據讀者喜好而發展的個人化電子報系統有其發展的必要。

本計畫將研究電子報系統中的個人化技術。我們所發展的個人化電子報系統將追蹤讀者的閱讀行為、學習讀者的喜好並且根據讀者的喜好，提供個人化新聞過濾、個人化評比、個人化主題分類及版面配置的功能。

此外，此系統也將提供社群推薦的功能。社群推薦的功能會根據相同族群的閱讀特徵，主動推薦相關的新聞。社群推薦的技術包括社群的形成與社群共同喜好的探勘。我們利用叢聚的技術，根據讀者的喜好或個人資料決定族群。對於族群喜好的探勘，我們利用關連法則來探勘同一族群讀者的共同喜好。

本計畫整合所發展的個人化技術，設計並實作此電子報系統。我們也測試並評估此個人化電子報系統的效果與效能。

關鍵詞：電子報、個人化、資料探勘、社群推薦、關連法則

Abstract

With the advance of web technology, e-news systems are becoming increasingly popular in our daily life. Users face the challenge on information overloading. It is necessary to develop the personalization e-news systems in order to reduce reader's information load with respect to reader's preference.

In this project, the personalization technologies in e-news system are investigated. The developed personalization e-news system will monitor the reader's behavior, learn reader's preference and support personalization filtering, ranking, topic classification and presentation according to reader's preference.

Besides, the system will also support the community recommendation function which recommends the news according to the access pattern of peer group. Two important design issues of community recommendation are the generation of peer groups and mining of the group preference. We develop the technique to determine the peer group, according to the preference or profile of readers, based on the clustering technology. For the mining of group preference, we use the association rule mining to discover the preference of users groups.

We design and implement the e-news system which integrates the developed personalization technologies. The effectiveness and efficiency of this personalization e-news system are also measured.

Keywords: E-News, Personalization, Data Mining, Recommendation, Association Rule

二、緣由與目的

隨著全球資訊網的發展，電子報系統也蓬勃發展。除了傳統的新聞報紙發展電子版的報紙，各式各樣不同主題的電子報也如雨後春筍般誕生。例如，PC Home **電腦報**(http://www.pchome.com.tw)**目前已有兩百萬的訂戶；又如台北市政府成立電子報發行中心的網站**(http://www.taipeilink.net)**，提供各個不同的社群或個人發行電子報。可以想見傳統由大眾媒體影響的社會將轉型成為分眾媒體的社會。**

目前國內現有電子報多提供兩種形式供讀者閱讀。第一種形式式網頁版的電子報。第二種形式是以 email **的方式，將電子報寄給訂閱的讀者。在** email **的內容中，除了本文之外，還提供鏈結到網頁版的新聞。**

但是，根據聯合報的新聞網站聯合新聞網(UDN News)**對訂閱讀者的調查**(http://www.udnnews.com/INFOTECH/INTERNET/245680.htm)**，讀者對於現有電子報的建議，主要包括**(1)**提供電子報匯集的網站，將新聞整理分類。**(2)**提供符合讀者興趣的新聞。**

因此，在新聞氾濫的現在，個人化電子報系統將有助於讀者閱讀新聞。

過去有關個人化電子報的研究，包括 push **與** pull **兩種不同的類型。其中麻省理工學院媒體實驗室的電子報系統** Fishwarp **的個人化功能，根據讀者的喜好呈現新聞。但是讀者的喜好是由讀者註冊時的問卷產生。另外** Fishwarp **也提供推薦新聞的功能，但是推薦是透過讀者票選的方式，將熱門的新聞推薦給每位讀者。因此，**Fishwarp **的個人化功能都必須藉助讀者主動的輸入相關資料。** NEC **也發展**

了一個個人化的電子報系統 ANATAGONOMY。ANATAGONOMY 觀察並根據讀者的閱報習慣，自動產生讀者的喜好資料，並依據讀者的喜好呈現新聞。但是，ANATAGONOMY 並沒有提供推薦的功能。

