

國 立 交 通 大 學
管 理 學 院 (資 訊 管 理 學 程) 碩 士 班
碩 士 論 文

結合案例推理與專家找尋之問題診斷支援機制

A Mechanism Combining CBR with Expert Finding for Problem

Diagnosis Support

研 究 生：張 宸 豪

指 導 教 授：李 永 銘 博 士

中 華 民 國 一 百 一 年 六 月

結合案例推理與專家找尋之問題診斷支援機制

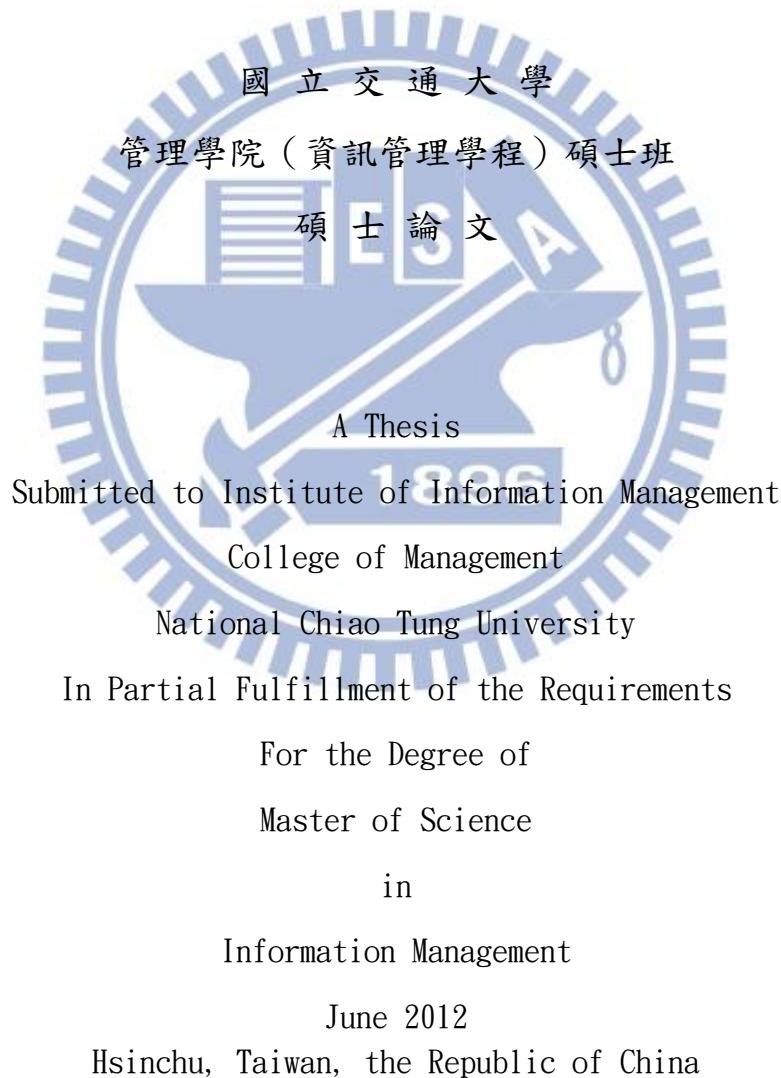
A Mechanism Combining CBR with Expert Finding for Problem Diagnosis Support

研究 生：張宸豪

Student : Chen-Hao Chang

指導教授：李永銘 博士

Advisor : Dr. Yung-Ming Li



中 華 民 國 一 百 一 年 六 月

結合案例推理與專家找尋之問題診斷支援機制

研究生：張宸豪

指導教授：李永銘 博士

國立交通大學管理學院（資訊管理學程）碩士班

摘要

如今先進製程控制與資訊科技深深影響著產品發展的各個面向。隨著產品越加複雜與全球化生產趨勢，其應用之支援資訊系統分工亦越趨精細、複雜。而企業有鑑於此，針對其投入之人力、時間與金錢成本亦越趨龐大。然而對於資訊人員流動快速、專業知識與異常處理經驗無法有效儲存、利用等問題仍舊是一項重要待解的議題。

在此，本研究嘗試以案例推理架構為主體結合文字探勘技術與專家找尋方法，來有效儲存、利用半結構化案例資訊，藉此解決案例庫儲存、利用困難等問題，並以更有效率的方式取得貼近問題本質之案例資料供處理人員以為新產生問題處理之依據。最後，並輔以基於社群網絡分析為基礎之專家找尋機制，建議經驗專家予處理人員諮詢，進而有效補足案例資訊不足之處。

最後，本研究將針對所提之相關方法學實作離型系統進行驗證，藉此期許能對企業內部資訊系統異常處理能力能有所提升，並強化其整體競爭力。

關鍵詞：案例推理、文字探勘、社群網絡、專家找尋

A Mechanism Combining CBR with Expert Finding for Problem Diagnosis Support

Student: Chen-Hao Chang

Advisor: Dr. Yung-Ming Li

Master Program of Institute of Information Management

College of Management

National Chiao Tung University

Abstract

Nowadays advanced manufacturing and information technologies have impacted on every aspect of product development significantly. With the increasing of product complexity and globalization manufacturing, enterprise manufacturing information change more and more complex. Therefore, sharing manufacturing knowledge understanding such as personal experience and exception handling is one important issue to be solved.

This paper introduces a research integrated text mining technique and top-k support documents expert search model based on Case-based reasoning (CBR) by using knowledge of the semantic structure of documents. CBR can use known experiences to solve new problems, we store the past problems as cases in a case base and a new case is classified by determining the most similar case from the case base. And the use of degree centrality of the expert candidate network can find the expert with the most influence through the experts score rank. Our approach of the CBR based expert recommends combined the reliability and influence between the experts can help users to get the expert support and resolve the potential problems of CBR such as lack of feedback and lack of a sufficiently rich case library. With these methodologies, we establish the Problem Support and Expert Recommend System (PSERS), which applied in the case base of fault diagnosis expert system of manufacturing information.

From the experimental results, these techniques are shown to be very effective in the modeling and extraction of the domain knowledge in the case base.

keyword : Case-Based Reasoning 、Text Mining 、Social Network 、Expert Finding

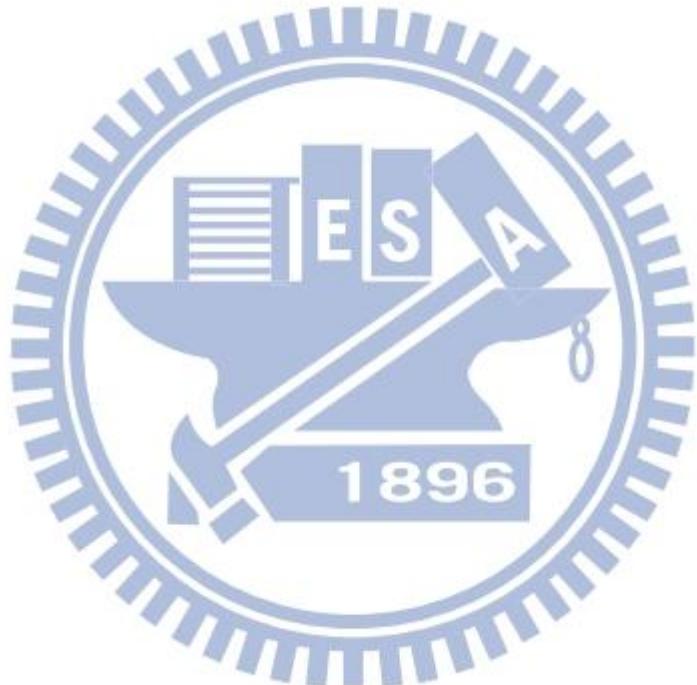
誌 謝

首先要感謝我的指導教授，李永銘老師。在這短短二年，努力督促我的學習與論文研究，讓我學習到嚴謹的研究方法、態度與細節，使得研究能夠如期完成。

此外，感謝撥空擔任口試委員的陳安斌教授、劉敦仁教授及楊耿杰教授，針對論文提供許多寶貴的意見，使學生獲益良多，也讓論文內容更為完整，思考更為全面。

在此，也感謝IEBI實驗室給我許多研究建議、提點我正確方法的學長們，以及同學鎮宇、婉瑜、育芳這二年來互相的幫忙與砥礪，很開心可以跟你們一起共渡學習、拼論文的時光。

最後要感謝我親愛的家人，特別是我的太太若辰以及處處幫忙的岳母大人，謝謝你們一路的關心與支持，讓我無後顧之憂而能全力以赴，完成碩士學業。



目 錄

摘要	i
Abstract	ii
目 錄	iv
圖 目 錄	vi
表 目 錄	viii
第一章 緒論	1
1.1 研究背景	1
1.2 研究動機與目的	2
1.3 研究範圍與限制	3
1.4 研究方法與章節架構	4
第二章 文獻探討	6
2.1 案例式推理(Case-Based Reasoning , CBR)	6
2.1.1 CBR 源起概念與定義	6
2.1.2 CBR 系統內涵	7
2.1.3 CBR 運作基本原理	8
2.1.4 CBR 優缺點	10
2.1.5 CBR 應用研究	11
2.2 文件資訊探勘	13
2.2.1 向量空間模型	14
2.2.2 中文斷詞	15
2.2.3 相似度衡量	16
2.2.4 K 最近鄰居法(K-Nearest Neighbor)	16
2.3 社會網路分析	17
2.3.1 類聚(Homophily)	18
2.3.2 關聯網路(Affiliation Network)	19
2.3.3 社會網路分析指標	22
2.3.4 總結	24
2.4 專家找尋(Expert Finding)	24
2.4.1 專家找尋途徑	24
2.4.2 專家評分方式	26
2.4.3 專家找尋機制	26
第三章 系統架構與設計	30
3.1 背景說明與系統架構	30
3.1.1 半導體製程	30
3.1.2 製造執行系統	31

3.1.3 作業流程	31
3.1.4 系統架構分析	34
3.2 問題診斷暨專家找尋模組	36
3.3 案例維護模組	43
3.4 系統管理模組	44
第四章 問題診斷暨專家推薦系統實作	45
4.1 系統開發工具與建置環境	45
4.2 系統操作流程與展示	47
4.2.1 使用者登入驗證	47
4.2.2 錯誤診斷暨專家推薦模組	49
4.2.3 案例維護模組	51
4.2.4 系統管理模組	53
4.3 效益分析	56
4.3.1 系統評估	56
4.3.2 系統驗證	57
第五章 結論與未來研究方向	60
5.1 研究結論	60
5.2 研究貢獻	60
5.3 未來研究方向與建議	62
參考文獻	64
附錄一：中研究平衡與料庫詞類標記集	67
附錄二：領域專家背景資料	68
附錄三：離型系統程式碼	69

圖 目 錄

圖 1.1 資訊系統值班紀錄表.....	1
圖 1.2 知識分類圖	2
圖 1.3 論文架構圖	5
圖 2.1 CBR 系統內涵組成圖	7
圖 2.2 案例式推理流程	8
圖 2.3 向量空間模型圖	14
圖 2.4 向量空間矩陣圖	14
圖 2.5 KNN 分類表示圖	17
圖 2.6 Wikipedia 作者間平均相似度	19
圖 2.7 二元網路圖	20
圖 2.8 關聯網路範例圖	20
圖 2.9 三元封閉域	21
圖 2.10 聚焦封閉域	21
圖 2.11 成員封閉域	22
圖 2.12 程度中心性示意圖	23
圖 2.13 個人化分析示意圖	25
圖 2.14 文件化分析示意圖	25
圖 2.15 問題與專家關係圖	27
圖 2.16 問題與專家網路圖	27
圖 3.1 SiView 架構圖	31
圖 3.2 企業現行問題解決流程圖	32
圖 3.3 系統功能結構圖	34
圖 3.4 系統架構圖	35
圖 3.5 問題診斷暨專家推薦模組流程圖	36
圖 3.6 案例維護模組關係圖	43
圖 3.7 系統管理模組關係圖	44
圖 4.1 離型系統 ER-Model	46
圖 4.2 系統登入畫面	47
圖 4.3 使用者系統功能畫面	48
圖 4.4 工程師系統功能畫面	48
圖 4.5 系統管理員系統功能畫面	49
圖 4.6 錯誤診斷暨專家推薦操作畫面	50
圖 4.7 錯誤診斷暨專家推薦結果畫面	50
圖 4.8 案例維護模組主畫面	51
圖 4.9 案例新增暨更新操作畫面	52

圖 4.10 案例刪除操作畫面.....	53
圖 4.11 權限管理操作畫面.....	54
圖 4.12 系統參數管理操作畫面.....	55
圖 4.13 同義詞管理操作畫面.....	56
圖 5.1 知識管理循環.....	61



表 目 錄

表 2.1 案例式推理系統整理.....	12
表 2.2 斷詞法優缺點整理表.....	15
表 2.3 網路中心性指標計算公式表.....	23
表 2.4 專家找尋方法優缺點比較.....	26
表 2.5 TOP-K 專家找尋公式參數整理表	28
表 3.1 案例一整理資訊.....	38
表 3.2 案例二整理資訊.....	38
表 3.3 關鍵字頻率整理表.....	38
表 3.4 新案例整理資訊.....	39
表 3.5 案例相似度比較表.....	39
表 3.6 近似案例維護人員資訊.....	40
表 3.7 專家初始分數計算範例.....	41
表 3.8 專家關係矩陣表.....	41
表 3.9 專家關係權重表.....	42
表 3.10 專家總合分數表.....	42
表 4.1 離型系統開發環境.....	46
表 4.2 樣本資料數統計表.....	57
表 4.3 文件分類判別表.....	57
表 4.4 案例分類驗證表	58
表 4.5 案例推理測試案例.....	58
表 4.6 案例推薦滿意調查表.....	59

第一章 緒論

社會自傳統經濟邁向知識經濟後，企業主要關鍵成功因素不再只是豐厚的資金、極大的產能、最好的品質。要持續保有競爭優勢，甚至更進一步創造新契機來擊敗競爭對手，取決於企業能否積極轉化並善用知識員工的智慧資本，並使其成為企業競爭的利器。因此，如何有效利用各種方法或流程將員工知識或經驗儲存並使其轉化為顯性知識，使之成為企業智慧資產而能有效利用與傳播，儼然成為現今知識管理重要議題，亦是企業為永續經營所必須面臨的挑戰。

1.1 研究背景

隨著資訊科技快速的進展，透過各種資訊系統的支援讓各產業得而從早期人工作業模式轉換為低人力使用率的自動化生產模式。而企業為保持其產業競爭力，除著重於產品或服務的創新之外，更必須致力於製造成本的降低與品質的提升。因此，隨著業務需求的開展與品質控制的要求，企業之資訊系統種類越趨繁雜、分工越趨精細，相對投入之開發與維護等人力、成本亦越趨龐大。

然企業內部資訊單位針對其所維護之資訊系統異常處理經驗與相關專業知識通常由各系統負責工程師所保存，而其儲存之形式通常為記憶、文字檔、表格資料等。如圖1.1所示即為某企業內部異常處理紀錄表。

	B Date Time	G Problem	H Cause	I Action	K Recorder
1	2006/12/16 0:00	Cannot reserve (told by TCS); User couldn't reserve FOUP	Error msg: Lot has not been reserved. FOUP in stocker, just waiting for transportation. User/TCS did not check transfer job on site and deleted reservation but transfer command still exists in MCSOPI	Ask AMHS member to delete the transfer job. EQP EGA01(FULL-A) then reserved fooup again.	
2	2007/1/4 15:00	User got ECMS alarm continuously.	It's ECMS Alarm, which ask user to call 81 when got the alarm continuously.	Notify user to check the engineering Lots she asked. If they(the engineering Lots she asked) had changed and got the alarm again. Please called us again. With other questions, please call	
3	2007/1/2 9:13	SCHDB Lock.	User disconnect SCH system abnormal.	Connect to SCH DB, and delete lock record.	
4	2007/1/2 10:20	BackupNGHold	SpecOverHold occur first. Cause BackupNGHold occur.	Release SpecOverHold and BackupNGHold. System will auto execute Backup step.	

圖 1.1 資訊系統值班紀錄表

資料來源：本研究整理

但是在上述之各種模式下並無法有效將工作經驗或知識分享予組織，使組織知識能與時俱進，亦無法讓相關人員快速的查詢與利用。因此，當資訊從業人員進行單位輪調、轉換跑道時，其專業知識與經驗亦隨之流失而對企業營運造成極大的損失。以半導體公司為例，因其產業特性、品質控管與客戶稽核等業務要求，通常必須透過核心製造執行系統來負責與外部各資訊系統溝通，並進行自動派貨、在製品生產、排程控管與生產資料拋轉等重要任務。當其發生異常時，常導致產線暫停或造成晶圓產出異常等重大損失，並進而造成營利虧損或產品無法準時交付等重大問題。因此，為避免企業因為資訊系統異常造成其營運損失，如何以有效率、有系統的方式儲存、利用以往處理經驗來協助解

決相關之系統異常，並快速做出正確回應便成為一個企業值得深入探討、研究的議題。

知識是一種有價值的智慧結晶，可以是資訊、經驗心得、抽象的觀念、標準作業程序、系統化的文件或以具體的技術等方式呈現，型態大致可分為隱性知識與顯性知識。如圖 1.2 所示，所謂隱性知識是指比較複雜，無法用文字描述的經驗式知識，不容易文件化與標準化的獨特性知識，以及必須經由人際互動才能產生共識的組織知識。通常其成本較高，可重複使用的機會較低；而顯性知識，是指可以文件化、標準化、系統化的知識。



因此，如何將隱性知識經由整理、歸納、分類、儲存等手段來達到顯性的程度，並且能夠十分便利的一再使用，是企業在知識管理活動中的重要工作。

然而為支援企業知識管理流程，一般建置符合企業流程與管理墓地之知識管理系統。雖隨著知識文件增加、更新能有效協助人員獲取有用之專業知識。但若知識庫中沒有極相似之知識案例則將會造成其瓶頸。因此，如何利用近似問題類別之案例並運用合適的技術來找出合適的專家來協助或提供諮詢以補知識庫不足便成企業另一重要課題。

1.2 研究動機與目的

案例式推理(Case-based Reasoning, CBR)系統，自 1982 年 Roger Schank[2]提出案例式推理後，廣泛應用於教育、軍事、法律、工程、醫學等，其中又以客服系統(Help Desk)應用最為理想[34]，因其對於資訊人員流動率過高、系統種類繁多、專業知識分散不易整合上，使用案例式推理系統為是較好的選擇。CBR 除了是一種合理的推論模式外，對知識學習的建構也是一個好的方法，適合於許多不同種類的推理工作，對工作績效的提昇與表現有正面的功能[45]。

然無論案例庫更新如何快速，搜尋方法如何精進，仍舊無法有效解決新產生之問題或相似度極低之案例等狀況。因企業領域專家針對其相關領域之工作經驗、創新思考等

專屬之內隱知識，亦隨從業時間增加而快速累積與發展，但並無對應知紀錄文件而無法有效利用。所以，針對領域內未知的新問題發生時仍須諮詢該領域內之經驗工作者或專家來進行相關問題判斷與處理。專家家找尋機制(Expert Finding)為近幾年一個新興的課題，有許多學者亦針對該領域付出相當心力並提出各種不同的方法，並進而協助使用者或企業能從內部網路與線上論壇找到特定主題之專家。

因此本研究將嘗試以案例推理機制為基礎，結合文字探勘(Text Mining)技術以及社會網路分析基礎下所發展之專家找尋機制等方法學來實作問題診斷之離型系統，來有效利用以往經驗解決類似問題，並根據案例描述等條件來找出可能之潛在專家或經驗人員提供人員諮詢以補案例之不足，並期能有效解決下列問題：

- (1) 資訊系統案例及異常處理經驗無法有效儲存、搜尋與利用。
- (2) 無法運用以往處理經驗支援未來產生之異常問題處理。
- (3) 資訊人員流動頻繁使其專業知識與處理經驗流失，進而造成營運之損失。

並進而能達成下列目標：

- (1) 提供企業資訊單位有效且便於使用之案例維護、查詢介面。
- (2) 資訊系統異常案例、處理經驗儲存中央化。
- (3) 降低查詢時間，提升問題處理速度。
- (4) 減輕因人力流失所造成之營運損失。

如此，將對企業組織效率之提升，知識經驗增加、人力資源之節省與異常損失之降低將有顯著的成效，並進而能協助提升企業整體競爭力。

1.3 研究範圍與限制

因本研究主要實驗之資料來源為半導體某企業內部之資訊系統，但資訊系統因支援性質或功能面不同而種類繁多、問題類別及對應之處理狀況亦各異。所以本研究知實驗範圍主要以企業內部之製造執行系統為對象，並根據其系統重要程度與發生頻率高低來篩選出適用之系統問題分類與發生頻率較高之案例作為研究樣本，而後使用文字探勘的方法針對案例描述之類別、異常描述等各項參數進行近似案例擷取，並配合以案例式推理架構來加強案例資料之儲存、修改與再使用等流程，最後再輔以基於社群網路之專家找尋機制來找尋適合之專家並推薦專家來補其案例不足之處。

1.4 研究方法與章節架構

本論文架構如下圖 1.3 所示，主要區分為五個章節。

第一章為緒論，說明本研究之背景、研究動機與目的、研究方法與章節架構。

第二章為文獻探討，針對案例推理、文字探勘、社會網路分析、專家找尋機制等相關文獻作整理與探討。

第三章將嘗試以案例推理結合文字探勘等方法學進行案例儲存、擷取與利用，並輔以專家找尋機制推薦潛在專家來進行系統設計，並參考某企業內部之系統異常作業需求及其所需知功能，加以規劃本研究之離型系統。

第四章則將針對系統設計之功能與規範進行系統的實作與測試，依其對應之方法學及設計之流程建立離型系統，展示系統各部份功能驗證系統架構可行性，並將篩選之各類別案例匯入案例庫。本研究將以 90% 資料為訓練案例，其餘 10% 為測試案例進行系統驗證，並利用傳統資訊擷取常用之正確率與召回率等指標進行擷取正確與否之驗證，並委請領域專家針對本研究所推薦之案例及潛在專家進行滿意度評估，以確認離型系統之資料擷取與專家推薦是否合於現況所用。

第六章為結論與建議，將根據實驗結果提出本研究的結論、建議、以及未來研究方向之建議。



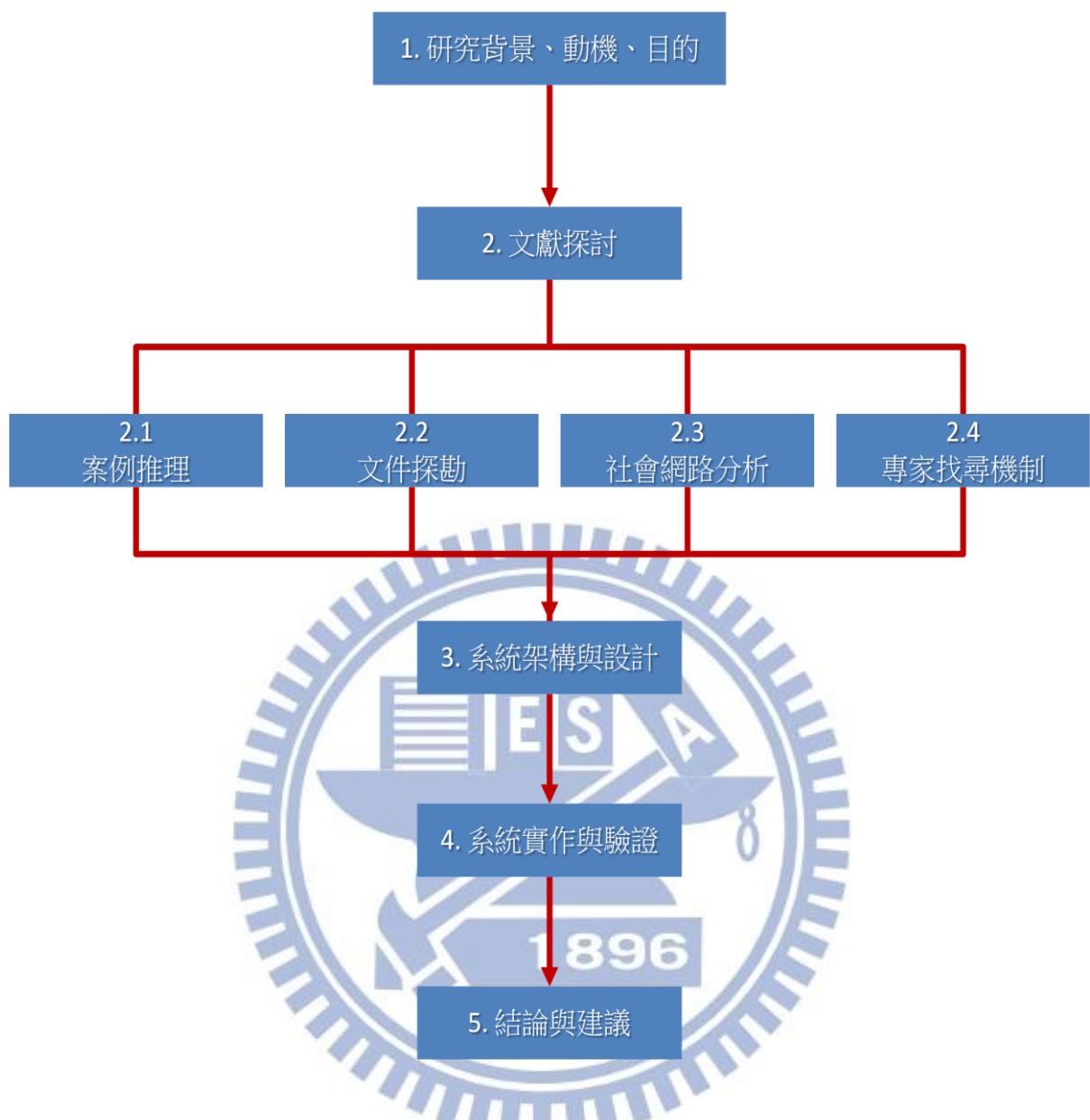


圖 1.3 論文架構圖

資料來源：本研究整理

第二章 文獻探討

本章節將針對本研究所使用到各種技術概念作一詳細介紹，包括案例式推理、文件探勘、社會網路分析及專家找尋機制等。

2.1 案例式推理(Case-Based Reasoning, CBR)

