The Personal Web-Based Bibliographic Management System

Student: Li-Chun Chen Advisor: Dr. Hao-Ren Ke, Dr. Wei-Pang Yang

Institute of Computer and Information Science

National Chiao Tung University

ABSTRACT

A bibliographic management system facilitates the storage and management of the bibliographic information on documents that are important for user. Features of a bibliographic information like authors, published year, and abstracts are stored in the system to represent one document. A user can personalize his/her folders in terms of user-configurable categories. In addition to the above-mentioned functions, the personal bibliographic management system that we design also recommends documents relevant to an individual user. The idea is to analyze semantic meaning of documents. Documents with the same semantic meaning will be grouped into a cluster. If someone has documents belonging to one cluster, our system will recommend him other documents belonging to this cluster. Word Sense Disambiguation (WSD) is employed to discover the semantic meaning of a document. In this thesis, we propose a new method for Word Sense Disambiguation. The method is based on lexical chains and employs the taxonomy of WordNet. On the basis of the strength of the relationship between words, automatic disambiguating of word semantics can be accomplished by giving different weights. An evaluation of the method was done on Semantic Concordance Corpus (SemCor). The average percentage of correct resolutions achieved was 65.26%.

Keywords: Bibliographic Management System; Personalized Recommendation; Semantic Analysis; Word Sense Disambiguation; Lexical Chain

線上個人化參考文獻管理系統

研究生: **陳莉君**　　**指導教授**: **柯皓仁博士，楊維邦博士**

國立交通大學資訊科學研究所

摘 要

參考文獻管理系統提供使用者儲存與管理文件的功能，文件儲存主要以記錄特徵(Feature)**的方式，例如文件作者、出版年份、內容摘要等來代表某篇文件。而蒐集的文件可以依照使用者自定的類別來歸類，達到個人化管理的功能。**

本論文提出的個人化文獻管理系統除了上述機制外，並且能夠做到個人化文獻推薦的功能，其概念為分析文件內容語意，亦即語意歧異解析(Word Sense Disambiguation, WSD)**，將相同語意的文件分在同一群，若使用者有文件屬於某一群，則系統能將此群中的其他文件推薦給該使用者。在語意歧異解析方面，本論文提出一個新的語意歧異解析方法來決定文件內容中字詞的語意，這個方法是以建立字詞語彙鍵結**(Lexical Chain)**為基礎，並搭配** WordNet **得到詞彙概念**(Concept)**，建立的語彙鍵結依概念關係的強度而有不同的權重，利用這些權重來判斷該字詞可能的語意。實驗中，我們採用** Semantic Concordance Corpus (SemCor) **文件集來評估語意歧異解析方法的好壞。結果顯示，我們所提的方法有不錯的表現，平均來說，正確率可以達到** 65.26%**，優於**[Suarez99]**所提出的** 59.11%**。**

關鍵字：參考文獻管理系統、個人化推薦、語意分析、語意歧異解析、語彙鍵結

致 謝

感謝指導教授柯皓仁老師及楊維邦老師的悉心指導，讓我學習到完成一篇論文或一項研究所需經歷的整個過程與自我挑戰的階段，也讓我了解到作為一個研究生所須具備的實事求是與追根究底的精神。

感謝實驗室的同學們，你們適時溫馨的幫助與實驗室和樂的氣氛是使我完成論文的一大動力，在那種環境下，具有讓人放鬆心情及專心思考的魔力。還有感謝圖書館參考諮詢組和計畫室的成員們，你們偶爾天南地北的閒聊和加油打氣是使我繼續衝刺的良方。

感謝親愛的家人永遠不變的支持與鼓勵，也感謝朋友的關懷。在研究所的這兩年當中，有摸索、有茫然、有討論、有歡笑、有瓶頸、也有解決方法，這段過程是獨一無二且滿懷回憶的。謝謝你們。

June, 2003

目 錄

圖目錄

表目錄

第一章 簡介

第一節 參考文獻管理系統

近年來隨著資訊數位化的技術逐漸成熟以及電腦科技的進步，幾乎各式各樣的資訊都能經由網路取得，網際網路的蓬勃發展使得數位圖書館的資源日益豐盈，而人們也可以方便地獲取知識。這些電腦與網路科技的發展已逐漸影響人類生活的許多方式，並改變了資訊與知識的產生、處理及傳播。舉例而言，早期使用者要利用圖書館資源只能實地到圖書館內尋找，而在「數位圖書館(Digital Library)」[26]這個概念被提出並且建置後，現在的使用者已經不一定要到圖書館內找資料了，他們可以透過網際網路來取得數位圖書館的資料。

一般來說，使用校園數位圖書館的目的不外乎館藏查詢、電子資料庫查詢及電子期刊查詢等，而會用到後兩項的使用者主要都是為了研究目的，我們稱之為學術研究者(Academic User)[22]，他們時常需要接觸並閱讀相關的研究論文與報告。然而，由於個人習慣的差異，有些人閱讀或下載後的文獻常放在零散的地方，導致將來想要整合時還得花費一番功夫。

有鑑於此，一個可依個人需求來調整且便於管理資料的文獻管理系統就顯得有需要了，目前文獻管理系統中較著名者包括線上參考目錄BoW(Bibliography on the Web)系統[24]以及個人線上參考文獻資料庫 RefWorks 系統[27]，總結這些系統的功能，一個好的文獻管理系統必須提供下列幾項服務：

- 不受時間與空間的限制，使用者可以在任何時間、任何地點使用文獻管理系統。
- 使用者可以方便地新增資料並且組織及管理個人目錄。
- 使用者可以藉由搜尋功能方便找到所需的文獻。
- 使用者可以利用唯讀方式將個人之文獻資料分享給其他的研究人員。

表格 1 是這兩個系統的主要功能比較。

文獻管理系統 / 功能	線上參考目錄 BoW	個人線上參考文獻資料庫 RefWorks
支援遠端存取	√	√
新增文獻資料	√	√
組織及管理資料(夾)		√
建立文獻索引(index)	√	
文獻搜尋服務	√	√
資源共享	√	√
自動文獻分群	√	

表格 1：已知參考文獻管理系統之功能比較

在表格 1 的這些功能中，支援遠端存取及新增文獻資料是一個文獻管理系統所需提供的基本功能，而一個貼心的文獻管理系統尚需讓使用者依自己的習慣組織及管理資料或資料夾，因此前三項服務是文獻管理系統的基本功能。而建立一個好的文獻管理系統的關鍵技術在於(1)如何建立文獻索引、(2)提升文獻搜尋結果之服務、(3)與他人共享資源及(4)自動文獻分群。

(1) 建立文獻索引：線上參考目錄 BoW 系統[24]提出的索引建立方法為階層式的概念索引，它從文件中的作者名、出版商、註解等挑選出關鍵字作為索引，在字詞的權重計算上採用資訊擷取(Information Retrieval)方法的 TF*IDF [1]。這種藉由計算 TF*IDF 的權重公式雖然簡單，但相對地，因為它並沒有判斷出文件字詞的語意，因此可能會導致非相關的文件搜尋結果或錯誤的文件分群。解決方法可以加入字詞的語意來建立文件索引，而在判斷字詞語意方面，包括字典導向方法 (Dictionary-Based Method)、監督式方法 (Supervised Method)、非監督式方法(Unsupervised Method)以及混合型方法(Hybrid Method)

都可以用來判斷字詞語意，其中字典導向方法是目前最多人採用且準確度較高的方法，一些相關研究如[3][4][5][6][7][8][19]等都是對照 WordNet 這個詞典(Thesaurus)，利用詞典中所定義的字詞關係來決定字詞間的語意。

(2) 文獻搜尋服務：良好的搜尋必須兼顧快速回傳資料及提升搜尋準確度，因此系統通常會先建立文件的索引，索引建立的方式依系統或使用者的需求而有所不同，若系統只提供字串比對的搜尋服務，則資訊擷取(Information Retrieval)方法[1]建立的索引就已足夠，但利用此方法建立的索引並不能保證搜尋的準確度；若系統希望滿足使用者搜尋的準確度，意即搜尋到語意相關的資料，則可以利用判斷出的字詞語意建立索引。

(3) 資源共享：線上參考目錄 BoW 系統[24]以及個人線上參考文獻資料庫 RefWorks 系統[27]可以藉由搜尋功能找出相關的文獻，並且提供使用者以唯讀方式共享他人的資源。但同樣地，系統提供的共享資源中可能有一些和使用者非相關的文獻，解決方法同樣可以加入字詞的語意來代表文獻。

(4) 自動文獻分群：文獻分群與文獻索引息息相關，計算所有已經建立好索引文獻的相似度，則可以達到自動文獻分群的功能，不過問題在於分在同群中的文獻是否真正相關，解決方法同樣可以加入字詞的語意來代表文獻。

以上所述的線上參考目錄 BoW 系統[24]以及個人線上參考文獻資料庫 RefWorks 系統[27]都沒有判斷出字詞的語意，因此我們開發的文獻管理系統將著眼於語意的分析且建立具語意的字詞索引。

第二節 研究動機

過去的文獻管理系統大多著重於提供一個便於操作的介面，這個介面可以讓使用者依個人需求來調整且管理所蒐集的文件，最多加上檢索的功能讓使用者能

快速搜尋文件及達到資源共享。有鑑於此，本論文研究的動機便是希望能夠提升文獻管理系統的價值，藉由分析使用者蒐集的文件，將文件分群並推薦相關的資料給使用者。因此我們認為一套好的文獻管理系統還必須滿足以下兩個條件：

- 分在一群的文件必須確實相關；
- 推薦的文件必須符合使用者的需求；

為了滿足上述的兩個要求，一套較佳的文獻管理系統必須要能夠理解文件的內容，即文件的真正語意，因此本論文提出一個可以判斷字詞語意的演算法，利用此演算法得到的字詞語意當作文件索引，進而將文件分群，分群後的結果並推薦給相關的使用者。

第三節 研究目的

綜合以上說明，本論文主要的研究在於判斷文件字詞語意，並利用分群方法將這些判斷出字詞語意的文件分群。實作的系統呈現(推薦)給使用者的是一個有組織、有架構的文件分群結果，使用者可以從這些推薦給他的文件中選取自己想要的資訊，並可以依照自己的喜好新增、修改、儲存整群文件等，而使用者所做的這些動作都將被系統記錄下來作為其個人設定檔(User Profile)的一部分，以應用到下次他的文件分群中。

整套個人化文獻管理系統提供的功能有：

- 輸入功能(Import)：讓使用者將欲儲存的文件輸入空白表單中－利用複製和貼上(Copy and Paste)。
- 管理資料夾功能(Organize Folders)：讓使用者自定資料夾，並將文件分門別類歸納入各個資料夾，並可對資料夾更名、新增、刪除、修改。
- 瀏覽功能：由使用者選擇排序欄位瀏覽儲存的文件。

- **搜尋功能：分為簡易搜尋及進階搜尋兩種；搜尋使用者儲存的文件。**
- **文件分群及推薦功能：系統將全部蒐集的文件經斷詞切字處理及字詞語意判斷後，進行文件分群，分群後的文件推薦給擁有該群某些資料的使用者。**
- **回饋**(Feedback)**及共享功能：使用者可以評估推薦結果，並將滿意度回傳給系統，系統可藉此調整分群演算法，進一步達到個人化服務的目的。**

我們並希望透過經由語意分析後分群好的文件，來達到以下目標：

- **自動文件分類：**

已知分群後的每群群中心，則新增的文件分別和每群群中心計算相似度，若最大相似度超過設定的門檻值(Threshold)**，則這個新增文件歸類在最大相似度的群中；若最大相似度沒有超過門檻值，則此新增文件單獨成一群。如此一來即可以做到自動文件分類。**

- **推薦使用者新進文件：**

這個目標可以利用與內容相關的方法(Content-Based Method)**來完成，若利用前述自動文件分類方法將某新增文件歸類到最大相似度的一群中時，這個新增文件同時會被推薦給擁有該群某些資料的使用者。**

綜合以上所述的功能，我們提出的 MyLibrary **文獻管理系統與線上參考目錄** BoW **系統**[24]**以及個人線上參考文獻資料庫** RefWorks **系統**[27]**比較的功能如表格** 2**：**

文獻管理系統 功 能	線上參考目錄 BoW	個人線上參考文獻資料庫 RefWorks	MyLibrary文獻管理系統
支援遠端存取	√	√	√
新增文獻資料	√	√	√
組織及管理資料(夾)		√	√
建立文獻索引(index)	√		√
文獻搜尋服務	√	√	√
資源共享	√	√	√
自動文獻分群	√		√
推薦文獻功能			√

表格 2：本論文提出的 MyLibrary 文獻管理系統與其他系統之比較

我們提出的 MyLibrary 文獻管理系統不僅提供基本功能，亦提供滿足並符合使用者需求的進階功能。

第四節 本論文內容與架構

本論文共分為五章，第二章介紹文獻管理系統之相關研究工作，包括判斷字詞語意以建立文件索引、文件分群和個人化參考文獻服務系統；第三章提出一個新的判斷字詞語意的演算法，利用語彙鍵結為基礎，改良字詞語意權重表示法並加入兩種策略來決定字詞語意，且對這個演算法及以這個演算法為核心技術的文件分群結果進行實驗分析與評估；第四章說明實作的「個人化參考文獻管理系統」，結合 MyLibrary@NCTU 呈現給使用者；第五章則歸納結論與未來研究方向。

第二章 文獻管理系統之相關研究工作

本論文提出的參考文獻管理系統著重於建立文獻語意索引及文獻分群與推薦功能，因此本論文主要的目的在於將使用者儲存的文獻資料加以語意分析並分群，然後將分群結果與使用者個人設定檔(User Profile)對照後推薦給使用者作為學術研究時的參考。

整個文獻管理系統的相關研究工作包括判斷字詞語意以建立語意索引；文件分群；以及個人化參考文獻服務系統，將在本章中逐一說明。

首先第一節介紹判斷字詞語意的問題－語意歧異解析及如何利用語意歧異解析的方法來選取關鍵字詞當作文件索引，第二節說明文件分群的方法及進一步達到個人化分群的技術，第三節描述個人化參考文獻服務系統的相關研究。

第一節 語意歧異解析

選取關鍵字詞來作為文件索引是分析文件的第一個步驟，本篇論文也不例外。關鍵字詞的選取是非常重要的，它關係著這些字詞是否能正確代表某篇文件。傳統純粹經由資訊擷取過程計算字詞頻率(Term Frequency)和該字詞出現文件數的反轉頻率(Inverse Document Frequency)所得到的關鍵字詞並不具有任何語意[1]。為了解加入語意的分群結果是否比傳統不具語意的分群結果為佳，本篇論文將比較兩者的分群結果且著重於具語意的關鍵字詞選取。

判斷字詞語意的問題稱為語意歧異解析(Word Sense Disambiguation，簡寫為WSD)，如何達成語意歧異解析是在處理自然語言時的一個常見問題，也是尚待解決的難題[3]。目前為止解決這個問題的方法主要分為四種類型：

1. 字典導向方法(Dictionary-Based Method)：這種方法又稱為知識導向(Knowledge-Driven)的語意歧異解析方法[8]，採用方式為對照一個具大量

詞彙及語意的字典(Dictionary)，常見的字典如 WordNet[28]，再藉由該字典所組織的相關詞彙語意集合找出某字詞的可能語意，如[3][4][5][6][7][8][19]。

2. 監督式方法(Supervised Method)：限定某類主題的文件集，且關於這類主題中的字詞已經訓練好語意。之後蒐集到的該類主題文件則依據已訓練好的字詞決定其語意，如[9][10][11]。
3. 非監督式方法(Unsupervised Method)：沒有採用任何資訊或知識，純粹由蒐集的文件去判斷字詞語意，如[12][13][14][15]。
4. 混合型方法(Hybrid Method)：任意結合上述三種類型的方法，例如結合某類文件集與字典的方法，如[9][10][16][17][18]。

我們依照年份及類型整理了關於語意歧異解析的相關研究工作，如圖 1：

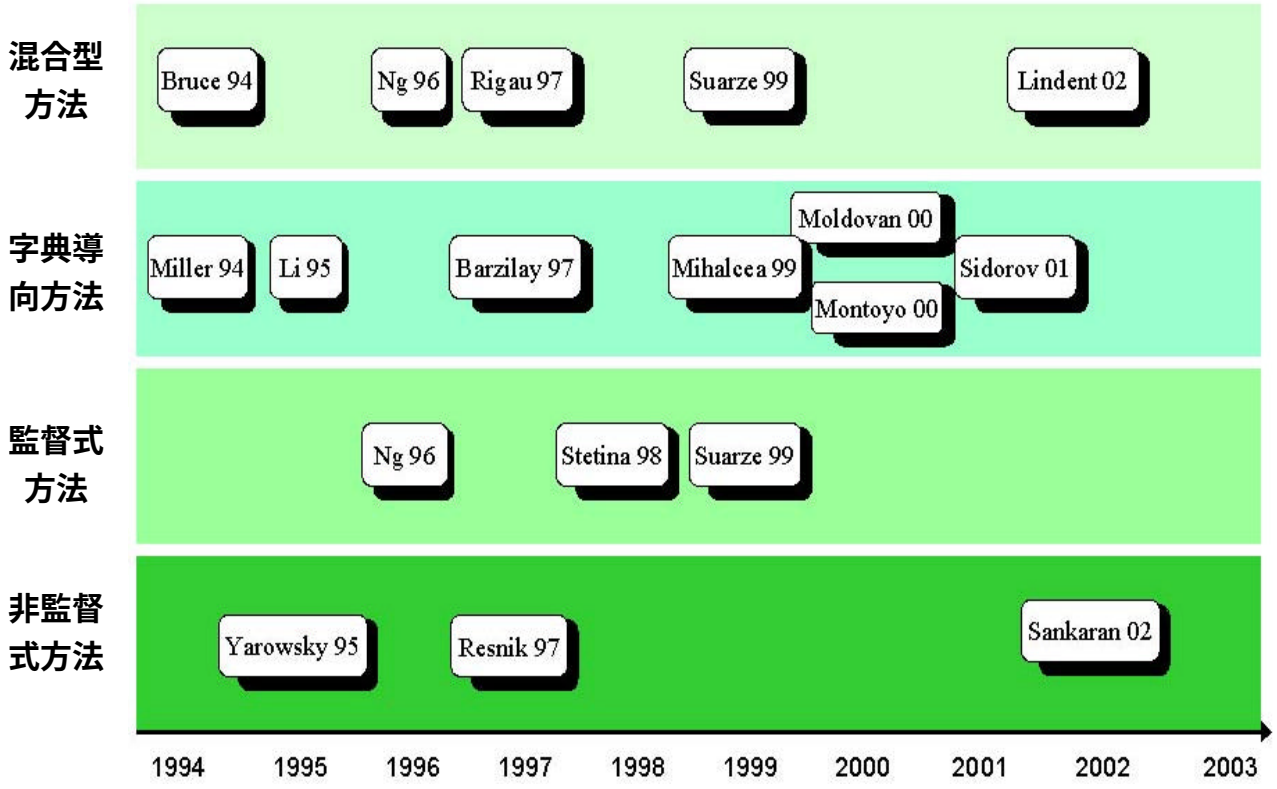

圖 1：語意歧異解析(WSD)的相關研究工作

由圖 1 可知字典導向方法是解決語意歧異解析最常用的方法，且最常被用來對照使用的字典是 WordNet，因此本篇論文也採用對照 WordNet 的方式。在對照 WordNet 的相關研究工作中，其技術包含：(1)語彙鏈結(Lexical Chains) [6]；(2)語意密度(Semantic Density) [3]。下面將依序介紹 WordNet 及利用 WordNet 解決語意歧異解析的技術。

2.1.1 WordNet

WordNet [2]是一個線上詞彙參考資料庫，它的設計靈感是從人類詞彙記憶中的心理語言學(Psycholinguistics)而來。在 WordNet 中，英文名詞、動詞以及形容詞組織成同義字集合(Synonym Sets, Synsets)，每個集合代表一個基本的詞彙概念，同義字集合間會以不同的關係串聯。

以英文名詞來說，在 WordNet 中定義了四種關係：

1. Synonym / Antonym 同義詞 / 反義詞關係
2. Hypernym / Hyponym (relation is a kind of) 上位詞 / 下位詞關係
3. Holonym (relation is part of) 完全關係
4. Meronym (relation parts of) 附屬關係

舉例而言，“car”這個英文名詞在 WordNet 中有五種語意，每種語意代表一種 synset，這五種語意及其相關解釋分別代表(1) auto 汽車；(2) railcar 火車車廂；(3) cable car 纜車；(4) gondola 氣球、氣船；(5) elevator car 升降廂，如下圖所示：

The **noun** "car" has 5 senses in WordNet.