因此，一個以個人化技術為重點（尤其是推薦的功能）的個人化電子報系統應具有下列功能，(1)主動收集各大新聞報紙網站、新聞社網站的新聞。(2)觀察並自動學習讀者的閱讀興趣。(3)提供個人化新聞的過濾與評比(personalization filtering and ranking)：提供符合讀者喜好的新聞，並根據讀者的興趣高低，將新聞依序排放。例如，讀者對家鄉的地方政治新聞有興趣，此類的新聞就會排在醒目的地方。(4)提供個人化的新聞主題分類(personalization topic classification)：根據讀者的興趣，將新聞分類。例如，龍應台擔任台北市文化局長，有的讀者將其視為政治新聞，有的讀者視為文化新聞。此外，在階層式分類的形況下，每個讀者的分類階層也會根據讀者的喜好而有所不同。(5)提供個人化的版面配置(personalization layout)：除了將新聞依讀者興趣排放在醒目的位置外，系統可以根據讀者對版面風格的喜好，將新聞做不同風格的版面配置。(6)提供社群推薦的功能(community recommendation)：根據社群同好的閱讀新聞行為，主動推薦給讀者。(7)提供分類整理後的舊聞查詢。

在個人化新聞的過濾與評比方面，information retrieval、information filtering、information agent 等領域已有不少的研究。大部分的研究以 content-based filtering 的方式處理。Content-based filtering 以 vector space model 表示讀者的喜好。也就是說，喜好是由關鍵詞或主題組成的向量。而每則新聞根據萃取出來的關鍵詞（或由關鍵詞而判斷而出的主題），也以向量的形式表示。每則新聞與讀者喜好之間的關係因此以兩向量之間的相似度來衡量。

在社群推薦方面，我們將研究利用資料探勘(data mining)的技術來達到社群推薦的功能。資料探勘技術的主要目的在從大量資料中尋找有價值的的資訊或知識。有關資料探勘的研究，根據所尋找出的資訊可概分為 association rule、characterization and summarization、classification、outlier analysis、clustering、time series analysis 等。

其中與社群推薦較為相關的研究是 clustering 與 association rule。Clustering 的目的在將性質相近的資料叢聚在一起。Clustering 的技術在 statistics、pattern recognition、machine learning 的領域，已有不少研究成果。Data mining 中有關 clustering 的研究主要的特性在於大量資料的處理。相關的研究可概分為 partitioning、hierarchical 、density-based、grid-based、 model-based clustering 等不同策略。

社群推薦的功能包括社群組成、社群喜好分析、推薦機制三個研究議題。社群的組成條件可以根據讀者輸入的個人基本資料，也可以根據讀者的閱讀興趣。至於社群組成的技術可以利用 clustering 的技術將相同條件的讀者叢聚在一起而歸類成同樣的社群。

至於社群共同喜好分析，我們可以利用 association rule 的技術，將每位讀者的喜好視為一筆交易，將每個關鍵詞視為商品。因此，本計畫的重點之一是利用 vector space model 之 association rule 演算法，以探勘出社群的共同喜好。一旦探勘出社群的喜好，推薦機制就可以根據社群與個別讀者的喜好推薦新聞。

本計畫第一年度將先建立個人化電子報系統的雛型，提供新聞主動收集、個人化新聞過濾與評比、社群推薦，研究其相關技術。尤其是社群推薦的功能。

三、結果與討論

（一）系統架構：

本計畫發展的個人化電子報系統，具有主動收集新聞、觀察並自動學習讀者的閱讀興趣、提供個人化新聞的過濾與評比、提供個人化的新聞主題分類、提供個人化的版面配置、提供社群推薦、提供舊聞查詢的功能。

本系統分成 server、client 兩部分，如圖 1 所示。Server 端負責所有計算、mining 及推薦的工作，包括取得新聞、進行社群推薦，以及當使用者上線時，幫使用者進行 content based filtering。Client 端只負責呈現個人化排列給使用者，並且記錄使用者的喜好。