案例式推理(Case-Based Reasoning, CBR)是近年來人工智慧領域中所新興的一項重要的推理方法，一種有效解決問題的方式。通常是藉由先前所遇到的近似問題來描述或解釋新問題，並利用以往經驗來解決目前所遇到之新產生的問題或困境，常被用來建構以知識為基礎的系統。

在習慣領域九大構想中的類推聯想[35]亦指出，當人們面對或遇到一個新事物、新問題或新概念時，大腦首先調查它的特徵或屬性，以便於與已經知道的事物、概念建立起關係。當關係一經建立後，則先前的記憶結構、知識就會自動被運用於新事物、新問題或新概念的認識和理解。因此學者提出案例式推理方法來輔助人類在理解與記憶上的弱點，並藉由電腦的輔助來突破人類記憶及查找能力的限制，以協助決策者利用以往發生的個案來找出可能之解決方案，讓決策者能根據問題的特徵與情境來決定其解法，進而提升其解決問題的正確度與創造力。

而 Watson[3]更提出案例式推理理應是一個方法論而不是一項技術。該學者亦藉由案例式推理週期的案例擷取、案例適應等四大階段之應用，細部展示其各階段分別可依需求或狀況採用多種技術讓案例推理更加成熟，亦能更加契合不同產業或狀況。

2.1.1 CBR 源起概念與定義

案例式推理(Case-based Reasoning, CBR)，起源於 Schank 和 Abelson 在 1977 年[1]所提出手稿(Script)模式來模擬人類的知識、記憶。將一般性知識記錄在一張張的手稿上，用來表示人類記憶，並能用來描述固定事件之資訊。但其互相獨立，只能用作知識的儲存無法交互查找出取得解決方案。而後 Roger Schank 於 1982 年[2]更提出動態記憶模式(Dynamic Memory Model)的概念，以加入索引(Index)概念來連結記憶、知識、經驗與學習過程，並以記憶庫(Memory-based)方法來進行推理，而被認定為最早的案例式推理。

此概念於 1983 年由 Kolodner 應用並發展出第一個案例式推理系統 CYRUS，以案例為知識基礎，允許使用者詢問相關問題來取得應答，該案例記憶模式亦成為後續案例式推理系統之開發基礎，如 MEDIATOR、JUDGE 等。

Riesbeck 與 Schank 指出[4]，人類的學習力與創造力是由之前所經歷的案例得來，越多的案例就擁有越多的參考案例。若將案例視為知識或記憶的基本單位，則累積的案例量越多，代表所能處理的問題經驗亦越多，那麼就越能解決所面臨到的更多樣性問題。Watson[3]也提到，案例式推理是人工智慧領域中研究了解人類如何記憶資訊、回憶資

訊，並藉由回憶過去類似問題的處理經驗來解決目前所遇到的問題。

由此可知，藉由案例庫與索引的結合與推理，可有效彌補人類限制並提高使用效率，且相當適合運用於以知識為基礎的應用系統。

2.1.2 CBR 系統內涵

案例式推理系統針對其運用領域之不同，其系統架構亦隨之不同，根據[26][27]所做的分析，說明了不同的案例推理系統的發展均有其對應之背景與重點，但基本上各系統的構成均包含有知識範疇、設計案例、推理方式、系統介面、案例庫等五大構面，如圖 2.1 所示。系統各構面則分別說明如下：



(1) 知識範疇(Knowledge Domain)：

每種知識的內涵都具有不同的屬性，而就不同特性的問題設計與範圍或功能來加以定義，例如衝突紛爭、醫療診斷、建築設計、結構設計、教學訓練等，就會有不同的推理方式與表現方式。

(2) 案例設計(Design Case)：

一般而言，案例的設計應含有案例的問題背景、成因、解答或與其他有關的影響因素。而案例式推理是將問題本身的形成過程、專家訪談所得經驗、實際運作情形及其他相關文獻資料，以案例方式組織整合起來，經由搜尋、選擇、運用於新的情境。

(3) 推理方法(Reasoning Methods)：

在案例系統中，案例式推理是主要的推理方式。不同領域的案例推理系統，就有不同的推理方法，其間也會混合著其他不同的推理方式，如規則式、歸納式、最鄰近值法、階層結構法等。

(4) 系統介面(System Interface)：

系統介面主要功能在於是使運用者方便將問題的描述輸入、迅速取得問題解答。至於設計形式，則取決於設計者對案例的屬性的認知與規劃。目前由於網際網路已是目前生活的重要一部分，利用全球資訊網提供案例資訊已是時勢所趨，因此，本研究將朝此方向來進行。

(5) 案例庫(Case Base)：

案例庫將項目繁多的各類案例組織整合，並分別給予索引與指標或特徵值，方便使用者取得與運用；同時必須時時充實案例庫內容，使之更豐富化，讓問題解答更順暢。

2.1.3 CBR 運作基本原理

根據 Klaus-Dieter[5]所整理，案例式推理流程共分為四個步驟如圖 2.2，包括擷取(Retrieve)、再利用(Reuse)、修改(Revise)與保留(Retain)，簡稱為 4R。

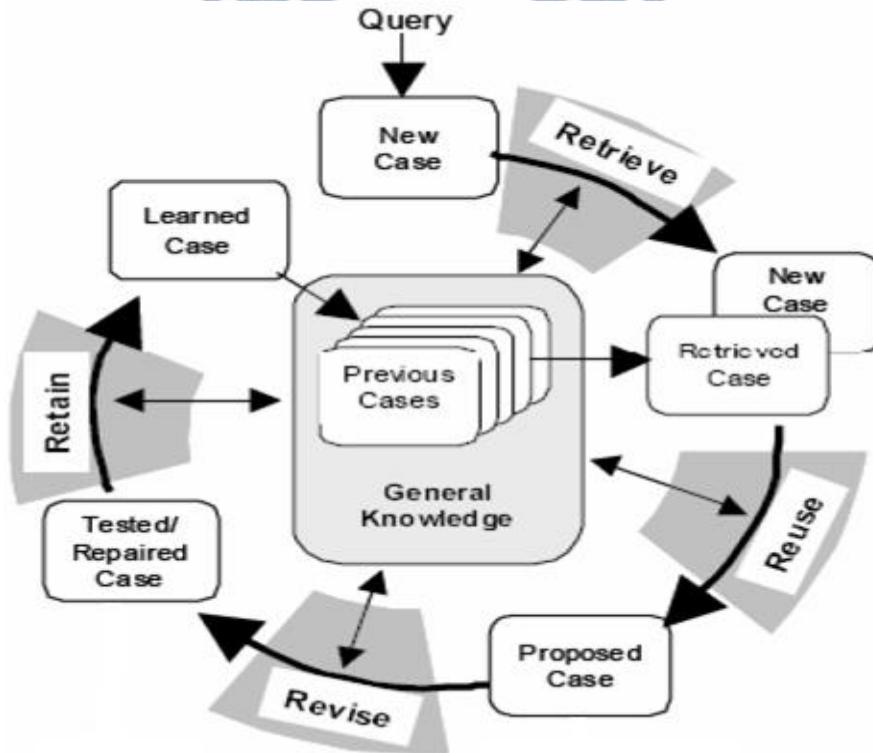


圖 2.2 案例式推理流程

資料來源：Klaus-Dieter Althoff

分別描述如下：

(1) 擷取(Retrieve)：

通常包含擷取過去相似案例與選取最佳案例兩部分。擷取過去案例在於從案例庫取得較近似的案例，亦即取得能夠對新案例進行描述並做出相關應對的案例，而選取最佳案例則目的在於自眾多相似案例中取出最貼近新案例狀況之候選案例。在此案

例擷取階段中，如何對案例間相似度判斷則為關鍵點所在。一般而言，沒有一個已存在的案例是正好完全滿足新的情況，因此必須經過搜尋的過程才能擷取案例，一般常用的搜尋方法大致有：序列搜尋(serial search)、階層搜尋(hierarchical search)、模擬平行搜尋(simulated parallel search)[28]。而案例間諸多之差異與相似點，用何種方法才能搜尋到合適的案例，大體而言有如下的方法[26][29]：

■ 最近鄰近值(nearest neighbor)：

CBR 主要的運算工具是靠電腦來執行，相似值是介於 0 與 1 之間。當運算值為 0 時，表示完全不合適，運算值為 1 時，則表示完全吻合，越接近 1 其相似性愈高。如果與現行問題狀況相類似的案例有多數時，就必須再對舊案例與新問題比對相似性，評估特徵相似性，取得加權值與相似度最高的案例。

■ 歸納法(induction)：

利用歸納法，決定那些特徵最能區分、辨別案例的差異，把案例加以分門別類，相似的歸納成一類，同類者再規劃成若干子類，如同決策樹結構來組織記憶這些案例，此種結構可以加速案例選取的效率。

■ 知識導引歸納法(knowledge guide induction)：

在資料庫中，以此方法與其他技術結合使用，用人工來識別案例特性，將知識應用到歸納法來處理。

■ 規則取得法(template retrieval)：

如同結構化查詢，經由規則取得法回饋全部符合案例的參數值，這種技巧常在其他技巧前使用，如在最鄰近法前使用，來限制搜尋案例的空間。

(2) 再利用(Reuse)：

由案例擷取階段所選出之案例，包含前人或自身的知識和經驗，來提供一個可行的建議或方向。由於新案例與舊有案例並非完全相同或適合，必須再加上自身的經驗或知識加以判斷與修改，就如同人類過去經驗與現在所遭遇問題並不完全相同，必須考量其他相關因素並加以修正，方能解決所遭遇之問題。

(3) 修改(Revise)：

因為案例式推理系統扮演的是輔助的角色，並不能真正的擬出正確的解決方案，最後決定權仍在於決策者本身。因此，在運用之前案例時必須利用比較的方式，將取得之案例與新問題進行分析。而實際運用後之結果，則能進行修改後存入案例庫幫助系統學習以形成新的經驗。一般而言，針對選取案例中不合適部分作修正，來滿足新問題的需求。修正的方法大致上有三種[26][29]：

■ 替換(substitution)：

在案例的解決方法中，一些元素不能滿足新的問題且局部不適用或尚未有答案，找一些能給解答的條件來替換使用。

■ 結構性修正法(structural adaptation)：

利用規則庫儲存的修正規則性直接就案例不合適部分進行修改。

■ 衍生修正法(derivational adaptation)：

在舊的案例中，一些元素不能滿足新案例需要，就重新評估之前案例的重要性，

為新的案例評估變化替代元素，以滿足新案例的要求。亦即選取一個案例後，經由案例評估得到不合適的部分，再進入案例中擷取可以修正不合適部分的案例，加以替代或增減，以滿足新問題的需求。

(4) 保留(Retain)：

儲存案例以增進效率的設計是 CBR 系統的重要概念。案例儲存必須考慮案例索引的特性，使之組織成一個易於辨識的結構，並且簡化儲存、擷取的方法，以達成有效率的搜尋與擷取案例[28]。當實際應用之前案例並經過修改等步驟滿足新問題，即成為一個新案例，它就是新的知識，將會儲存於案例庫供其他決策者作為參考，同時完成學習的功能。而案例庫則同資料庫分門別類，運用案例索引來進行搜尋與使用。當案例庫越大時，其包含案例越多，則推理所得之結果亦越佳。

2.1.4 CBR 優缺點

針對案例式推理，整理 Fritz[6]、Mehdi & Owrang[7]，Ralph[8]等學者提出之相關優缺點如下。

優點：

(1) 協助人類處理事情能力：

CBR 系統運作模式類似於人類思考方式，但藉由索引更能有效尋找以往的經驗。與其他類似支援系統比較，CBR 系統更能協助決策者以更徹底、更中立的方式評估各種可能性與制定決策。

(2) 案例庫容易被理解：

在系統中遵循邏輯與直覺使用的方式進行設計，較不易讓使用者產生認知誤差。

(3) 獲取資料容易：

案例庫沒有複雜的規則影響，案例與案例之間不會互相影響。因此可以很方便且快速的上線，隨著案例庫中案例量的增加，其學習效率也隨之增加。

(4) 案例庫學習能力：

案例庫可不斷新增案例資訊，不需要經過判讀與比較，隨著案例增加即能有效增加案例推理能力。

(5) 促進新知識的結合：

新的知識或經驗可以持續的新增至案例庫，不會影響任何已存案例，且更容易維護及擴充。

(6) 失敗經驗亦可記錄：

與專家系統相較之下，除成功案例外；CBR 系統亦可記錄失敗案例以供後續之經驗，並能藉此累積知識。

缺點：

(1) 當案例庫數量大量增加時，需要更高容量與計算資源。

(2) 當案例數量較少時，其準確度較低。

(3) 案例的修正必須仰賴人為修正，以符合實際狀況。

2.1.5 CBR 應用研究

案例所描述的是一種問題情境，案例推理嘗試提供一個問題解答的管道，呈現一種努力的成果及經驗分享與創新。以取得知識觀點而言，案例資料庫其多樣性的案例，是諸多專家解決問題的經驗與眾人知識的集合。其運用之目的在於有效的保存與取得相關案例，且能評估和決定如何將案例應用於新的問題上，來協助使用者更有效率的解決問題，隨著問題解決的經驗累積，案例資料庫不斷的充實多樣化的案例，案例愈多，愈有助於人們處理新的問題，在實際運用上更有其價值[46]。

案例式推理自被Roger Schank提出後，應用相當廣泛，早期多應用在設計(Design)、規劃(Planning)、仲裁(Arbitration)、法律推理(Legal Reasoning)、診斷(Diagnosis)、教導(Tutoring)、建築及教學訓練等領域，於近年則更衍伸至電腦系統Help Desk服務與健康科學[33]等領域。根據[36]所整理，如：Bain(1986)發展的JUDGE系統則應用於法案判例；Kolodner與Simpson (1989)發展MEDIATOR系統，化解兩個對立之間的衝突；Koton (1989)發展CHSEY為疾病診斷系統，將病患出現的症狀與以前的案例做比對，來診斷患者的疾病。Paek等學者(1996)將CBR應用於資料庫的設計；Kwong等學者(1997)將CBR應用於射出模型的參數獲取；Shi與Yeh(1999)將CBR系統運用在土地之發展管制等等；Haque等學者(2002)將CBR應用於產品發展同步工程的決策中。表2.1更彙整出自1980年以後所發展出之代表性案例式推理系統[34][9]。



表2.1 案例式推理系統整理

系統名稱	年份	應用領域
CYRUS	1980年	記憶處理模式
MEDIATOR	1985年	調解衝突
JUDGE	1986年	犯罪判例
CHEF	1986年	烹飪
HYPO	1987年	法律
TRUCKER	1988年	排程
PRODIGY	1988年	數學
CHSEY	1988年	醫藥
IVY	1989年	醫藥
SWALE	1989年	案例解釋
SMART	1992年	客戶服務
MODELER	1993年	模具設計
CoolAir	1999年	銷售支援
CLAVIER	1994年	材料製造
FormTool	2005年	塑料上色

資料來源：本研究整理參考資料來源，[34][9]

由此可以發現，隨著各種方法學的演進與電子化的趨勢，案例式推理隨著各種不同知識範疇與特定之設計技巧則能夠有效運用於不同的產業狀況，如醫學、法律、工業、客服等。

2.2 文件資訊探勘

傳統資料探勘技術主要針對資料庫中結構化的資料進行分析、推論，來得到有用的規則或關聯來幫助決策或形成知識。但近年來隨著資訊科技與網路服務的快速發展，使得資訊文件的儲存數量、傳播的速度與使用者取得的方式皆產生劇烈的改變，因而造成資料超載等狀況。因此，現今所面臨的問題並不在如何取得資訊，而是該如何從電子媒體中快速流動、擴張的各種非結構或半結構化文件取得我們想要的或有興趣的目標。

也正因此，文件探勘在近年隨著電子媒體的擴張亦成為一門新興的學問，Dan Sullivan[10]將其定義為「一種編輯、組織及分析大量文件的過程，為了要提供使用者特定的資訊，以及發現某些特徵及其間之關聯」，巫啟台[37]亦定義其為「從非結構性的文字或半結構性文件中發掘出有用或是有趣的片段、模型、方向、趨勢或規則」，其中所使用技術包括資訊檢索(Information Retrieval, IR)、資訊萃取(Information Extraction, IE)、自然語言處理(Natural Language Processing, NLP)、資料探勘等。文件探勘則主要應用這些技術自非結構或半結構化之各類文件或電子資料中挖掘出隱含於內而相對有用的資訊或規則。

文件探勘為針對各類文件、文字進行處理，其整合各種資訊檢索技術，包括關鍵字萃取、文件自動分類、自動摘要等。下面則針對幾個主要之功能進行介紹[11]：

(1) 叢集化(Clustering)：

用來將一個龐大的文件集合切分為數個小叢集，並找出各叢集的主題。即從整個文件集合中，根據文件特徵、相似度進行劃分來形成小叢集，直到叢集內部文件相似程度最大化，而不同叢集間之相似程度最小化為止。

(2) 類別化(Categorization)：

利用文件中擷取之特徵來進行分類作業。和叢集化差別在於分類並非自動產生，而是以預先定義之架構與內存的訓練文件為基礎，透過訓練的方式來改進分類的結果，使之更接近實際目標。

(3) 特徵擷取(Feature Extraction)：

特徵擷取技術則用來辨識文件重要之字彙項目，比如說人名、組織、關係、地點、詞性等。這些關鍵詞彙並非事先定義，而是透過特定之方法自動從文件所得。

(4) 摘要(Summarization)：

將文件中關鍵性的句子找出來，並組成該文件簡化的總結，亦即產生其文件摘要。建立摘要時會根據文件中語句與文件相關性，以及在文件內之位置來進行評比。評比後，根據相關性高低來產生文件摘要。

2.2.1 向量空間模型

向量空間模型(Vector Space Model, VSM)概念最早由 Salton 於 1975 年所提出[12]，主要概念是將文件以向量來表示，不僅可以方便表現出各文件間的關係，亦更容易計算彼此間之相似程度。向量中每一維度代表一個有意義的詞彙，每一維度之屬性值則代表該詞彙在文件中所代表的重要性高低程度，而詞彙加權(Term Weighting)主要目的是為了區分文件間之差異性及反應該詞彙在此文件中的相對重要程度。

如圖 2.3 所示，空間中的三篇文章(D_1 、 D_2 、 D_3)皆由三個不同索引詞彙(T_1 、 T_2 、 T_3)所組成，依其文件詞彙權重不同，在空間中之表示亦不同。

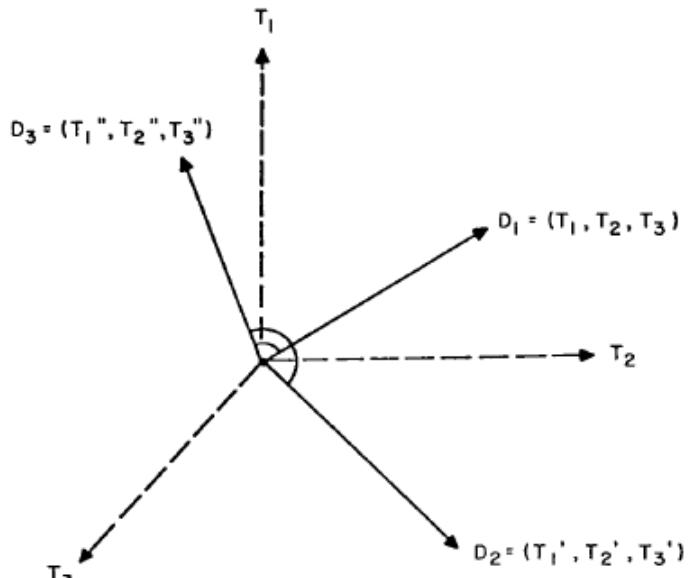


圖 2.3 向量空間模型圖

資料來源：G. Salton[12]

以圖 2.3 例，若延伸至 n 維空間，文件(D_i)則可以表示為 $D_i = (W_{1i}, W_{2i}, \dots, W_{ni})$ 。而一個具有 m 篇文章和 n 個不同關鍵詞彙之空間向量則可以表達如下圖 2.4 所示，其中 W_{ij} 則表示第 i 個關鍵詞彙在第 j 個文件所代表的權重。

$$\begin{bmatrix} W_{11} & W_{21} & \dots & W_{n1} \\ W_{12} & W_{22} & \dots & W_{n2} \\ \dots & \dots & \dots & \dots \\ W_{1m} & W_{2m} & \dots & W_{nm} \end{bmatrix}$$

圖 2.4 向量空間矩陣圖

資料來源：本研究整理

2.2.2 中文斷詞

在進行文件資訊探勘工作前，都必須先進行萃取文章字詞才能進行下一步處理。而中文句子通常是由字(word)與詞(term)所組成，雖可以單一字或合併字詞來表達其意，但大部分仍須包含二個字或以上，才能形成一個有意義的詞。

而所謂之「中文斷詞」則是將中文文件中一連串的「字」轉換成「詞」的方法，每個詞是根據特定之斷詞規則將文件中「一個字」或「多個字」進行組合而成，同一句子依使用之斷詞可能會有不同的結果。

大致來說，中文斷詞方法有：辭庫斷詞法(Dictionary Based Approach)、統計式斷詞法(Statistical Based Approach)與文法剖析法[38]。

(1) 辭庫斷詞法：

主要是利用現有之詞庫，依據法則來比對輸入之文件，進而擷取出文件中有出現在辭庫的字詞，通常該斷詞法所依據的辭庫要有相當的權威性，因辭庫內容是否完備直接影響斷詞結果。除此之外，為確保斷詞正確性，亦必須耗費時間與人力長期進行各類新詞句的蒐集來維護詞庫品質。如中央研究院所提供之斷詞工具[39]則是透過大量語意資料統計所得之法則，並結合中央研究院詞庫小組所維護之中文辭庫所形成之斷詞法。

(2) 統計式斷詞法：

通常有意義的詞彙出現頻率比無意義的詞組來的高，所以透過語意資料統計頻率後，將出現頻率較高之字組是為有意義之詞彙，並將其視為斷詞的依據。該斷詞法優點在於不需搭配詞庫，遇到未蒐集的新詞也能正確斷出；缺點則是容易將頻率較高之無意義字組視為有效詞或忽略頻率較低之有意義詞彙。

(3) 文法剖析法：

在曾元顯[40]研究中提到文法剖析法是利用詞庫比對的方式，剖析出文件中的各詞性片語，再透過一些方法和準則過濾掉不適合的字組，擷取出有意義的詞彙。因其亦須借助事先建立的詞庫，缺點亦同辭庫斷詞法，包括比對速度、需耗費人力維護詞庫等。

由上述介紹之各種斷詞法來看，各類斷詞法優缺點如表 2.2 所示：

表2.2 斷詞法優缺點整理表

	辭庫斷詞法	統計斷詞法
優點	準確率較高。	不需維護辭庫。
缺點	耗費人力維護辭庫。	會斷出頻率高之無意義詞。

資料來源：本研究整理

詞庫式斷詞法優點在於準確率較高，缺點則是要耗費相當人力來維護詞庫；統計式斷詞法優點在於不需建立詞庫可有效降低成本，缺點則是容易將頻率較高之無意義詞斷出，造成錯誤狀況。在提高斷詞準確率與降低維護成本前提下，因中央研究院權威性詞庫能有效協助斷詞準確度，又不需耗費成本來進行相關詞庫之維護工作，所以本研究將應用中央研究院所提供之斷詞服務來進行中文斷詞相關工作。

2.2.3 相似度衡量

在資訊檢索方法研究中，透過相關的特徵擷取工具進行關鍵詞彙選取後，通常必須比對兩文件間之相似度，以做為文件群集或分類之依據。在向量空間模型中，兩文件間相似度的測量通常是計算各文件所建立的特徵向量之餘弦相似度(Cosine Similarity)來判別。首先，透過工具從文件中萃取出關鍵字詞或片語，並形成其代表之特徵向量，而後計算兩文件特徵向量間的餘弦夾角($\cos\theta$)。餘弦相似度公式如公式(2.1)所示：

$$Sim(d_i, d_j) = \frac{\bar{d}_i \bullet \bar{d}_j}{|\bar{d}_i| |\bar{d}_j|} \quad (\text{式 2.1})$$

其中 d_i, d_j 代表兩個文件之特徵向量， $\bar{d}_i \bullet \bar{d}_j$ 則為兩向量間之內積， $|d_i|, |d_j|$ 分別定義為 $|d_i| = \sqrt{\sum a_k^2}$ ， $|d_j| = \sqrt{\sum b_k^2}$ ， a_k, b_k 分別代表兩文件向量內之關鍵字，而 k 則為文件中關鍵字之個數， $k=1, 2, 3, \dots, n$ 。

若兩文件向量間夾角越小，則代表兩份文件之間相似度越高。如當夾角為 0 度時，其 $\cos\theta$ 值為 1，代表兩文件向量重疊，相似度最高。而夾角為 90 度時，則 $\cos\theta$ 為 0，代表兩文件極度不相似。

2.2.4 K 最近鄰居法(K-Nearest Neighbor)

在文件資訊探勘中，文件分類法中比較常使用的方法大致上有中心向量分類(Centroid Vector)、K 最近鄰居法(K-Nearest Neighbor)、決策樹(Decision Tree)等。本研究在此僅針對 K 最近鄰居法進行介紹。

KNN 簡單來說就是套用所謂「物以類聚」的道理，以最接近自己本身的 K 個鄰居之類別來當成自己所歸屬之類別。不利用事先所定義之分類資訊，而是針對每一個測試文件，透過選取相似度最高的 K 個訓練文件，並取得多數歸屬之類別來成為該測試文件之類別。

如圖 2.5 所示，以圓圈中帶叉叉符號圖樣為測試文件，向量空間中分別有紅色類別、藍色類別與綠色類別。若設定 K 值為 5 時，則可以發現有 3 個紅色類別與 2 個綠色類別最靠近，依 KNN 演算法則可得到測試文件為紅色類別。