1. **car**, auto, automobile, machine, motorcar -- (4-wheeled motor vehicle; usually propelled by an internal combustion engine; "he needs a car to get to work")
2. **car**, railcar, railway car, railroad car -- (a wheeled vehicle adapted to the rails of railroad; "three cars had jumped the rails")
3. cable car, **car** -- (a conveyance for passengers or freight on a cable railway; "they took a cable car to the top of the mountain")
4. **car**, gondola -- (car suspended from an airship and carrying personnel and cargo and power plant)
5. **car**, elevator car -- (where passengers ride up and down; "the car was on the top floor")

圖 2：car**在** WordNet **中定義的五種語意**

"car"**在** WordNet **中定義的四種相關語意關係如圖 3 所示：**

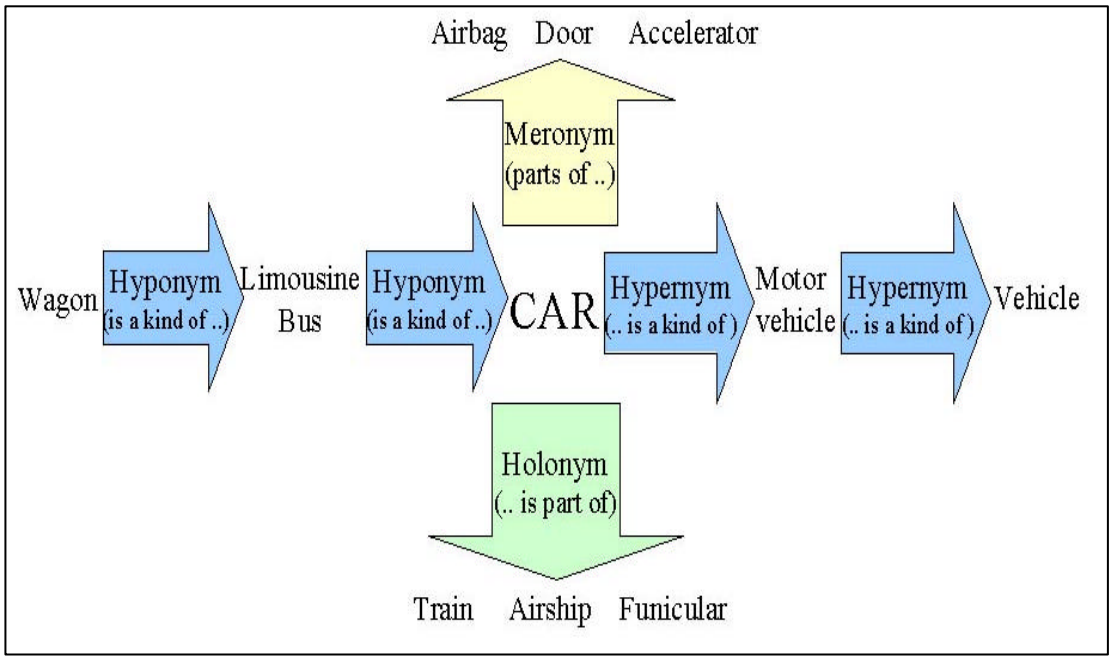

圖 3：名詞 "car" **的語意關係**[10]

由圖 3 中可以看出名詞 "car" **與其他字詞間的關係，例如**

- Limousine is a kind of car, limousine **是** car **的下位詞**(Hyponym)。
- Car is a kind of Motor vehicle, motor vehicle **是** car **的上位詞**(Hypernym)。

- Car is part of train, train 和 car 屬於完全關係(Holonym)，且是部份完全關係(Member Holonym)。
- Airbag, door, accelerator 都是 car 的附屬物件，所以它們和 car 屬於附屬關係(Meronym)。

由於 WordNet 具備豐富的詞彙語意集合及關係，因此它是最常用來判斷字詞語意，解決語意歧異解析的工具。接下來的兩個小節介紹利用 WordNet 解決語意歧異解析的技術。

2.1.2 語彙鏈結(Lexical Chain)

語彙鏈結(Lexical Chain) [6]是文章中具有相同意義的字詞所構成的集合，每個語彙鏈結代表文章中所描述的一個概念(Concept)。一般來說，建構語彙鏈結的程序可分為下列三個步驟：

1. 挑選候選的字詞。
2. 對於每個候選的字詞，針對每個語彙鏈結，衡量該字詞所代表的語意與語彙鏈結中每個字詞的語意關聯度，藉此找出相關聯的語彙鏈結。
3. 如果找到適當的語彙鏈結，便將該字詞加入語彙鏈結中；如果沒有找到的話，便建構新的語彙鏈結。

上述步驟中，用來衡量語意相關的方法，乃是利用 WordNet 來判斷字詞間的關係。主要的關係定義有三種：(1) Extra-strong (定義字詞與其同義字詞間的關係)，(2) Strong (定義兩個字詞在 WordNet 中存在直接關聯的關係)，(3) Medium-strong (定義兩個字詞在 WordNet 中存在間接關聯的關係)。

在建構過程中，給予鍵結一強度值，用來表示字詞語意關聯的程度：若某鍵結為同義詞關係(Synonym)，則給予 10 分；鍵結為完全關係(Holonym)，則給予

7 分；鍵結為上位詞關係(Hypernym)，則給予 4 分。語彙鍵結的強度值是衡量一個字詞語意的主要指標。

以下例說明如何建構語彙鏈結。其中，粗體字型為挑選出的候選字詞，注意挑選出的候選字詞以名詞為主。

> ***Mr.*** *Kenny is the* ***person*** *that invented an anesthetic* ***machine*** *which uses* ***micro-computers*** *to control the rate at which an anesthetic is pumped into blood. Such* ***machines*** *are nothing new. But his* ***device*** *uses two* ***micro-computers*** *to achieve much closer monitoring of the* ***pump*** *feed the anesthetic into patient.*

對於第一個字詞 "Mr."，首先建構語彙鏈結 [lex "Mr.", sense {mister, Mr.}]。接著，考慮第二個字詞 "person"，由 WordNet 中可知 person 具有三種不同的涵義，分別為(1)"human being"；(2)"a person's body"及(3)"grammatical category of pronouns and verb forms"。為了正確地選擇區別字詞的真正涵義，建構過程便需考慮所有可能的鏈結組合，如圖 4 所示，Mr.及 person 會產生三種不同的語彙鏈結組合。

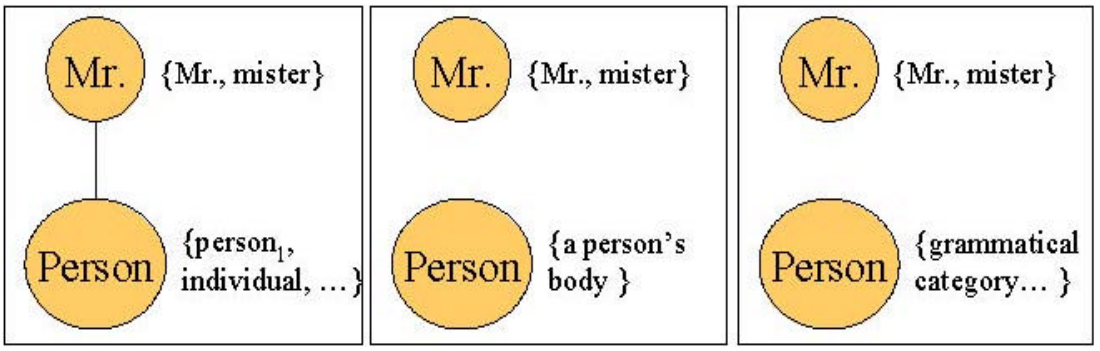

圖 4：考慮 Mr.及 person 所產生的不同語彙鏈結組合

其中，涵義為 "human being"的 person 是 Mr.的上位詞，因此它們之間的鍵結強度為 4 分。再接著考慮第三個字詞 "machine"，由 WordNet 中可知 machine 具有五種不同的涵義，第 1 種涵義為 "an efficient person"，以 $machine_1$ 表示，此

種涵意是 holonym of person。因此 $machine_1$ 和 person 之間的鍵結強度為 7 分。而 machine 的其他四種語意 $machine_2$ $machine_5$ 和 Mr.以及 person 都沒有其他關係。

Mr., person 及 machine 產生的所有可能的語彙鏈結組合如圖 5 所示。

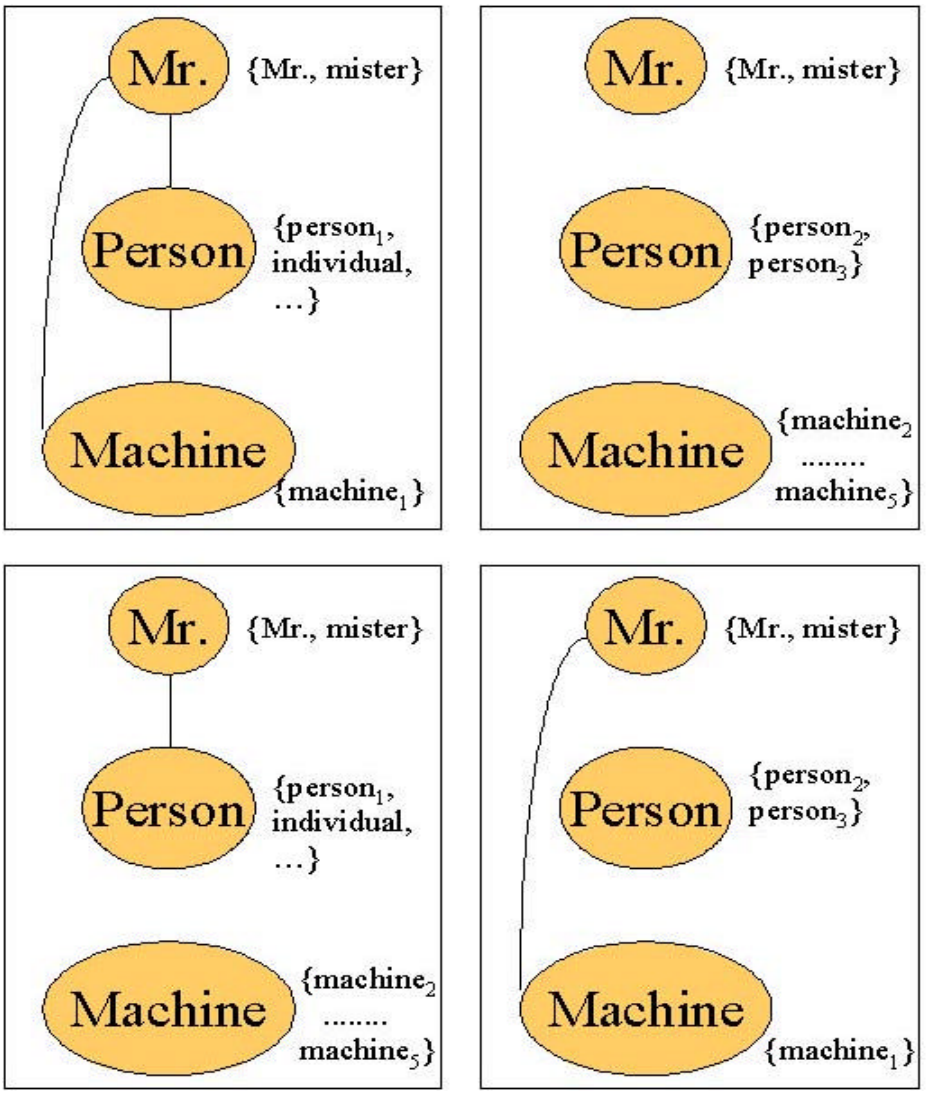

圖 5：Mr., person 及 machine 所產生的不同語彙鏈結組合

上述這樣的作法保留所有可能的鏈結組合，相對地，在建構過程中便產生許多無意義的語彙鏈結；因此必須衡量每個語彙鏈結的重要性，以刪去無用的語彙鏈結。計算語彙鏈結中所有字詞間的相關連結及程度來衡量該語彙鏈結的重要性，便可以將不具代表意義的語彙鏈結刪除，以便有效且快速地建構語彙鏈結。

上例中最後建構出的語彙鏈結有兩種可能，如圖 6 及圖 7 所示，衡量 machine 在兩圖中的鍵結強度：11 分及 30 分，因此選擇圖 7 為語彙鍵結結果。圖 7 中清楚地看到 Mr.與 person 被歸類在相同的語彙鏈結，其所要表達的概念為『人』；Machine，Micro-computer，Device 以及 Pump 則被歸類在另一個語彙鏈結中，其所要表達的概念為『機器』。由此可知，語彙鏈結確實可以反映出某字詞在文件中的語意。

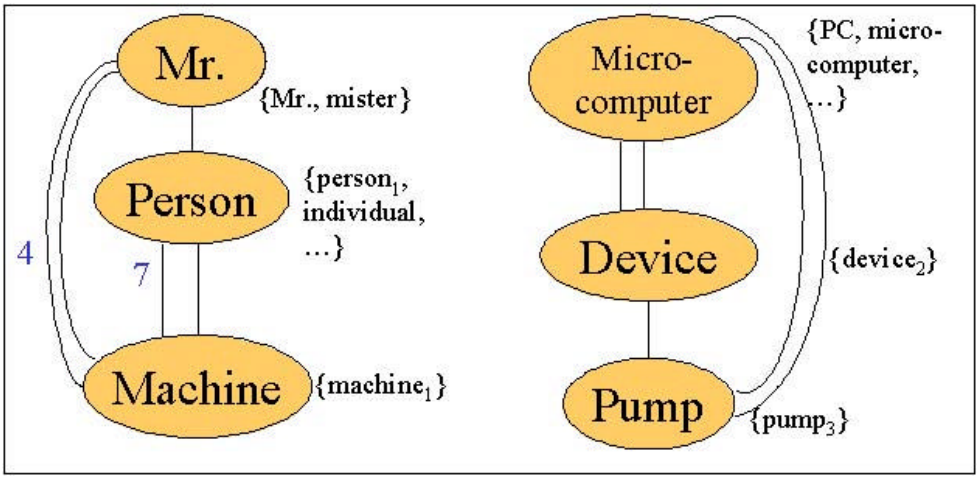

圖 6：語彙鏈結的第一種可能建構結果

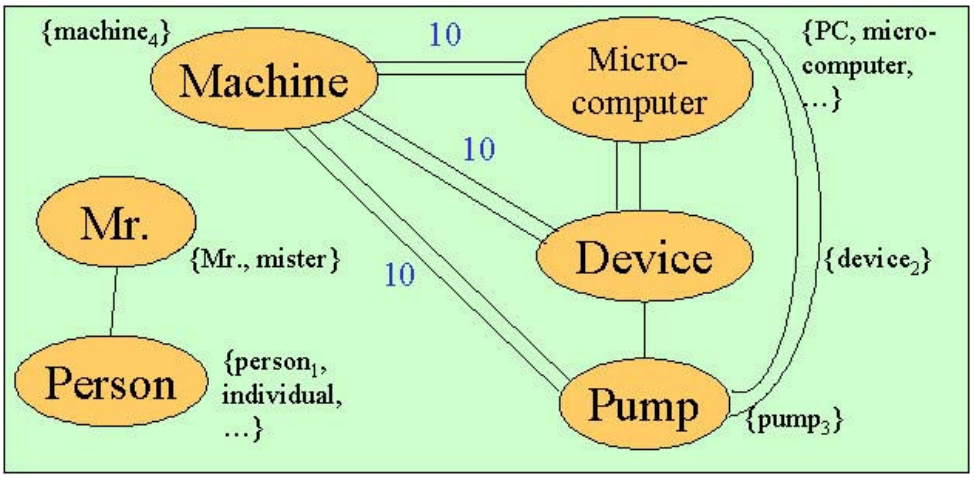

圖 7：語彙鏈結的第二種可能建構結果

美中不足的是，藉助 WordNet 以衡量兩兩語詞間語意的關聯程度，可能因為某個語詞的語意辨認錯誤，而產生錯誤的語彙鏈結；如此，該語彙鏈結所要表達的知識概念便可能偏離原文所要表達的涵義。

2.1.3 語意密度(Semantic Density)

語意密度(Semantic Density) [3]**的定義為：兩個或多個字詞間的語意距離內，相同字詞出現的個數。若兩個字詞的語意關係愈接近，其語意密度愈高。在上述定義中，有關度量語意密度時所採用的字詞詞性必須先解釋清楚：**

語意密度度量的是一對動詞-名詞(Verb-noun Pair)**間的語意距離，語意距離通常以一個句子**(Sentence)**為單位，因為一個句子一般而言都可以簡短地用動作與物體**(Action-object Pair)**來表示。舉例而言：***He has to investigate all the reports*** **可以簡短地用** *investigate-report* **這個動作-物體對來描述。**

在度量語意密度時，首先利用判斷詞性的演算法，如 Part-of-speech Tagging**，標示出字的動詞、名詞等詞性，然後考慮一個句子中的動詞-名詞配對，再利用** WordNet **找出該動詞與名詞的所有語意。若動詞有** k **個語意，名詞有** m **個語意，以** $<v_1, v_2, \ldots v_k>$ **及** $<n_1, n_2, \ldots n_m>$ **表示，對每一個可能的** v_i - n_j pair**，計算語意密度的演算法如以下步驟：**