本系統之所以選擇如此的方式，是考慮到如果把新聞過濾器或是把社群推薦引擎移到 client 端的話，server 端必須把有關使用者的資訊，以及全部未過濾的新聞都先傳給 client 端，才能在 client 端進行過濾新聞或是社群推薦引擎的工作。然而，這樣會把不必要的新聞也傳給 client 端，額外的增加網路頻寬以及網路等待的時間。因此我們選擇 thin-client 的方式，雖然 server 端會因為同時上線人數的增加而加重 load，不過可以因為節省下不必要的網路傳輸，相對之下節省使用者等待的時間。

所以，本系統的 server 端分成兩個 database 以及五個 module。兩個 database 是 User Database 與 News Database。User database 負責記錄使用者的相關資訊，例如：帳號、密碼、使用者喜好（user profile）、閱讀記錄等。News database 負責儲存本系統的所有新聞以及新聞特徵（news profile）。五個 module 分別為新聞蒐集器（News Collector）、新聞分析器（News Parser）、社群推薦引擎（Mining and Recommend engine）、新聞過濾器（News Filter）以及學習引擎（Learning Engine）。新聞蒐集器負責蒐集新聞，再將新聞傳給新聞分析器，從新聞內文擷取出 news profile。社群推薦引擎負責將使用者分群後利用 mining association rule 進行社群推薦。新聞過濾器負責利用 content based filtering 幫使用者過濾新聞。學習引擎負責在使用者結束閱讀新聞後更新 user profile。

Client 端分成使用者識別者（User Identifier）、呈現者（Presenter）及觀察者（Observer）等三個部分。使用者識別者負責辨別出使用者。呈現者負責將 server 端過濾後的新聞，以個人化排列的方式

呈現出來。觀察者負責觀察使用者的喜好，並將結果傳回 server 端的學習引擎。

（二）系統流程：

本系統主要的運作流程可分為三部分，第一為加入新聞的流程，第二為個人化推薦的流程，第三為社群推薦的流程。

第一部分為加入新聞的流程。新聞蒐集器（News Collector）利用 spider 從外面電子報網站（如中國時報，聯合新聞網）蒐集新聞或是利用 innbbsd 從附近的 news server 抓取中央社的 news group 回來後，把新聞傳給新聞分析器（Parser）。Parser 接到從 News Collector 傳來的新聞後，利用中文關鍵詞擷取的技術，擷取出每篇新聞的關鍵詞。並將每篇新聞的關鍵詞、報紙名稱、新聞標題、系統時間、新聞的內文，加入 News Database 之中。

第二部分為個人化推薦的流程。當使用者從 client 端的使用者識別者（User Identifier）登入後，server 端的 News Filter 就會根據 User Database 中該使用者的喜好（user profile），來過濾送往 client 端的新聞。News Filter 此時會把新聞的 news profile 跟 user profile 的關鍵詞向量做向量內積，再依內積結果過濾新聞。之後再把過濾完的結果傳給 client 端的呈現者（Presenter）。呈現者依 Filter 計算的內積結果的高低產生個人化版面配置，分開呈現今日新聞及社群推薦的新聞，如圖 2 所示。並即時的從 server 端取得 user 想看的新聞內文，並呈現新聞如圖 3。在呈現者呈現新聞時，觀察者（Observer）開始觀察使用者閱過那些新聞，並且喜歡什麼新聞。當使用者結束閱讀新聞後，觀察者將觀察結果送回 server 端，此時學習引擎（Learning Engine）會依據觀察者的觀察結果，更新 user profile。

第三為社群推薦的流程。當使用者從 client 端的使用者識別者登入後，server 端的 news filter 就會根據 user database 中該使用者的社群推薦，來過濾送往 client 端的新聞。News filter 此時會把新聞的 news profile 跟社群推薦的關鍵詞向量做向量內積，再依內積結果過濾新聞。之後再把過濾完的結果傳給 client 端的呈現者（Presenter）。呈現者同樣地依 Filter 計算的內積結果的高低產生個人化版面配置，並以相同的介面，呈現社群推薦的新聞。並即時的從 server 端取得 user 想看的新聞內文。在呈現者呈現新聞時，觀察者（Observer）開始觀察使用者閱過那些新聞，並且喜歡什麼新聞。當使用者結束閱讀新聞後，觀察者將觀察結果送回 server 端，此時學習引擎（Learning Engine）會依據觀察者的觀察結果，更新 user profile。