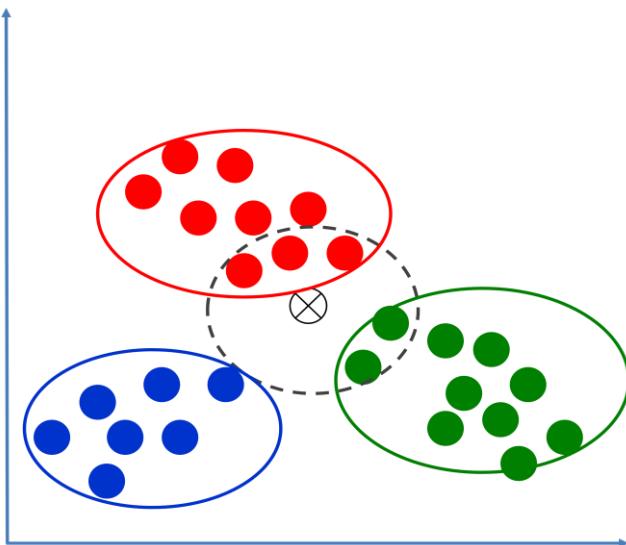


圖 2.5 KNN 分類表示圖

資料來源：本研究整理

KNN 優點在於不需對訓練文件作事前處理，只要向量形成後便可以利用訓練文件進行測試，並進行文件分類。其缺點則因無事先處理，執行效率可能較差，而當訓練文件數量太少時，會影響準確度。最後，K 值的選取也會影響文件分類之正確性。當 K 值太大時，較不敏感，對分類結果影響較低；K 值太小，則較為敏感，導致分類結果會變動較為頻繁。

2.3 社會網路分析

社會網路通常是指獨立個體間所形成的的某種鏈結關係，而獨立個體可以是人、群體或組織。而組成網路的兩個最基本元素分別為節點(node)與連結(tie)，換句話說，即節點與節點之間相互連結所形成的結構，就稱之為網路。在社會網路的研究當中，節點就代表著研究的獨立個體，而連結則用來表示個體與個體或個體與群體之間的某種社交關係，如朋友、會員、員工等。Granovetter[13]進一步描述出人際互動連結的強度，依其強度區分為強連結(strong tie)與弱連結(weak tie)。

(1) 強連結(strong tie)：

通常是指那些跟我們親近、興趣相近且互動頻繁的人，如親人、朋友等。一般來說，在強連結關係下資訊可以快速流通，而個體間之互相影響與信賴程度也最高。但因其個體間之興趣、資源與生活背景重疊性高，較不容易獲得新資訊。

(2) 弱連結(weak tie)：

專指那些認識但關係較疏遠的人，如公司同事、點頭之交。弱連結則能有更廣的社會循環或其於網路上有更多的潛在資源。因此，弱連結更容易接觸到不同的資源，而能替網路帶來較新的資源或創新的想法。

羅家德[41]認為社會網路分析重點在強調節點之間的關係、關係隱含意義以及關係網路結構對現實社會現象的解釋。靜態的社會網路分析，著重分析在什麼樣的關係網路中、什麼樣的結構特性將會如何影響節點的行為。而動態的社會網路分析，是分析社會網路結構的形成與未來發展趨勢，以及結點的行為是如何影響社會網路的架構。

而社會網路的研究大略可分為二個類型，第一種是自我中心社會網路，第二種是整體社會網路[14]。

(1) 自我中心社會網路：

較為強調人際關係間之內涵所帶來的影響。一般稱兩個人有關係，泛指兩個人之間可能在血緣上(例如親子)或者在組織中(例如員工、師生、朋友)有直接的關聯。Bian[14]認為華人的關係可以分成三種：第一種是「間接關係」，是兩個人之間有共同認識的第三者；第二種關係是「直接關係」，是指兩個人之間會有直接的關聯或接觸；第三種是與某個人之間具有強力的、堅固的關係。

(2) 整體社會網路：

著重在網路結構所造成的影響。網路結構的分析則多採用圖形理論的觀念，如整體網路結構的緊密程度、節點在圖形網路中所代表的中心性(centrality)以及中介性(betweenness)等問題。若節點具有較高之中心性，在社會學的研究中，因為累積有較多的社會資本，所以通常伴隨擁有較高的權力、影響力與資源。而中介性則專指節點扮演溝通橋梁之重要程度。若中介性高的節點試圖壟斷、或拒絕協助溝通的時，其他節點之間訊息流通的溝通成本將會因此而提高。

2.3.1 類聚(Homophily)

類聚由 Lazarsfeld and Merton[15]首先提出，係指從社會互動中之頻率及強度觀察，在人際互動中，特質、態度、生活型態同質性高的人，往往傾向會選擇相近似的生活、社會及工作環境，亦即所謂之「物以類聚」。而後 McPherson 等人[15]針對此一現象，亦於研究中驗證此現象。根據 Lazarsfeld and Merton 所提，類聚之類型約略可分為二種：第一種為狀態類聚(status homophily)，若獨立個體間具有相似的特性或背景，有較多的機會可以產生關係；第二種為價值類聚(value homophily)，若獨立個體間有共同的想法、目標或興趣，比較有機會可以產生關係。

在類聚之概念下，則產生選取(selection)與社會影響(social influence)兩種機制[16]。選取：藉由不易變動的特徵，人們通常趨向於與那些跟自己近似性高的人產生關係，進而形成一個小群組；社會影響：人們通常會修改自身的行為或興趣來使自己與其自身之朋友更為一致，或影響朋友，使其行為或興趣產生變化而與自己更為親近。亦即，透過獨特的特徵來驅使獨立個體間產生連結，而透過社會影響力，那些已形成關係的個體會互相影響而至想法、行為或興趣產生變化，讓關係更為密切。而以上所述機制也由 David Crandall 等學者[16]於 2008 年利用 Wikipedia 資料驗證確認，如圖 2.6 所示：

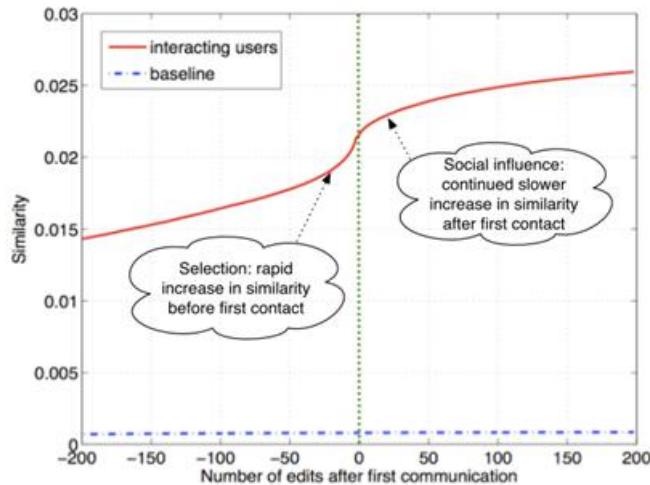


圖 2.6 Wikipedia 作者間平均相似度

資料來源：[16]

由上圖可知，當無關係的兩個作者，隨共筆的數量增加，其相似度亦會隨之增加。

2.3.2 關聯網路(Affiliation Network)

依 Duncan[47]著作中所說，關聯網路本身非常值得探索，不僅因為它是形成其他社會關係(如朋友、商務往來等)的基礎，同時也能廣泛地應用在許多非社交性但又深具社會價值的網路。例如，當於線上網站購買書籍時，其選單上會列出「買這本書的人也買了…」等推薦資訊，而這就是一個關聯網路。藉由買書的行為，一個人將和其他買同樣書籍的人產生關聯，就好像加入一新的群組。由此可知，關聯網路永遠是由兩種型態的節點所形成，通常稱之為演員(actors)和群組(groups)，而該類網路最好的表示方法可稱之為二分(bipartite)或二元(two-mode)網路。如圖 2.7 所示，在網路中，兩組結點被分開來，只有不同型態之結點才能相連，其相對應的關係可以說是隸屬(belonging to)或者選擇(choosing)。所以演員只能連向群組，群組亦只能連向演員。其中若兩個演員共同隸屬於同一個群組，則他們之間就會產生關聯，進而形成演員關聯網路。群組之間亦會因擁有共同成員而產生關聯，通常若至少有一位演員同時分屬兩個群組，則兩個群組相互交疊(overlap)或連動(interlock)，進而形成群組交疊網路。因此，一個關聯網路通常包含了所有關於演員關聯網路及群組交疊網路資訊。

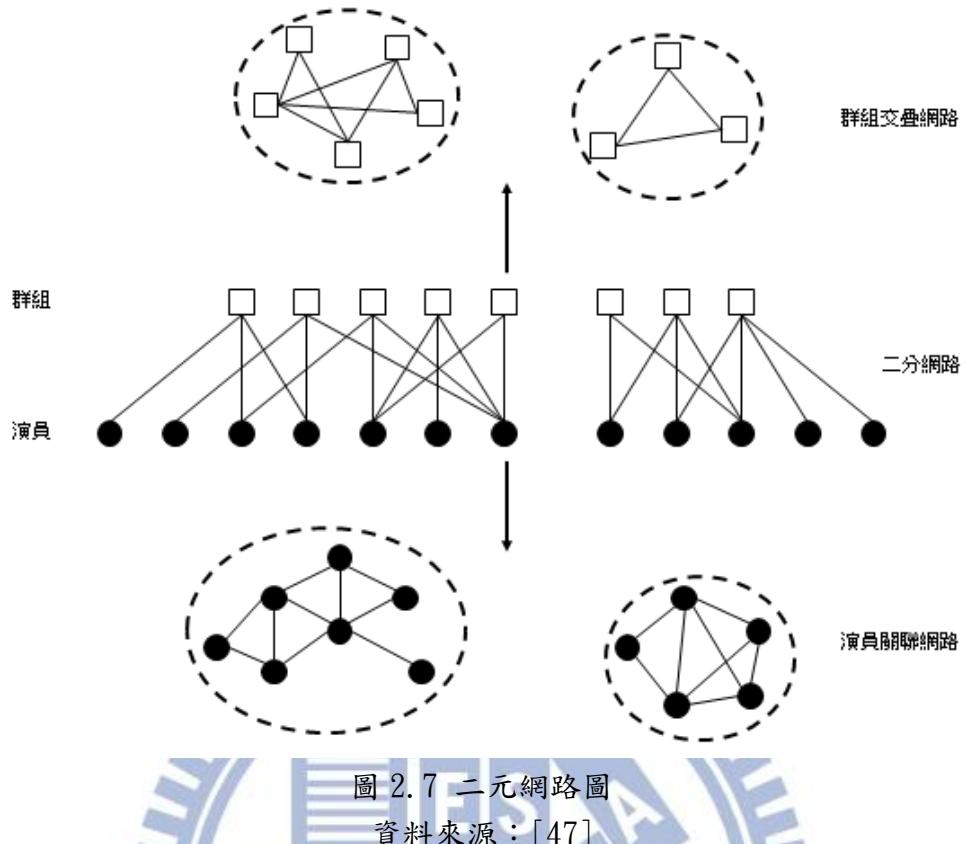


圖 2.7 二元網路圖

資料來源：[47]

而在 David Easley [16] 文中則更加簡略闡述出，所謂關聯網路是由一特定集合的與其所針對之特定社交關注點(focal point of social interaction, foci)所形成的網路，換句話說，亦即由那些參與共同目標或特定社交活動的人們所形成的集合。如圖 2.8 所示：

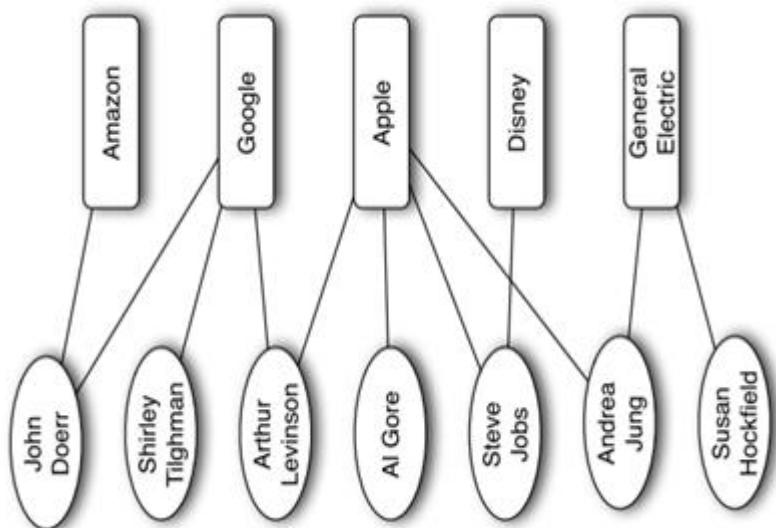


圖 2.8 關聯網路範例圖

資料來源：[16]

在關聯網路中，節點(node)可以是個人或品牌，更或者是其他有意義的物件，如文章、俱樂部或讀書會等。而連結(edge)則表示節點間之關係，如友誼關係(Friendship)、會員關係(Membership)或員工關係。

而依其不同結構與機制所產生之連結都可以被視為形成封閉域(Closure)之流程，亦即人們會因不同機制產生連結，並進而形成一個集合。依[16]所整理，依其形成方式可分為：三元封閉域(Triadic Closure)、聚焦封閉域(Focal Closure)與成員封閉域(Membership Closure)。

(1) 三元封閉域(triadic closure)：

這是一種人與人之間的行為現象。當某兩個完全不認識的人之間有一位共同的朋友，那這兩個人會認識的機率就會提高。如圖 2.9 所示：

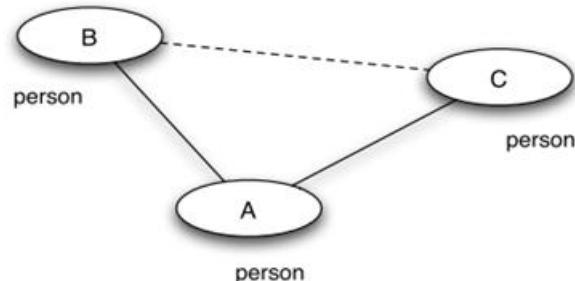


圖 2.9 三元封閉域

資料來源：[16]

B 與 C 都有 A 這個朋友，那麼透過 A 的介紹或互動機會，則 B 與 C 容易會變成朋友。

(2) 聚焦封閉域(focal closure)：

當某兩個完全不認識的人若共同對某件事物或組織有興趣，透過該平台互動，那這兩個人會認識的機率就會提高。如圖 2.10 所示：

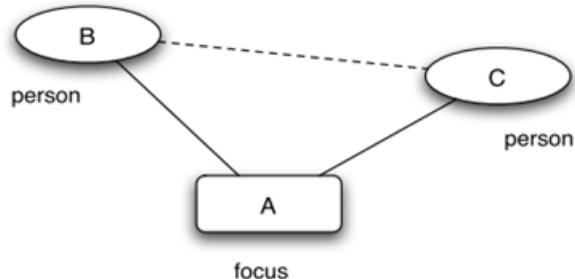


圖 2.10 聚焦封閉域

資料來源：[16]

B 與 C 都參加 A 線上虛擬社群，那麼透過平台間回文或參加所舉辦活動之機會，則 B 與 C 容易會變成朋友。

(3) 成員封閉域(membership closure)：

當某個人對某件某件事物或組織並不熟悉，若有一關係較密切之朋友洽有參與該組織，那麼透過朋友的介紹，這個人與該事物或組織產生關係的機率就會提高。如圖 2.11 所示：

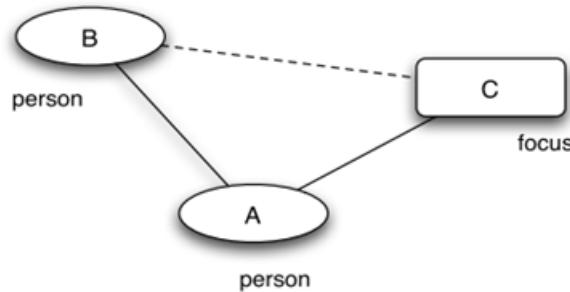


圖 2.11 成員封閉域

資料來源：[16]

A 參加 C 線上虛擬社群，而 B 與 A 是朋友，那麼透過朋友 A 的影響與介紹，則 B 便容易加入 C 線上虛擬社群。

2.3.3 社會網路分析指標

衡量社會網路的分析指標大致可分為三種：社會網路規模(size)、社會網路密度(density)、網路中心性(centrality) [17][18]。社會網路規模是指特定社會網路中所有節點的數量，社會網路的規模是影響節點間關係的重要因素，因為節點彼此間關係的建立會受到資源多寡的限制，若是節點越多表示可用的資源可能也越多。社會網路密度是用來表示節點間的關係是否緊密或是該社會網路中節點之間的連結程度，通常密度越高代表節點與節點之間的關係越緊密。

網路中心性則可用來衡量個人在網路中影響力的大小，而一般而言分析網路中心性指標可分為三種：程度中心性(degree centrality)、親近中心性(closeness centrality)、中介中心性(betweenness centrality) [19]。根據 Mitchell, J. C. [31] 與 Scott, J. [32]所述，程度中心性是利用相鄰的節點進行計算而未考慮間接節點的關係，找出來的重要節點比較屬於區域性。而親近中心性與中介中心性會計算網路中每一個節點與節點之間的相關距離，找出來的重要節點比較屬於全域性。因本研究著重於利用案例及作者來建構出局部網路來進行量測，因此將利用程度中心性來找出網路中較具影響力的節點。以下則針對程度中心性之計算做相關的說明，其餘僅進行說明：

(1) 程度中心性(degree centrality)：

利用相鄰節點的數量來衡量社會網路的區域中心性，節點程度中心性值越高表示這個節點的影響力可能越大，就有可能成為這個網路中的代表人物。如圖2.12所示，節點4因連入關係最多，據程度中心性之定義可知其值最高，因此節點4便為此網路之中心節點。

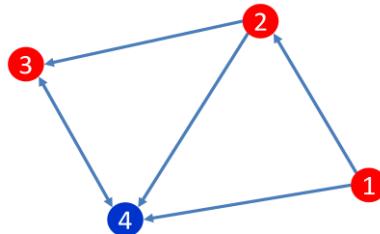


圖 2.12 程度中心性示意圖

資料來源：本研究整理

(2) 親近中心性(closeness centrality)：

主要是測量節點與其他節點的接近緊密程度，接近中心性值越高表示該節點影響其他節點的速度較快並強烈。

(3) 中介中心性(betweenness centrality)：

衡量某一點存在於其他任兩點路徑上的重要程度。當中介中心性的值越高時，表示該節點是位於溝通與橋梁的地位，也代表其在網路中是個重要的節點。

針對網路中心性各類指標計算公式，如表 2.3 所示。

表2.3 網路中心性指標計算公式表

網路中心性指標	計算公式	參數	
程度中心性	$C_D(n_i) = \frac{d(n_i)}{g-1}$	n_i	: 欲計算之節點
		$d(n_i)$: 節點於其他節點關係量
		g	: 網路所有節點量
親近中心性	$C_c(n_i) = \frac{g-1}{\sum_{j=1}^g d(n_i, n_j)}$	n_i	: 欲計算之節點
		$d(n_i, n_j)$: 兩節點間距離
		g	: 網路所有節點量
中介中心性	$C_s(n_i) = \frac{\sum_{j \neq i} g_{jk}(n_i)}{(g-1)(g-2)}$	$g_{jk}(n_i)$: 通過 i 連結 j 及 k 最短路徑數
		g_{jk}	: 連結 j 及 k 最短路徑數
		g	: 網路所有節點量

資料來源：本研究整理

2.3.4 總結

由前述之網路分析、形成機制與分析指標之描述，可以發現在社會網路中若藉由特定之特徵、興趣所形成的網路屬於一種強連結型態的封閉域，而在網路內資訊流通非常迅速、成員間亦會互相影響。因此，若能在網路中找到最具影響力之節點，則代表其擁有最多的資產，亦更能提供更多的知識。

因此，專家推薦機制不僅需針對問題相似度來進行評分，更應該透過專家間的關係來找出專家網路中最富影響力、擁有資源最多的人。若能綜合此兩方面來進行評比，那麼即能針對問題推薦出該領域之最貼近現況、擁有最多資源的專家來提供協助。

2.4 專家找尋(Expert Finding)

近幾年來，專家找尋(Expert Finding)成為一個新興的課題，有許多學者亦針對該領域付出相當心力並提出各種不同的方法，並進而協助使用者或企業能從內部網路與線上論壇找到特定主題之專家。

2.4.1 專家找尋途徑

在最早期的想法是透過資料庫內存之專家資訊(Expert Profile)來進行專家找尋工作[20]，但問題在於此方法必須耗費相當之人力與成本來進行專家資訊的維護工作。而相關研究指出即使利用資源來定期更新專家資訊，並無法提供有效並正確的資訊，原因在於被賦予更新專家資訊工作的人並無法清楚了解並定義專家的知識，而專家本身亦過於忙碌無法時常更新自身資訊。因此，手動維護專家資訊以提供專家找尋方法使用並不是一個可行的策略。

為降低維護成本並提供正確、有效之搜尋結果，隨著社會網路分析的發展與研究，進而發展出兩類基於社會網路分析的專家找尋方法，其一為個人化分析(Personal Profiling)；二為文件化分析(Document Profiling)[21]。

(1) 個人化分析(Personal Profiling)：

透過各種包含專家描述的文件來擷取專家相關資訊，並進而形成專家個人描述檔。當個人描述檔完備後，針對所輸入之特定關鍵字，搜尋引擎則其查詢領域與專家描述進行比對，並輸出相關之專家列表。流程如圖 2.13 所示：

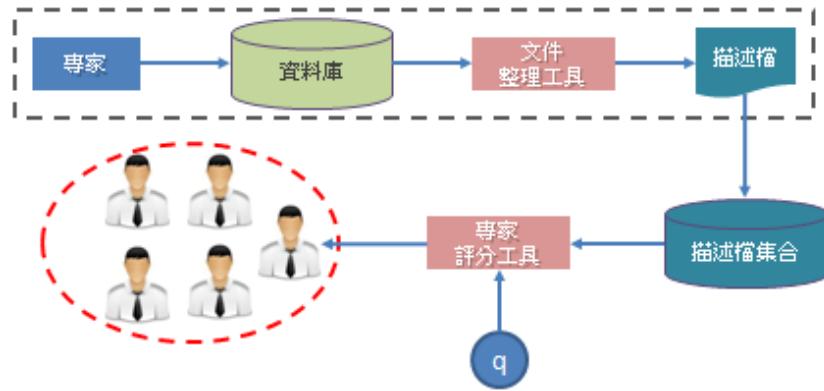


圖 2.13 個人化分析示意圖

資料來源：本研究整理

如 Nick Craswell 等學者[22]所提之演算法則屬個人化分析的一種，其方式是將專家所發表過之文章進行收集，並將相關資訊整理後產生專家個人所屬之檔案，而後當輸入查詢關鍵字後，系統會以關鍵字與專家個人檔案進行比對並進行分數評比後輸出專家清單以供使用。

(2) 文件化分析(Document Profiling)：

針對所輸入之特定關鍵字，查詢資料庫所有文件並取得與查詢相關之文件。而後根據過這些相關文件之內容萃取出可能之專家列表。大致如圖 2.14 所示：

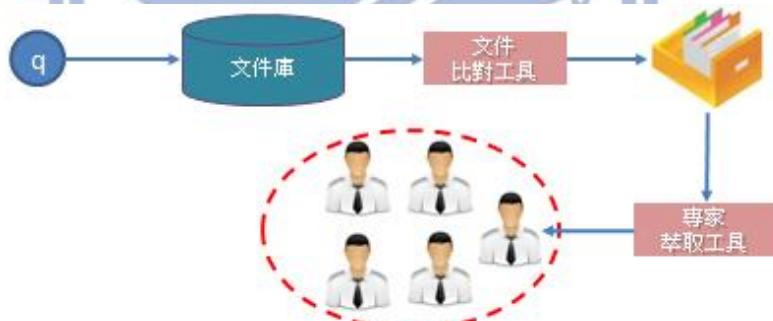


圖 2.14 文件化分析示意圖

資料來源：本研究整理

如 Christopher 等人[23]所提出之方法即是屬於文件化分析，其方式是從企業郵件資料庫取得於輸入查詢條件相關之所有電子郵件列表，而後系統從電子郵件萃取出所有人員列表並依其關係建立依網路圖，待網路圖建立後再利用演算法對所有專家進行評分，最後輸出依評分高低排序之專家列表。

針對上述之個人化分析(Personal Profiling)與文件化分析(Document Profiling)架構，本研究根據[21]內容整理其相關優缺點如表 2.4 所示。

表2.4 專家找尋方法優缺點比較

	優點	缺點
個人化分析 (Personal Profiling)	1.不須維護專家資料。 2.離線作業。 3.運作較有效率。	1.須持續收集專家資訊。
文件化分析 (Document Profiling)	1.不須維護專家資料。 2.準確定較高。	1.耗費時間較多。

資料來源：本研究整理

2.4.2 專家評分方式

然無論使用何種找尋專家的方法學，都必須對專家清單中所有專家進行評分，以檢查其相關程度高低，而能提供或建議相關性最高之專家予企業或發問者使用。依[42]研究所整理，專家評分大部分以機率運算為基礎，如以專家出現之機率來進行評分。運用其出現在相關文件中的頻率來進行計算分數，其結果會因文件數量有所限制。而後為改善此狀況，學者們開始使用以圖論為基礎之評分方式來進行專家評分與定義專家等級，如 PageRank、HITS 等方法。在另一方面，有一些學者也開始利用社會網路分析的方式來進行專家找尋與評分，他們認為獨立個體間的知識轉換並非都是以正規的組織架構或企業來流程進行，通常是以一種人與人之間的非正式的溝通來進行，所以使用社會網路分析對於評比專家重要程度也是一個非常有效的工具。

2.4.3 專家找尋機制

綜合 2.4.1 與 2.4.2 節之討論，可以發現針對專家找尋此一課題，已發展出以統計為基礎或以探討網路關係為主之相關找尋方式。本研究主要目的在協助管理人員進行資訊系統問題之診斷，為提高專家推薦之正確度並希望能參考專家個人影響力以為重要項目，本研究將以文件化分析類型之專家找尋機制為主，並參考相關研究加入社會網路分析之要素來進行專家評分，並期待能產生更好的推薦效果進而能有效協助企業解決系統異常問題。