1. **在** WordNet **中，動詞** synset **的註解會提供該動詞的上下文相關名詞。找出該動詞的相關名詞及這些名詞的** synset **與** hypernym set, **組成包含該動詞** v_i **的階層**(Hierarchy)**。**
2. **利用** WordNet **找出名詞** n_j **的** synset **與** hypernym set, **組成包含該名詞** n_j **的階層。**
3. **比較動詞與名詞的階層，計算這兩個階層相同字詞出現的個數** *Cij***。在此** *Cij* **的算法為：**

$$C_{ij} = \frac{\sum_{k}^{|cd_{ij}|} w_k}{\log(desc_j)}$$

- 對動詞 v_i 的階層而言，w_k 代表與該動詞有關的名詞 sv_k 在階層中的權重(Weight)，在這裡權重用階層的高度(Level)來計算。
- cd_{ij} 是在動詞階層與名詞階層中，相同概念出現的個數。
- $desc_j$ 是名詞 n_j 的階層中的總個數。

計算所有 v_i 和 n_j 所形成的 C_{ij} 之後，最大的 C_{ij} 即代表動詞最可能的語意 i 與名詞最可能的語意 j，因此決定了字詞的語意。

第二節 文件分群方法

本篇論文希望呈現並推薦給使用者的是一個有組織、有架構的文件分群後的結果，分好的每群以概念來描述它，以方便使用者瀏覽與管理。呈現結果類似圖 8 所示。

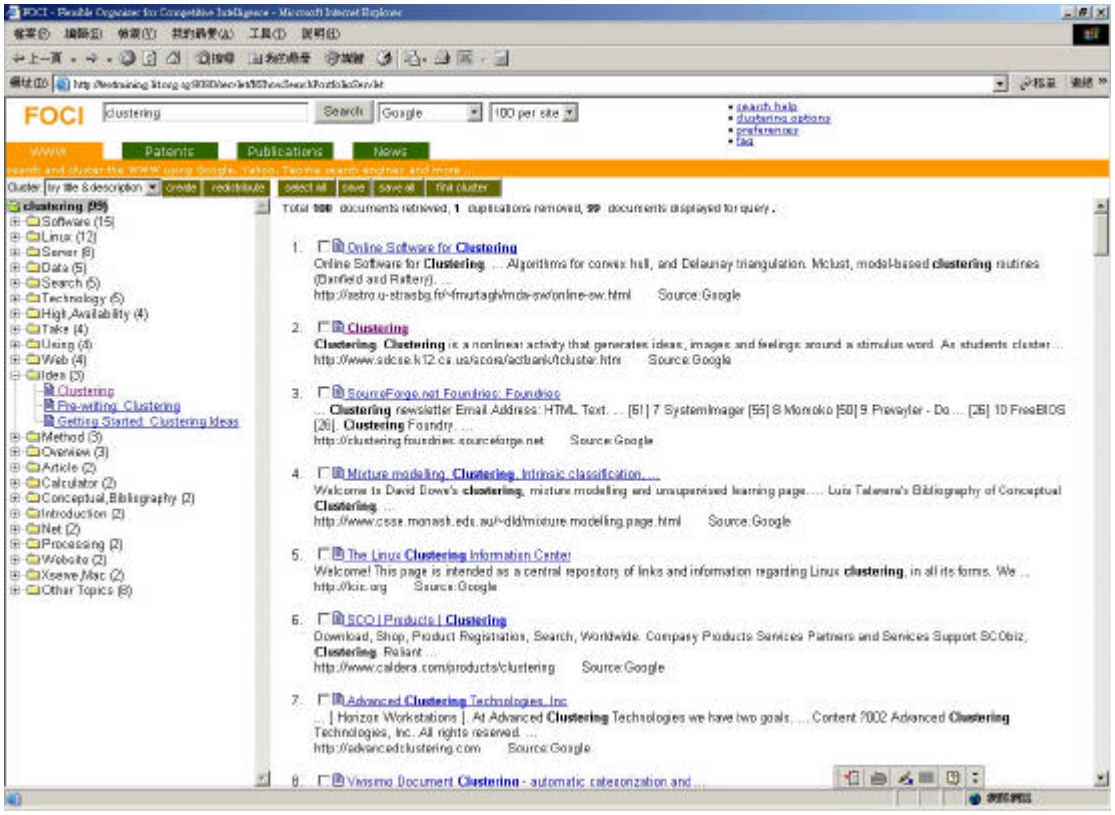

圖 8：方便使用者瀏覽與管理的分群結果

其中，左框架的每個資料夾(目錄)代表一群，每群以一個概念(Concept)表示之，概念後的小括弧中代表的數字為該群包含的文件數，例如第一群的概念為 Software，包含 15 篇文件；第二群的概念為 Linux，包含 12 篇文件，以此類推。右框架則顯示左框架每群中包含的文件詳細資料。

分群方法主要分為兩種類型：(1)非監督式分群法(Unsupervised Clustering)和(2)不完全監督式分群法(Semi-supervised Clustering)。非監督式分群法會在演算過程中自動達到一個最佳的分群數目，並且將資料指定到某群中，然而要達到一個最佳的分群數目並不容易，因此研究如何達到一個最佳的分群數目仍然是一個重要的議題[20][21]。不完全監督式分群法則結合非監督式分群法及一些已知的資訊，這些資訊包含分群的數目及每群中已有的一些資料，因此不完全監督式分群法主要是用來判斷出某個不知道類別的資料該歸屬於哪一類。

在[Tan02]中提到，使用者可以對呈現給他的分類資料加以新增或刪除，而使用者的這些動作會被儲存起來，作為下次分類資料的依據，換句話說，使用者能夠自行調整分群結果來滿足自己的需求，亦即達到個人化分群的目的。圖 9 為使用者可自行調整的分群系統示意圖：

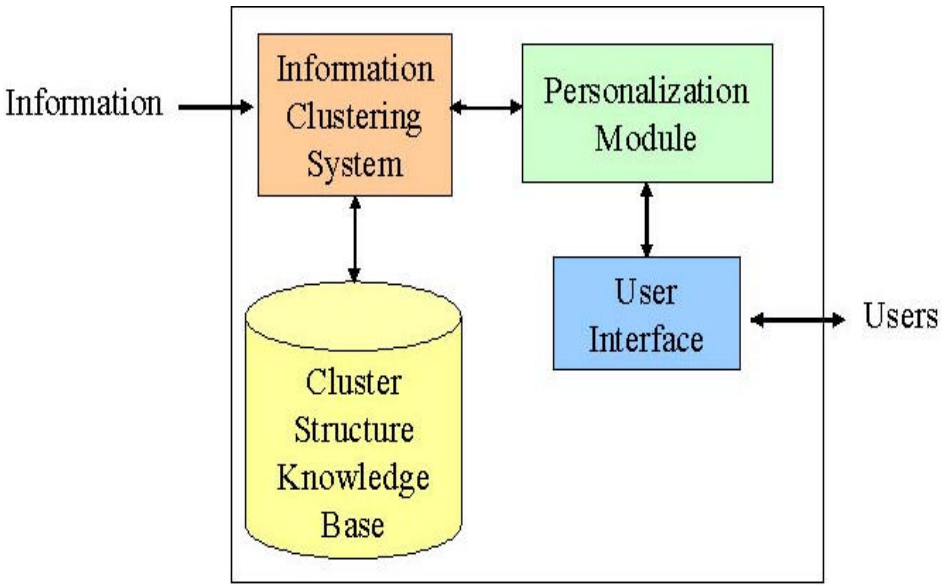

圖 9：使用者可自行調整的分群系統[22]

由圖 9 中可知，當蒐集的資訊，如文件等經過分群系統組織分群後，這些分群好的資訊會先透過個人化模組加以調整，個 人化模組儲存使用者的個人設定檔等，而系統加以調整好的資訊才會藉由使用者介面呈現給使用者以期滿足個人需求。

第三節 個人化參考文獻服務系統

有關個人化參考文獻服務的研究方面，線上參考目錄 BoW 系統[24]以及個人線上參考文獻資料庫 RefWorks 系統[27]提出個人化參考文獻服務的架構，概括來說，所謂的個人化參考文獻服務，應該要具備以下條件：(1) 儲存使用者所蒐集的文獻，以便於組織及管理資料；(2) 支援使用者以唯讀模式讀取他人的資料庫，以達到資源共享的目的；(3) 提供個人化的搜尋服務，快速找到自己蒐集的參考資料，並可以依主題搜尋他人的參考資料。

由 Cambridge Scientific Abstracts (CSA) 所發展的個人線上參考文獻資料庫 RefWorks 系統[27]，架構於全球資訊網(WWW)，主要提供個人參考文獻管理與編製的解決方案，它具備以下特色：

- 無需安裝任何軟體工具：它是一 WWW-based 的服務，適用於任何平台上的主要瀏覽器。
- 不受時間與空間的限制：使用者只需要透過網際網路，就可以在任何時間、任何地點連接到 RefWorks.
- 與多數電子資源系統相容：使用者可以快速且方便地匯入主要電子資源系統及參考文獻編輯工具所產生之參考文獻資料。
- 個人化服務：RefWork 為個人化的參考文獻資料庫管理工具，使用者只需以個人使用者帳號及密碼登入，即可享受個人化的服務。

- **資源分享：可以透過匯出參考文獻記錄或使用唯讀密碼將個人之參考文獻資料分享給其他的研究人員。**
- **多國語言介面：目前提供九種國際間常用的語言介面，使用者可以在最習慣的語言環境中使用** RefWorks **提供的服務。**

RefWorks **的主要提供功能有：**

- **匯入資料；**
- **組織及管理個人資料；**
- **將文獻中的參考文獻以不同的樣式格式化；**
- **匯出資料；**
- **產生文稿的參考文獻；**
- **將引文（篇號）寫入文稿中。**

RefWorks **的操作介面如圖** 10：

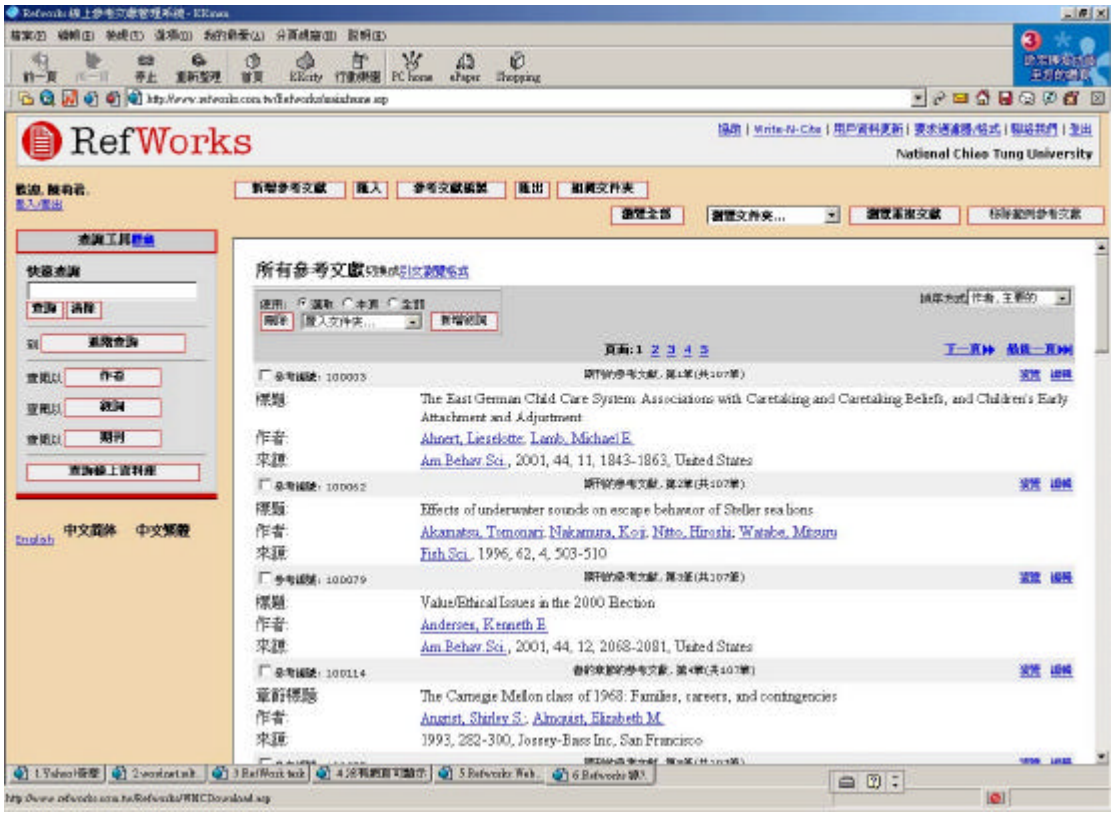

圖 10：RefWorks **登入後提供的操作介面**

在圖 10 的左框架為登入者姓名及各種查詢方式的介面，右框架的上方為 RefWorks 提供的幾項功能，依照各種功能的結果呈現在右框架的下方中，舉例而言，若使用者點選"新增參考文獻"，則如圖 11 所示：

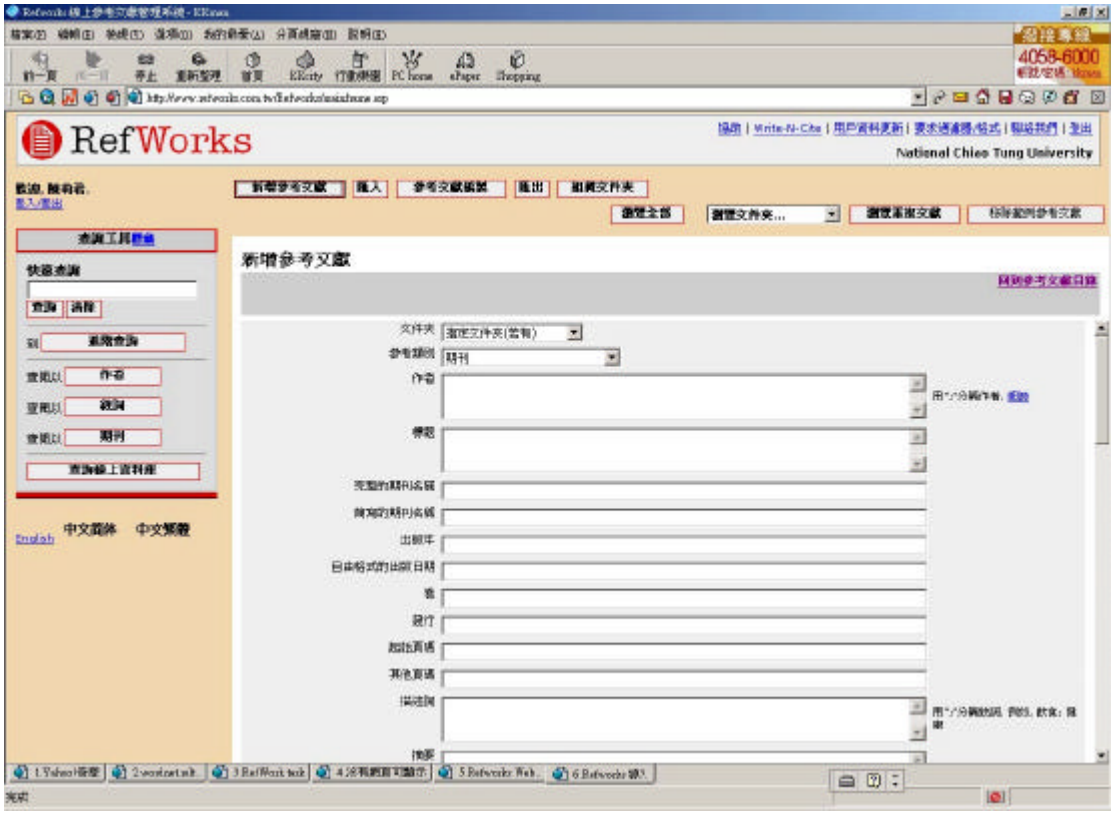

圖 11：RefWorks 的新增參考文獻介面

RefWorks 的新增參考文獻可以選擇要新增到哪一個文件夾，並且輸入參考文獻的一些相關資料。

線上參考目錄 BoW(Bibliography on the Web)系統[24]是依據階層式概念索引連結物件的一個線上書目儲存系統，BoW 建立的主要目的為創造一個方便使用與管理線上書目資料的環境。在階層中每一個節點稱為概念頁面(Concept Pages)，亦即 BoW 所蒐集的文獻資料，愈上層頁面所代表的概念愈廣泛，而包含某些概念頁面節點及其子樹的頁面稱為主題(Topic)。BoW 取概念頁面的作者名、出版商、註解等當作索引，建構出階層式的概念索引，以提高搜尋的正確性。

BoW **提供的主要功能有：**

- **新增資料**
- **依照使用者新增的資料建立概念式的階層索引**
- **階層索引中的文件瀏覽**
- **匯出資料**
- **搜尋功能**

BoW **的操作介面如圖** 12**。下方網頁的左框架為階層索引，範圍愈大的索引包含的頁面愈多，右框架則是階層索引中的文件瀏覽。上方的網頁為新增資料的操作介面。**

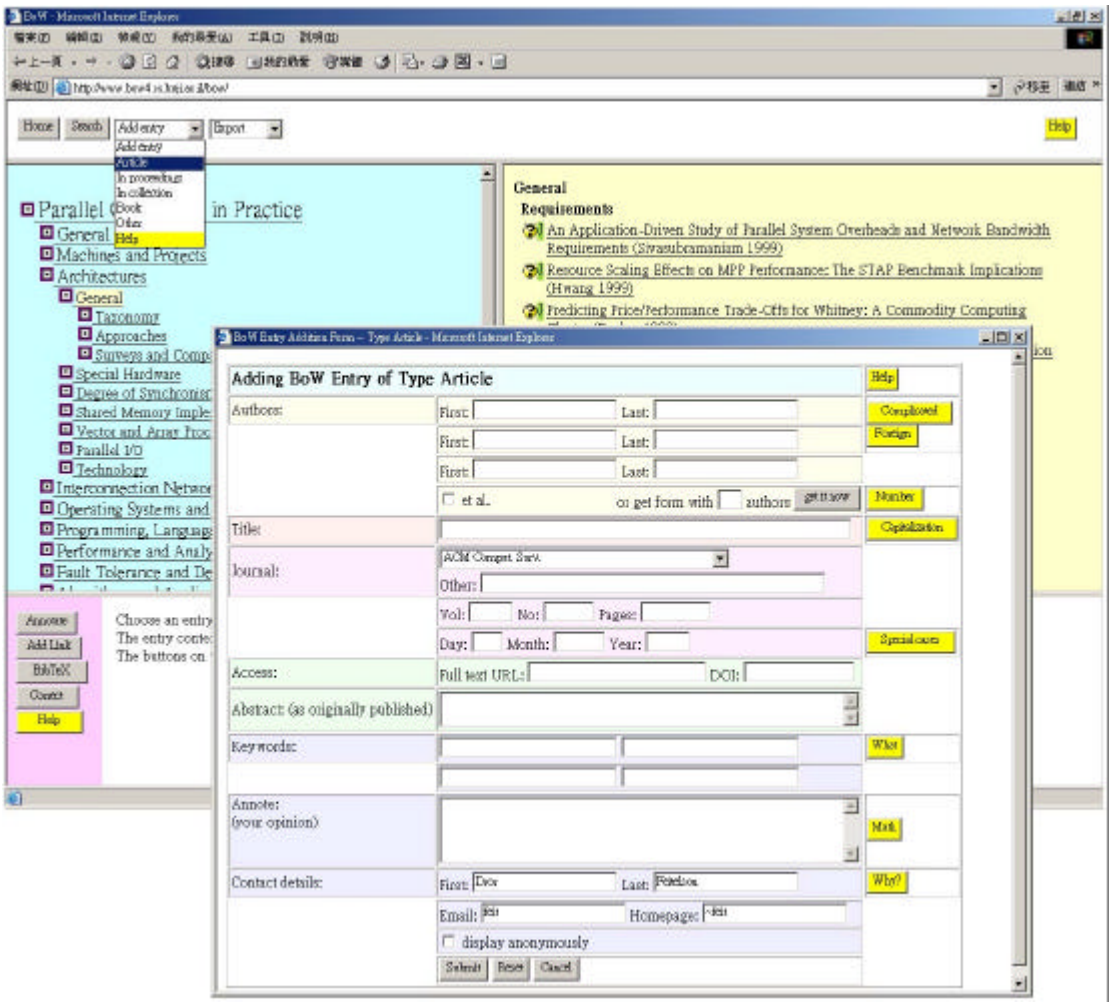

圖 12**：**BoW **的階層索引瀏覽功能**

第三章 改良型語意歧異解析演算法

在本章中，我們針對參考文獻的特性，選擇適合的語意歧異解析演算法。由於在文獻管理系統中，文件儲存主要是以記錄特徵(Feature)的方式，例如文件作者、出版年份、內容摘要等來代表某篇文件。其中我們選取 "摘要" 中的字詞進行語意歧異解析，因為摘要通常不會太長，且在簡短的摘要中，相同的字詞通常具有相同的語意。