（三）作業系統與程式語言

本系統的伺服器端在 FreeBSD 的平台，並利用 C 語言來實作。而本系統的客戶端選擇用跨平台的 Java Applet 實作。至於 client 與 server 溝通的部分，我們選擇用一個簡化的 protocol，而不用 HTTP protocol。因為如果選用 HTTP protocol 以及 CGI 的方式，會增加 http server 的 load。於是我們把 server 端的各個必須要跟 client 連線傳送資料的 component 都獨立成 socket server 來接受 client 端的連線。

（四）伺服器端元件

如前所述，本系統的 server 端分成兩個 database 以及五個 module。

在我們的 user database 中主要由一個 index 檔案跟使用者相關目錄所組成。在 index 檔案中記錄 user 登記的 username 以及編碼過的密碼。至於其他相關的資訊則放在使用者相關目錄下。存放在個人目錄下的相關資訊有：（1）在 48 小時之內，該使用者閱讀過的新聞。（2）該使用者的 user profile，這個 structure 中有兩個欄位，一個是 keyword 的編號，另一個是記錄 keyword 的加權值。（3）由社群推薦所產生的關鍵詞向量，格式同上面的 user profile。

News database 是由一個主要的 index 檔案及兩個目錄所組成的。兩個目錄分別存放新聞的內文，以及經由新聞分析器所擷取出來的關鍵詞。每當加入一則新聞時，便在 index 檔案中加入該新聞的標題，新聞加入的時間，及所屬的報紙名稱。並在此兩目錄下存入新聞內文及擷取出來的關鍵詞。