再者，根據 Marco van der 等人[30]提出的理論，認為在擁有相同特性或共同作者的小型社會網路中，具有強連結的節點較為重要。因此本研究將以 Ji-Meng Chen 等人[24]所提出之專家找尋方法為基礎提出依改良之專家找尋模型與案例式推理結合來支援資訊系統問題診斷。根據[24]研究，認為利用查詢條件從文件庫找出相關信高的文件後，若能從文件中萃取出現在文件中之專家則可形成列表使用，再深入探究專家間之共同出現(co-occurrence)之關係則能建置成一個小型網路並利用社會網路分析方法找出

最具影響力之專家進行推薦。概念圖則如圖 2.15 所示：

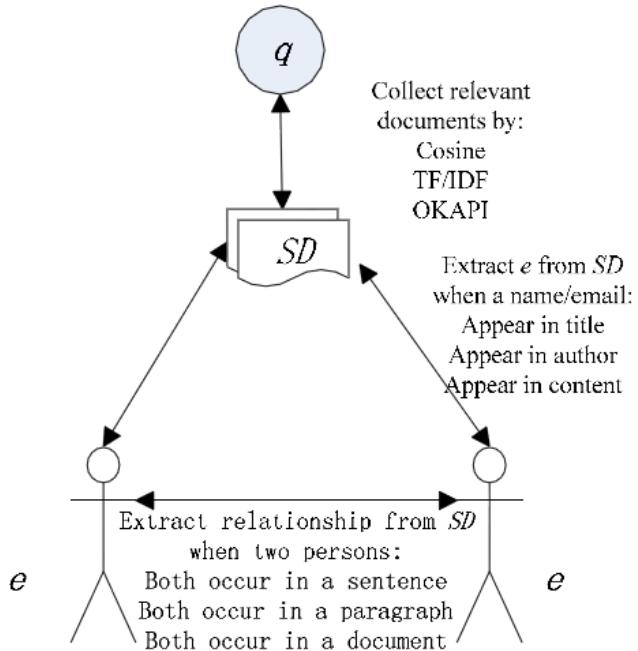


圖 2.15 問題與專家關係圖

資料來源：[24]

根據 2.3.2 節說明，上圖兩專家皆共同出現於支援案例文件，以其型態可以發現隨著接觸機會增加，專家間會逐步形成一個聚焦封閉域(focal closure)，透過其間之強連結(strong tie)作用而能互相交流或影響。若再將其簡圖再擴展來看，如圖 2.16 所示：

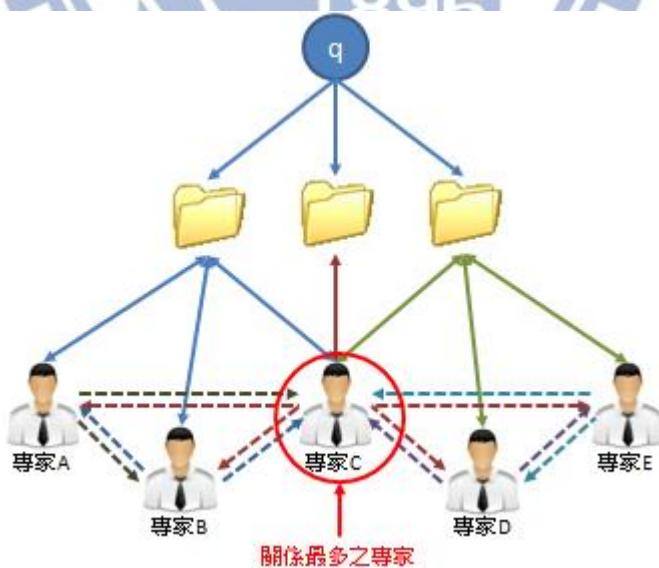


圖 2.16 問題與專家網路圖

資料來源：本研究整理

可以發現當利用問題查詢取得相關案例集合後，出現在各文件之專家群透過其共同撰寫之案例之關係來形成一個小型的社會網路。依 2.3.3 節所述，在該網路中找出關係最多的專家，那麼此專家亦會是這一群專家中最具影響力的專家。

針對以上想法，參考 Ji-Meng Chen 所提出之專家評分模型並結合程度中心性指標所提出提出之改良型專家找尋模型如公式(2.3)所示：

$$S_{n+1}(e, q) = \gamma C_D(e) S_n(e, q) + (1 - \gamma) \sum_{(e, e') \in E} W_{e, e'} \cdot S_n(e', q) \quad (\text{式 2.3})$$

$S_n(e, q)$ 即為專家 e 於第 n 計算所得之分數，而專家初始分數則以頻率計算為基礎架構來評分，如公式(2.4)所示：

$$S_1(e, q) = \frac{pf^2}{tf} \sum_{d \in SD} p(d | q) p(e | d, q) \quad (\text{式 2.4})$$

而 $W_{e, e'}$ 是指兩專家之間的關係權重，計算方式如公式(2.5)所示：

$$W_{i,j} = \frac{cr(e_i, e_j)}{|E_i| \cdot |E_j|} \quad (\text{式 2.5})$$

以上計算公式對應之參數說明，如表 2.5 所示。

表2.5 改良型TOP-K專家找尋公式參數整理表

	計算公式	參數
專家初始分數	$S_1(e, q) = \frac{pf^2}{tf} \sum_{d \in SD} p(d q) p(e d, q)$	pf : 出現在該文件次數 tf : 出現在文件集合總次數 SD : 支援文件集合 $p(d q)$: 文件與案例相似度 $p(e d, q)$: 專家出現在該文件機率 $W_{e, e'}$: 專家關係權重
關係權重	$W_{i,j} = \frac{cr(e_i, e_j)}{ E_i \cdot E_j }$	$cr(e_i, e_j)$: 兩專家共同出現文件數和 $ E_i , E_j $: 與多少個專家有關係和 n : 計算次數
專家總合評分	$S_{n+1}(e, q) = \gamma C_D(e) S_n(e, q) + (1 - \gamma) \sum_{(e, e') \in E} W_{e, e'} \cdot S_n(e', q)$	e, e' : 表示兩個不同專家 γ : 表示專家影響因子 E : 表示專家清單

資料來源：本研究參照[24]整理

可以發現研究[24]透過頻率計算方式找出與問題相關性最大的專家外，亦同時利用社會網路分析來評比專家間關係權重並將之加總來代表專家影響力，那麼就能從專家群中將最貼近問題、擁有關係最多、影響力最大的專家找出來。



第三章 系統架構與設計

本章節將首先介紹半導體製程及製造執行系統範疇，緊接著針對某企業內部自動化部門解決系統問題之作業流程進行探討，而後再依據其流程與需求進行離型系統規劃與設計。

3.1 背景說明與系統架構

半導體是一種電性介於導體與非導體之間的材料，廣泛應用於積體電路(Integrated Circuit)、太陽能電池等領域。其中積體電路即是將電子元件及線路製作在非常小面積的一種電子產品，例如微處理器或記憶體，各項製程主要有薄膜沉積(Thin Film Deposition)、微影(Photolithography)、蝕刻(Etching)與化學機械研磨(Chemical Mechanical Polish)等[43]。因其製程繁複、生產設定資訊各異，為有效控制其生產流程與記錄各製程量測數據，半導體公司通常會導入電腦整合製造(CIM)技術來進行控制，其中製造執行系統位處邏輯核心，其地位更顯重要。

3.1.1 半導體製程

針對薄膜沉積(Thin Film Deposition)、微影(Photolithography)、蝕刻(Etching)與化學機械研磨(Chemical Mechanical Polish)等製程說明如下：

(1) 薄膜沉積(Thin Film Deposition)：

薄膜沉積的步驟包括長晶、晶粒成長、晶粒聚結、縫道填補與沉積膜成長，薄膜沉積的方式大致有二種，分別是物理氣相沉積(Physical Vapor Deposition, PVD)與化學氣相沉積(Chemical Vapor Deposition, CVD)，大致可分為常壓、低壓及電漿三種方式。目的皆在於讓晶圓表面形成薄膜，其耐熱、耐蝕特性來有效避免其他製程造成之損害。

(2) 微影(Photolithography)：

使用半聚合(Semi-coherent)光照射以玻璃為主體的光罩，將光罩上的線路圖案投射至晶圓上的感光材料，其主要製程包含光阻覆蓋、曝光及顯影。目的在於將電子線路圖像生成在經元上以利後續製程使用。

(3) 蝕刻(Etching)：

目的在於去除晶圓上未被光阻覆蓋的薄膜，以保留自光罩轉移的電子線路圖案。通常可分為濕式蝕刻(Wet Etching)與乾式蝕刻(Dry Etching)。

(4) 摻雜(Doping)：

通常使用擴散法(Diffusion)與離子植入法(Ion implantation)等方式，將其物質摻雜於晶圓本體以調整其濃度與導電能力。

(5) 化學機械研磨(Chemical Mechanical Polish)：

積體電路中層層堆疊的結構，容易讓晶圓表面變得不平坦導致製程品質下降，利用佈滿顆粒的研磨墊和化學助劑(Reagent)使晶圓表面平坦化(Planarization)。

3.1.2 製造執行系統

製造執行系統是一從接到訂單從事生產開始到產品完成，將產品在生產線上生產的即時資訊提供給使用者監看，並將工廠生產的即時資料回應於報表上，呈現的結果會隨著狀況的不同迅速改變，其目的在於控制生產流程、降低人員誤操作等，並提高生產效率。以 IBM 所提供之製造執行系統(SiView)為例[25]，其架構如圖 3.1 所示：

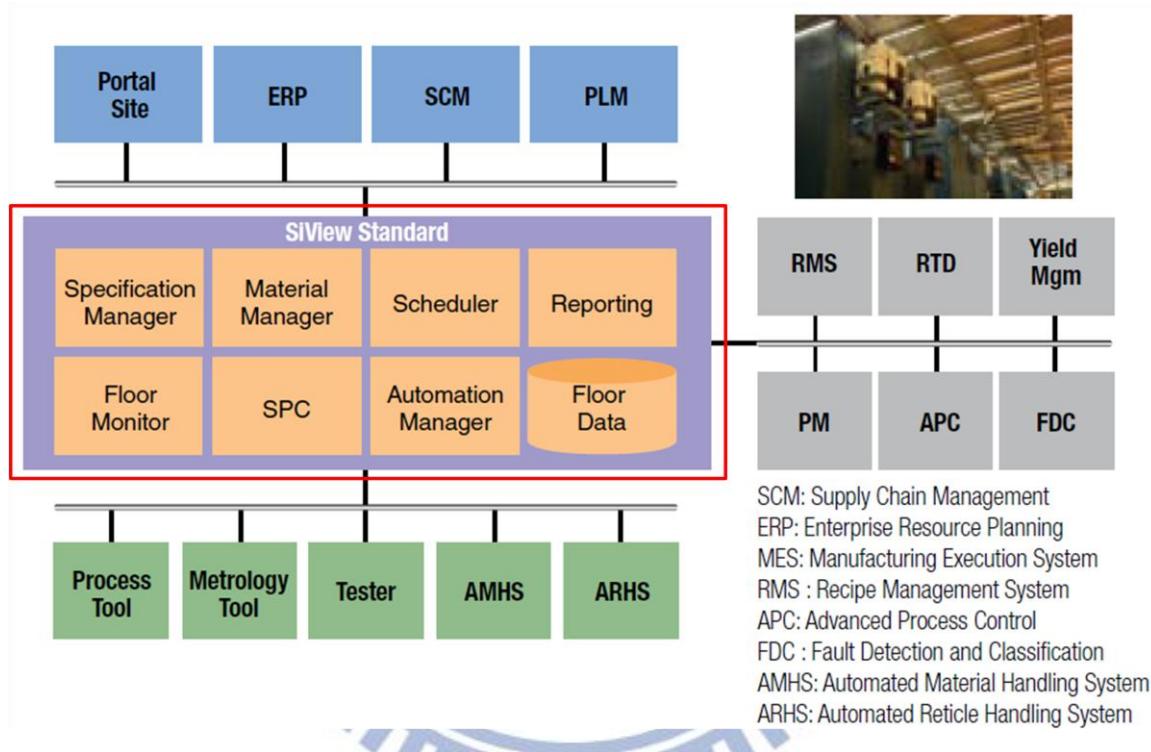


圖 3.1 SiView 架構圖

資料來源：[25]

可以發現其位於上游(如 EPR、SCM)與下游(如 AMHS)之間，扮演一個重要的溝通橋梁與核心角色，並提供即時生產資訊予後端先進製程控制(APC)或資訊報表系統。因此，本研究將針對製造執行系統問題處理流程與需求進行設計一離型系統。

3.1.3 作業流程

企業針對與營運有直接相關之資訊系統皆會安排值班人員進行異常排除與解答使用者問題。半導體亦不例外，針對自動化生產相關系統來說亦會安排值班人員 24 小時於線上協助使用者排除遇到之異常狀況。一般來說，使用者在進行產品設定、生產參數調整或機台自動上貨等流程產生異常時，線上使用者會透過電話聯絡自動化各系統相關

人員協助處理。此時，處理人員會針對線上反應之異常狀況來判定可能發生原因並加以處理，若順利解決問題則將處理資訊紀錄以備存；反之，若無法處理則將問題反映給專家(系統負責人或資深工程師)進行處理，問題解決後亦將其處理資訊紀錄備存；若仍無法解決則可能問題是由二個或以上系統交互作用引起，需進行研究找出影響系統後再委請不同領域之專家共同解決之。以上處理步驟則如圖 3.2 所示：

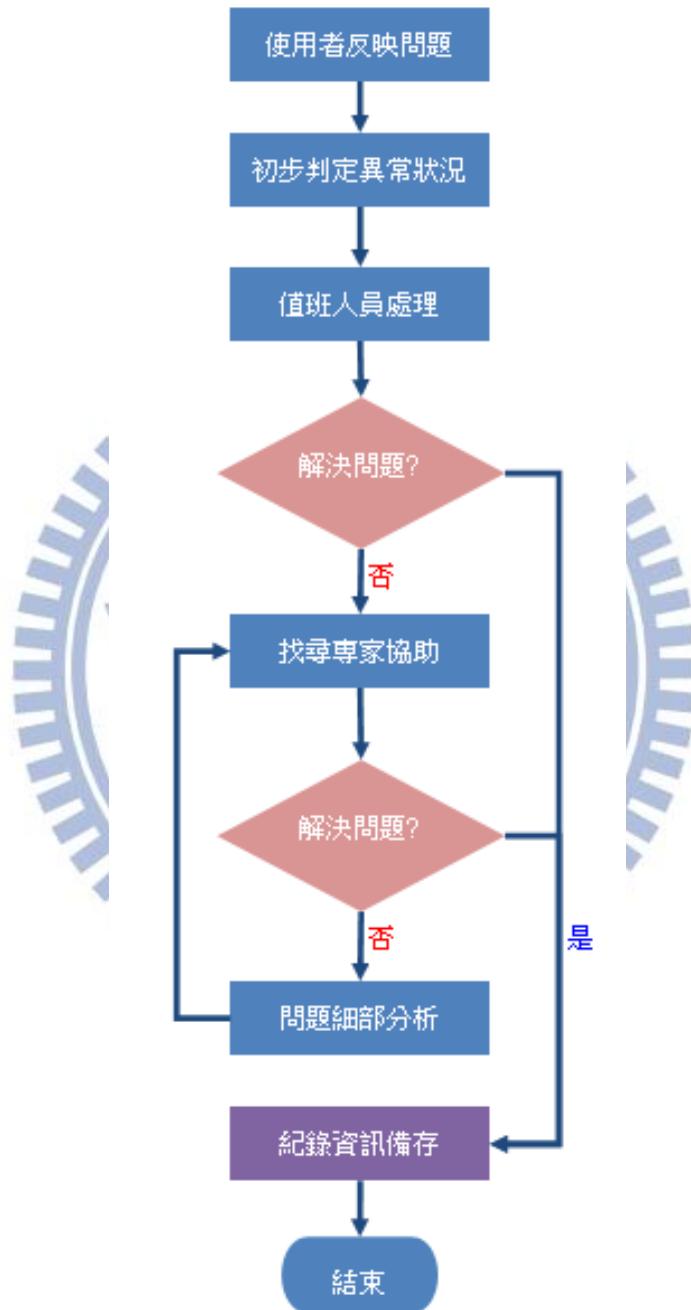


圖 3.2 企業現行問題解決流程圖

資料來源：本研究整理

由以上流程可知在企業現行作業流程中所衍生之問題，包括(1)問題判斷正確度與處理人員經驗具極大相關性，誤判將造成處理時間增加與人力量浪費。(2)類似問題將因不同處理人員而有處理流程與觀點之差異。(3)問題描述與處理資訊並無統籌管理，造成

資料過於分散，缺乏效率和彈性。(4)系統專家離職造成相關知識流失。

針對可能發生之問題，本研究結合可用之方法學提出改善現行流程架構，並實作離型離型系統和原先人工作業與問題解決作驗證比較。離型系統之規格則依實際訪談後整理出相關需求如下：

一般使用者：

- (1) 簡單的操作介面。
- (2) 問題能獲得快速、正確之回應。

處理人員：

- (1) 提供系統化的知識庫。
- (2) 搜尋問題之近似案例能更加有效率。
- (3) 容易進行處理資訊維護工作。

系統管理員：

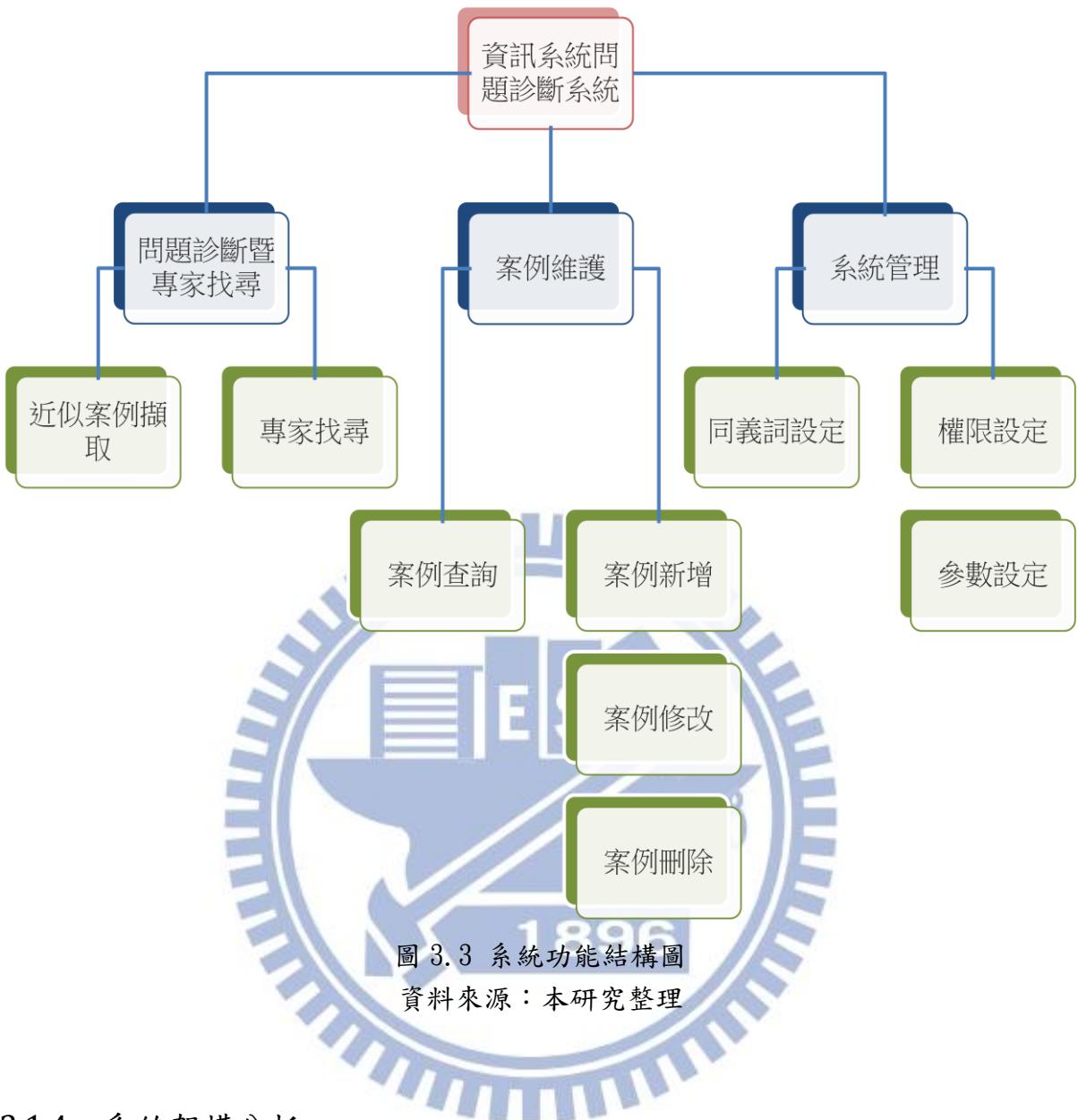
- (1) 需有使用者權限控管。
- (2) 能夠以簡易介面進行系統設定。

為符合上列需求，本研究將系統規畫成三大模組，分別為問題診斷暨專家找尋模組、案例維護模組及系統管理模組。

問題診斷暨專家找尋模組主要使用者為一般使用者，主要是將問題描述資料整理並擷取其關鍵字後，將之轉換為空間向量與案例庫中的案例進行相似度計算，取得相似度較高的前幾筆案例。在完成案例擷取後，則據其撰寫者資訊形成專家列表並行評分。最後，將支援案例資訊與推薦之專家提供予使用者讓其按照相關處理資訊進行問題排除。

案例維護模組主要使用者為處理人員與系統管理員，根據需求提供一個較為結構化之介面提供人員進行問題描述與處理資訊輸入工作，並提供案例庫搜尋功能以利進行相關維護工作，如新增、修改、刪除等。

系統管理模組主要使用者則為系統管理員，主要提供進行個人或部門之權限設定，系統相關參數設定與同義詞維護。同義詞維護則主要在輔助案例進行相似度比對時之依據，在於使系統能更貼近各系統專業名詞而能有效提升案例判斷之準確度。系統功能結構如圖 3.3 所示：



3.1.4 系統架構分析

本研究之離型系統為便於企業使用，採行 WEB 化人機介面。首先，必須經過系統驗證機制以取得使用者資訊與其使用身分。進入系統後，根據使用者角色給予不同使用介面，如一般使用者只能使用問題診斷暨專家找尋模組；處理人員除一般使用者權限外，可以再使用案例模護模組；系統管理原則可以使用包含系統管理模組在內之所有功能。系統架構圖如圖 3.4 所示。

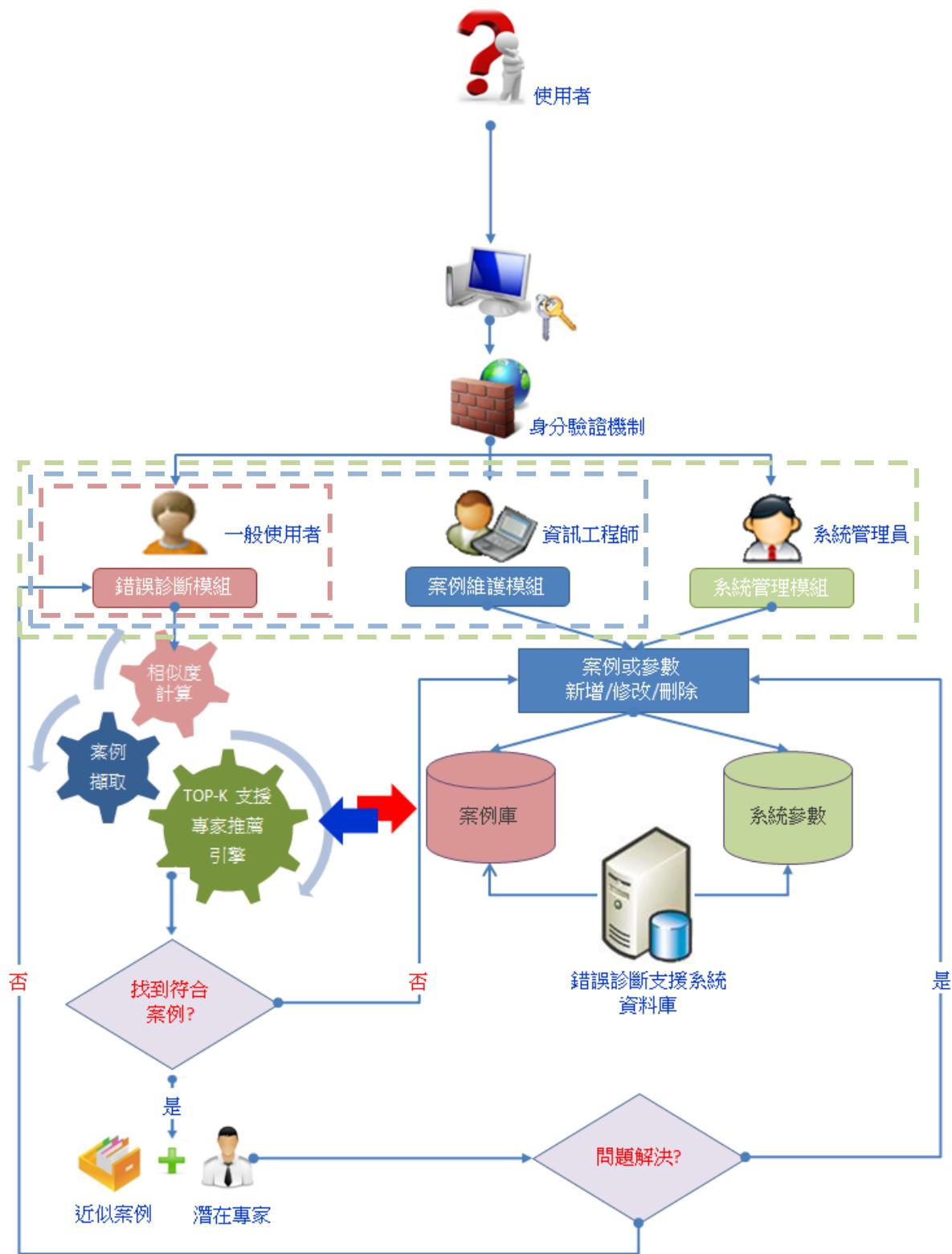


圖 3.4 系統架構圖

資料來源：本研究整理

3.2 問題診斷暨專家找尋模組

問題診斷暨專家找尋模組，包括兩大主要功能，分別為案例擷取與專家找尋。主要在利用問題相關描述，透過案例推理過程來找出相似度較高之前 K 個案例(TOP-K)，並藉由案例撰寫或修改清單取得專家列表後，進行評分。最後，系統輸出可參考之案例與可協助解答問題之專家資訊予處理人員參考。詳細步驟如圖 3.5 所示：