針對上述特性，我們假設在簡短而有限的一段摘要中，其內的字詞只會擁有一種語意。因此我們以會判斷出字詞最有可能語意的 "語彙鍵結(Lexical Chains)" 方法[6]為基礎，分析文件中「名詞」的語意，並提出幾點改進：(1) 複合語意與權重：針對每個名詞的涵義，考量最後建構出的幾個語彙鍵結，將每個名詞的語意依照鍵結強度賦予不同的權重。(2) 鍵結擴充：為了防止語彙鍵結過少，加入一些策略讓每個名詞盡量能判斷出語意，以降低無法判斷語意的比率。我們將這種方法稱之為改良型語意歧異解析方法。

在本章中，第一節及第二節分別介紹改良型語意歧異解析方法的複合語意權重表示法及鍵結擴充，第三節說明整個改良型語意歧異解析演算法的流程，第四節計算實驗數據並評估本論文提出的改良型語意歧異解析方法的效能。

第一節 名詞語意歧異解析與複合語意權重表示法

為了方便說明所提出的改良型語意歧異解析方法，在此以相關研究工作中語彙鍵結方法(Lexical Chain)的範例來驗證。

在此範例中，分析的文件內容為：

Mr. *Kenny is the* ***person*** *that invented an anesthetic* ***machine*** *which uses* ***micro-computers*** *to control the rate at which an anesthetic is pumped into blood. Such* ***machines*** *are nothing new. But his* ***device*** *uses two* ***micro-computers*** *to achieve much closer monitoring of the* ***pump*** *feed the anesthetic into patient.*

圖 13：語意歧異解析的範例 1

分析此文件中挑選出來的候選字詞，在此候選的字詞皆為名詞，如黑體字所示，其最後建構出來的語彙鏈結有兩種可能，如圖 14 及圖 15 所示：

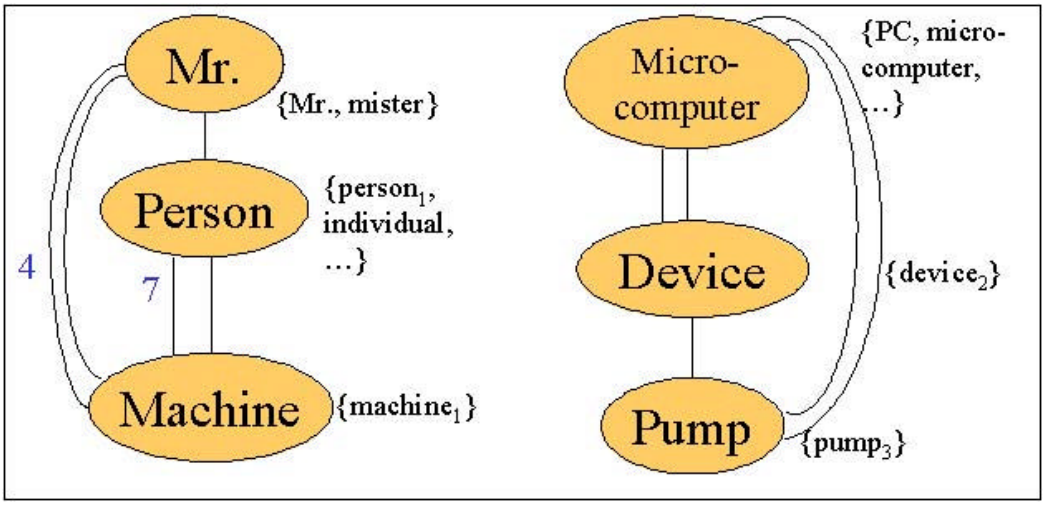

圖 14：語彙鏈結的建構結果 1

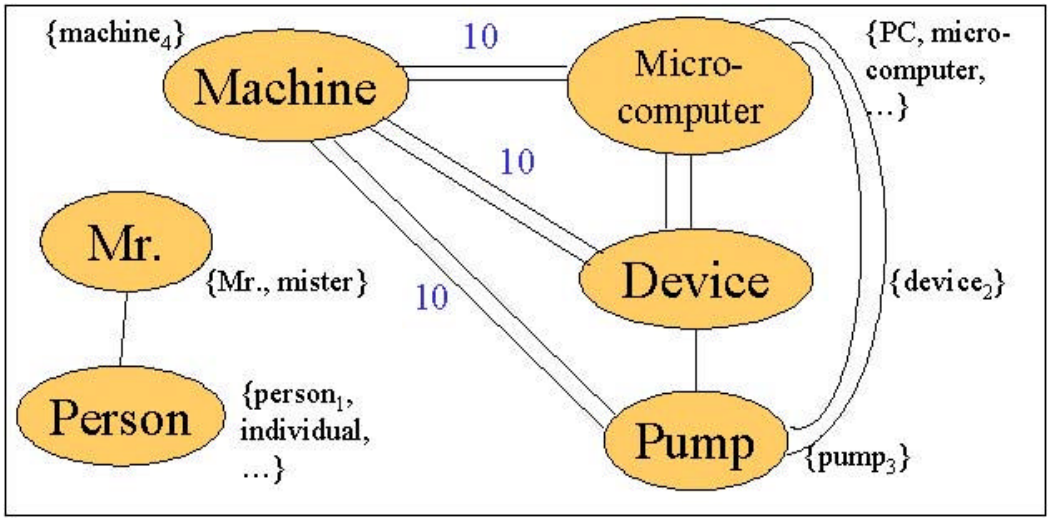

圖 15：語彙鏈結的建構結果 2

machine 這個名詞在圖 14 中所表示的語意為 $machine_1$，其鍵結強度為 11 分；而在圖 15 的語意則為 $machine_4$，其鍵結強度為 30 分，因此最後 machine 會被判斷成 $machine_4$ 這個語意，也就是指定它的概念為「機器」。

但這種只單取一個語意的決定似乎有點專斷，因為還是有可能 machine 的真正語意是 $machine_1$ 而不是 $machine_4$。以下面圖 16 這段文字而言：

> ***Mr.*** *Kenny is not only the boxer, a magnificent fighting* ***machine****, but also the* ***person*** *that invented an anesthetic mechanism which uses* ***micro-computers*** *to control the rate at which an anesthetic is pumped into blood. His* ***device*** *uses two* ***micro-computers*** *to achieve much closer monitoring of the* ***pump*** *feed the anesthetic into patient.*

圖 16：語意歧異解析的範例 2

最後建構出來的語彙鍵結結果仍然是圖 14 與圖 15，但是 machine 的真正語意卻應該是鍵結強度較弱的 $machine_1$ 而非 $machine_4$。為了考量類似的情形並使得語彙鍵結方法更精確，我們將此方法加以改良，使得具有不同語彙鍵結的字詞以「複合語意」來表示，而只有一個語彙鍵結的字詞則直接指定其語意。

以圖 13 的範例而言，我們讓 machine 這個名詞以複合語意來表示，並賦予每個語意強度值，成為 machine{$machine_1$：11；$machine_4$：30}，而其他的字詞 Mr., person, micro-computer, device, pump 則可以直接指定語意為{mister, $person_1$, micro-computer, $device_2$, $pump_3$}。因此整篇文件中，我們以 Mr. {mister}；person {$person_1$}；machine {$machine_1$：11/41；$machine_4$：30/41}；micro-computer {micro-computer}；device {$device_2$}；pump {$pump_3$}這些名詞語意來代表此文件。

假設在挑選這些候選字詞時，已經利用 TF*IDF 計算出權重，即(Mr. person, machine, micro-computer, device, pump)的權重為(0.4, 0.3, 0.9, 0.7, 0.8, 0.7)，則加入語意後形成的向量變成(Mr., $person_1$, $machine_1$, $machine_4$, micro-computer,

$device_2$, $pump_3$)**的權重為**(0.4, 0.3, 0.9*11/41, 0.9*30/41, 0.7, 0.8, 0.7)**，下面表格** 3 **顯示此文件經語意判斷前後的字詞權重向量。**

	Mr.	person	machine		micro-computer	device	pump
未決定語意前計算好的權重(weight)	0.4	0.3	0.9		0.7	0.8	0.7
判斷出的語意	**Mr.**	**$person_1$**	**M_1**	**M_4**	**micro-computer**	**$device_2$**	**$pump_3$**
分析語意後的權重	0.4	0.3	$0.9 * \frac{11}{41}$	$0.9 * \frac{30}{41}$	0.7	0.8	0.7

表格 3**：字詞語意判斷前後的向量表示**

我們將每篇文件挑選出的候選名詞判斷出的語意依照此方式來代表文件向量的表示式，有了這些具語意的文件向量表示法，即可將這些文件分群，並進行後續的其他工作。

第二節 鍵結擴充－語意歧異解析的策略

在語彙鍵結方法中，若某一候選字詞與其他候選字詞皆沒有語意上的關聯，則這個候選字詞無法形成任何鍵結，也就是說，無法判斷出這個候選字詞的語意。

為了讓每個名詞能盡量判斷出語意，我們參考[8]**提出的改善方法，觀察文件中名詞的特性，提出兩點策略如下：**

1. **與定義相關的策略**(Strategy of Definition)

   這裡的定義(Definition)**指的是** WordNet **中關於字詞的註解**(Gloss)**，也可以說是註解的策略**(Strategy of Gloss)**。**

若某字詞的語意在 WordNet 註解中含有其他字詞，我們可以說這兩個字詞間含有某種程度的關聯。舉例而言，若某篇文件中含有(sister, person)這兩個字詞，它們在 WordNet 中並沒有任何關係，也就是說不會出現這兩個字詞互相連接的語彙鍵結。但 sister 在 WordNet 中有四種語意：

The **noun** "sister" has 4 senses in WordNet.

1. **sister**, sis -- (a female ***person*** who has the same parents as another ***person***; "my sister married a musician")
2. Sister -- ((Roman Catholic) a title given to a nun (and used as a form of address); "the Sisters taught her to love God")
3. **sister** -- (a female ***person*** who is a fellow member of a sorority or labor union or other group; "none of her sisters would betray her")
4. baby, **sister** -- ((slang) sometimes used as a term of address for attractive young women)

圖 17：sister 在 WordNet 中定義的四種語意

其中，sister 的第一種語意和第三種語意的註解都包含 “person”，第一種語意($sister_1$)有 2 個 “person”；第三種語意($sister_3$)有 1 個 “person”，因此在“person” 所建構出來的語彙鍵結下，我們可以加入 person 與 sister 的連結，其強度單位指定為 3 分，出現次數愈多則強度值愈大。

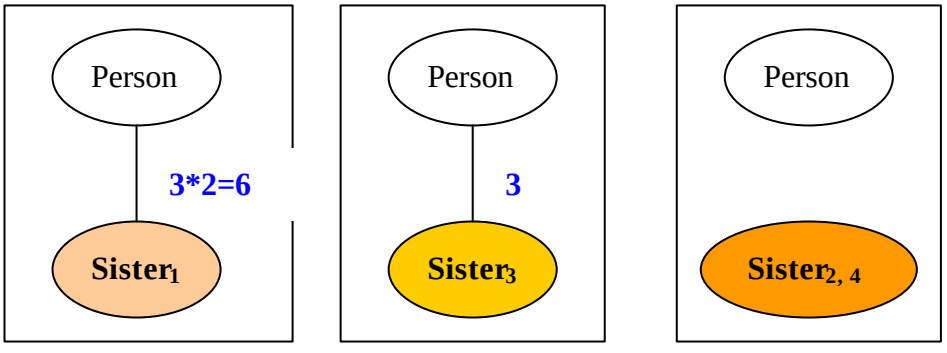

圖 18：與定義相關的策略的範例圖

圖 18 顯示 sister 的四種語意和 person 之間的鍵結關係與強度示意圖。

2. **一般性**(Common)**字詞的策略**

這個策略主要是用來解決語意差異性過小的問題。舉例而言，(year, month) **這種字詞，它們在** WordNet **中定義的語意彼此間的差異性很小，因此實在不需要再判斷出哪一個才是正確語意。在實作上，雖然這些字詞有多個語意，不過可以直接將它們指定為某一種語意。**

year **在** WordNet **中定義的** 4 **種語意如下：**

The **noun** "year" has 4 senses in WordNet.

1. **year**, twelvemonth, yr -- (a period of time containing 365 (or 366) days; "she is 4 years old"; "in the year 1920")
2. **year** -- (a period of time occupying a regular part of a calendar year that is used for some particular activity; "a school year")
3. **year** -- (the period of time that it takes for a planet (as, e.g., Earth or Mars) to make a complete revolution around the sun; "a Martian year takes 687 of our days")
4. class, **year** -- (a body of students who graduate together; "the class of '97"; "she was in my year at Hoehandle High")

圖 19：year **在** WordNet **中定義的四種語意**

month **在** WordNet **中定義的** 2 **種語意如下：**

The **noun** "month" has 2 senses in WordNet.

1. calendar month, **month** -- (one of the twelve divisions of the calendar year; "he paid the bill last month")
2. **month** -- (a time unit of 30 days; "he was given a month to pay the bill")

圖 20：month **在** WordNet **中定義的二種語意**

由圖 19 **和圖** 20 **可以看出** year **的四種語意都是代表"年"，**month **的兩種語意都是代表"月"。**

這個策略由於是藉由人為判斷某字詞的語意是否差異性不大，因此它的執行過程主觀性較強，也比較需要花費人力，我們規定此策略必須等前述第一節複合語意權重表示法及第二節的與定義相關的策略執行完後才可以依照需求繼續執行此策略。

將第一節中提出的複合權重表示法加入上述兩點鍵結擴充策略，我們希望能夠提高語意歧異解析的正確性，並減少無法判斷出字詞語意的比率。

第三節 改良型語意歧異解析演算法

總結上述兩節和範例說明，整個改良型語意歧異解析演算法分為五個步驟及兩個策略，其描述如下：

- Step1：利用 TF*IDF 計算文件中的名詞權重，挑出權重>threshold 的名詞當作候選字詞。Document={W_1, W_2, …W_n}
- Step2：對每一個候選字詞 W_i，查詢 WordNet 找出其所有語意。S_i={S_{i1}, S_{i2}, …S_{im}}
- Step3：

```
For each word {
    For each sense {
        和現存的語彙鍵結比較其涵義：
        若屬於 WordNet 定義的字詞間的關係，則連接現存的語彙
        鍵結並存在一鍵結強度值；
        否則無法連接，建立一個新的語彙鍵結。
    }
}
```

- Step4：

  對每一個語彙鏈結，計算其中所有字詞間的相關連結及程度來衡量該語彙鏈結的重要性，將不具代表意義的語彙鏈結刪除。

- Step5：

  計算保留下來的語彙鍵結中每個字詞所屬的強度值。
  若某個字詞具有兩個以上語彙鍵結，則按照強度值分配權重；
  否則直接指定其語意。

- **與定義相關的策略** (Strategy of Definition)：

  For all non-disambiguation senses {
  　For all words that belong to the context {
  　　搜尋此 sense **的註解中是否包含這些字詞，**
  　　若有，則兩者存在一鍵結且按照出現次數給予鍵結強度。
  　}
  }

- **一般性** (Common) **字詞的策略：搜尋執行完上述五個步驟及與定義相關的策略後仍未決定語意的字詞，查詢** WordNet **這些字詞的語意，以專家的角度判斷某字詞的多個語意是否差異性不大，若語意間差異性不大，則指定該字詞為第一個語意，並將此字詞儲存起來，以後遇到相同情況時則直接指定該字詞的語意。**

透過這個改良型語意歧異解析演算法分析文件後，下一節我們將計算實驗數據並評估我們提出的改良型語意歧異解析方法的效能。

第四節 實驗結果分析與評估

本節闡述上述所提出的改良型語意歧異解析方法的實驗結果及相關討論。3.4.1 小節簡介實驗文件集；3.4.2 小節說明評估的方法，包括針對語意歧異解析的評估方法與針對文件分群的評估方法；3.4.3 小節討論改良型語意歧異解析方法的效益評估；3.4.4 小節討論以改良型語意歧異解析方法來執行文件分群的可行性評估。

3.4.1 實驗資料說明

本實驗利用的文件集為 Semantic Concordance Corpus (簡寫為 SemCor)。SemCor 包含 500 篇文件，每篇文件的字數均超過 2000 字，這 500 篇文件分為 15 大類 (A~R 類)，可以作為評估文件分群時的一個標準文件集。15 大類的主題如圖 21 所示：

A. PRESS: REPORTAGE (44 texts)
B. PRESS: EDITORIAL (27 texts)
C. PRESS: REVIEWS (17 texts)
D. RELIGION (17 texts)
E. SKILL AND HOBBIES (36 texts)
F. POPULAR LORE (48 texts)
G. BELLES-LETTRES (75 texts)
H. MISCELLANEOUS: GOVERNMENT & HOUSE ORGANS (30 texts)
J. LEARNED (80 texts)
K. FICTION: GENERAL (29 texts)
L. FICTION: MYSTERY (24 texts)
M. FICTION: SCIENCE (6 texts)
N. FICTION: ADVENTURE (29 texts)
P. FICTION: ROMANCE (29 texts)
R. HUMOR (9 texts)

圖 21：SemCor 的 15 類文件主題

為了評估語意歧異解析的結果，本實驗從 SemCor 中挑選 186 篇文件來測試 (SemCor 中對文件的名詞指定好語意的只有 186 篇)。SemCor 中儲存文件的字詞詞性與語意，它的字詞語意標示是以 WordNet 1.7 版為依據，和本論文建立語彙