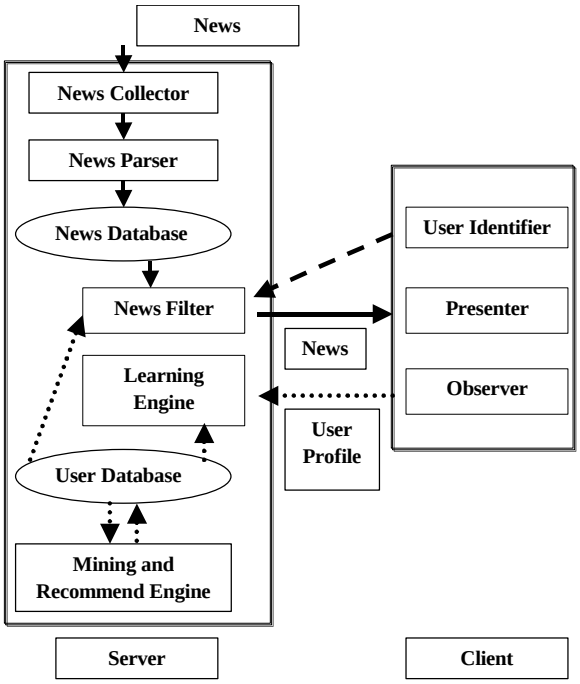

圖 1 系統架構圖

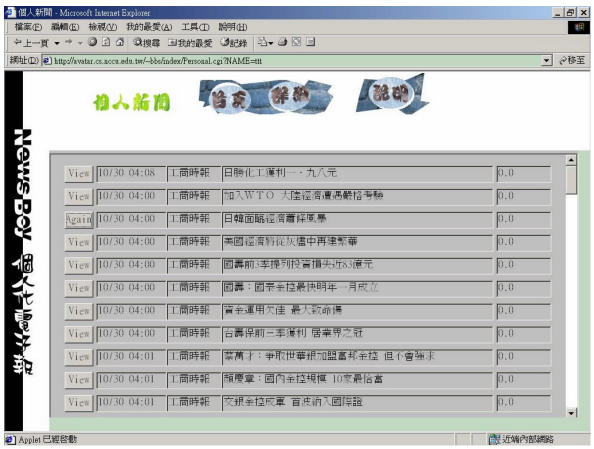

圖 2 呈現者呈現個人化推薦新聞

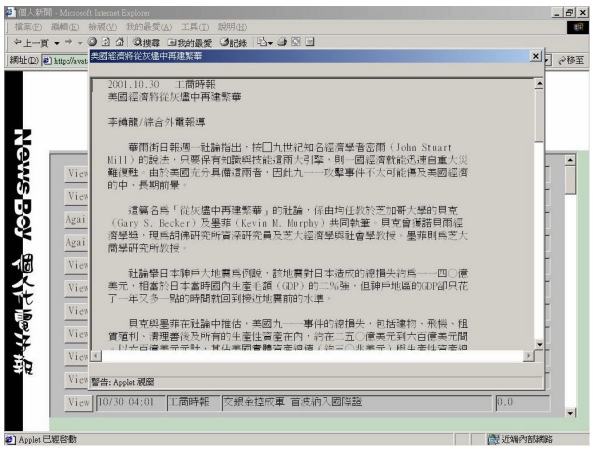

圖 3 呈現者呈現新聞內容

新聞蒐集器（News Collector）實作出兩種蒐集新聞的方式，第一種是利用 http protocol 到新聞網站抓取新聞，另外一種則是利用 nntp protocol 抓取 newsgroup 上的新聞群組。目前利用第一種經由 http protocol 的方式抓取新聞的報紙有中國時報、中國晚報及工商時報。而經由 nntp protocol 方式抓取新聞的有中央社的簡明新聞稿。

新聞分析器（News Parser）擷取關鍵詞的方法是應用常見中文關鍵詞擷取技巧。現今主要有三種中文關鍵詞擷取的方法，詞庫比對法，文法剖析法及統計分析法。第一種為詞庫比對法：從已經有的詞庫中，一一的比對關鍵詞有否出現在輸入的文句中。這種比對法不需要自然語言的技術，可是無法擷取出在該詞庫未出現的關鍵詞，例如現代的人名或新設立的機關名。第二種為文法剖析法：利用自然語言處理技術的文法剖析程式，來剖析出文件中的名詞片語。由於因為運用到自然語言處理技術，因此如果是出現在標題，書目等的單獨不成句的關鍵詞就無法擷取出來。第三種方法為統計分析法：經由分析文件，計算每一個字詞出現的頻率，取出大於某一頻率的字詞，就是關鍵詞。但是因為沒有參考辭庫，所以可能發生取出無意義的字詞。可是卻可以利用來擷取出辭庫所未收集的專業用語。

而本系統則採用其中的辭庫比對法，因為本系統主要注重在個人化的方面。而辭庫比對法，是較為簡單的方法。於是，我們使用了由 University of Illinois Urbana-Champaign 的蔡志浩先生編輯的中文辭庫，全辭庫共有 137, 450 筆詞彙。在找出關鍵詞之前，我們先以逗號、句號等標點符號為斷句的單位，再以長詞優先法找出每個句子內的關鑑詞。當把整篇文章的關鍵詞都找出來後，再把重覆過二次以上的關鍵詞連同本文分別存入 news database 中。

在社群推薦引擎（Mining and Recommend engine）開始社群推薦時，首先要決定的參數是利用 clustering 方法時，共要分成幾組，目前本系統先設定分成 10 組，亂數選出十個 user 的 user profile 當 seed。再依這十個 seed 組成 10 組 cluster。之後再利用 mining association rules 的方法分別找出這十組 cluster 的 large itemsets 後，就可以很直覺的產生該組 cluster 的 association rules。再依據該組 cluster 的 association rules 更新使用者的社群推薦的關鍵字向量。考慮系統效率，本系統並非即時而是每天做一次社群組成與社群喜好分析。