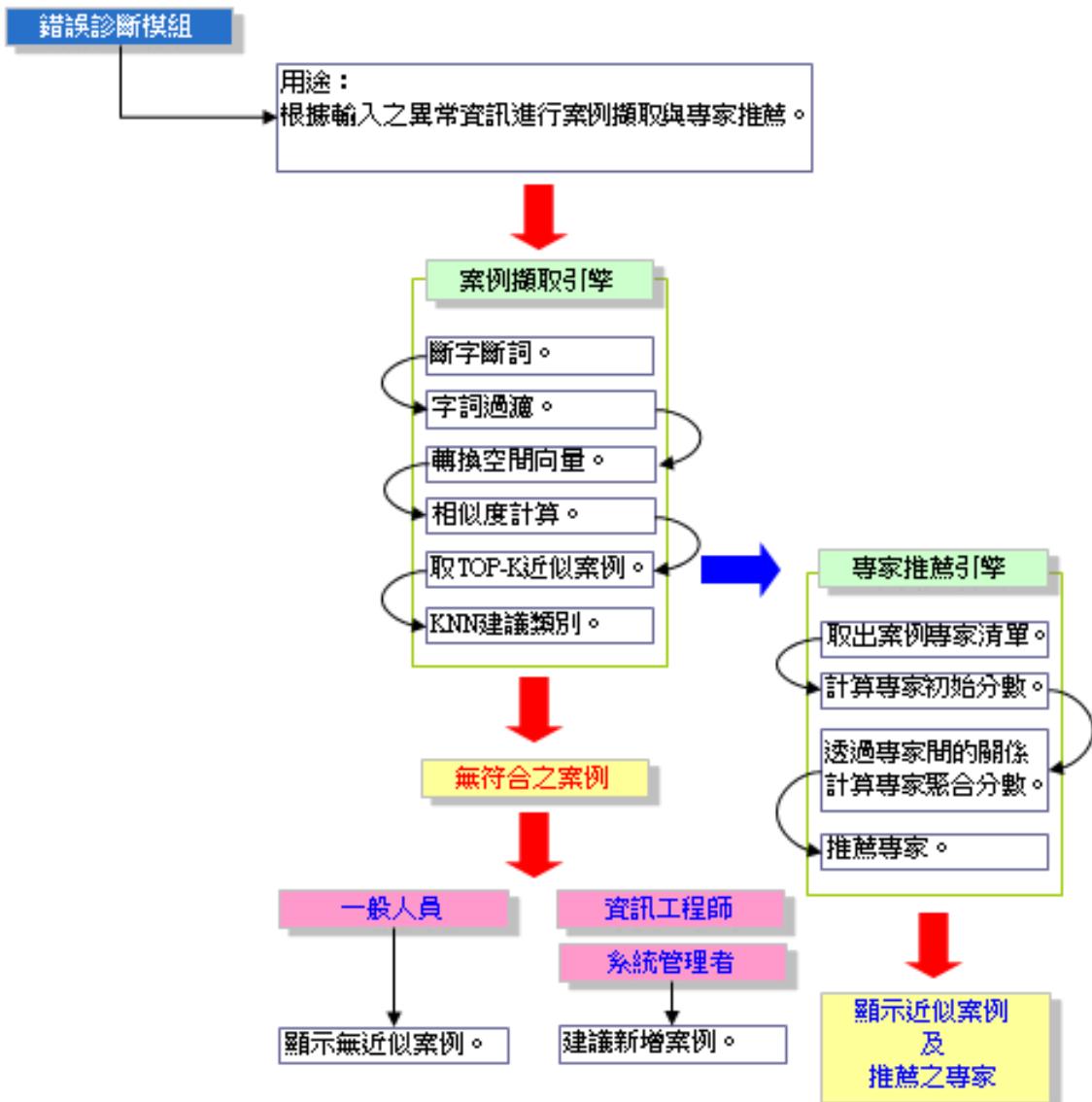


圖 3.5 問題診斷暨專家推薦模組流程圖

資料來源：本研究整理

以下將針對各步驟細項與實作方法進行解釋。

(1) 斷字斷詞：

由於本研究處理之案例資料以中文為主，因此必須將中文敘述加以斷詞即進行詞性分析。所以本研究依循中研院資訊所 CKIP 小組公布之斷詞服務原則與實作方式實作斷詞工具，透過此工具便能將中文敘述句有效轉換為帶有詞性的分詞序列句。如「註冊 SEO 後，一直沒上機台進行更換 FOUP 動作」，經由斷字斷詞後顯示結果為「註冊(VA) SEO(FW) 後(Ng)」，(COMMACATEGORY) 一直(D) 沒(D) 上(VCL) 機台(Na) 進行(VC) 更換(VC) FOUP(FW) 動作(Na)」，其中「註冊」、「SEO」、「後」、「一直」等皆為斷詞後所抽取之字詞，而每一個字詞後面英文簡碼則為該詞之詞性。各類詞性對照表請參考附錄一。

(2) 字詞過濾：

由附錄一可以知道斷詞工具所標記的詞性分類相當細緻。在此階段為了避免過多詞性造成特徵向量維度與複雜度增加，經由問題分析後決定僅先設定以名詞、動詞及專有名詞之英文字作為關鍵字依據。其中詞性過濾參數設定於系統管理模組之參數設定，可根據使用狀況來予以修改符合所需，以期斷詞結果更能有效率的擷取出較符合資訊系統實際異常狀況之描述。例如，「註冊 SEO 後，一直沒上機台進行更換 FOUP 動作」經由字詞過濾步驟後所得到之關鍵字為「註冊(VA) SEO(FW) 機台(Na) 進行(VC) 更換(VC) FOUP(FW) 動作(Na)」等共七個關鍵字詞。

(3) 轉換空間向量：

將案例庫已存在之案例進行斷詞，並收集過濾後之關鍵字詞以為特徵向量之維度。各案例則依維度依序檢查並計算其維度值，待完畢後該案例即可轉換為一獨特之案例特徵向量。

(4) 相似度計算：

已存在案例完成關鍵詞擷取並轉換為特徵向量後，新案例找尋近似案例所利用之方法為餘弦相似度(Cosine Similarity)。其方式是透過兩向量間之夾角或距離來進行相似度計算。餘弦相似度公式如公式(3.1)所示：

$$Sim(d_i, d_j) = \frac{\vec{d}_i \bullet \vec{d}_j}{|\vec{d}_i| |\vec{d}_j|} \quad (\text{式 3.1})$$

若現有兩案例分別為 A,B，則透過關鍵詞轉為特徵向量。那麼根據餘弦相似度公式，

$$Sim(A, B) = \frac{\sum a_i b_i}{\sqrt{\sum a_i^2} \sqrt{\sum b_i^2}} \text{，其中 } \vec{A} = [a_1, a_2, \dots, a_n], \vec{B} = [b_1, b_2, \dots, b_n] \text{。}$$

針對計算方法，本研究舉例說明如下：

案例一(C1)：類別為「OPI」，功能「SEO」，錯誤訊息「*」，錯誤碼「*」，異常描述「註冊 SEO 後，一直沒上機台進行更換 FOUP 動作」。

案例二(C2)：類別為「OPI」，功能「SEO」，錯誤訊息「*」，錯誤碼「*」，異常描述「FOUP 無法註冊 SEO 以進行 SPLIT 動作」。根據斷字斷詞、字詞過濾步驟整理案例

一及案例二結果如表 3.1 與表 3.2 所示：

表3.1 案例一整理資訊

案例一(C1)			
類別	OPI	功能	SEO
錯誤訊息	*	錯誤碼	*
異常描述	註冊 SEO 後，一直沒上機台進行更換 FOUP 動作		
斷字斷詞	註冊(VA) SEO(FW) 後(Ng) ,(COMMACATEGORY) 一直(D) 沒(D) 上(VCL) 機台(Na) 進行(VC) 更換(VC) FOUP(FW) 動作(Na)		
字詞過濾	註冊(VA) SEO(FW) 上(VCL) 機台(Na) 進行(VC) 更換(VC) FOUP(FW) 動作 (Na)		

資料來源：本研究整理

表3.2 案例二整理資訊

案例二(C2)			
類別	OPI	功能	SEO
錯誤訊息	*	錯誤碼	*
異常描述	FOUP 無法註冊 SEO 以進行 SPLIT 動作		
斷字斷詞	FOUP(FW) 無法(D) 註冊(VA) SEO(FW) 以(P) 進行(VC) SPLIT(FW) 動作(Na)		
字詞過濾	FOUP(FW) 註冊(VA) SEO(FW) 進行(VC) SPLIT(FW) 動作(Na)		

資料來源：本研究整理

由表 3.1 和表 3.2 可知，兩筆案例可取出{註冊(VA), SEO(FW), 上(VCL), 機台(Na), 進行(VC), 更換(VC), FOUP(FW), 動作(Na), SPLIT(FW)}共 9 個關鍵字，根據出現頻率整理成表 3.3。

表3.3 關鍵字頻率整理表

	註冊	SEO	上	機台	進行	更換	FOUP	動作	SPLIT
C1	1	1	1	1	1	1	1	1	0
C2	1	1	0	0	1	0	1	1	1

資料來源：本研究整理

由關鍵字頻率表則可以整理出案例特徵向量矩陣 X 如下所示：

$$X = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}$$

若現產生一新案例(C3)：類別為「OPI」，功能「SEO」，錯誤訊息「*」，錯誤碼「*」，異常描述「FOUP 註冊 SEO 後，一直沒上機台進行更換」。整理資訊如表 3.4 所示：

表3.4 新案例整理資訊

新案例(C3)			
類別	OPI	功能	SEO
錯誤訊息	*	錯誤碼	*
異常描述	FOUP 註冊 SEO 後，一直沒上機台進行更換		
斷字斷詞	FOUP(FW) 註冊(VA) SEO(FW) 後(Ng) ,(COMMACATEGORY) 一直(D) 沒(D) 上(VCL) 機台(Na) 進行(VC) 更換(VC)		
字詞過濾	FOUP(FW) 註冊(VA) SEO(FW) 上(VCL) 機台(Na) 進行(VC) 更換(VC)		

資料來源：本研究整理

根據案例一及案例二形成之向量元素進行整理，則能將新案例轉換為一特徵矩陣 Y 如下所示：

$$Y = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0]$$

利用餘弦相似度計算案例間相似度則可以得到相似度比較資訊，如表 3.5 所示：

表3.5 案例相似度比較表

	案例一(C1)	案例二(C2)	新案例(C3)
案例一(C1)	1	0.72	0.93
案例二(C2)	0.72	1	0.62
新案例(C3)	0.93	0.62	1

資料來源：本研究整理

由表 3.5 可知新案例(C3)和案例庫中案例一(C1)最為相似。

(5) KNN 分類：

案例庫隨時間推移，其案例數量亦隨之遞增，在硬體架構未更新情況之下，推理之效能會隨之遞減而不敷使用。為有效解決此問題，本研究於異常描述外更增加類別與功能等額外資訊以為案例篩選之條件，進而有效減少訓練案例數量來提升推理效率。倘若使用者無法清楚判斷類別與功能資訊則可能造成推理效能低落與推理結果異常等狀況。因此，本研究使用資訊探勘之 KNN 方法來協助使用者判斷其問題之分類，並藉此以為篩選條件而有效提升推理效率與結果正確性。

(6) 取出專家清單：

在進行專家找尋前，必須先從近似案例取出可用之專家清單，而後才能以該清單針對其出現頻率、關係重要性等進行評分。假設現取出支援案例共二筆，根據其維護人員清單則可以取出專家共三人，如表 3.6 所示：

表3.6 近似案例維護人員資訊

案例一(C1)			
類別	OPI	功能	SEO
錯誤訊息	*	錯誤碼	*
異常描述	註冊 SEO 後，一直沒上機台進行更換 FOUNP 動作		
維護歷史	EXPA,EXPC		
案例二(C2)			
類別	OPI	功能	SEO
錯誤訊息	*	錯誤碼	*
異常描述	FOUP 無法註冊 SEO 以進行 SPLIT 動作		
維護歷史	EXPA,EXPB,EXPA		

資料來源：本研究整理

依表 3.6 資訊，則可以取出專家列表為 {EXPA, EXPB, EXPC} 共三人。

(7) 專家初始分數：

取出支援案例專家清單後，根據專家初始化計算公式計算各專家初始分數，如公式(3.2)所示

$$S_1(e, q) = \frac{pf^2}{tf} \sum_{d \in SD} p(d | q) p(e | d, q) \quad (\text{式 3.2})$$

其中 pf 為出現在該文件次數， tf 為專家出現在文件集合總次數， SD 為支援文件集合， $p(d | q)$ 為文件與案例相似度， $p(e | d, q)$ 為專家出現在該文件的機率。

根據表 3.5 整理之文件相似度與表 3.6 整理之專家清單，依公式(3.2)計算則可以得到專家處始分數如表 3.7 所示：

表3.7 專家初始分數計算範例

	專家一(EXPA)	專家二(EXPB)	專家三(EXPC)
C1-出現次數	1	0	1
C2-出現次數	2	1	0
出現總次數	3	1	1
C1-出現機率	0.33	0	1
C2-出現機率	0.67	1	0
C1-案件相似度	0.93	0.93	0.93
C2-案件相似度	0.62	0.62	0.62
專家初始分數	0.65	0.62	0.93

資料來源：本研究整理

由上表可知，若以單純之案例相似度與出現機率計算專家分數，則可發現得分最高為專家三(EXPC)。

(8) 專家總合分數：

取得各專家初始分數後，根據本研究所提出之改良型TOP-K專家總合分數評估模型，如內公式(3.3)所示：

$$S_{n+1}(e, q) = \gamma C_D(e) S_n(e, q) + (1 - \gamma) \sum_{(e, e') \in E} W_{e, e'} \cdot S_n(e', q) \quad (\text{式 3.3})$$

依[24]實驗結果，設定 γ 為0.7， n 為2。

依據表3.6案例之專家間共同出現之關係可建置專家關係矩陣表，如表3.8所示：

表3.8 專家關係矩陣表

	專家一(EXPA)	專家二(EXPB)	專家三(EXPC)
專家一(EXPA)	1	1	1
專家二(EXPB)	1	1	0
專家三(EXPC)	1	0	1

資料來源：本研究整理

並據[24]所提出之權重計算公式，如公式(3.4)所示：

$$W_{i,j} = \frac{cr(e_i, e_j)}{|E_i| \cdot |E_j|} \quad (\text{式 3.4})$$

則我們可以計算出專家間關係權重如表3.9所示：

表3.9 專家關係權重表

	專家一(EXPA)	專家二(EXPB)	專家三(EXPC)
專家一(EXPA)		0.5	0.5
專家二(EXPB)	0.5		0
專家三(EXPC)	0.5	0	

資料來源：本研究整理

那麼依照公式(3.3)、表3.7和表3.9資訊，則我們可以計算出專家總合分數，以專家一(EXPA)為例：

$$S_2(EXPA, q) = 0.7 \times 1 \times S_1(EXPA, q) + 0.3 \times [(0.5 \times S_1(EXPB, q)) + (0.5 \times S_1(EXPC, q))] \\ S_2(EXPA, q) = (0.7 \times 1 \times 0.65) + 0.3 \times [(0.5 \times 0.62) + (0.5 \times 0.93)] \approx 0.69$$

專家總合分數表則如表3.10所示：

表3.10 專家總合分數表

	專家一(EXPA)	專家二(EXPB)	專家三(EXPC)
專家初始分數	0.65	0.62	0.93
程度中心性	1	0.5	0.5
原始專家分數(n=2)	0.69	0.53	0.75
改良專家分數(n=2)	0.69	0.32	0.42
原始專家分數(n=3)	0.68	0.48	0.63
改良專家分數(n=3)	0.59	0.22	0.25

資料來源：本研究整理

由推演結果可以發現原始模型於第三次運算後才找到正確專家，且與次高分專家分數差距僅0.05，易造成系統判斷失效；而新模型於第二次運算即找到正確專家，且與次高分專家分數差距僅0.27，於第三次運算時差距則擴大至0.34，充分表現出網路富者越富的現象，且能充分提高系統判斷準確度。

因此專家找尋模型加入程度中心性後，能夠在更少的遞迴次數下更快找出具有較多關係，且貼近問題之正確專家。並且當遞迴次數增多時，正確專家與其他專家分數也形成一個明顯差距以利系統判斷。因此，相較於原始模型方法，當運用新模型來進行專家找尋則能夠有效提升運算速率與推薦準確率。

3.3 案例維護模組

使用案例式推理方式來協助解決問題，首先必須要有可以參考之案例庫，案例內容則須包含問題描述與解決方法。然而，推理效果好壞取決於案例庫中案例數目的多寡。通常當各類型案例數量越多時，則案例推理效果越好，涵蓋範圍亦越廣；反之，若案例數不多或缺乏維護，那麼就會導致案例庫數目未隨時間推移而增加或案例內容缺乏維護而不能與時俱進，並進而造成案例式推理成效低落。其關係如圖 3.6 所示：

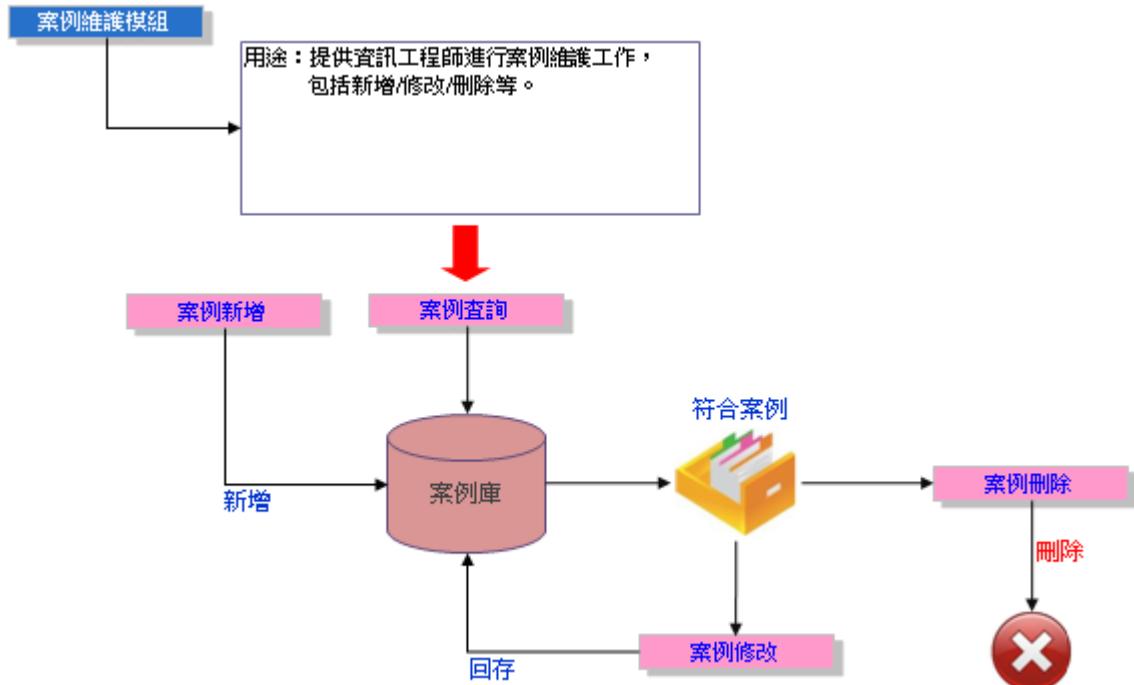


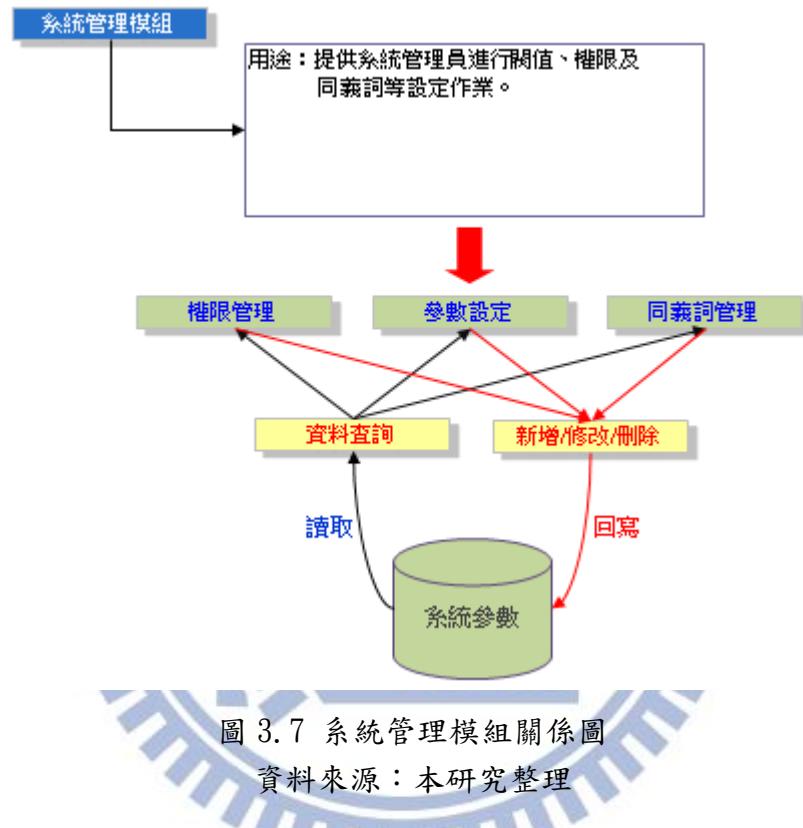
圖 3.6 案例維護模組關係圖

資料來源：本研究整理

當處理人員進行案例維護時，首先輸入查詢條件並從案例庫中擷取出相關案例集，而後選取特定案例進行處理動作，如修改或刪除。當完成維護動作後即將維護後案例寫回案例庫或將老舊不合時宜資料進行刪除。

3.4 系統管理模組

如圖 3.7 所示，系統管理模組共包含同義詞設定、權限設定與參數設定三大功能面。其中權限設定主要協助系統管理員進行個人或部門使用權限；參數設定則讓系統管理員針對各方法學細項參數依狀況進行微調，如近似閥值、中研院斷詞伺服器網路位址、專家影響力參數與系統所需運作參數等。功能面均支援新增、刪除與修改等維護必須功能。



而同義詞設定為此模組中最重要之部分，因本研究資料以中文為主，而中文字常有字詞不同但意義相近之情況。為避免測試資料在關鍵字擷取與近似值計算過程中同義字詞被捨棄，導致推理過程產生誤差之狀況，因而提供同義詞設定予管理人員依據企業內部專有名詞或同義詞部分進行維護動作。

舉例來說，若測試資料經過斷詞與字詞過濾程序後產生一個關鍵詞為「拆批」，在訓練案例資料中並無該關鍵詞。若無同義詞設定，二個詞將被認定為不同，並進而使案例相似度降低；反之，若同義詞設定「拆批」與訓練案例之「SPLIT」相等，那麼在進行案例相似度計算時就會被列入計算，而能降低系統誤判之狀況。

第四章 問題診斷暨專家推薦系統實作

本章主要是以第三章討論之系統架構與設計方向為基礎，實作一個結合案例推理、資料探勘與專家找尋功能的問題支援離型系統。將針對離型系統環境架構、開發工具及操作功能介面進行介紹，最後以實際案例擷取與專家推薦之結果對系統成效進行評估與驗證。

4.1 系統開發工具與建置環境

本研究離型系統主要在全球資訊網(WWW)基礎上架構網頁應用程式，而採用網頁應用程式架構理由則在於：

- (1) 程式包更新只需於伺服器端進行，並不需要對使用者端進行維護動作，可以有效降低維護人力與時間。
- (2) 使用者端僅需要有網頁瀏覽器就能使用系統。

本研究之離型系統以三層式架構(3-Tier)為主。前端主要為資料顯示與使用者互動介面，主要以微軟所提供之 IE6 以上或相容之瀏覽器為存取工具。中介層則為網頁伺服器，使用微軟 Windows2003 及 IIS6.0 建置，其主要功能在接收使用者之要求(Request)並進行相關處理後將結果回傳(Response)予使用者。後端則為資料庫系統，主要在儲存使用者更新項目或相關之系統參數設定值，並能針對使用者查詢條件進行處理回傳資料集。本研究之離型系統資料庫採用甲骨文所提供之 Oracle 10g XE 版本，其主要優點在於能方便移轉至企業正式之 Oracle 10g 或以上版本。

至於軟體開發工具部分，選擇 Microsoft Visual Studio 2005 為離型系統之網頁應用程式開發工具，使用語言為 C#，函式庫則以.NET FrameWork 2.0 為主，並搭配 AJAX 與 jQuery 工具來實做操作頁面。而採用.NET FrameWork 2.0 並結合 AJAX 與 jQuery 原因如下：

(1) .NET FrameWork 2.0

- 部署容易：
.NET 2.0 網站部署簡單而且容易使用。
- 系統化整理：
網頁程式碼與商業邏輯程式碼分開，方便修改。而不同物件依其種類分存於不同資料夾，方便管理。
- 高生產力：
無論在開發階段的智慧提醒或於測試階段之伺服器模擬功能、除錯工具皆能有效降低程式開發人員之負擔、耗費時間而能提升其生產力。