鍵結所對照的 WordNet 版本一致，因此我們拿它來作為字詞語意判斷的文件集。

圖 22 是 SemCor 中 J 大類第 59 篇文件的第一個句子的表示方式。

```
1 <contextfile concordance=brown>
2 <context filename=br-j59 paras=yes>
3 <p pnum=1>
4 <s snum=1>
5 <wf cmd=done pos=RB lemma=rather wnsn=1 lexsn=4:02:02::>Rather</wf>
6 <wf cmd=done pos=RB ot=notag>than</wf>
7 <wf cmd=done pos=VBG ot=notag>being</wf>
8 <wf cmd=done pos=VB lemma=deceive wnsn=1 lexsn=2:41:00::>deceived</wf>
9 <punc>,</punc>
10<wf cmd=ignore pos=DT>the</wf>
11<wf cmd=done pos=NN lemma=eye wnsn=2 lexsn=1:09:00::>eye</wf>
12<wf cmd=done pos=VBZ ot=notag>is</wf>
13<wf cmd=done pos=VB lemma=puzzle wnsn=1 lexsn=2:31:00::>puzzled</wf>
14<punc>;</punc>
15<wf cmd=done pos=RB lemma=instead wnsn=1 lexsn=4:02:00::>instead</wf>
16<wf cmd=ignore pos=IN>of</wf>
17<wf cmd=done pos=VB lemma=see wnsn=1 lexsn=2:39:00::>seeing</wf>
18<wf cmd=done pos=NN lemma=object wnsn=1 lexsn=1:03:00::>objects</wf>
19<wf cmd=ignore pos=IN>in</wf>
20<wf cmd=done pos=NN lemma=space wnsn=1 lexsn=1:03:00::>space</wf>
21<punc>,</punc>
22<wf cmd=ignore pos=PRP>it</wf>
23<wf cmd=done pos=VB lemma=see wnsn=1 lexsn=2:39:00::>sees</wf>
24<wf cmd=done pos=NN lemma=nothing wnsn=1 lexsn=1:23:00::>nothing</wf>
25<wf cmd=done pos=RB lemma=more wnsn=2 lexsn=4:02:01::>more</wf>
26<wf cmd=done pos=JJ ot=notag>than</wf>
27<punc>-</punc>
28<wf cmd=ignore pos=DT>a</wf>
29<wf cmd=done pos=NN lemma=picture wnsn=1 lexsn=1:06:00::>picture</wf>
30<punc>.</punc>
31</s>
32</p>
```

圖 22：SemCor 文件的範例

圖 22 **的具體說明如下：**

- **每行前面的數字是為了解說方便而標示的，與原來的文件無關。**
- **第** 2 **行的** filename=br-j59 **代表** SemCor **中** J **大類第** 59 **篇文件。**
- <p pnum=1> **的** p **代表段落，**pnum=1 **是第** 1 **段。**
- <s snum=1> **的** s **代表句子，**snum=1 **是第** 1 **句。**
- **因圖** 22 **為** SemCor **中** J **大類第** 59 **篇文件的第一個句子。接下來的第** 5 **行到第** 30 **行就是這個句子所包含字詞的相關分析。也就是說整個句子為：**

Rather than being deceived, the eye is puzzled; instead of seeing objects in space, it sees nothing more than-a picture.

- **以第** 5 **行為例，***Rather* **這個字詞的** cmd=done **代表它不是停用字**(Stop Word)**，而若** cmd=ignore**，如第** 10 **行的** *the***，則為停用字；**pos=RB **的** pos **是** part-of-speech **的縮寫，亦即該字的詞性，詞性對照表為：**

NN(noun)**：名詞**		
VB(verb)**：動詞**	VBG**：現在分詞**	VBZ**：助動詞**
RB**：副詞**	WRB**：疑問副詞**	
JJ**：形容詞**	PRP**：代名詞**	
IN**：介系詞，前置詞**	TO**：介系詞，後置詞**	
CD**：連接詞**	DT**：冠詞**	

lemma=rather **的** lemma **為** *Rather* **這個字的標題字，亦即該字經** stemming **後的詞幹；**wnsn=1 **代表該字的語意是對照** WordNet 1.7 **版得到的第** 1 **個** sense**。**

3.4.2 **實驗評估方法**

本小節將分別介紹針對語意歧異解析的評估方法和針對文件分群的評估方法。

📖 針對語意歧異解析的評估方法

在比較改良型語意歧異解析方法與 SemCor 所指定語意的差異時，我們分別計算幾種比率如下：

- 正確率：針對多語意的名詞(Polysemous Nouns)來計算，計算方式為：與 SemCor 指定的語意相同的多語意名詞數目/所有多語意名詞的數目。
- 全正確率：以全部的名詞來計算，計算方式為：與 SemCor 指定的語意相同的名詞數目(包含只有一個語意的名詞)/全部名詞的數目。
- 錯誤率(Incorrect)：與 SemCor 指定的語意不同的數目/全部名詞的數目。
- 無法判斷率(Ambiguous)：無法判斷出語意的數目/全部名詞的數目。

由於本論文只針對名詞的語意，因此在比較時僅比較名詞語意歧異解析的效能。

📖 針對文件分群的評估方法

分群好的文件，我們分別計算 inter-cluster 值與 intra-cluster 值，這兩個評估值是傳統上評估分群方法好壞的依據，其描述如下：

- inter-cluster：計算整個分群體中群與群之間的平均分開程度。若計算的是群與群間的距離，距離愈大，表示群與群愈不相似，因此這個值要愈大愈好；但若計算的是群與群間的相似度，則這個值要愈小才能表示群與群愈不相似。在此我們是用群與群間的相似度來計算，計算公式如下：

$$\text{inter} = \sqrt{\frac{1}{n-1}\sum\left(X_{ij} - \overline{X}\right)^2} = \sqrt{\frac{1}{n-1}\left[\sum\left(X_{ij}\right)^2 - n\overline{X}^2\right]} \qquad i, j \text{ are two cluster}$$

$$X_{ij} = Sim(C_i, C_j) \quad i \neq j \qquad n = \frac{(\#\text{of cluster})(\#\text{of cluster} - 1)}{2} \qquad \overline{X} = \frac{\sum X_{ij}}{n}$$

$$Sim(C_i, C_j) = \frac{\overrightarrow{d_i} \bullet \overrightarrow{d_j}}{\left|\overrightarrow{d_i}\right| \times \left|\overrightarrow{d_j}\right|} = \frac{\sum_{k=1}^{t} w_{k,i} \times w_{k,j}}{\sqrt{\sum_{k=1}^{t}\left(w_{k,i}\right)^2} \times \sqrt{\sum_{k=1}^{t}\left(w_{k,j}\right)^2}}$$

- intra-cluster：**計算整個分群體中每群內的相似度，即群內個體的緊密程度。因為是用相似度來計算，因此這個值要愈大愈好。**

$$\text{intra} = \frac{\sum \text{average similarity of each cluster}}{\#\text{ of cluster}}$$

3.4.3 改良型語意歧異解析方法之評估

下面表格 4**－表格** 6 **的數據是對** SemCor **中部分文件**(br-j03 br-j09)**所實驗的結果，其中表格** 4 **是參考**[10]**中抄錄下來的數據，**[10]**的語意歧異解析是一種監督式方法**(Supervised Method)**，此方法先利用** br-j01 **與** br-j02 **選取出來的名詞當作訓練資料**(Training Data)**，而** br-j03 br-j09 **則依據已訓練好的名詞決定其語意，它是用來對照語彙鍵結結果及本論文所提出的改良型語意歧異解析的結果。**

表格 5 **是利用語彙鍵結**(Lexical Chain)**得到的數據；而表格** 6 **則是利用未加入鍵結擴充策略的改良型語意歧異解析方法所得到的數據。由於本論文提出的改良型語意歧異解析方法將一篇文件中的名詞以複合語意來表示，但為了計算正確率和全正確率，我們將同一個名詞的不同語意取最大權重的那個語意來計算。**

文件名稱	名詞個數	多語意名詞 (poly.)	判斷正確的語意 (對 poly.)	正確率 % (針對 poly.)	全部正確的語意	全正確率 %
br-j03	356	280	195	**69.64%**	271	**76.12%**
br-j04	340	266	130	**48.87%**	204	**60.00%**
br-j05	492	407	131	**32.19%**	216	**43.90%**
br-j06	455	339	140	**41.30%**	256	**56.26%**
br-j07	529	413	216	**52.30%**	332	**62.76%**
br-j08	427	362	156	**43.09%**	221	**51.76%**
br-j09	435	306	145	**47.39%**	274	**62.99%**
				47.83%		**59.11%**

表格 4：參考[10] (Suarez93)**得到的數據**

文件名稱	名詞個數	多語意名詞 (poly.)	判斷正確的語意 (對 poly.)	正確率 % (針對 poly.)	全部正確的語意	全正確率 %
br-j03	356	280	158	**56.43%**	234	**65.73%**
br-j04	340	266	139	**52.26%**	213	**62.65%**
br-j05	492	407	240	**58.97%**	325	**66.06%**
br-j06	455	339	214	**63.13%**	330	**72.53%**
br-j07	529	413	224	**54.24%**	340	**64.27%**
br-j08	427	362	228	**62.98%**	293	**68.62%**
br-j09	435	306	162	**52.94%**	291	**66.90%**
				57.28%		**66.68%**

表格 5：採用語彙鍵結(Lexical Chain)**得到的數據**

文件名稱	名詞個數	多語意名詞 (poly.)	判斷正確的語意 (對 poly.)	正確率 % (針對 poly.)	全部正確的語意	全正確率 %
br-j03	356	280	174	**62.14%**	250	**70.22%**
br-j04	340	266	152	**57.14%**	226	**66.47%**
br-j05	492	407	254	**62.41%**	339	**68.90%**
br-j06	455	339	228	**67.26%**	344	**75.60%**
br-j07	529	413	242	**58.60%**	358	**67.67%**
br-j08	427	362	242	**66.85%**	307	**71.90%**
br-j09	435	306	170	**55.56%**	299	**68.74%**
				61.42%		**69.93%**

表格 6：未加入策略的改良型語意歧異解析方法得到的數據

表格的第一行為實驗的文件名稱，因為在[10]**中挑選** SemCor **的** J **大類** 03 **篇到** 09 **篇來測試，為了與**[10]**比較結果，我們也挑選這七篇文件來實驗。第二行代表每篇文件包含的名詞個數；第三行代表每篇文件中多語意名詞**(Polysemous Nouns)**的個數；第四行是針對多語意的名詞中，各種演算法判斷出正確語意的個數；第五行的正確率是計算多語意名詞的正確率，計算方式為第四行除以第三行；全部正確的語意個數包含只有單一語意的名詞及判斷正確的多語意的名詞；全正確率計算全部正確的語意除以名詞個數。**

由表格 4 **中可知，** [10]**提出的語意歧異解析方法的結果並不穩定，有時好有時差，這是因為它採用的是監督式方法，用** br-j01 **和** br-j02 **當訓練資料，而** br-j03 br-j09 **則依據已訓練好的名詞決定其語意，因此若有一篇文件與訓練資料不太相關，則其結果將較差。**

而由上面三個表格中的數據，我們可以看出和語彙鍵結相關的方法得到的數據較平均。這是因為它考慮的是整篇文件，用整篇文件中字與字在 WordNet **中的關聯來決定語意，不需要訓練資料的介入。這種特性適宜用在文獻管理系統。**

由表格 6 也可看出本論文提出的未加入鍵結擴充策略的改良型語意歧異解析方法比語彙鍵結方法的正確度多了 4.14%。

以上三個表格只呈現關於正確語意的部分，我們加入錯誤率和無法判斷率的結果，其數據如表格 7 所示：

	正確率	全正確率	錯誤率	無法判斷率
監督式方法[10]	47.83%	59.11%		
語彙鍵結	57.28%	66.68%	20.84%	12.48%
改良型(未加策略)	61.42%	69.93%	17.59%	12.48%

表格 7：三種語意歧異解析方法的比較

未加入鍵結擴充策略的改良型語意歧異解析方法比語彙鍵結方法的錯誤率少，但二者的無法判斷率是相同的，為了降低無法判斷率，本論文加入鍵結擴充策略並實驗評估加入策略的結果。

接下來呈現加入與定義相關的策略(策略 1)以及加入兩個策略後的數據。

文件名稱	名詞個數	多語意名詞 (poly.)	判斷正確的語意 (對 poly.)	正確率 % (針對 poly.)	全部正確的語意	全正確率 %
br-j03	356	280	177	**63.21%**	253	**71.07%**
br-j04	340	266	156	**58.66%**	230	**67.65%**
br-j05	492	407	259	**63.64%**	344	**69.92%**
br-j06	455	339	231	**68.14%**	347	**76.26%**
br-j07	529	413	248	**60.05%**	364	**68.81%**
br-j08	427	362	244	**67.40%**	309	**72.37%**
br-j09	435	306	175	**57.19%**	304	**69.89%**
				62.61%		**70.85%**

表格 8：加入策略 1 的改良型語意歧異解析方法得到的數據

由表格 8 可知加入與定義相關的策略後的改良型方法比語彙鍵結的方法正確度多了 5.32%；比未加入任何策略的改良型方法正確度多了 1.19%。

文件名稱	名詞個數	多語意名詞 (poly.)	判斷正確的語意 (對 poly.)	正確率 % (針對 poly.)	全部正確的語意	全正確率 %
br-j03	356	280	191	**68.21%**	267	**75.00%**
br-j04	340	266	171	**64.29%**	245	**72.06%**
br-j05	492	407	278	**68.30%**	363	**73.78%**
br-j06	455	339	247	**72.86%**	363	**79.78%**
br-j07	529	413	271	**65.62%**	387	**73.16%**
br-j08	427	362	262	**72.38%**	327	**76.58%**
br-j09	435	306	197	**64.38%**	326	**74.94%**
				68.01%		**75.04%**

表格 9：加入策略 1 和策略 2 的改良型語意歧異解析方法得到的數據

由表格 9 可知加入策略 1 和策略 2 的改良型語意歧異解析方法比語彙鍵結的方法正確度多了 10.73%；比未加入任何策略的改良型方法正確度多了 6.59%；比加入策略 1 的改良型方法正確度多了 5.4%。

表格 10 呈現上述方法的正確率、全正確率、錯誤率和無法判斷率的完整資訊。

	正確率	全正確率	錯誤率	無法判斷率
監督式方法[10]	47.83%	59.11%		
語彙鍵結	57.28%	66.68%	20.84%	12.48%
改良型(未加策略)	61.42%	69.93%	17.59%	12.48%
改良型+策略 1(S1)	62.61%	70.85%	18.82%	10.33%
改良型+S1+S2	68.01%	75.04%	20.86%	4.1%

表格 10：語意歧異解析方法的完整比較資訊

加入鍵結擴充策略的訴求是要降低無法判斷字詞語意的比率，但因為加入策略 1 後，每篇文件可以新增判斷出 10±3 個名詞，不過因為判斷正確的比率小於一半，所以非正確率提高。同樣地，加入策略 2 後，每篇文件可以新增判斷出 30±5 個名詞，但也會導致非正確率提高。

在評估本論文提出的改良型語意歧異解析方法中，我們亦針對每個名詞的語意數計算正確率，表格 10 是從 SemCor 中的 15 大類挑選出的 186 篇文件中，每篇文件的名詞數和語意歧異解析的正確比率。

表格 10 中的第二行表示 15 大類(A　R)中選取的文件數，第三行到第八行分別針對多語意的名詞進行語意歧異解析計算，例如在 A 大類中，第三行顯示有兩個語意的名詞的個數為 492 個，利用改良型語意歧異解析方法判斷正確的有 415 個，所以大約有 0.8335 的正確率。

語意愈多的名詞，理論上可以正確判斷出語意的比率愈少。表格 10 的結果亦符合此一現象。

	docum_#	2 sense	3 sense	4 sense	5 sense	6~10	>=11	poly_num	total_num
A	7	492	343	241	196	630	242	2144	2901
正確個數		415	255	165	147	265	22	1269	2026
比率		0.843495935	0.743440233	0.684647303	0.75	0.420634921	0.090909091	59.19%	69.84%
B	2	124	109	99	70	177	44	623	843
正確個數		91	64	75	21	74	14	339	559
比率		0.733870968	0.587155963	0.757575758	0.3	0.418079096	0.318181818	54.42%	66.31%
C	3	208	147	124	114	264	80	937	1218
正確個數		182	118	63	54	87	9	513	794
比率		0.875	0.802721088	0.508064516	0.473684211	0.329545455	0.1125	54.75%	65.19%
D	4	253	191	229	154	380	154	1361	1655
正確個數		209	123	120	63	137	44	696	990
比率		0.826086957	0.643979058	0.524017467	0.409090909	0.360526316	0.285714286	51.14%	59.82%
E	14	1092	981	786	561	1671	405	5496	7010
正確個數		797	641	518	346	666	141	3109	4623
比率		0.72985348	0.653414883	0.659033079	0.616755793	0.398563734	0.348148148	56.57%	65.95%
F	19	1339	1060	1014	744	1772	579	6508	8700
正確個數		1203	827	682	397	723	244	4076	6268
比率		0.898431665	0.780188679	0.672583826	0.533602151	0.408013544	0.421416235	62.63%	72.05%
G	18	1205	944	1049	750	1609	576	6133	7650
正確個數		905	616	701	264	588	123	3197	4714
比率		0.751037344	0.652542373	0.668255481	0.352	0.365444375	0.213541667	52.13%	61.62%
H	12	922	666	689	544	1406	384	4611	5996
正確個數		764	409	355	244	418	145	2335	3720
比率		0.828633406	0.614114114	0.515239478	0.448529412	0.297297297	0.377604167	50.64%	62.04%
J	43	3249	2370	1977	1867	5113	1453	16029	20801
正確個數		2996	2009	1427	1269	1843	610	10154	14926
比率		0.922129886	0.847679325	0.721800708	0.679700054	0.360453745	0.41982106	63.35%	71.76%
K	29	1544	1102	1279	916	2207	1071	8119	10480
正確個數		1162	670	857	487	657	418	4251	6612
比率		0.752590674	0.607985481	0.67005473	0.531659389	0.297689171	0.390289449	52.36%	63.09%
L	11	415	367	449	261	774	357	2623	3392
正確個數		357	242	277	136	267	116	1395	2164
比率		0.860240964	0.659400545	0.616926503	0.521072797	0.34496124	0.324929972	53.18%	63.80%
M	2	113	71	86	43	123	47	483	627
正確個數		87	44	51	14	48	15	259	403
比率		0.769911504	0.61971831	0.593023256	0.325581395	0.390243902	0.319148936	53.62%	64.27%
N	10	428	399	299	369	839	430	2764	3456
正確個數		342	236	238	267	315	126	1524	2216
比率		0.799065421	0.591478697	0.795986622	0.723577236	0.375446961	0.293023256	55.14%	64.12%
P	6	281	203	222	182	394	218	1500	1962
正確個數		203	123	125	61	181	93	786	1248
比率		0.722419929	0.60591133	0.563063063	0.335164835	0.459390863	0.426605505	52.40%	63.61%
R	6	339	246	243	153	477	120	1578	2123
正確個數		249	182	165	72	165	11	844	1389
比率		0.734513274	0.739837398	0.679012346	0.470588235	0.34591195	0.091666667	53.49%	65.43%
Total	186								