當新聞過濾器（News Filter）被啟動後，除了會從 user database 那邊取出 user profile 及社群推薦的關鍵字向量外，還會從 news database 那邊取出使用者尚未閱讀的新聞標題及 news profile。然後再以 content based filtering 的方法，一一的把 news profile 跟 user profile 及社群推薦的關鍵字向量做向量內積。再依向量內積的結果過濾新聞。過濾後剩下的新聞再交由 client 端的呈現者（Presenter）呈現給使用者。

當使用者結束某次 session 的連線之時，學習引擎（Learning Engine）就會被啟動。這個時候，學習引擎會依據 client 端上觀察者的觀察，將使用者的相關資訊加以更新，將使用者的相關資訊更新完後，再將結果送回 server 端的 user database 儲存。

而更新的情形可以分為兩種，第一種情形是使用者閱讀過的新聞。由於我們認為閱讀的順序也會隱含著喜好，越先閱讀的新聞，使用者的喜好程度越高。因此，我們會依據閱讀的順序來調整 news profile 的 weight。

$$u = u + \sum_{i=1}^{t} n_i \cdot r_i \cdot w_i$$

其中，u 代表 user profile，t 代表使用者在這個 session 中閱讀的新聞總數，n_i 代表使用者閱讀的第 i 則新聞的 profile，並且依使用者閱讀的順序排列，r_i 代表觀察者給第 i 則新聞的 weight，w_i 代表系統給第 i 則新聞的 weight。而 w_i 可以由下面的公式得到。

$$w_i = N - i$$

其中，N 是呈現給使用者的新聞總數。

而更新的另一種情形，則是使用者並未閱讀的新聞。這類新聞，我們認為使用者不喜歡，因此系統會依下面的公式調整 user profile。

$$u = u - \sum_{j}^{N-t} n_j \cdot w$$

其中，u 代表 user profile，N-t 代表使用者在這個 session 中未閱讀的新聞總數，n_j 代表第 j 則新聞的 profile，w 代表系統給未閱讀新聞的 weight。而本系統目前的 weight 選為 0.2。

（六）客戶端元件

Client 端分成使用者識別者（User Identifier）、呈現者（Presenter）及觀察者（Observer）等三個部分。以下將分別詳細介紹。

當使用者一連進本系統時，就會叫出使用者識別者（User Identifier），提示使用者登入本系統。使用者登入後，使用者識別者就會啟動新聞過濾器，依 user profile 及社群推薦的關鍵字向量過濾新聞。

呈現者（Presenter）在把所有從經新聞過濾器過濾過的新聞，依據使用者設定的個人化版面配置，一一呈現，而且是個人化推薦和社群推薦的新聞分開呈現後。在使用者按下閱讀的按鈕之後，呈現者會馬上連回 server 端的 news database，取得使

用者想看的新聞內文，並跳出另外一個視窗，把新聞內文顯示在裡面。並且啟動觀察者（Observer），觀察使用者。

觀察者（Observer）得知使用者喜愛那種類型的新聞可能有幾種方法，其中一個方法是直接問使用者。可是使用者不一定會回答。另一個方法就是觀察法。我們假設，當所有的新聞都被呈現者呈現給使用者後，使用者會選他們有興趣的新聞看。不看的就是對該則新聞沒興趣。而且，看新聞的順序中也隱含有喜好的順序。因此，觀察者會記錄使用者看了那些新聞，連同看的順序會會記錄下來。當使用者結束這一次的閱讀時，觀察者就會連上 server 端的學習引擎，並把觀察的結果傳送給學習引擎。

（五）討論

決定使用本系統的使用者心中的滿意度，可以取決於兩個面向：一是 Server 端過濾新聞的速度。二是個人化新聞推薦的滿意度。

關於第一點，我們測量在 server 端，從查詢使用者是否在 user database 中到把使用者相關資訊及新聞相關資訊經由 socket 傳送完的時間。而本報告中測量的方法是同時產生出 50 到 300 個 tasks，同時跟 server 端要求過濾新聞，測量四次後再計算 server 上每一個 task 的 Turnaround Time 平均值。