(2) AJAX

- 數據量減少、回應更快

AJAX 應用可以僅向伺服器發送並取回必需的數據，因為在伺服器和瀏覽器之間交換的數據大量減少，進而讓回應更快。

(3) jQuery

- 語法簡單

透過其框架能以更簡單的語法來達成原有 JavaScript 繁複的邏輯撰寫。

- 提供方便的使用介面模塊

提供套裝化的樣式表與介面選項，讓程式開發人員能方便引用來達成更為多彩多姿的操作介面。

透過以上各種開發工具的使用則能協助減少開發時程，並能以更為簡潔之程式碼來達成目的，並以強大的測試與除錯工具來釐清錯誤與改正缺點。最後，針對上述之離型系統開發環境等相關資訊，整理如表 4.1 所示：

表4.1 離型系統開發環境

環境項目	使用之工具或軟體版本
網頁伺服器作業系統	Windows 2003 SP2
網頁應用伺服器	Internet Information Service 6.2
系統開發工具	.NET FrameWork 2.0，ASP.NET 2.0 with C#，AJAX，jQuery
資料庫系統	Oracle 10g Express Edition

資料來源：本研究整理

而離型系統後端資料庫系統之 ER-Model 則如圖 4.1 所示：

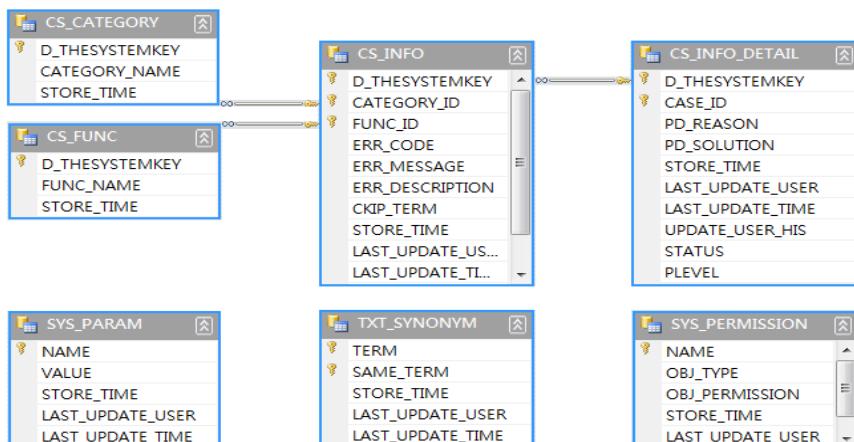


圖 4.1 離型系統 ER-Model

資料來源：本研究整理

如 ER-Model 所示，CS_INFO 為雛型系統核心資料表，主要用來儲存案例相關描述資訊與斷詞結果，而 CS_INFO_DETAIL 則透過索引連結主資料表，主要用來儲存案例對應之可能原因與解決方案等資訊。除此之外，針對 SYS_PARAM、TXT_SYNONYM 與 SYS_PERMISSION 則分別用來儲存系統參數、同義詞資訊與系統權限等資訊。

4.2 系統操作流程與展示

本節主要介紹已實作之資訊系統問題診斷支援系統，透過圖文方式對系統功能與操作流程進行完整說明與介紹。

4.2.1 使用者登入驗證

雛型系統登入畫面如圖 4.2。進入系統前皆須使用特定帳號及專屬密碼進行驗證措施，以利系統擷取使用者資訊及驗證可使用之功能清單。步驟一：輸入員工編號與密碼，如圖 4.2.1 為「980099」，密碼則以「*」表示。步驟二：輸入完成後，點選「確定」按鈕執行使用者驗證程序。



圖 4.2 系統登入畫面

資料來源：本研究整理

經過系統驗證與比對後，系統會根據不同使用權限顯示不同之畫面。例如，當身分為「使用者」，系統畫面僅顯示問題診斷模塊，如圖 4.3 所示。

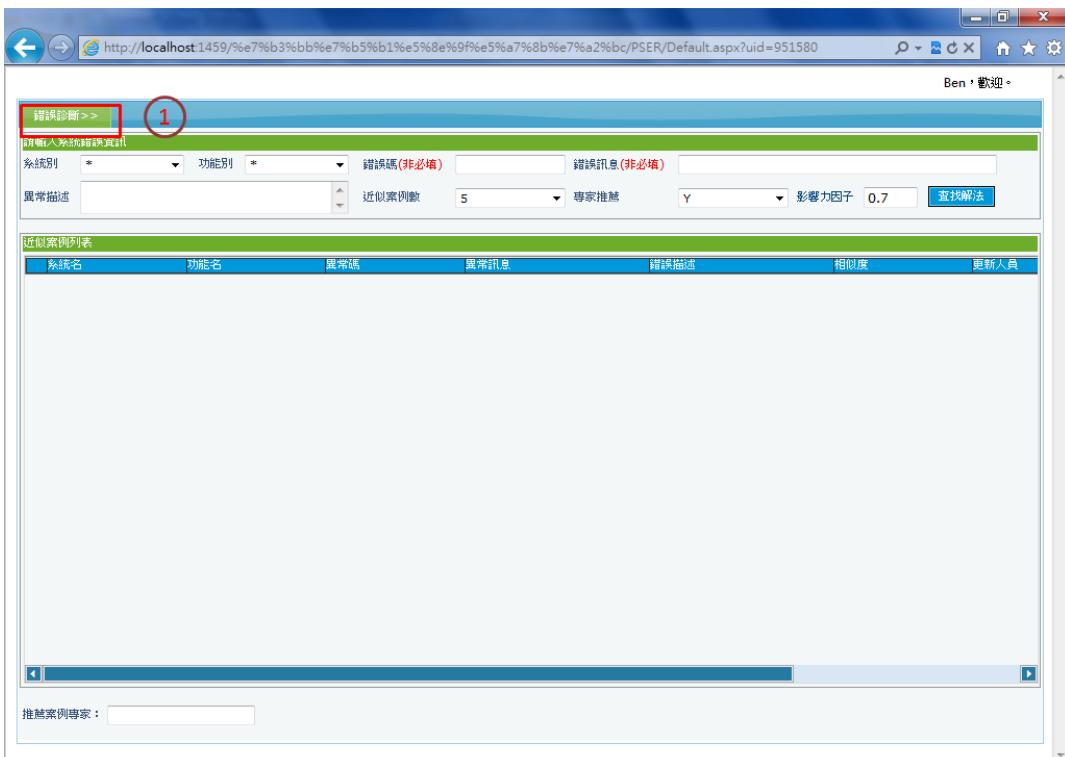


圖 4.3 使用者系統功能畫面

資料來源：本研究整理

當身分為「工程師」，系統畫面則顯示問題診斷模塊與案例維護模塊，如圖 4.4 所示。

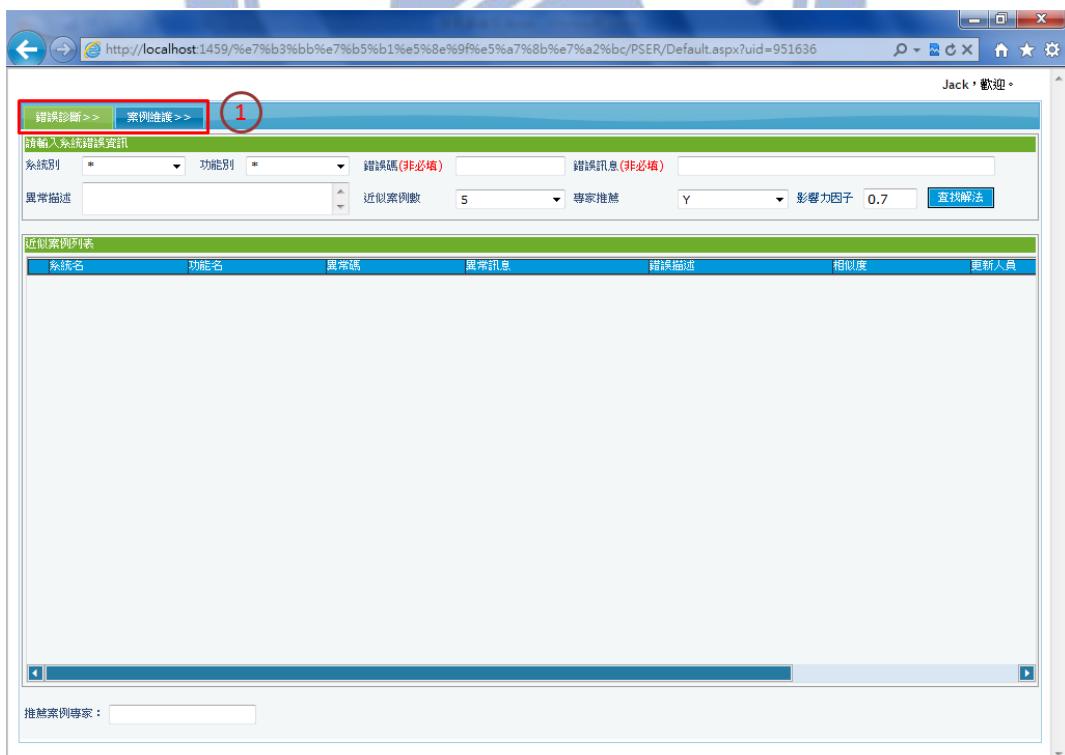


圖 4.4 工程師系統功能畫面

資料來源：本研究整理

當身分為「系統管理員」，系統畫面則顯示所有功能模塊，如圖 4.5 所示。

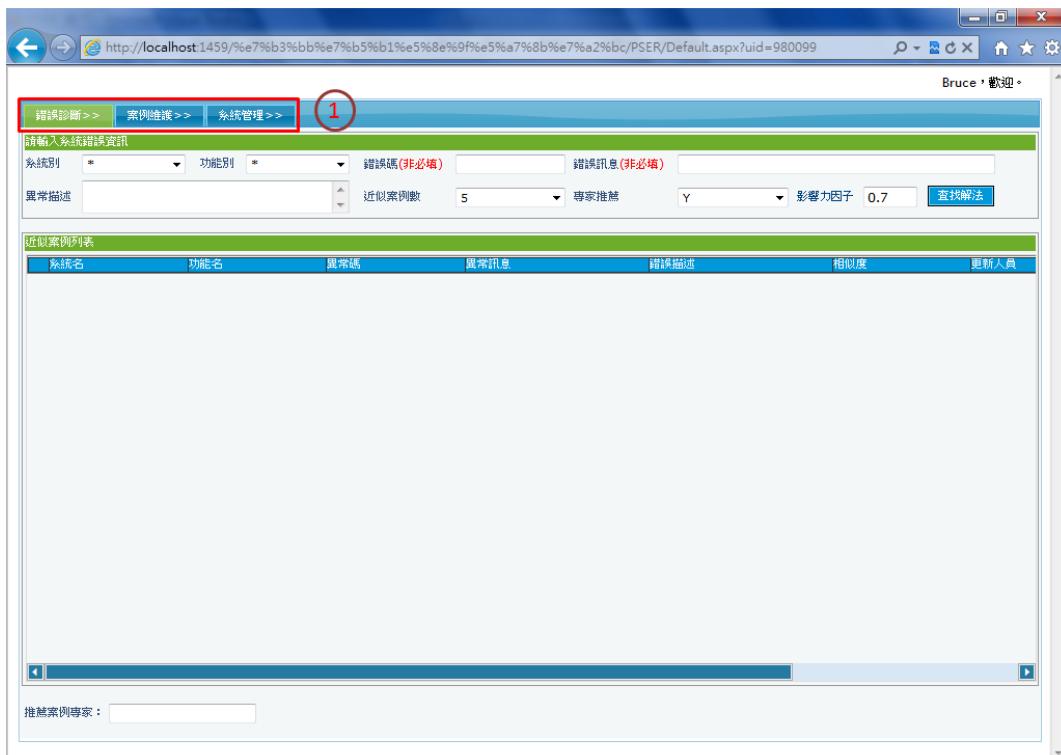


圖 4.5 系統管理員系統功能畫面

資料來源：本研究整理

4.2.2 錯誤診斷暨專家推薦模組

錯誤診斷暨專家推薦模組，顧名思義即其中包括兩大項功能，一個為問題分類判斷，根據輸入之新案例描述與條件進行相似案例比對，取出近似案例以協助處理人員處理資訊系統問題。另一個為專家推薦，根據取出之近似案例列表之撰寫人員資訊所形成之專家候選人中，以本研究改良之評分模型進行計算並取最高分之專家為推薦專家，意在利用相關案例之經驗專家來補足案例庫之不足。

其操作流程如圖 4.6 所示。使用者執行錯誤診斷與專家推薦功能前，必須(1)將問題產生之「系統別」、「功能別」資訊輸入，以利系統根據此範圍尋找訓練案例，可有效降低計算量與案例特徵向量之維度量。若使用者未能清楚判定「系統別」、「功能別」相關資訊，則可依序填入「*」、「*」，系統則會根據異常描述進行案例特徵向量相似度比較，並根據 KNN 演算法進行自動分類。(2)完成輸入案例異常描述後，使用者必須選取支援案例數以告知系統近似案例最大量，此參數值後續會使用於 KNN 分類方法與專家找尋模型中。(3)選取專家推薦之設定值，「Y」系統必須依近似案例推薦專家以供諮詢，「N」系統僅需提供近似案例清單與相關資訊。(4)根據實際狀況與系統特性，設定專家影響因子來明確判別專家供線之影響力。(5)最後，完成所有輸入與設定之動作後，點選「查找解法」按鈕驅動程式進行案例推理與專家找尋程序運算。

This screenshot shows the 'Error Diagnosis and Expert Recommendation' interface. At the top, there are three tabs: '錯誤診斷 >>', '案例推陳 >>', and '系統管理 >>'. The main area has a green header bar with several input fields and dropdown menus. A red box highlights the first input field, which contains '系統別 *' and '功能別 *'. Five numbered circles (1 through 5) are placed around the interface to point out specific elements: circle 1 points to the '系統別' dropdown; circle 2 points to the '錯誤碼(非必填)' input field; circle 3 points to the '錯誤訊息(非必填)' input field; circle 4 points to the '影響力因子' input field; and circle 5 points to the '查詢解法' button. Below this is a table titled '近似案例列表' (Similar Case List) with columns: 系統名 (System Name), 功能名 (Function Name), 非常碼 (Abnormal Code), 非常訊息 (Abnormal Message), 調試描述 (Debug Description), 相似度 (Similarity), and 更新人員 (Update Person). The bottom of the interface features a search bar labeled '推薦案例專家:' and a status message '推薦案例專家：' followed by a placeholder '980099'.

圖 4.6 錯誤診斷暨專家推薦操作畫面

資料來源：本研究整理

This screenshot shows the same interface as Figure 4.6, but it displays results. The input fields at the top remain the same, with a red box highlighting the '系統別' dropdown set to 'OPI' and '功能別' dropdown set to 'LOT'. A red circle labeled '1' is placed over the '系統別' dropdown. In the '近似案例列表' table, there is one row highlighted with a red box, showing 'OPI' in the '系統名' column and 'LOT' in the '功能名' column. A red circle labeled '2' is placed over this row. At the bottom, a red box highlights the '推薦案例專家:' input field, which contains the value '980099'. A red circle labeled '3' is placed over this field.

圖 4.7 錯誤診斷暨專家推薦結果畫面

資料來源：本研究整理

如上圖 4.7 則顯示系統完成錯誤診斷暨專家推薦後之結果顯示畫面，如圖 4.6 所示，使用者因無法清楚判斷問題產生系統與所屬功能別，輸入「*」。而(1)系統則利用使用者輸入之問題描述等相關資訊進行 KNN 分類後，確定其可能所屬之系統別為「OPI」、功能別為「LOT」，並回填資訊。(2)則顯示由系統利用餘弦相似度計算範圍內之已存案例，並依相似度而大至小排列後，根據使用者設定之近似案例數來取得顯示之案例數。(3)若使用者設定專家推薦為「Y」，則依據本研究提出之改良專家找尋模型，系統依近似案例之撰寫人清單進行計算後，並將得分最高之專家填入該區域。

4.2.3 案例維護模組

案例推理正確性取決於案例數量與案例內容之維護，因此本研究離型系統提供案例維護模組予處理人員，如「工程師」、「系統管理員」等，進行案例查詢、新增、刪除與修改等維護功能。如圖 4.8 所示，(1)為案例查詢條件設定區，使用者可依需求分別設定系統、功能、處理人員或日期等條件進行查詢，該區域亦配置「新增案例」按鈕以利使用者點選進行案例新增。(2)根據設定條件，自案例庫查詢出符合條件之「已結案」之案例，並顯示其相關異常描述、可能原因及解決方案。按鈕列「U」更新案例、「D」刪除案例、「H」隱藏案例解決方案資訊、「E」顯示隱藏案例解決方案資訊。(3)依登入使用者資訊，自案例庫取得該使用者「未結案」之案例資訊。

The screenshot shows the 'Case Maintenance Module' interface. At the top, there are tabs for 'Error Diagnosis >>', 'Case Maintenance >>', and 'System Management >>'. Below this is a search bar with fields for 'System Type', 'Function Type', 'Error ID', 'Error Message', and 'Processor ID'. A red circle labeled '1' highlights the 'Search Case' and 'Add Case' buttons.

The main area is divided into two sections:

- Section 1 (Top):** Shows a table of closed cases. The columns are 'System Name', 'Function Name', 'Abnormal ID', 'Abnormal Message', and 'Error Description'. One row is highlighted with a red box, showing 'OPI' and 'LOT' as system and function names respectively. A red circle labeled '2' is at the bottom right of this section.
- Section 2 (Bottom):** Shows a table of unresolved cases. The columns are 'System Name', 'Function Name', 'Abnormal ID', 'Abnormal Message', and 'Error Description'. One row is highlighted with a red box, showing 'OPI' and 'SEO' as system and function names respectively. A red circle labeled '3' is at the bottom right of this section.

圖 4.8 案例維護模組主畫面

資料來源：本研究整理

以下則分別針對案例維護之新增、修改與刪除等案例維護功能之操作畫面進行詳細介紹與操作步驟解說。

如圖 4.9 所示為案例新增操作畫面，新增案例介面共分系統錯誤資訊與處理資訊兩大區塊。使用流程步驟(1)使用者依問題發生之系統選擇既有系統資訊或新增。(2)依系統區分其細部功能別，亦可選擇既有功能或新增之。(3)根據異常發生之錯誤訊息、異常描述等依序填入。(4)若已知發生原因與解決方案則填入相關資訊，若未知則可不填。(5)依據機密程度，使用者可設定處理資訊可觀看層級。最後，點選「儲存」按鈕進行案例儲存。儲存時，若相關資訊欄位皆已填具則案例狀態為「已結案」；反之，若有資訊尚未填具則案例狀態為「未結案」。

新增/更新案例資訊

系統錯誤資訊

OP (非必填)

SEO (非必填)

錯誤訊息
FOUP無法註冊 SEO

處理資訊

發生原因
FOUP其狀態可能不為SI

解決辦法
將FOUP置入STOCKER後再行設定

處理人員
980099

檢視層級
工程師

儲存 取消

圖 4.9 案例新增暨更新操作畫面

資料來源：本研究整理

案例更新畫面亦同圖 4.9，系統將已存案例資訊填入相對應欄位，使用者可依最新處理狀況進行各欄位資訊更新並儲存。同新增案例功能，若有資訊尚未填具則案例狀態變更為「未結案」。圖 4.10 則顯示案例刪除成功之畫面，操作步驟(1)針對已存在案例庫，且狀態為已結案之案例。若欲刪除之，則點選畫面「D」之按鈕驅動系統刪除功能。(2)刪除完成後，系統會顯示「刪除案例成功。」訊息。最後，則根據設定條件進行更新。

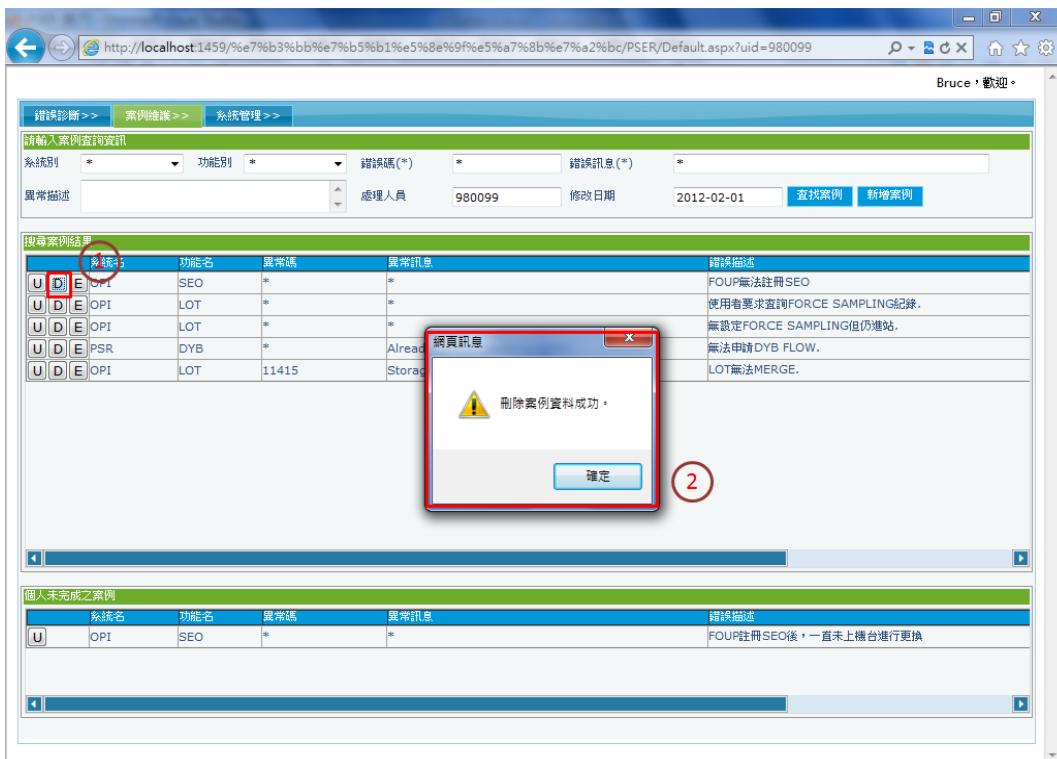


圖 4.10 案例刪除操作畫面

資料來源：本研究整理

4.2.4 系統管理模組

系統管理模組主要功能在於提供系統管理人員一個方便維護使用權限、系統參數與同義詞設定等相關設定，其中又以同義詞管理最為重要。以下將針對此模塊三大功能進行介紹。

如圖 4.11 所示即為權限管理操作畫面，本研究將使用者物件分為「個人」或「部門」兩大類。通常設定模式以群組設定方式來進行群組權限授與，而個人層級則高於群組權限。例如，某 A 部門設定權限為「使用者」，則該部門人員皆僅能操作錯誤診斷及專家推薦模塊功能，若有一隸屬於 A 部門之使用者 A1 設定權限為「工程師」，那麼 A1 就可以操作案例維護模塊。

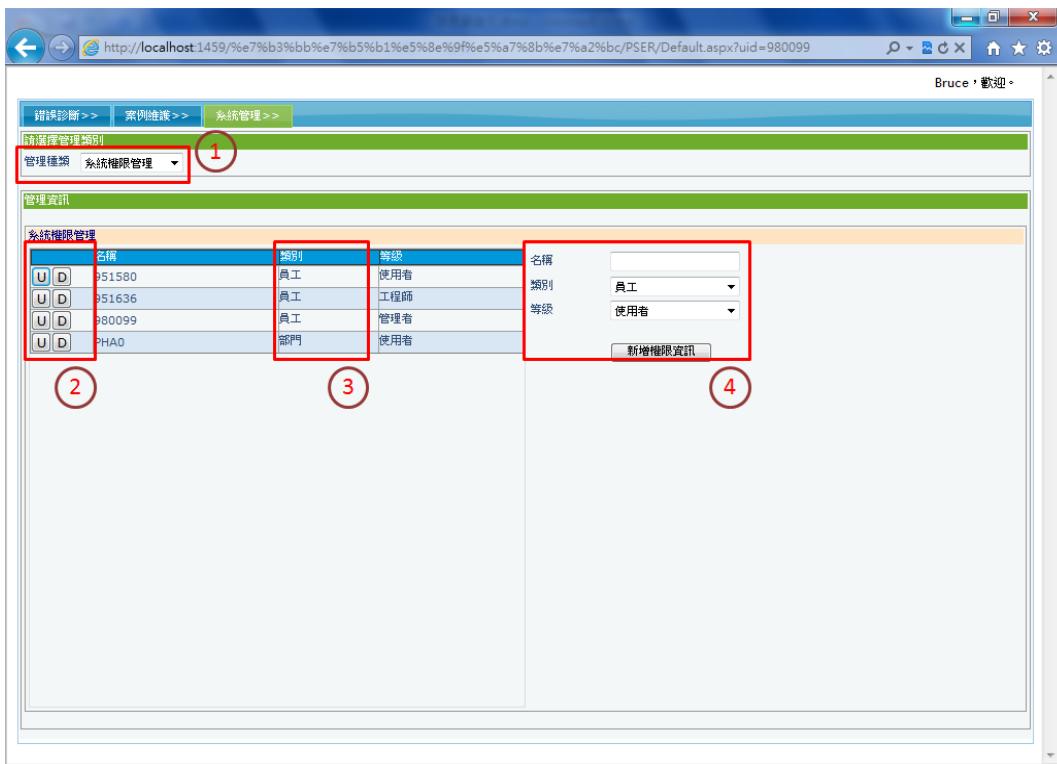


圖 4.11 權限管理操作畫面

資料來源：本研究整理

如上圖，(1)為管理種類選擇清單，共「系統權限管理」、「系統參數設定」與「同義詞類管理」三類。(2)則為特定紀錄之維護按鈕，「U」為紀錄屬性更新、「D」為記錄刪除功能。(3)為物件類別，由畫面可知分為「個人」與「部門」兩類。(4)為資訊顯示區塊，管理員可於此處新增權限或檢視權限資訊並更新。

如圖 4.12 所示即為系統參數管理操作畫面，本研究提供此介面予管理者進行參數維護功能，如新增、刪除或修改。一般來說，此部分參數存放資訊通常為可能隨時間變遷而改變之參考資訊，如中研院斷詞服務伺服器網路位址、相似度擷取過濾值或企業郵件伺服器位址等。