表格 11：利用改良型語意歧異解析方法判斷多語意名詞的正確比率

每大類主題因為文件內容不同而有相異的判斷正確率，不過大致上來說正確率(針對多語意的名詞)可以達到 55%，其中 J 大類的正確率最高。全正確率(針對全部的名詞)則可以達到 65%，平均為 65.26%。

3.4.4 採用改良型語意歧異解析方法來分群文件之評估

從 SemCor 中挑選 186 篇文件，每類(群)挑選的文件數如下所示，第一行表示類別，第二行為每類別挑選出的文件數。

A	B	C	D	E	F	G	H	J	K	L	M	N	P	R
7	2	3	4	14	19	18	12	43	29	11	2	10	6	6

分群後的文件，主要比較下列四種分群方法，說明如下：

1. 基準分群(Base-cluster)：針對這 186 篇文件，我們分別計算 inter-cluster 和 intra-cluster 的值，當作基準值。
2. 無語意分群：利用資訊擷取方法得到的關鍵字做文件分群，計算 inter-cluster 和 intra-cluster 的值與 1.比較。
3. 語彙鍵結分群：利用語彙鍵結方法判斷出語意的關鍵字做文件分群，計算 inter-cluster 和 intra-cluster 的值與 1.比較。
4. 改良型語意歧異解析分群：利用改良型語意歧異解析方法得到的關鍵字做文件分群，計算 inter-cluster 和 intra-cluster 的值與 1.比較。

不論要計算 inter-cluster 值或 intra-cluster 值，在每一種分群方法中代表一篇文件的向量維度(Dimension)必須要一致，這樣計算出來的評估值才具有意義。因此我們將每篇文件的關鍵字向量(Keyword Vector)加以修改，修改過程如下：

- 無語意分群(關鍵字沒有詞性分別，不過為了方便計算，還是只挑名詞當作關鍵字)

- **若有一篇文件的關鍵字向量表示法為：**D (n1, n2, n3, n4)
- **為了改變向量維度，將其中沒有語意的名詞指定為最常使用的語意，而權重也分配到這個最常使用的語意上。**
- **若** n1 **有兩種語意**(Sense)**，且最常使用的語意為** n11**；**n2 **有四種語意，且最常使用的語意為** n23**；**n3 **有一種語意；**n4 **有三種語意，且最常使用的語意為** n42**。則** D **修改為**(n11, n12, n21, n22, n23, n24, n31, n41, n42, n43)**，其中這個向量權重**≠0 **的是** n11, n23, n31, n42**。**

- **語彙鍵結分群**(**關鍵字具有語意，且只有名詞**)

  - **某篇文件經由語彙鍵結方法得到的關鍵字向量為：**D (n1, n21, n31, n42)**，其中** n_i **表示第** i **個名詞，**n_{ij} **表示第** i **個名詞的第** j **個語意。**
  - **只有判斷出的關鍵字語意才有權重，如果這個關鍵字無法判斷語意，例如** n1**，則** n11, n12 **的權重相同**(**指定為原來的一半**)**。**
  - D **修改為**(n11, n12, n21, n22, n23, n24, n31, n41, n42, n43)**，其中** n11, n12 **的權重為** n1 **的一半，**n22, n23, n24, n41, n43 **的權重**=0**。**

- **改良型語意歧異解析分群**(**關鍵字以複合語意表示，且只有名詞**)

  - **某篇文件經由改良型語意歧異解析方法得到的關鍵字向量為：**D (n1, n21, n22, n23, n31, n41, n42)
  - **每個關鍵字語意的權重值按照建構出來的語彙鍵結所佔的鍵結關係程度的比率來分配，如果這個關鍵字無法判斷語意，例如** n1**，則** n11, n12 **的權重相同**(**指定為原來的一半**)**。**
  - D **修改為**(n11, n12, n21, n22, n23, n24, n31, n41, n42, n43) **，其中** n11, n12 **的權重為** n1 **的一半，**n24, n43 **的權重**=0**。**

四種分群方法計算出來的評估數據如表格 12 所示：

Method (Keyword_threshold=0.05)	Intra-cluster	Inter-cluster
基準分群(SemCor)	**0.091**	**0.0157**
無語意分群	**0.126**	**0.0128**
語彙鍵結分群	**0.183**	**0.0104**
改良型語意歧異解析分群	**0.196**	**0.0101**

表格 12：各種文件分群方法之比較

Intra-cluster **計算整個分群體中每群內的相似度，即群內個體的緊密程度，因此這個值要愈大愈好。結果顯示利用改良型語意歧異解析方法計算出來的** intra-cluster **值是最大的。**

而 inter-cluster **代表整個分群體中群與群之間的平均分開程度。我們採用的計算方式為群與群間的相似度，則這個值要愈小才能表示群與群愈不相似，亦即群與群之間愈分開。結果顯示利用改良型語意歧異解析方法計算出來的** iner-cluster **值是最小的。**

因此在分群應用上，利用改良型語意歧異解析方法將文件中的字詞分析其語意，並以之代表一篇文件的關鍵字向量，這種方法可得到較佳的分群結果。

第四章 個人化參考文獻管理系統之實作

本章描述我們實作的個人化參考文獻管理系統，第一節詳細介紹系統流程與採用的機制；第二節呈現應用於交大浩然圖書館個人化環境 MyLibrary@NCTU 的參考文獻管理系統。

個人化參考文獻管理系統的流程分為兩個部分：第一個部分為文件分析，第二個部分為文件分群及推薦。文件分析包含系統對使用者蒐集的文獻資料的語彙處理－詞幹轉換、計算文件字詞權重以得到索引關鍵字、以及利用第三章提出的改良型語意歧異解析方法為基礎的關鍵字語意判斷。文件分群及推薦則包含計算文件的語意相似度來分群以及推薦使用者相關的文獻資料。

第一節 系統流程與機制

4.1.1 文件分析的流程

文件分析即文件字詞的語意分析，其流程如圖 23 所示：

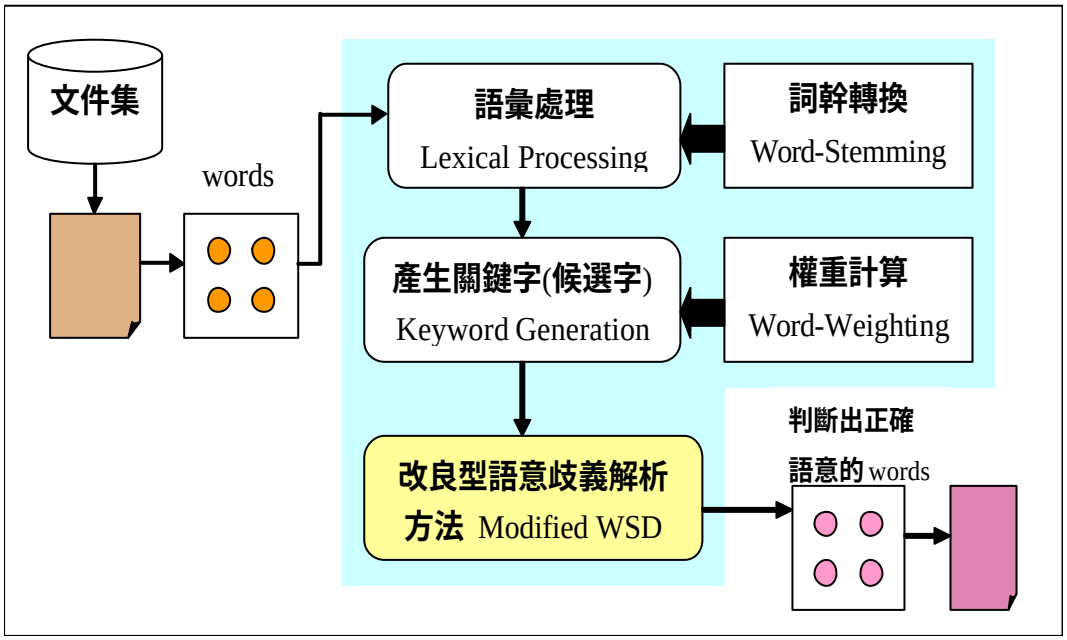

圖 23：文件字詞語意分析的流程

在使用者蒐集的文獻中，首先依據整篇文件的空格(空白鍵)與標點符號切出字詞(Word)，這些字詞若是屬於事先就指定好的停用字(Stop Word)，則將之捨棄，反之則留下來以備後續轉換詞幹(Stemming)與計算權重(Weighting)的處理，然後挑出權重大於某個門檻值(Threshold)的字詞當作關鍵字。

挑選出來的關鍵字利用改良型語意歧異解析的方法對照 WordNet 來判斷語意，當系統對每篇文件的關鍵字判斷出語意後，再進行個人化參考文獻管理系統的第二部分流程－文件分群與推薦(在 4.1.2 小節介紹)。

接下來詳細說明文件分析流程中每個步驟的機制。

- **詞幹轉換(Word Stemming)：**

此步驟將字詞轉換成它的詞幹，例如：複數的字詞轉成非複數 (plans → plan)、過去式動詞或現在分詞轉成 "原形(Root)" 動詞 (asked → ask；yielding → yield) 等。詞幹轉換的技術採用 Porter's Algorithm[25]，這個演算法制定了某些形式的字詞該如何轉換的規則，這些規則分為 1-5 個階段。若以 *V* 代表母音、*C* 代表子音、*L* 代表一般的字母(*V* or *C*)、任何 *C, V, L* 的組合稱為樣型(*Patterns*)，則此演算法五階段的規則轉換如下：

  - 階段 1：這個階段主要是處理複數形或過去式(plurals and past participles)。例如：plastered → plaster, motoring → motor
  - 階段 2：處理具有常見字尾的樣型(pattern matching on some common suffixes)。例如：relational → relation
  - 階段 3：處理特殊字尾(special word endings)。例如：hopeful→ hope
  - 階段 4：為了避免複合字詞，拆解好的 word 必須檢查更多字尾。例如：allowance → allow, inference → infer
  - 階段 5：檢查拆解好的字詞是否以母音作為結尾，並且適當地修正。

Porter's algorithm **詳細規則轉換的程序請參照附錄** A**。**

- **權重計算(Word Weighting)：**

將字詞轉換成它所屬的詞幹後，接下來就要計算這些詞幹的權重，在此我們只挑出「名詞」的詞幹來計算權重(以下仍用 "字詞" 這種說明代表已經經過 stemming **後得到的詞幹)。傳統資訊擷取過程中計算字詞權重的方式為** weighting = TF*IDF (Term Frequency * Inverse Document Frequency)**，**TF **表示該字詞在某篇文件中的出現頻率；**IDF **表示該字詞出現過的文件數的反轉頻率。其公式如下：**

$$tf_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}}$$ $freq_{i,j}$**代表** k_i **這個字在文件** d_j **中出現的頻率**

$$idf_i = \log_2 \frac{N}{n_i}$$ N**：文件個數；** n_i**：** k_i**這個字出現過的文件數**

$$w_{i,j} = tf_{i,j} \times idf_i$$ $w_{i,j}$**：** k_i **在文件** d_j **的權重**

這種考量 IDF **計算出來的權重會導致一些問題，例如當某字詞幾乎在每篇文件都出現過，不過卻在其中一篇出現了很多次時，此時** IDF **會很小，但對出現很多次該字詞的那篇文件來說，這個字詞可能會有某種程度的重要性，舉下面表格** 13 **的例子來說：**

word	Document				totfreq	IDF=log(N/n)	Entropy (Noise)	Signal
		D2	D3	D4		**計算出現的文件數**	**計算文件中的集中程度**	
W1	3	2	2	3	10	0	0.593315	0.406685
W2	0	5	1	1	7	log(4/3)=0.124939	0.345834	0.499264
W3	2	0	5	0	7	(log4/2)=0.30103	0.259825	0.585273
W4	0	3	0	0	3	log(4/1)=0.60206	0	0.477121

表格 13：計算字詞的 IDF 與 Signal 值之比較

Word2(W2)**因為出現在三篇文件中，所以計算出來的** IDF **值蠻低的，但出現在第二篇文件中的次數相較於第三篇與第四篇的次數則蠻多的。而** Signal **值可以強調這一點。**

Signal **與** Noise **是兩個相對的值，它們之間的轉換公式為：**

$\text{Signal}_i = \log(totalfreq_i) - \text{Noise}_i$ **，而** Noise **的算法為：**

$$E_{ij} = \sum p_i \log_2 \frac{1}{p_i} \qquad p_i = \frac{freq\ of\ \ k_i\ in\ Document\ D_j}{totalfreq_i}$$

在此，Noise **也可解釋為** Entropy **(亂度值)，它是一個介於** 0 **到** 1 **之間的數值，當某個字詞平均分佈在每篇文件中時，它的** Entropy **值會很高。舉例而言，假設現在只有兩篇文件，而** word_i **在這兩篇文件中的出現次數相同，則** $E = \frac{1}{2}\log_2 \frac{1}{\frac{1}{2}} + \frac{1}{2}\log_2 \frac{1}{\frac{1}{2}} = 1$ **；若** word_i **只出現在一篇文件中，則** $E = 1 \cdot \log_2 \frac{1}{1} = 0$ **。因此** Entropy **值愈大，表示該字詞愈不重要。**

在計算文件中的字詞權重時，本論文結合 IF-IDF **和** Signal**，使得權重的計算公式為：**

$$w_{i,j} = tf_{i,j} \times \left(\frac{idf_i + \text{Signal}}{2} \right)$$

每篇文件中的字詞(名詞)計算出權重後，我們選擇權重值大於 0.2 的名詞組成該篇文件的關鍵字向量 (Keyword Vector)。

- **改良型語意歧異解析(Modified WSD)：**

挑選出來的關鍵字(名詞)當作建立改良型語意歧異解析第一個步驟的候選字詞，對於每個候選的字詞，針對每個語彙鏈結，衡量該字詞所代表的語意與語彙鏈結中每個字詞的語意關聯度，利用複合語意的計算和鍵結擴充策略重新修改關鍵字向量的權重。其建構的過程和詳細說明請參考第三章的方法。

透過文件分析的過程，當每篇文件的關鍵字判斷出語意後，系統會對具語意關鍵字的文件分群。在推薦時，若使用者蒐集的文獻包含在某群中，則將該群的其他文獻推薦給使用者。

4.1.2 文件分群及推薦的流程

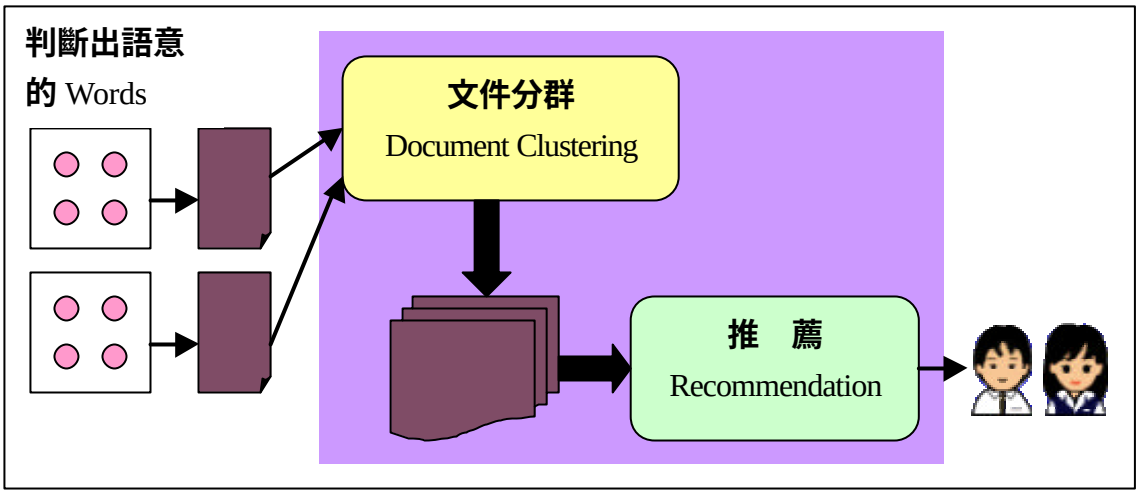

圖 24：文件分群及推薦的流程

當系統對每篇文件的關鍵字判斷出語意後，系統針對每篇文件的向量關鍵字將文件分群，分群好的文件依照使用者曾經蒐集的資料來推薦。

➢ **文件分群(Document Clustering)：**

文件分群的方法採用 Bottom-up Hierarchy Method**。其步驟如下：**

1. **首先每篇文件即代表一群；**
2. **計算兩兩群的相似度，相似度的計算公式為：**

$$sim(d_j, d_k) = \frac{\vec{d_j} \bullet \vec{d_k}}{\left|\vec{d_j}\right| \times \left|\vec{d_k}\right|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,k}}{\sqrt{\sum_{i=1}^{t} (w_{i,j})^2} \times \sqrt{\sum_{i=1}^{t} (w_{i,k})^2}}$$

3. **計算出來的相似度按照大小排序，若最大的相似度≧某個門檻值** (**我們的實驗採用** 0.1)**，則該兩群結合為一群，並重新計算群中心來代表這群更新過後的向量。**
4. **重複步驟** 2-3 **直到最大的相似度沒有大於門檻值為止。**

其示意圖如圖 25**：**

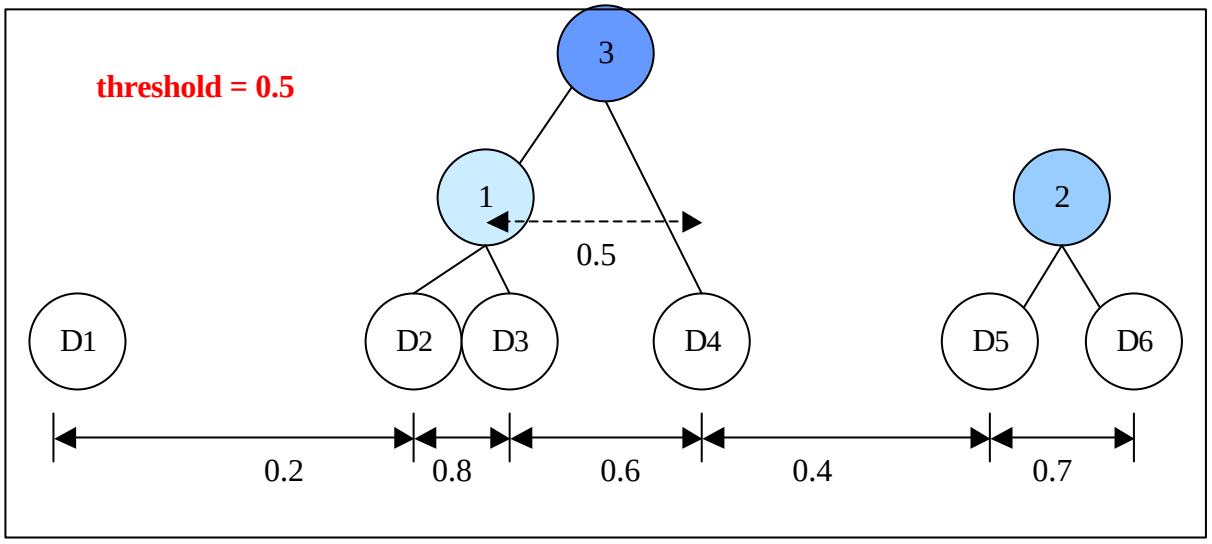

圖 25**：**Bottom-Up Hierarchy **分群方法示意圖**

圖 25 **最下面一行的數字代表兩兩群的相似度，文件** 2 **和文件** 3 **的相似度最大，且相似度** 0.8 **大於門檻值** 0.5**，因此它們最先被合為一群，重新計算群中心**