本實驗在 server 端的配備為：Pentium III 450 的 CPU，512MB RAM，作業系統為 FreeBSD 5.0-Current，網路環境為 100 Mb/s Fast Ethernet，次要儲存媒介為容量 13Gb 的 IBM-DTTA-350640 with UltraDMA-33。而平均每次傳輸的使用者相關資訊共 11472 bytes，625 個關鍵詞。

根據本文的作法，是先由 Server 端把所有新進，使用者尚未閱讀的新聞都讓過濾器進行過濾。在表 1 中可以看到使用者尚未閱讀的新聞數從 50 則漸增到 300 則新聞時，以及同時連線的使用者數從 50 人增加到 250 人時，本系統在 server 端的每一 task 的 Turnaround Time 的變化情形。

至於第二個面向，因為使用者的反應普遍不錯，而且又缺乏客觀的評量方式，因此目前我們並沒有做更具體的比較。

表 1 Server 端的平均 turnaround time（單位：秒）
* P 代表新聞則數，U 代表同時上線人數

U \ P	50	100	150	200	250	300
50	0.574	1.343	6.428	8.559	10.914	12.635
100	1.154	2.431	9.743	12.808	15.975	19.679
150	2.300	4.664	13.176	16.957	21.281	25.556
200	3.447	6.935	16.765	21.723	26.760	32.483
250	4.640	9.228	19.752	26.275	33.704	38.592

四、計畫成果自評

本研究計畫發展了個人化電子報系統的雛形。本系統所具有的功能包括：

1. 主動收集各大新聞報紙網站、新聞社網站及 newsgroup 的新聞。
2. 觀察並自動學習讀者的閱讀興趣。
3. 提供個人化新聞的過濾與評比。
4. 提供社群推薦的功能。

五、參考文獻

[1] Pascal R. Chesnais, Matthew J. Mucklo, and Jonathan A. Sheena, The Fishwarp Personalized News System. Proceedings of the 1995 Second International Workshop on Community Networking, Princeton, NJ, June, 1995.

[2] David Goldberg, David Nichols, Brain M. Oki, and Douglas Terry, Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM, Vol.35, No.12, pp. 61-70, December, 1992.

[3] Tomonari Kamba, Krishna Bharat and Michael C. Albers, The Krakatoa Chronicle - An Interactive, Personalized, Newspaper on the Web. Proceedings of the 1995 Fourth International World Wide Web Conference,Boston, MA, December, 1995.

[4] Brian Kantor and Phil Lapsley, Network News Transfer Protocol - A Proposed Standard for the Stream-Based Transmission of News Transmission of News. RFC 977, February, 1986.

[5] Kaoru Kobayashi, Yasuyuki Sumi, and Kenji Mase, Information Presentation Based on Individual User Interests. Proceedings of 1998 Second International Conference on Knowledge-Based Intelligent Electronic System, IEEE, April, 1998.

[6] Berners-Lee, R. Fielding, and H. Frystyk, Hypertext Transfer Protocol - HTTP/1.0. RFC 1945, May, 1996.

[7] Hidekazu Sakagami and Tomonari Kamba, Learning Personal Preferences on Online Newspaper Articles from User Behaviors. Proceedings of the 1997 Sixth International World Wide Web Conference, Senta Clara, CA, April, 1997.

[8] Hidekazu Sakagami, Tomonari Kamba, Atsushi Sugiura, and Yoshiyuki Koseki, Effective Personalization of Push-Type Systems Visualizing Information Freshness. Proceedings of the 1998 Seventh International World Wide Web Conference, Brisbane, Australia, April,1998.

[9] Philip S. Yu, Data Mining and Personalization Technologies. Proceedings of International Conference on Database Systems for Advance Applications, HsinChu, Taiwan, ROC,1999.

[10] 會元顯，關鍵詞自動擷取技術之探討，中國圖書館學會會訊 106 期，九月，1997。