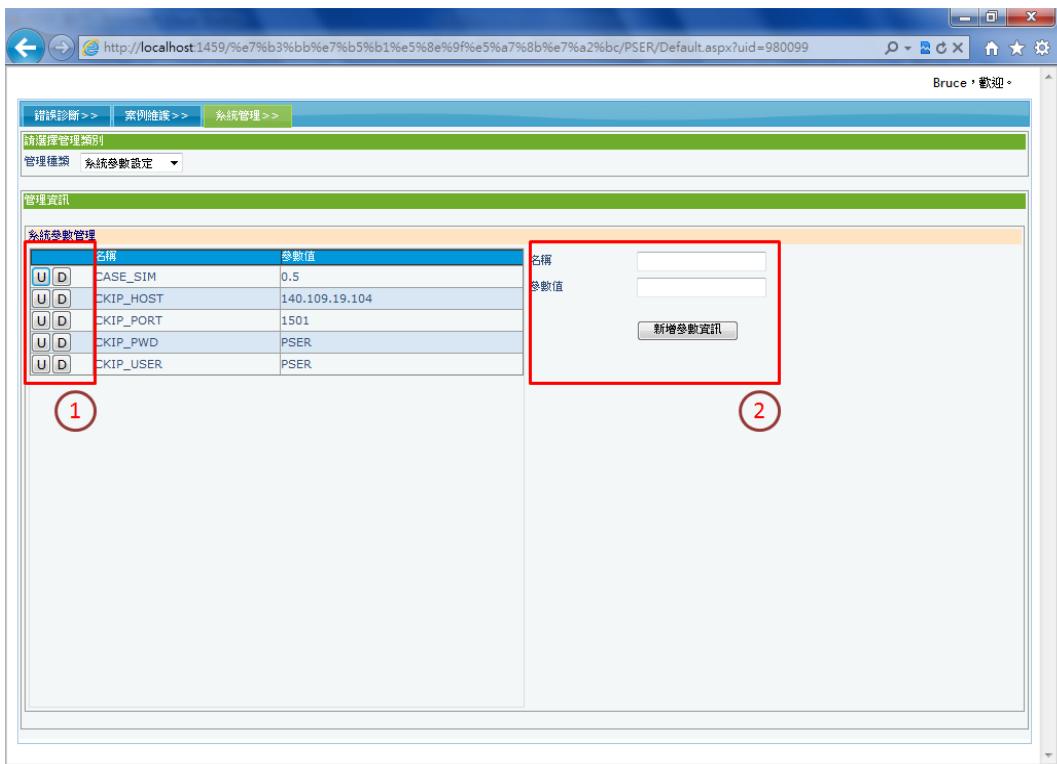


圖 4.12 系統參數管理操作畫面

資料來源：本研究整理

如上圖，(1)則為紀錄之維護按鈕，「U」為參數值更新、「D」為參數刪除。(2)為資訊顯示區塊，管理員可於此處新增資訊或檢視參數資訊並更新。

同義詞管理主要功能在於輔助案例進行相似度比對時之重要依據。透過設定，進而能協助案例比對與降低誤判。如圖 4.13 所示即為同義詞管理作畫面，畫面左方為已設定之同義詞資訊，而(1)為系統提供之刪除與修改功能按鈕，管理員可透過此功能進行資料維護。(2)則為系統顯示資訊或管理員新增同義詞輸入之欄位。如畫面範例，管理員設定「不行」與「無法」屬於同義詞，因此於案例相似度比較時會將其視為相等之關鍵詞。

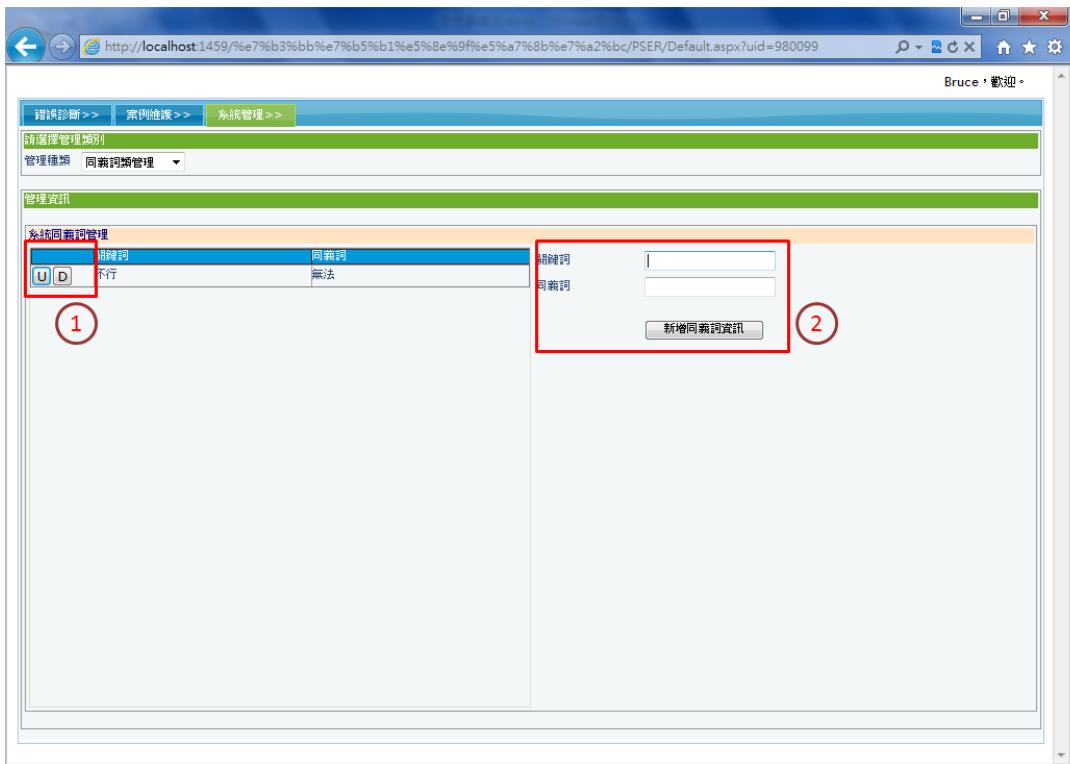


圖 4.13 同義詞管理操作畫面

資料來源：本研究整理

4.3 效益分析

本研究目的在於評估基於案例推理並結合資料探勘與專家找尋機制是否能有效協助使用者有效解決資訊系統問題。因此針對離型系統之效益評估與驗證可用性則分成兩個階段：首先將案例資料類別與其案例區分為訓練與測試兩組資料，並選取企業內部該領域之專家委請評估推理結果正確性。第二階段則利用訓練案例形成特徵向量維度，並以選定之維度將案例轉換成向量，而後以餘弦相似度計算案例間相似度來取出前 K 筆最相近之案例。當近似案例擷取完成後，再行利用相近似案例經過專家找尋機制來評算出推薦之專家。最後，則將其系統擷取案例及推薦專家之結果反應給選定之專家進行人工判斷是否合於實際現況。

4.3.1 系統評估

本研究測試資料來源是以某半導體公司內部之 MES 系統值班紀錄中資料為主，利用工具從其固定之文件資料分別將值班過程所遭遇之問題資訊、解決方案與處理人員擷取後儲存至案例資料庫，並以其三大系統之問題為代表性類別，分別為 MM、OPI、ALH 等。為驗證系統之可行性，本研究將以案例之 90%為訓練資料，餘下之 10%則為測試資料，如表 4.2 所整理。

表4.2 樣本資料數統計表

	樣本數	訓練樣本數	測試樣本數
MM	137	123	14
OPI	115	103	12
ALH	103	93	10
總計	355	319	36

資料來源：本研究整理

除以原有案例測試外，針對三大類型之分類，本研究亦委請企業內部此三大系統之領域專家協助評估系統分類準確性、案例擷取近似性、推薦專家可信度與處理時間之驗證與評分。領域專家背景資料如附錄二。

4.3.2 系統驗證

(1) 案例分類評估

針對案例分類準確度評估方面，本研究以正確率與召回率作為評估指標[44]。以分類而言，文件分類狀況如表 4.3 所示，屬於該類的文件中被系統正確分類有 P 篇、沒有被系統分類則有 Q 篇；不屬於該類的文件中被系統分為該類有 R 篇、沒有被分為該類的有 S 篇。

表4.3 文件分類判別表

	系統分為該類	系統不分為該類
屬於該類別	P	Q
不屬該類別	R	S

資料來源：本研究整理

正確率(Precision Rate)計算方式如下：

$$\text{正確率(Precision Rate)} = \frac{\text{人工判定為該類} \cap \text{系統判定為該類}}{\text{系統判定為該類}} \times 100\% \quad (\text{式 4.1})$$

召回率(Recall Rate)計算方式如下：

$$\text{召回率(Recall Rate)} = \frac{\text{人工判定為該類} \cap \text{系統判定為該類}}{\text{人工判定為該類}} \times 100\% \quad (\text{式 4.2})$$

本研究將測試資料逐一輸入系統分類所得結果、再與各類別領域專家評估分類正確性相關數據，並依正確率(公式 4.1)及召回率(公式 4.2)整理如表 4.4 所示。

表4.4 案例分類驗證表

類別	測試數	專家歸類	系統歸類	正確數	錯誤數	正確率	召回率
MM	14	13	10	7	3	70%	53.8%
OPI	12	10	8	7	1	87.5%	70%
ALH	10	10	9	8	1	88.8%	80%

資料來源：本研究整理

由表 4.3.3 之實驗結果可以發現，離型系統利用分類方法學縮小資料搜尋範圍，使用斷詞工具並進行詞性過濾來進行案例擷取，可以讓案例擷取之正確性高於 70% 以上之合理可使用範圍，其相關召回率部分亦屬偏好之程度。唯 MM 類別因案例數多，產生較多關鍵字而致使維度複雜化，進而導致系統分類產生誤判而讓召回率較差。

(2) 案例推理暨推薦專家評估

針對案例推理所擷取出之近似案例與推薦專家之可信度，主要以領域專家評分為依據，依其評分高低來驗證系統處理結果是否可行。實驗程序如下：首先從三個類別案例中選取測試案例如表 4.5 所示，緊接利用案例推理與專家找尋機制來取得前 5 筆近似案例與推薦專家，最後將所得相關結果反應給領域專家，並由專家群針對推薦案例、專家進行評分。評分標準為 1-5 分，分數由小至大分別為「非常不滿意」、「不滿意」、「尚可」、「滿意」、「非常滿意」。

表4.5 案例推理測試案例

系統別	功能別	問題描述
MM	EQP	查詢 EQP 資訊，系統產生錯誤無法顯示資訊
OPI	SEO	註冊 SEO 後，FOUP 一直未上機台更換。
ALH	SPLIT	執行 SPLIT 動作，產生 ALFH 異常碼。

資料來源：本研究整理

驗證結果如表 4.6 所示，案例平均滿意分數各為「4.2」、「4.4」、「4」，推薦專家滿意分數各為「4」、「4」、「5」，專家平均滿意度為「4.3」，皆介於滿意至非常滿意之間，代表各領域專家認為離型系統依據案例推理與專家找尋機制所擷取出之近似案例與專家大致符合現況，可以提供給相關使用人員作為問題診斷及處理決策輔助之工具。

表4.6 案例推薦滿意調查表

測試資訊				
系統別	功能別	問題描述		
MM	EQP	查詢EQP資訊，系統產生錯誤無法顯示資訊		
系統處理結果				
系統別	功能別	問題描述	解決方案	評分
MM	EQP	查詢機台，系統無法顯示資訊	機台內LOT資訊遺漏，補足即可	5
MM	EQP	查詢EQP INFO跳出錯誤訊息	兩廠之機台設定不一致，重設即可	4
MM	EQP	查詢EQP時無法顯示NearByStocker資訊	基準情報未設定完成，請基準情報處理	4
MM	EQP	查詢機台時，OPI產生錯誤無法顯示	機台內LOT資訊遺漏，補足即可	5
MM	EQP	查詢機台資訊時，機台狀態異常	資訊殘留，請TCS重置即可	3
推薦專家			評分	
951580				4

測試資訊				
系統別	功能別	問題描述		
OPI	SEO	註冊SEO後，FOUP一直未上機台更換。		
系統處理結果				
系統別	功能別	問題描述	解決方案	評分
OPI	SEO	BAY635.000註冊SEO很久，一直未更換	將FOUP傳至GE605即可	5
OPI	SEO	ForceExchang註冊後一直未更換	將Source與Destination傳送相同Stocker	4
OPI	SEO	註冊SEO約2小時，FOUP一直未更換	確認位置並手動傳送對應Stocker	5
OPI	SEO	於SEO註冊Merge，實體一直未更換	將Source與Destination傳送相同Stocker	4
OPI	SEO	註冊SEO後，RTD一直未派工	確認位置正確性並聯絡RTD值班確認	4
推薦專家			評分	
951636				4

測試資訊				
系統別	功能別	問題描述		
ALH	SPLIT	執行SPLIT動作，產生ALFH異常碼。		
系統處理結果				
系統別	功能別	問題描述	解決方案	評分
ALH	SPLIT	執行時產生ALFH，RTD被關閉訊息	將LOT合併後再做一次	4
ALH	SPLIT	產生Source not match Destination異常碼	手動將LOT傳至對應Stocker即可	3
ALH	SPLIT	執行過程產生ALFH異常碼	LOT狀態異常，修改狀態即可	5
ALH	SPLIT	SPLIT時產生ALFH異常碼異常終止	拆批動作異常，將已拆出合併再做一次	5
ALH	SPLIT	產生ALFH，Interfab Command stop訊息	手動將LOT傳至對應Stocker即可	3
推薦專家			評分	
980099				5

資料來源：本研究整理

第五章 結論與未來研究方向

本章分為四個部份，第一部份為本研究所提出之結論，第二部份及第三部份則分別說明本研究的貢獻以及對於後續研究之建議與方向。

5.1 研究結論

本研究是一種以案例式推理方法為基礎並整合文件探勘與專家找尋機制來實作一個離型系統，用來輔助企業將內隱知識外顯化來讓內部資訊人員對於資訊系統異常問題診斷與決策能快速回應與解決。離型系統實作後經過案例擷取驗證與委請專家進行人工驗證，並根據正確率與召回率與各領域專家針對推薦結果之滿意度評估等結果，就問題支援案例擷取正確性、潛在專家推薦可用度與案例維護處理時間來看，本研究所實作之離型系統確實可有效提供近似案例予管理人員進行新問題處理，更能依據案例庫萃取出可用之專家以供諮詢來補足案例資訊不足之處。而案例維護模組則能以規格化的輸入方式來將較無結構化之處理經驗轉化為可重複使用之顯性知識，並協助企業有系統的儲存異常處理經驗與傳承。

5.2 研究貢獻

本研究結合案例推理、資料探勘與專家找尋機制來實作一資訊系統問題診斷系統，其為企業及其資訊人員帶來貢獻分別有：

- (1) 便利之操作介面，協助資訊人員有效率的進行案例處理經驗之儲存與以往案例處理經驗之利用，以標準化輸入元件協助相關人員有系統的將內隱知識進行外化，而能迅速增加其組織知識的存量與價值。透過案例推理流程與以往處理經驗之回饋，則能協助企業單位提升資訊服務之績效、降低系統異常造成之成本浪費，並進而能加強企業整體對外競爭力與面對競爭環境快速變遷之調適力。
- (2) 藉由案例維護機制與便利的查詢功能，能促進企業內部異常處理經驗流通，並得而提升成員獲取異常處理經驗之效率，進而降低異常處理時間。且本研究之離型系統能有效整合組織內各成員對於資訊系統異常處理經驗與其相關知識，並藉由專家找尋機制推薦合適之經驗專家加入異常處理團隊協助處理異常問題並激發組織合作之潛能，以社會化方式與團隊實踐模式來進行專家內隱知識之轉換與傳承。
- (3) 本研究結合各方法學所實作之離型系統能有效儲存工程師專業知識與經驗而能輕易利用，並減輕因資訊工程師流動所造成之知識流失與營運成本增加等問題。

而根據 TURBAN 等人[48]所整理，知識管理是組織將重要資訊、組織記憶及組織中非結構化的各種專業知識，進行辯識、選擇、組織、傳播與轉換的過程。而知識管理的目標在於：認知到個人與群體知識的存在，使組織可以善用所擁有的知識。而知識管理

系統是以現代化科技系統化地提升及促進組織內及組織間的知識管理，亦即該系統為了將組織人力資本的專業知識廣泛傳播、協助處理問題及因應快速變遷及永續經營等壓力所建立。

基本上，知識管理系統會依照六個步驟來循環運作。如圖 5.1 所示。循環原因在於知識會隨著時間演進而逐漸精煉。當環境變遷時，知識必須隨之更新以反應變化。

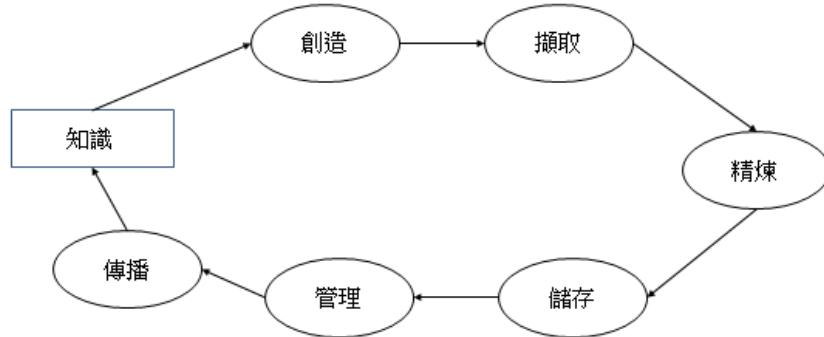


圖 5.1 知識管理循環

資料來源:[48]

本研究之離型系統能有效協助企業以知識管理之六大步驟來進行異常問題處理經驗與知識之管理與應用，詳述如下：

(1) 創造知識：

創造知識在於發現新的作法、發展新的技術或是以各種方式將組織內人員之內隱知識將之轉化而得。本研究提供的離型系統能協助企業資訊相關人員以一種較系統化的方式將異常處理經驗轉化為外顯知識以供組織利用。

(2) 擷取知識：

知識必須要確認具有價值，並以合理的方式來進行呈現。本研究以資料探勘等相關方法來協助企業擷取合適之異常處理紀錄以供利用。

(3) 精煉知識：

知識必須具有可執行的內容，才能使人們可以理解並精化知識。本研究提供之離型系統即透過案例維護之更新功能讓相關使用者可以進行案例資訊更新，以符合最新之狀況。而透過專家推薦，則能讓使用者於案例之外找到較具經驗之專家，透過詢問等社會化方式來精煉知識並進而轉化。

(4) 儲存知識：

有用的知識必須具有可執行的以合理的形式儲存於知識庫，以利事後存取。本研究針對資訊系統異常所需資訊，透過分析後建置相關資料表以利進行案例存放與便利操作之用。

(5) 管理知識：

知識必須隨時更新，亦必須隨時檢視其相關性與正確性。本研究提供之案例維護功能及協助使用者能隨時檢視案例資料與其解決方案，並視實際運作狀況與系統邏輯來進行相關維護動作。

(6) 傳播知識：

知識必須以方便取得的方式呈現，以利組織每個人都可以取用。本研究提供之系統相關介面基於使用者權限劃分，能有效協助使用者取得他們所需之相關處理案例與知識。

由上可知，本研究之離型系統不僅能協助資訊人員有效進行異常處理外，更能有效協助企業以合理的知識管理循環步驟進行資訊人員所專有的內隱知識與日常維護與異常處理經驗之轉化與儲存相關事務，並進而能使之成為組織知識以為所用。

5.3 未來研究方向與建議

本研究雖結合案例推理、資料探勘與專家找尋機制實作離型系統驗證其對於企業問題的輔助能力，但仍有須進行功能加強或修改之處，如：

- (1) 本研究在案例推理流程中，僅針對案例擷取(Retrieve)、再使用(Reuse)與保留(Retain)加以探討，但並未針對案例校正部份進行特別處理。一般來說，案例推理流程通常在案例再使用即回存後會針對案例進行調整。但本研究較著重於處理經驗等較無結構性之資料並無相關之量化資料可進行修正，所以並無針對案例校正之流程加以處理。
- (2) 資料樣本數並非相當龐大，案例向量維度普遍較少而能提升案例相似度與專家找尋計算速度。然而系統相關之計算與評分機制可能因資料數增大而有相當之影響。
- (3) 當案例資料量增多時，案例特徵向量維度往往過於複雜進而影響相似度計算效率與相似度偏低之情況。因此，針對關鍵詞過濾設定與其連帶相關參數會連帶影響案例維度數目之增減。
- (4) 測試案例專家間共同撰寫情況較少，進而造成社群網路結構密度較低，可能影響專家評分準確度。

針對以上未盡完善之處，本研究建議：

- (1) 應針對案例處理經驗等較無結構化等資料進行分析並開發對應之自動搜尋更新案例背景程式來協助進行案例自動修正等流程，以求案例庫資訊與時俱進。
- (2) 應將該問題診斷系統結合至企業異常處理流程以持續更新案例資訊並快速增加案例之數量，如此即能使案例擷取更為準確，對於案例推理之可用度能有較佳結果。
- (3) 應透過企業實際專家定期針對已分類之案例進行關鍵詞分析，藉由分析結果進行關鍵詞組的設定來形成各類別之特徵向量，進而有效降低向量維度來提升案例相似度比對與推薦之準確率。
- (4) 擴展案例分享及共筆平台，協助相關人員維護案例資訊、分享案例與互相補充案例以使案例資訊更加完善，並形成案例維護歷史與共筆清單，讓文件之專家關係更形密切、網路更為緊密而能提升專家評分機制之準確率。
- (5) 關鍵詞過濾與同義詞之設定，應由企業內部指派之專家視實際狀況與其領域專業知

識進行辨識來加以微調，進而能提升計算效率與正確率。

- (6) 應將本研究所實作之離型系統與企業知識管理流程相結合，如此可透過日常作業維護與異常處理紀錄之新增使案例庫持續更新，而讓資訊相關人員與系統或專家間互動能更加緊密、系統輔助決策亦能更加即時與有效。



參考文獻

英文部分：

- [1] Schank, R. C., Abelson, R. P., “Scripts, Plans, Goals and Understanding” , Lawrence Erlbaum Ass. Hillsdale, New Jersey, 1977.
- [2] Schank, R. , “Dynamic Memory: A Theory of Learning in Computers and People” , New York: Cambridge University Press, 1982.
- [3] Watson, I., “Case-based reasoning is a methodology not a technology.” , Knowledge-Based Systems, vol. 12, pp. 303–308, 1999.
- [4] Christopher Reisbeck, Roger Schank. , “Inside Case-Based Reasoning.” , Lawrence Erlbaum Ass. Hillsdale, New Jersey, 1989.
- [5] Klaus-Dieter Althoff, “Case-Based Reasoning” , Handbook on Software Engineering and Knowledge Engineering, River Edge, 2002.
- [6] Fritz, H. G., “Case-Based Reasoning Applying Past Experience to NewProblems” , Information System Management, pp. 77–80, Spring, 1993.
- [7] Medhbi, M., and Owrang, O., “Case Discovery in Case-Based Reasoning” , Information System Managements, pp. 74–78, Winter, 1998.
- [8] Ralph, Barletta, “An Introduction to Case-Based Reasoning” , AI Expert , pp. 43–49, August, 1991.
- [9] Case-based reasoning, Wikipedia,
http://en.wikipedia.org/wiki/Case-based_reasoning
- [10]Sullivan, D., “Document Warehousing and Text Mining” , Wiley Computer Publishing, 2001.
- [11]IBM, “Intelligent Miner for Text : Getting Started” , IBM Corp, 1998.
- [12]G. Salton, A. Wong, and C. S. Yang, “A Vector Space Model for Automatic Indexing” , Communications of the ACM, vol. 18, pp. 613–620, 1975.
- [13]M. S. Granovetter, “The Strength of Weak Ties, American Journal of Sociology” , vol. 78, no. 6, pp. 1360–1380, 1973.
- [14]Y. J Bian, “Guanxi and the allocation of jobs in urban China.” , The China Quarterly, vol. 140, pp. 971–999, 1994.
- [15]McPherson, M., Smith-Lovin, L., & Cook, J. M., “Birds of a Feather: Homophily in Social Networks.” , Annual Review of Sociology, vol. 27, pp. 415–444, 2001.
- [16]David Easley, “Networks, Crowds, and Markets: Reasoning About a Highly Connected World” , Cambridge University Press, 2010.
- [17]Faust, K., “Centrality in affiliation networks.” , Social Networks, vol. 19, pp. 157–191, 1997.
- [18]Scott, J., “Social Network Analysis : A Handbook” , London, Sage, 2000.

- [19]Freeman, L. C., "Centrality in Networks Conceptual Clarification." , Social Networks, vol. 1, pp. 215–239, 1979.
- [20]Y. Fu, R. Xiang, M. Zhang, Y. Liu, S. Ma, "A PDD-based Searching Approach for Expert Finding In Intranet Information Management." , The 3rd Asia Information Symposium, AIRS06, vol. 4182, pp. 43–53, 2006.
- [21]Parin Shah, "Expert Finding Using Social Networking" , SJSU, Master Theses, 2009.
- [22]Nick Craswell, David Hawking, Anne-Marie Vercoustre, Peter Wilkins, "P@NOPTIC Expert: Searching for Experts not just for Documents" , CSIRO Mathematical and Information Sciences, 2001.
- [23]Christopher S. Campbell, Paul P. Maglio, Alex Cozzi, and Byron Dom, "Expertise identification using email communications" , CIKM of ACM, pp. 528–531, 2003.
- [24]Ji-Meng Chen, Jie Liu, Ya-Lou Huang, Min Lu, "Efficient top-k support documents for expert search using relationship in a social network" , ICMLC, vol. 4, pp. 1479–1484, 2011.
- [25]Industrial Software and Services – Factory Control Software, IBM, <ftp://ftp.software.ibm.com/common/ssi/pm/br/n/ets00015usen/ETS00015USEN.PDF>
- [26]Kolodner, J., "Case-Based Reasoning." , San Francisco, C.A : Maorgan Kaufmann Publishers. Inc., 1993.
- [27]Maher, M. L., Balachandran, M. B., Zhang, D. M., "Case-Based Reasoning in Design." , Lawrence Erlbaum Ass. Mahwah, New Jersey, 1995.
- [28]Watson, I., Marir, F., "Case-Based Reasoning: A Review." , The Knowledge Engineering Review, vol. 9, no. 4, 1994.
- [29]Watson, I. D., "Applying Case-Based Reasoning : Techniques for Enterprise Systems." , San Francisco, California : Morgan Kaufmann Publishers, Inc., 1997.
- [30]Marco van der Lei j, Sanjeev Goyal, "Strong Ties in a small world" , Tinbergen Institute Discussion Paper, TI 2006-008/1., 2006.
- [31]Mitchell, J. C., "Social networks and urban situations." , England: Manchester University Press, 1969.
- [32]Scott, J., "Social Network Analysis: Critical Concepts in Sociology" , New York, Routledge, 2002.
- [33]Begum S., M. U Ahmed, P. Funk, Ning Xiong, M. Folke, ""Case-Based Reasoning Systems in the Health Sciences: A Survey of Recent Trends and Developments" , IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 41 (4), vol4., pp. 421–434, 2011.