和該群中心與其他群的相似度；接下來文件 5 和文件 6 的相似度 0.7 為最大，且 0.7 大於門檻值 0.5，它們也被合為一群，重新計算群中心和該群中心與其他群的相似度；接著文件 2 和文件 3 形成的新群和文件 4 的相似度 0.5 為最高，因此這三篇文件合為一群，重新計算群中心和該群中心與其他群的相似度。至此沒有任兩群的相似度大於門檻值了，所以分群步驟結束，分好的群為{D1}, {D2, D3, D4}, {D5, D6}。

- **推薦(Recommendation)：**

依據文件分群後的結果來推薦使用者相關的資料，若使用者蒐集的文獻包含在某群中，則將該群的其他文獻推薦給使用者。我們預期推薦的結果都含有某種語意程度上的關聯。

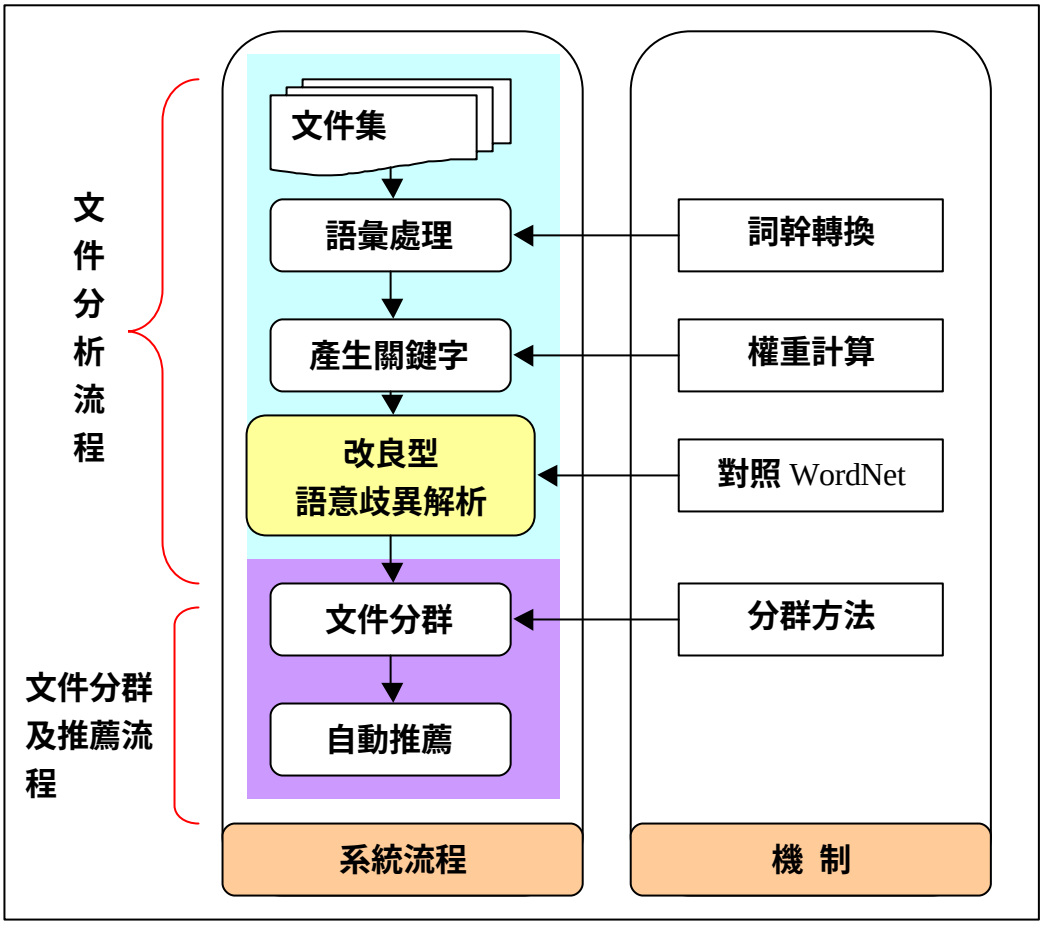

圖 26：系統流程圖

圖 26 為整個個人化參考文獻管理系統的流程，包含文件分析流程與文件分群及推薦流程。

第二節 應用於個人化環境

依照上述第一節介紹的個人化參考文獻管理系統的流程與機制，本論文設計的系統能輔助使用者管理其蒐集之參考文獻，並將分析使用者蒐集文獻的語意，藉此推薦與使用者所蒐集文獻具有相同語意的其他文獻，進而節省使用者檢視與蒐集類似文獻的時間。

本論文實作的個人化參考文獻管理系統結合交大個人化環境(MyLibrary@NCTU)，其介面提供以下功能：

- 新增文獻(輸入功能)：讓使用者將欲儲存的文件(包含期刊論文與會議記錄)輸入空白表單中。如圖 27 及圖 28 所示。

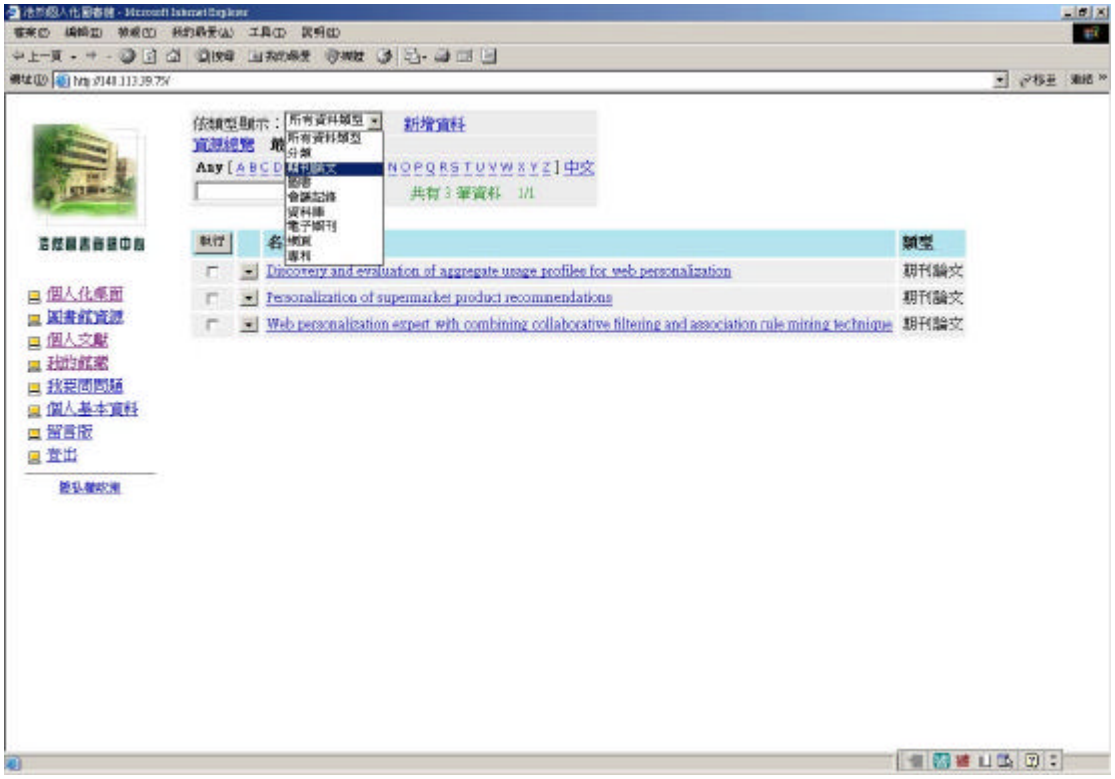

圖 27：選擇新增各種文獻類型

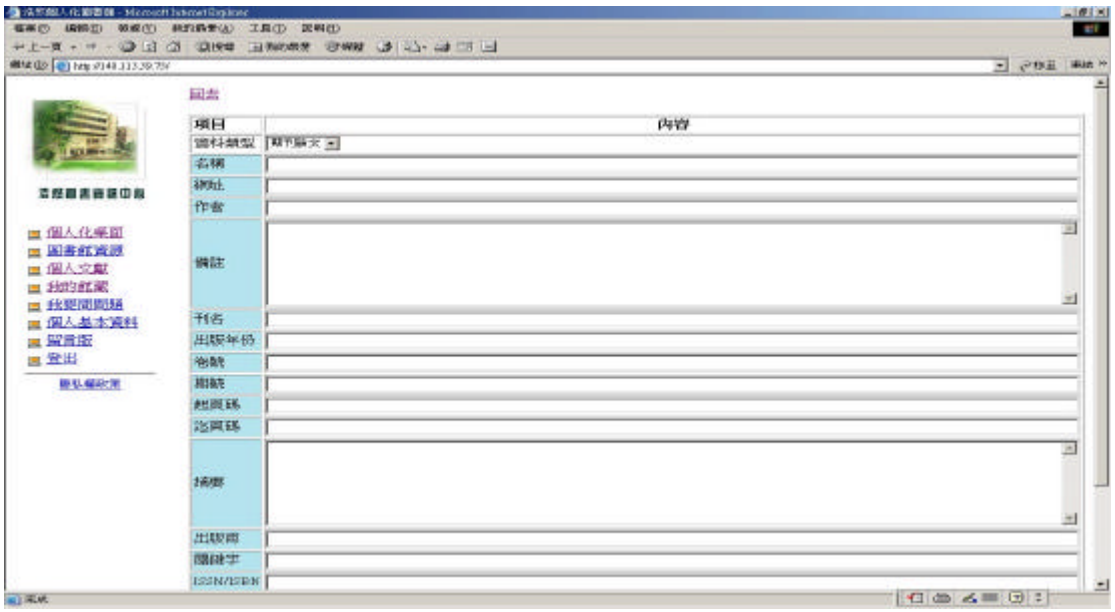

圖 28：新增期刊論文

系統因為新增的資料類型不同而出現不同格式的資料欄位，例如選擇新增期刊論文(如圖 27)，則出現期刊論文的欄位(圖 28)。

- 組織管理資料(夾)功能：讓使用者自定(新增)資料夾，如圖 29；對文獻或資料夾瀏覽、編輯、刪除、複製，如圖 30、圖 33 及圖 31；並可將文獻分門別類歸納入各個資料夾，如圖 33：