中文部分：

- [34]廖嘉偉，「前導性協同預測架構與實施系統之研究」，碩士論文，民國 91 年。
- [35]游伯龍，《行為的新境界》，1987 年。
- [36]沈靖傑，「光碟製程設備中射出成型機遠距維護與故障診斷系統初步探討」，碩士論文，民國 95 年。
- [37]巫啟台，「文件之關聯資訊萃取及其概念圖自動建構」，碩士論文，民國 91 年。
- [38]陳冠憲，「利用文件探勘技術來自動分類民眾陳情案件」，碩士論文，民國 98 年。
- [39]中央研究院中文詞知識庫小組，中文斷詞研究概況
<http://rocling.iis.sinica.edu.tw/CKIP/wordsegment.htm>
- [40]曾元顯，「關鍵字自動擷取技術與相關詞回饋」，中國圖書館學會會報，第 59 期，1997 年。
- [41]羅家德，《清華社會學講義—社會網分析》，社會科學文獻出版社，2005 年。
- [42]廖子鳳，「知識分享推薦機制應用於線上論壇社群之研究」，碩士論文，民國 98 年。
- [43]莊達人，《VLSI 製造技術》，高立出版社，民國 91 年。
- [44]曾元顯，「文件主題自動分類成效因素探討」，中國圖書館學會會報，第 66 期，2002 年。
- [45]胡瑋珊譯，Davenport, T. H., Pausak, L. 著，《知識管理》，中國生產力中心，民國 88 年。
- [46]蕭碧茹、洪振方，「案例式推理與科學教學」，物理教育季刊，第 3 期，頁 57-74，民國 89 年。
- [47]傅士哲 謝良瑜譯，Duncan J. Watts 著，《6 個人的小世界》，大塊文化出版社，2009 年。
- [48]邱亦仁 楊世偉譯，Turban, Leidner, McLean, Wetherbe 著，《資訊科技與管理》，雙葉書廊有限公司，2008 年。

附錄一：中研究平衡與料庫詞類標記集

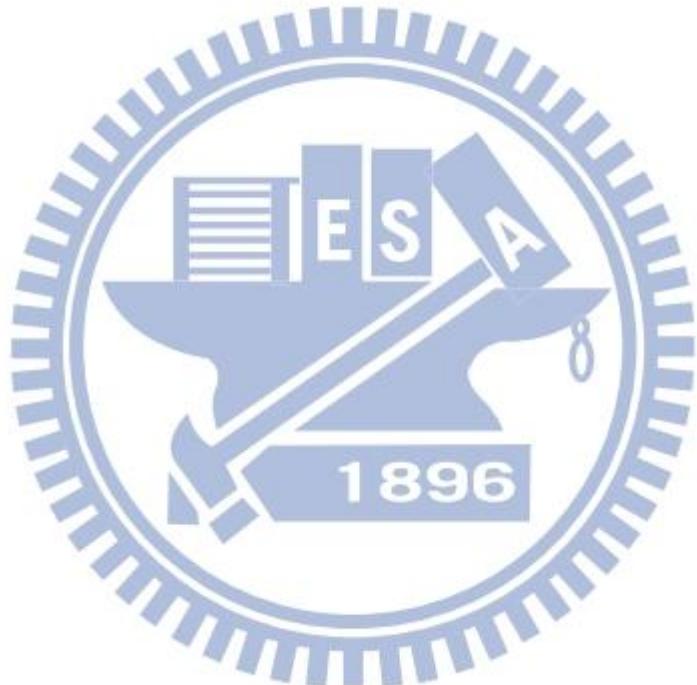
精簡詞類	簡化標記	對應的CKIP詞類標記
A	A	/*非謂形容詞*/
C	Caa	/*對等連接詞，如：和、跟*/
POST	Cab	/*連接詞，如：等等*/
POST	Cba	/*連接詞，如：的話*/
C	Cbb	/*關聯連接詞*/
ADV	Da	/*數量副詞*/
ADV	Dfa	/*動詞前程度副詞*/
ADV	Dfb	/*動詞後程度副詞*/
ASP	Di	/*時態標記*/
ADV	Dk	/*句副詞*/
ADV	D	/*副詞*/
N	Na	/*普通名詞*/
N	Nb	/*專有名稱*/
N	Nc	/*地名詞*/
N	Ncd	/*位置詞*/
N	Nd	/*時間詞*/
DET	Neu	/*數詞定詞*/
DET	Nes	/*特指定詞*/
DET	Nep	/*指代定詞*/
DET	Neqa	/*數量定詞*/
POST	Neqb	/*後置數量定詞*/
M	Nf	/*量詞*/
POST	Ng	/*後置詞*/
N	Nh	/*代名詞*/
Nv	Nv	/*名物化動詞*/
T	I	/*感嘆詞*/
P	P*	/*介詞*/
T	T	/*語助詞*/
Vi	VA	/*動作不及物動詞*/
Vt	VAC	/*動作使動動詞*/
Vi	VB	/*動作類及物動詞*/
Vt	VC	/*動作及物動詞*/
Vt	VCL	/*動作接地方賓語動詞*/
Vt	VD	/*雙賓動詞*/
Vt	VE	/*動作句賓動詞*/
Vt	VF	/*動作謂賓動詞*/
Vt	VG	/*分類動詞*/
Vi	VH	/*狀態不及物動詞*/
Vt	VHC	/*狀態使動動詞*/
Vi	VI	/*狀態類及物動詞*/
Vt	VJ	/*狀態及物動詞*/
Vt	VK	/*狀態句賓動詞*/
Vt	VL	/*狀態謂賓動詞*/
Vt	V_2	/*有*/
T	DE	/*的, 之, 得, 地*/
Vt	SHI	/*是*/
FW	FW	/*外文標記*/
COLONCATEGORY		/*冒號*/
COMMACATEGORY		/*逗號*/
DASHCATEGORY		/*破折號*/
ETCCATEGORY		/*刪節號*/
EXCLINATIONCATEGORY		/*驚嘆號*/
PARENTHESISCATEGORY		/*括弧*/
PAUSECATEGORY		/*頓號*/
PERIODCATEGORY		/*句號*/
QUESTIONCATEGORY		/*問號*/
SEMICOLONCATEGORY		/*分號*/
SPCHANGECATEGORY		/*雙直線*/

資料來源：中研院資訊所CKIP小組整理

附錄二：領域專家背景資料

	專家A	專家B	專家C
職稱	技術副理	主任工程師	資深工程師
年資	9	6	5
專長	MM	WebOPI、ALH	OPI
學歷	研究所	大學	大學

資料來源：本研究整理



附錄三：離型系統程式碼

```
//以中研院斷詞服務進行斷詞功能之函式
public string SendCKIPRequest(string sINPUTText)
{
    Socket CKIPconn = null;
    IPAddress CKIPIPAddress = null;
    IPEndPoint CKIPIPEndPoint = null;
    List<string> sCKIPResult = null;

    string sCKIPReturnValue = "";
    int iRC = 0;

    try
    {
        CKIPIPAddress = IPAddress.Parse(ConfigurationManager.AppSettings["CKIP_HOST"]);
        CKIPIPEndPoint = new IPEndPoint(CKIPIPAddress,
            Int32.Parse(ConfigurationManager.AppSettings["CKIP_PORT"]));

        CKIPconn = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);
        CKIPconn.Connect(CKIPIPEndPoint);
    }
    catch (Exception ex)
    {
        throw (new System.Exception("連結中研院斷詞服務發生錯誤。", ex));
    }

    try
    {
        string xmlString = SetXmlString(sINPUTText);
        byte[] msg = Encoding.Default.GetBytes(xmlString);
        byte[] bytes = new byte[1024];

        iRC = CKIPconn.Send(msg);

        //取得伺服器回傳之資料
        iRC = CKIPconn.Receive(bytes);

        string recieve = Encoding.Default.GetString(bytes);
        string sucessMsg = "<processstatus code=\"0\">Success</processstatus>";

        if (!recieve.Contains(sucessMsg))
        {
            string[] seperator = { ">", "</processstatus>" };
            string[] sepResult = recieve.Split(seperator, StringSplitOptions.RemoveEmptyEntries);

            throw (new System.Exception("取得斷詞結果發生錯誤：" + sepResult[2]));
        }

        sCKIPResult = ParseXML(recieve);

        string[] sCKIPTermNeedAry =
            ConfigurationManager.AppSettings["CKIP_TERM_NEED"].ToString().Split(',');
    }

    //過濾不需要的詞性資料
    string sCKIPItemList = "";
    string[] sCKIPItemAry = null;

    foreach (string s in sCKIPResult)
    {
        if (sCKIPItemList == "") sCKIPItemList = s.Replace(" ", " , ");
        else sCKIPItemList += " , " + s.Replace(" ", " , ");
    }

    sCKIPItemAry = sCKIPItemList.Split(',');
}

for (int i = 0; i < sCKIPItemAry.Length; i++)
{
    for (int j = 0; j < sCKIPTermNeedAry.Length; j++)
    {
        if (sCKIPItemAry[i].IndexOf(sCKIPTermNeedAry[j]) >= 0)
```

```

        if (sCKIPReturnValue == "") sCKIPReturnValue = sCKIPItemAry[i];
        else sCKIPReturnValue += "," + sCKIPItemAry[i];
        break;
    }
}
return sCKIPReturnValue;
}
catch (SocketException ex)
{
    throw (new System.Exception("取得斷詞結果發生錯誤：" + ex.Message));
}
finally
{
    CKIPconn.Close();
}

//計算專家間權重函式
//docInfoAry: 相似度, 案例專家列表 E1,E2,E3...
public static double GetAssociationWeight(string expertA, string expertB, string[,] docInfoAry)
{
    double associationW = 0;
    string expARelationList = "";
    string expBRelationList = "";
    string[] expRelationAry = null;
    int crCount = 0;

    for (int i = 0; i < docInfoAry.Length / 2; i++)
    {
        //取得與專家A有關係的專家清單
        if (docInfoAry[i, 1].IndexOf(expertA) >= 0)
        {
            expRelationAry = docInfoAry[i, 1].Split(',');
            for (int j = 0; j < expRelationAry.Length; j++)
            {
                if (expRelationAry[j] != expertA)
                {
                    if (expARelationList == "") expARelationList = expRelationAry[j];
                    else
                    {
                        if (expARelationList.IndexOf(expRelationAry[j]) < 0)
                            expARelationList += "," + expRelationAry[j];
                    }
                }
            }
        }

        //取得與專家B有關係的專家清單
        if (docInfoAry[i, 1].IndexOf(expertB) >= 0)
        {
            expRelationAry = docInfoAry[i, 1].Split(',');
            for (int j = 0; j < expRelationAry.Length; j++)
            {
                if (expRelationAry[j] != expertB)
                {
                    if (expBRelationList == "") expBRelationList = expRelationAry[j];
                    else
                    {
                        if (expBRelationList.IndexOf(expRelationAry[j]) < 0)
                            expBRelationList += "," + expRelationAry[j];
                    }
                }
            }
        }

        //總和專家A與專家B共同出現的文章數
        if (docInfoAry[i, 1].IndexOf(expertA) >= 0 && docInfoAry[i, 1].IndexOf(expertB) >= 0) crCount++;

    }

    int expARelCount = 0;
    int expBRelCount = 0;

    expARelCount = expARelationList.Split(',').Length;

```

```

expBRelCount = expBRelationList.Split(',').Length;
associationW = crCount / (expARelCount * expBRelCount);
return associationW;
}

//錯誤診斷模塊函式
protected void answerSButton_Click(object sender, EventArgs e)
{
    DataTable dtDocInfo = new DataTable(); //D_THESYSTEMKEY, CKIP_TERM

    string sSQL = "";
    string sWHRCmd = "";
    string alarmScript = "";

    //取得輸入案例描述之斷詞結果
    cCKIP oCKIP = new cCKIP();

    string sINPDocCKIP = "";

    try
    {
        sINPDocCKIP = oCKIP.SendCKIPRequest(errDescSTextBox.Text.ToUpper());
    }
    catch (Exception ex)
    {
        alarmScript = "alert('" + ex.Message + "');";
        ScriptManager.RegisterStartupScript(this, this.GetType(), "alarmScript", alarmScript, true);
        return;
    }

    //形成查詢指令
    sSQL = "SELECT a.D_THESYSTEMKEY, a.CKIP_TERM FROM CS_INFO a ";
    if (catSDropDownList.Text != "*")
    {
        sSQL += ", (SELECT D_THESYSTEMKEY FROM CS_CATEGORY WHERE CATEGORY_NAME = '" +
        catSDropDownList.Text + "') b ";
        sWHRCmd = "WHERE a.CATEGORY_ID = b.D_THESYSTEMKEY ";
    }
    if (funcSDropDownList.Text != "*")
    {
        sSQL += ", (SELECT D_THESYSTEMKEY FROM CS_FUNC WHERE FUNC_NAME = '" + funcSDropDownList.Text +
        "' ) c ";
        if (sWHRCmd == "") sWHRCmd = "WHERE a.FUNC_ID = c.D_THESYSTEMKEY ";
        else sWHRCmd += "AND a.FUNC_ID = c.D_THESYSTEMKEY ";
    }
    if (errCodeSTextBox.Text != "")
    {
        if (sWHRCmd == "") sWHRCmd = "WHERE ERR_CODE LIKE '%" + errCodeSTextBox.Text.ToUpper() + "%' ";
        else sWHRCmd += "AND ERR_CODE LIKE '%" + errCodeSTextBox.Text.ToUpper() + "%' ";
    }
    if (errMessageSTextBox.Text != "")
    {
        if (sWHRCmd == "") sWHRCmd = "WHERE ERR_MESSAGE LIKE '%" + errMessageSTextBox.Text.ToUpper() +
        "%' ";
        else sWHRCmd += "AND ERR_MESSAGE LIKE '%" + errMessageSTextBox.Text.ToUpper() + "%' ";
    }

    //根據設定條件取得所有文章之斷詞結果
    try
    {
        dtDocInfo = cUIOPE.GetCaseInfo("D_THESYSTEMKEY, CKIP_TERM", sSQL + sWHRCmd);
    }
    catch (Exception ex)
    {
        alarmScript = "alert('" + ex.Message + "');";
        ScriptManager.RegisterStartupScript(this, this.GetType(), "alarmScript", alarmScript, true);
        return;
    }
}

```

```

}

//根據取得之各文章斷詞內容形成向量元素
string[] sSVMItemAry = cDM.GetSVMItemByCASE(1, dtDocInfo);
//記錄各文章向量數值，最後1筆為輸入案例
double[,] sDocSVMary = new double[dtDocInfo.Rows.Count, sSVMItemAry.Length];
//記錄各文章向量數值，最後1筆為輸入案例
double[,] sINPDocSVMary = new double[1, sSVMItemAry.Length];
string[,] sDocCosineValue = new string[dtDocInfo.Rows.Count, 2]; //紀錄各文章分數，文章鍵值

//形成各文章向量
for (int i = 0; i < dtDocInfo.Rows.Count; i++)
{
    for (int j = 0; j < sSVMItemAry.Length; j++)
    {
        if (dtDocInfo.Rows[i][1].ToString().IndexOf(sSVMItemAry[i]) >= 0) sDocSVMary[i, j] = 1;
        else sDocSVMary[i, j] = 0;
    }
}

//形成輸入文章之向量
for (int j = 0; j < sSVMItemAry.Length; j++)
{
    if (sINPDocCKIP.IndexOf(sSVMItemAry[j]) >= 0) sINPDocSVMary[0, j] = 1;
    else sINPDocSVMary[0, j] = 0;
}

//計算各文章相似度
sDocCosineValue = cDM.GetDocCOSValue(0, dtDocInfo, sDocSVMary, sINPDocSVMary);

//從大到小排序
string sTmpCOSValue = "";
string sTmpKEYValue = "";

for (int i = (sDocCosineValue.Length / 2) - 1; i > 0; i--)
{
    for (int j = 0; j < i; j++)
    {
        if (double.Parse(sDocCosineValue[j, 0]) < double.Parse(sDocCosineValue[j + 1, 0]))
        {
            sTmpCOSValue = sDocCosineValue[j, 0];
            sTmpKEYValue = sDocCosineValue[j, 1];

            sDocCosineValue[j, 0] = sDocCosineValue[j + 1, 0];
            sDocCosineValue[j, 1] = sDocCosineValue[j + 1, 1];

            sDocCosineValue[j + 1, 0] = sTmpCOSValue;
            sDocCosineValue[j + 1, 1] = sTmpKEYValue;
        }
    }
}

//根據使用者輸入條件，取前K筆近似案例
string sCASEKeyList = "";
string sCASECosValueListForEXP = "";
string sCASEKeyListForEXP = "";

int needSupportDocNum = 0;
needSupportDocNum = Int32.Parse(topKSDropDownList.Text);

if ((sDocCosineValue.Length / 2) > needSupportDocNum)
{
    for (int i = 0; i < sDocCosineValue.Length / 2; i++)
    {
        if (i < needSupportDocNum && double.Parse(sDocCosineValue[i, 0]) > 0)
        {
            if (sCASEKeyList == "") sCASEKeyList = "" + sDocCosineValue[i, 1] + "";
            else sCASEKeyList += "," + sDocCosineValue[i, 1] + "";

            if (sCASECosValueListForEXP == "") sCASECosValueListForEXP = sDocCosineValue[i, 0];
            else sCASECosValueListForEXP += "," + sDocCosineValue[i, 0];

            if (sCASEKeyListForEXP == "") sCASEKeyListForEXP = sDocCosineValue[i, 1];
            else sCASEKeyListForEXP += "," + sDocCosineValue[i, 1];
        }
    }
}

```

```

    else
    {
        for (int i = 0; i < sDocCosineValue.Length / 2; i++)
        {
            if (double.Parse(sDocCosineValue[i, 0]) > 0)
            {
                if (sCASEKeyList == "") sCASEKeyList = "" + sDocCosineValue[i, 1] + "";
                else sCASEKeyList += "," + sDocCosineValue[i, 1] + "";

                if (sCASECosValueListForEXP == "") sCASECosValueListForEXP = sDocCosineValue[i, 0];
                else sCASECosValueListForEXP += "," + sDocCosineValue[i, 0];

                if (sCASEKeyListForEXP == "") sCASEKeyListForEXP = sDocCosineValue[i, 1];
                else sCASEKeyListForEXP += "," + sDocCosineValue[i, 1];
            }
        }
    }

    DataTable dtTOPKCaseInfo = new DataTable();

    sSQL = "SELECT " +
        "c.D_THESYSTEMKEY, a.CATEGORY_NAME, b.FUNC_NAME, c.ERR_CODE, c.ERR_MESSAGE,
        c.ERR_DESCRIPTION, c.LAST_UPDATE_USER, c.LAST_UPDATE_TIME, " +
        "d.PD_REASON, d.PD_SOLUTION, d.LAST_UPDATE_USER, d.LAST_UPDATE_TIME, d.PLEVEL,
        d.UPDATE_USER_HIS" +
        "FROM " +
        "CS_CATEGORY a, CS_FUNC b, CS_INFO c, CS_INFO_DETAIL d " +
        "WHERE " +
        "a.D_THESYSTEMKEY = c.CATEGORY_ID AND b.D_THESYSTEMKEY = c.FUNC_ID AND d.CASE_ID =
        c.D_THESYSTEMKEY AND d.STATUS = 'Y' AND c.D_THESYSTEMKEY IN (" + sCASEKeyList + ")";

    try
    {
        dtTOPKCaseInfo = cUIOPE.GetCaseInfo("D_THESYSTEMKEY, CATEGORY_NAME, FUNC_NAME, ERR_CODE,
        ERR_MESSAGE, ERR_DESCRIPTION, LAST_UPDATE_USER, LAST_UPDATE_TIME, PD_REASON, PD_SOLUTION,
        PD_LAST_UPDATE_USER, PD_LAST_UPDATE_TIME, PLEVEL, UPDATE_USER_HIS", sSQL);

        //將資料繫結至控制項
        cCOMM.BindGridView(ref dtTOPKCaseInfo, ref caseListInfoSGridView, "", "");

        for (int i = 0; i < caseListInfoSGridView.Rows.Count; i++)
        {
            if (i % 2 == 0)
            {
                ((HiddenField)caseListInfoSGridView.Rows[i].FindControl("caseDKEYSHiddenField")).Value =
                    dtTOPKCaseInfo.Rows[i][0].ToString();
                ((Label)caseListInfoSGridView.Rows[i].FindControl("cateSLabel")).Text =
                    dtTOPKCaseInfo.Rows[i][1].ToString();
            }
            else
            {

                ((TextBox)caseListInfoSGridView.Rows[i].FindControl("casePDRReasonSTextBox")).Text =
                    dtTOPKCaseInfo.Rows[i - 1][8].ToString();

                if (dtTOPKCaseInfo.Rows[i - 1][12].ToString() == "USER")
                    ((TextBox)caseListInfoSGridView.Rows[i].FindControl("casePDSolutionSTextBox")).Text = "請將可能發生原
                    因告知值班人員，以獲得更快速的服務。";
                else
                    ((TextBox)caseListInfoSGridView.Rows[i].FindControl("casePDSolutionSTextBox")).Text =
                        dtTOPKCaseInfo.Rows[i - 1][9].ToString();

                ((Label)caseListInfoSGridView.Rows[i].FindControl("casePDUpdUserSLabel")).Text =
                    dtTOPKCaseInfo.Rows[i - 1][10].ToString();
                ((Label)caseListInfoSGridView.Rows[i].FindControl("casePDUpdTimeSLabel")).Text =
                    dtTOPKCaseInfo.Rows[i - 1][11].ToString();

                ((HiddenField)caseListInfoSGridView.Rows[i].FindControl("casePDLvlSHiddenField")).Value =
                    dtTOPKCaseInfo.Rows[i - 1][12].ToString();
            }
        }
    }
    catch (Exception ex)
    {
        alarmScript = "alert('" + ex.Message + "');";
        ScriptManager.RegisterStartupScript(this, this.GetType(), "alarmScript", alarmScript, true);
    }
}

```

```

        return;
    }

//根據設定，判斷是否推薦專家
string[,] docInfoAry = null;
string[] docCosValueAry = null;
string[] docKeyValueAry = null;

string totalEXPList = "";
string[] docEXPListAry = null;
string[] totalEXPListAry = null;

if (expertNeedSDropDownList.Text == "Y")
{
    docCosValueAry = sCASECosValueListForEXP.Split(',');
    docKeyValueAry = sCASEKeyListForEXP.Split(',');

    docInfoAry = new string[docKeyValueAry.Length, 2];

    for (int i = 0; i < docInfoAry.Length / 2; i++)
    {
        for (int j = 0; j < dtTOPKCaseInfo.Rows.Count; j++)
        {
            if (dtTOPKCaseInfo.Rows[j][0].ToString() == docKeyValueAry[i])
            {
                docInfoAry[i, 0] = docCosValueAry[i]; //文章近似值
                docInfoAry[i, 1] = dtTOPKCaseInfo.Rows[j][13].ToString(); //專家更新文章記錄

                docEXPListAry = dtTOPKCaseInfo.Rows[j][13].ToString().Split(',');

                for (int k = 0; k < docEXPListAry.Length; k++)
                {
                    if (totalEXPList == "") totalEXPList = docEXPListAry[k];
                    else
                    {
                        if (totalEXPList.IndexOf(docEXPListAry[k]) < 0)
                            totalEXPList += "," + docEXPListAry[k];
                    }
                }
                break;
            }
        }
    }
}

totalEXPListAry = totalEXPList.Split(',');

double[] expScoreAry = new double[totalEXPListAry.Length];

int maxIterationCount =
    Int32.Parse(ConfigurationManager.AppSettings["MAX_ITERATION_CNT"].ToString());

double expCoefficient = double.Parse(expertCoefficientTextBox.Text);

double otherExpScoreSum = 0;

//計算各專家分數
for (int i = 0; i < maxIterationCount; i++)
{
    for (int j = 0; j < totalEXPListAry.Length; j++)
    {
        //取得專家初始分數
        if (i == 0)
        {
            expScoreAry[j] = cTOPKEXP.GetExpertInitialScore(totalEXPListAry[j], docInfoAry);
        }
        //取得專家複合分數
        else
        {
            expScoreAry[j] = expCoefficient * expScoreAry[j];
        }
        otherExpScoreSum = 0;

        for (int k = 0; k < totalEXPListAry.Length; k++)
        {
            if (totalEXPListAry[k] != totalEXPListAry[j])
            {

```

```

        otherExpScoreSum += expScoreAry[k] *
cTOPKEXP.GetAssociationWeight(totalEXPListAry[j], totalEXPListAry[k], docInfoAry);
    }
}

expScoreAry[j] += (1 - expCoefficient) * otherExpScoreSum;
}
}

//取最高分數專家
double highestEXPScore = 0;
string highestScoreEXP = "";

for (int i = 0; i < totalEXPListAry.Length; i++)
{
    if (i == 0)
    {
        highestEXPScore = expScoreAry[i];
        highestScoreEXP = totalEXPListAry[i];
    }
    else
    {
        if (expScoreAry[i] > highestEXPScore)
        {
            highestEXPScore = expScoreAry[i];
            highestScoreEXP = totalEXPListAry[i];
        }
    }
}
caseExpertSTextBox.Text = highestScoreEXP;
}
}

```