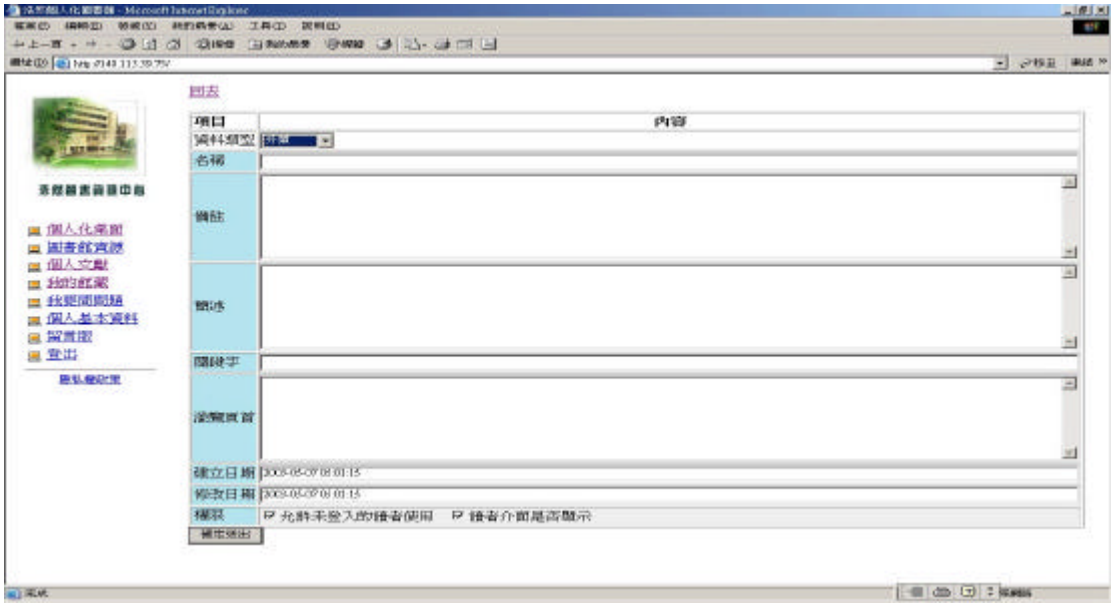

圖 29：自定(新增)資料夾

點選每篇文獻前面的▼按鈕，則可瀏覽、編輯、刪除該篇文獻。

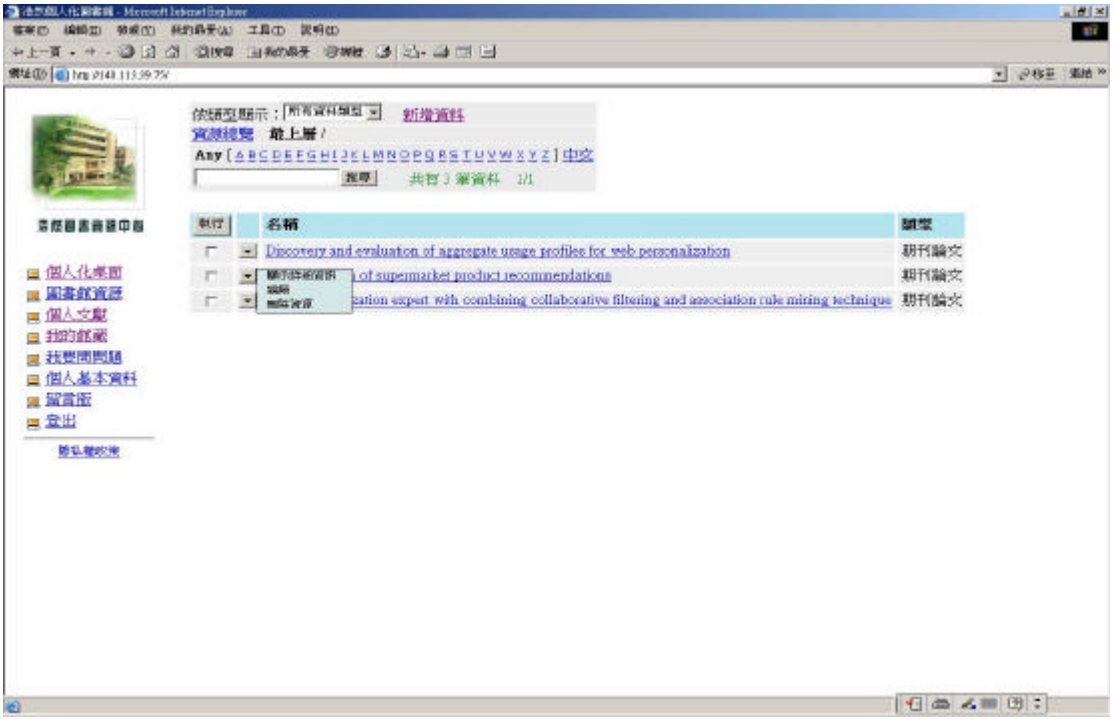

圖 30：選擇瀏覽、編輯、刪除文獻

- **瀏覽功能：使用者可以瀏覽文獻。**

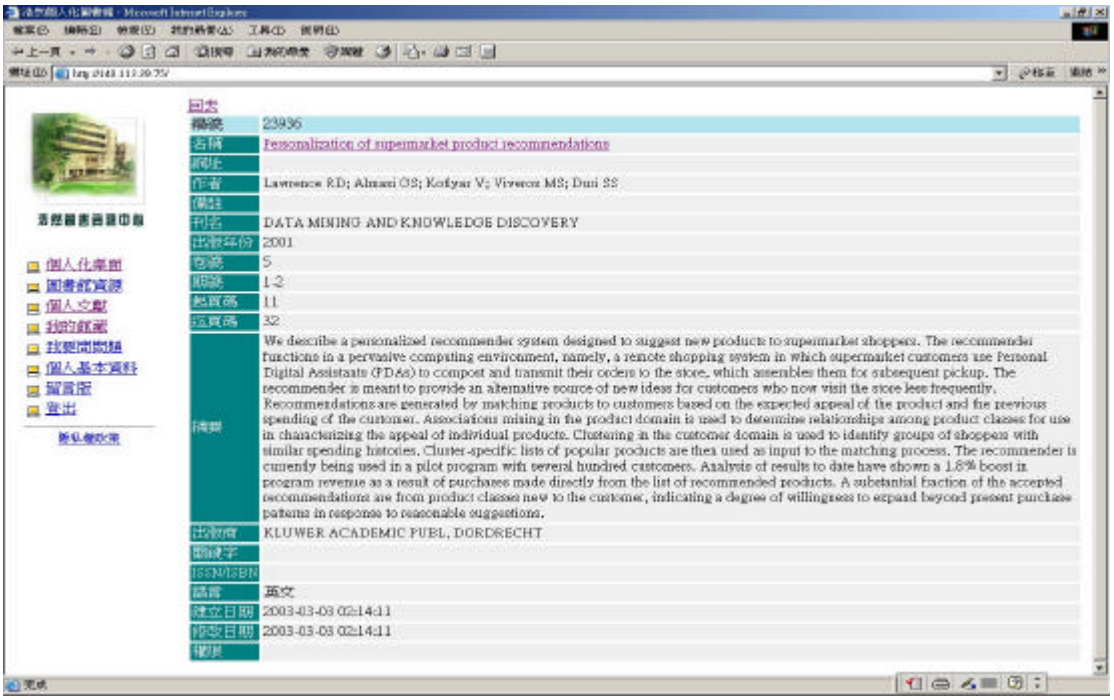

圖 31：瀏覽及顯示文獻詳細資訊

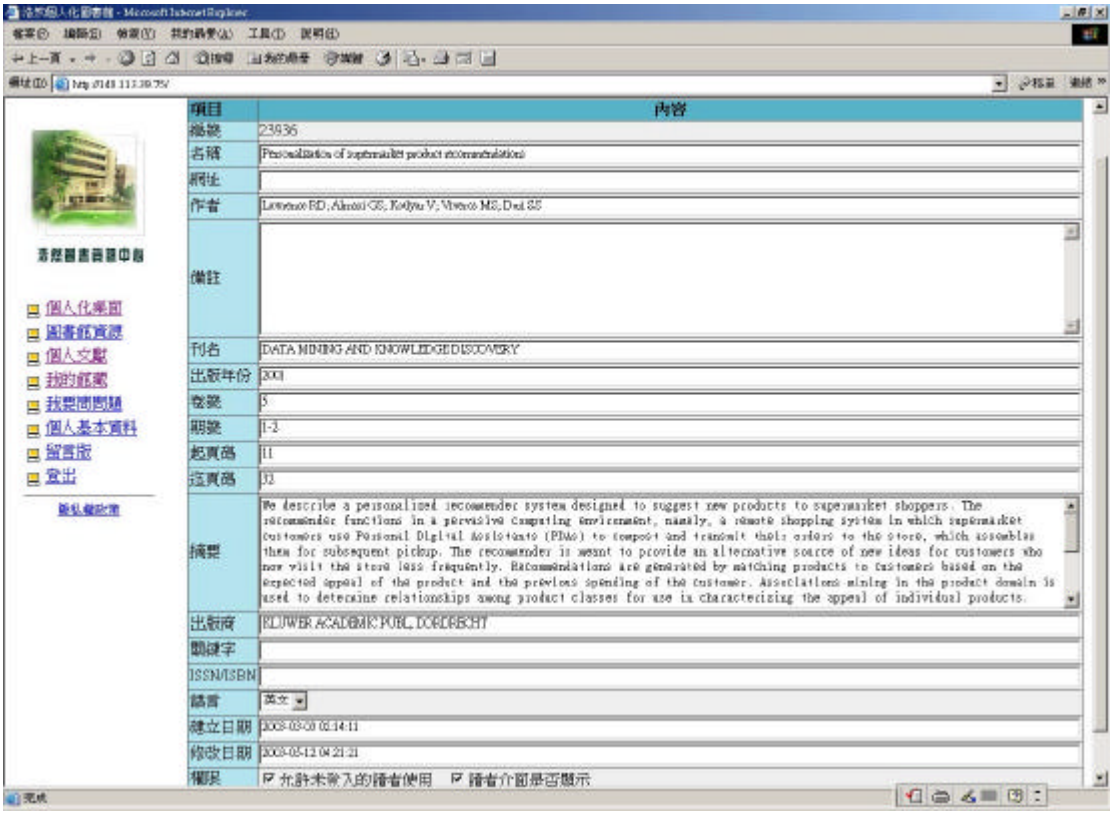

圖 32：編輯(更新)文獻資料

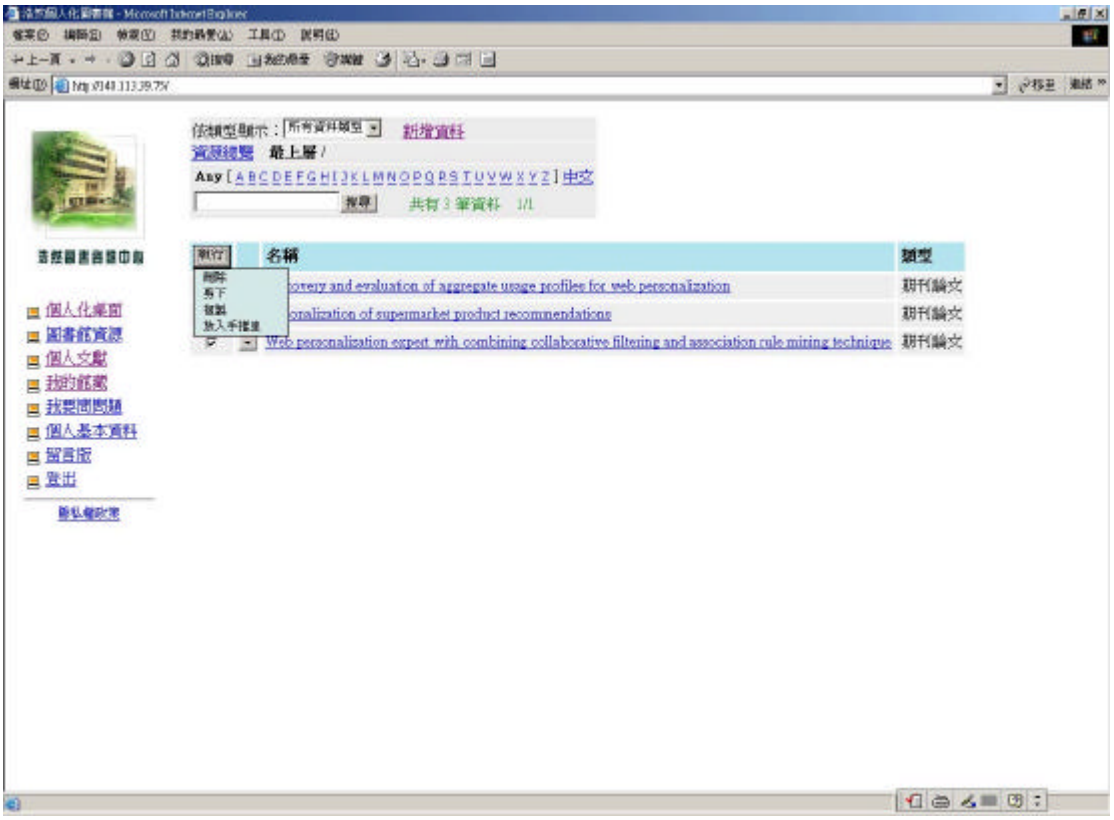

圖 33：刪除、剪下、複製文獻，並可將文獻歸類入資料夾(手推車功能)

- **搜尋功能：分為簡易搜尋及進階搜尋兩種；搜尋使用者儲存的文件。**

簡易搜尋(如圖 34 A)提供單一欄位讓使用者鍵入關鍵字來搜尋文獻；進階搜尋則提供多個欄位及邏輯運算子(AND, OR, NOT)的搜尋。

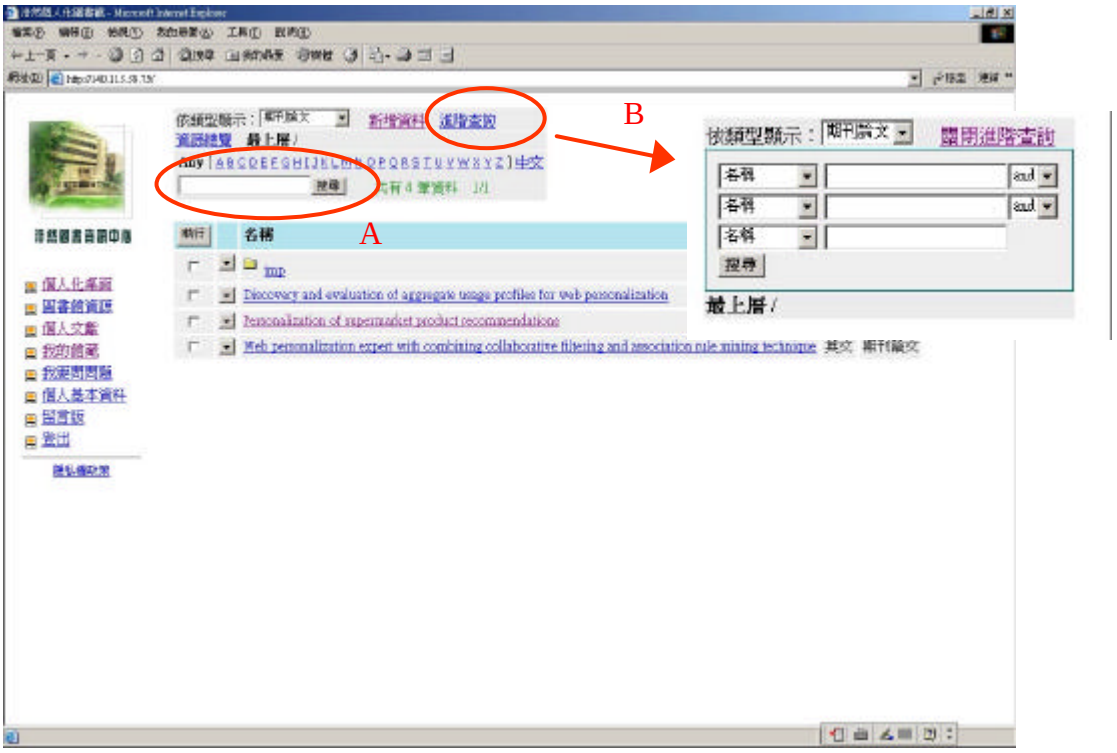

圖 34：簡易搜尋及進階搜尋功能

選擇資料類型後，出現進階查詢的選項(如圖 34B)，使用者可查詢多個欄位並選擇使用哪個邏輯運算子。

- **文獻分群及推薦功能：系統將使用者蒐集的文獻經斷詞切字處理及字詞語意判斷後，進行文獻分群，分群後的文獻推薦給擁有該群某些資料的使用者，如圖 35。**

圖 35：系統推薦使用者文獻

第五章 結論與未來研究方向

本章總結本論文並描述未來的研究方向。第一節說明本論文所提的方法應用在字詞語意判斷及文件分群上的效益，並總結個人化參考文獻管理系統的應用；第二節則說明未來可能的研究發展方向。

第一節 結論

本論文提出一套可達到自動文獻推薦功能的參考文獻管理系統，推薦文獻的主要概念為分析文件內容語意，將相同語意的文件分在同一群，並將同群中的文件推薦給相關的使用者。

在分析文件內容語意上，本論文提出一個新的語意歧異解析(WSD)的方法，以字詞語彙鍵結為基礎，改良為複合語意權重表示法並加入鍵結擴充的兩點策略－與定義相關的策略及一般性字詞的策略。實驗的結果證實本論文提出的改良型語意歧異解析方法比其他的方法來得好，且隨著策略的加入而提高語意歧異解析的正確率和降低無法判斷語意的比率。實驗出的結果亦證實加入語意後的文件分群結果比不具語意的文件分群結果來得佳。

我們以此改良型語意歧異解析方法為基礎實作出一套「個人化參考文獻管理系統」，主要功能有：(1)新增文獻資料；(2)組織及管理資料夾；(3)文獻搜尋與瀏覽功能；(4)資源共享與回饋；(5)利用字詞語意自動文獻分群；以及(6)推薦相關的文獻給使用者。

第二節 未來研究方向

本論文將傳統資訊擷取用於建立文件索引的 TF*IDF 方法從不具語意提升至具有語意的文件索引，在判斷語意時利用語彙鍵結並加以改良，提出複合語意權重表示法以判斷出字詞的語意，進而提昇文件分群的正確性，並作為推薦使用者文獻的依據。

未來我們將針對以下幾點進行更深入的研究：

1. 新增語意歧異解析的策略：本論文提出關於鍵結擴充的兩個策略，其目的為減少無法判斷出語意的比率，但相對地會增加判斷錯誤的比率。未來可以研究朝這兩方面同時改善的策略著手。
2. 加入動詞語意歧異解析：本論文只針對文件中的名詞進行語意歧異解析，不過我們提出的方法只要有一完善的字典且該字典定義豐富的詞彙語意集合關係可供對照的話，則可以適用在任何詞性中。若一篇文件能夠同時判斷出名詞與動詞的語意，則此文件內容的涵義將更容易了解。
3. 擴充使用者可自定分群的功能：在自定分群的功能上，未來可以加入更彈性的自定分群設計，例如使用者可以選擇保留一整群的文件內容，或保留某群中的某些文件，或者使用者可以將感興趣的文件組織成一群等。
4. 使用者推薦滿意度調查：系統推薦給使用者的文獻，若使用者回傳一些資訊給系統的話，藉由不斷的改善及學習，可以讓系統更為完善。

附錄 A：Stemming－Porter's Algorithm [25]

V **代表母音**(vowel, **是為** *a, e, i, o, u*)**；***C* **代表子音** (consonant)**；***L* **代表一般的字母** (vowel or consonant)**，任何** *C, V, L* **的組合稱為樣型** (*patterns*)**；Ø 代表空字串** (one with no letters)**；* 代表重複** 0 **次以上的樣型；+ 代表重複** 1 **次以上的樣型。**

select rule with longest suffix {
    sses → ss;
    ies → i;
    ss → ss;
    s → 0;
}
select rule with longest suffix {
    if ((C)*((V)$^+$(C)$^+$)$^+$(V)*eed) then eed → ee;
    if (*V*ed or *V*ing) then {
        select rule with longest suffix {
            ed → 0;
            ing → 0;
        }
        select rule with longest suffix {
            at → ate;
            bl → ble;
            iz → ize;
            if ((*C_1C_2) and ($C_1 = C_2$) and ($C_1 \notin$ {l, s, z})) then C_1C_2 → C_1;
            if (((C)*((V)$^+$(C)$^+$)$C_1V_1C_2$) and ($C_2 \notin$ {w, x, y})) then $C_1V_1C_2$ → $C_1V_1C_2$e;
        }
    }
}
if (*V*y) then y → i;
if ((C)*((V)$^+$(C)$^+$)$^+$(V)*) then select rule with longest suffix {
    ational → ate;
    tional → tion;
    enci → ence;
    anci → ance;
    izer → ize;
    abli → able;
    alli → al;

entli → ent;
eli → e;
ousli → ous;
ization → ize;
ation → ate;
ator → ate;
alism → al;
iveness → ive;
fullness → ful;
ousness → ous;
aliti → al;
iviti → ive;
biliti → ble;

}

if ((C)*((V)$^{+}$(C)$^{+}$)$^{+}$(V)*) then select rule with longest suffix {

icate → ic;
ative → Ø;
alize → al;
iciti → ic;
ical → ic;
ful → Ø;
ness → Ø;

}

if ((C)*((V)$^{+}$(C)$^{+}$)((V)$^{+}$(C)$^{+}$)$^{+}$(V)*) then select rule with longest suffix {

al → Ø;
ance → Ø;
ence → Ø;
er → Ø;
ic → Ø;
able → Ø;
ible → Ø;
ant → Ø;
ement → Ø;
ment → Ø;
ent → Ø;
ou → Ø;
ism → Ø;
ate → Ø;
iti → Ø;

ous → Ø;

ive → Ø;

ize → Ø;

if (*s or *t) then ion → Ø;

}

select rule with longest suffix {

if ((C)*((V)$^{+}$(C)$^{+}$)((V)$^{+}$(C)$^{+}$)$^{+}$(V)*) then e → Ø;

if (((C)*((V)$^{+}$(C)$^{+}$)(V)*) and not ((*$C_1V_1C_2$) and ($C_2 \notin$ {w, x, y}))) then e→ nil;

}

if ((C)*((V)$^{+}$(C)$^{+}$)((V)$^{+}$(C)$^{+}$)+V*ll) then ll → l;

參考文獻

[1] G. Salton, *Automatic text processing*. Addison-Wesley, 1989.

[2] G.A. Miller, "WordNet: An On-line Lexical Database," *International Journal of Lexicography*, vol. 3, no. 4, pp.235-312, 1990.

[3] R. Mihalcea and D.I. Moldovan, "Word Sense Disambiguation Based on Semantic Density," *Use of WordNet in National Language Processing Systems: Proceedings of the Conference*, 1999.

[4] G.A. Miller, M. Chodorow, S. Landes, C. Leacock and R.G. Thomas, "Using a Semantic Concordance for Sense Identification," *Proceedings of the ARPA Human Language Technology Workshop*, pp.240-243, 1994

[5] X. Li, S. Szpakowicz and S. Matwin, "A WordNet-Based Algorithm for Word Semantic Sense Disambiguation," *Proceedings of the 14th International Joint Conference on Artificial Intelligence IJCAI-95*, pp.1368-1374, 1995.

[6] R. Barzilay and M. Elhadad, "Using Lexical Chains for Text Summarization," *ACL/EACL Workshop on Intelligent Scalable Text Summarization*, 1997.

[7] D.I. Moldovan and R. Mihalcea, "Using WordNet and Lexical Operators to Improve Internet Searches," *Internet Computing, IEEE*, vol. 4, no. 1, pp.34-43, 2000.

[8] A. Montoyo and M. Palomar, "Word Sense Disambiguation with Specification Marks in Unrestricted Texts," *Proceedings of 11th International Workshop on Database and Expert Systems Applications, IEEE*, pp.103-107, 2000.

[9] H.T. Ng and H.B. Lee, "Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Examplar-Based Approach," *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL-96)*, pp.40-56, 1996.

[10] A. Suarez, M. Noeda and M. Palomar, "A Method of Restricted Knowledge Acquisition from WordNet," *Proceeding of the 3rd International Conference on Knowledge-Based Intelligent Information Engineering System, IEEE,* pp.38-41, 1999.

[11] J. Stetina, S. Kurohashi and M. Nagao, "General Word Sense Method Based on a Full Sentential Context," *Use of WordNet in National Language Processing Systems: Proceedings of the Conference*, 1998.

[12] D. Yarowsky, "Unsupervised Word Sense Disambiguation rivaling Supervised Method," *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pp.189-196, 1995.

[13] P. Resnik and D. Yarowsky, "A Perspective on Word Sense Disambiguation Methods and Their Evaluation," *Proceedings of ACL Siglex Workshop on Tagging Text with Lexical Semantics, Why, What and How?* Washington, pp.79-86, 1997.

[14] P. Resnik, "Selectional Preference and Sense Disambiguation," *Proceedings of ACL Siglex Workshop on Tagging Text with Lexical Semantics, Why, What and How?* Washington, 1997.

[15] B. Sankaran and V. Vaidebi, "Role of Collocations and Case-Markers in Word Sense Disambiguation: A Clustering-Based Approach," *IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, pp.625-630, 2002.

[16] R. Bruce and J. Wiebe, "Word Sense Disambiguation using Decomposable Models," *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pp.139-146, June 1994.

[17] G. Rigau, J. Atserias and E. Agirre, "Combining Unsupervised Lexical Knowledge Methods for Word Sense Disambiguation," *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, 1997.

[18] K. Lindent and K. Lagus, "Word Sense Disambiguation in Document Space," *IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp.288-293, 2002.

[19] G. Sidorov and A. Gelbukh, "Automatic Detection of Semantically Primitive Words Using Their Reachability in an Explanatory Dictionary," *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 3, pp.1683-1687, 2001.

[20] A. Hardy, "On the Number of Clusters," *Computational Statistics and Data Analysis*, vol. 23, pp.83-96, 1996.

[21] G. Milligan and M. Cooper, "An Examination of Procedures for Determining the Number of Clusters in a Data Set," *Psychometrika*, vol. 50, no. 2, pp.159-179, 1985.

[22] A.H. Tan, "Personalized Information Management for Web Intelligence," *Proceedings of the 2002 IEEE International Conference on FUZZ-IEEE'02*, vol. 2, pp.1045-1050, 2002.

[23] A.H. Tan, FOCI (search, cluster and personalize WWW, Patents, Publication and News). Available at http://textmining.krdl.org.sg/FOCI/

[24] M. Geffet and G. Feitelson, "Hierarchical Indexing and Document Matching in BoW," *Joint ACM/IEEE Conf. Digital Libraries*, pp.259-267, 2001. Available at http://www.bow4.cs.huji.ac.il/bow/

[25] B.Y. Richard and R.N. Berthier, *Modern Information Retrieval*. Addison-Wesley, ACM Press New York, 1999

[26] Digital Library Federation, "A Working Definition of Digital Library," 1998. Available at http://www.clir.org/diglib/dldefinition.htm

[27] RefWorks (Your Personal Web-based Database and Bibliography Creator). Available at http://www.refworks.com.tw/

[28] WordNet (a lexical database for the English language). Available at http://www.cogsci.princeton.edu/~wn/