# 國立交通大學

## 電機資訊學院 資訊學程

## 碩 士 論 文

基於SAO結構之中文專利文件自動摘要技術研究

**Design and Study of Automated Text Summarization for Extracting SAO Structures from Chinese Patent Documents**

研 究 生 ： 劉翰卿

指導教授 ： 楊維邦 博士

蒙以亨 博士

中 華 民 國 九 十 四 年 一 月

# 基於SAO結構之中文專利文件自動摘要技術研究

# Design and Study of Automated Text Summarization for Extracting SAO Structures from Chinese Patent Documents


研 究 生: 劉翰卿　　Student : Han-Ching Liu

指導教授: 楊維邦　博士　　Advisor : Dr. Wei-Pang Yang

蒙以亨　博士　　Dr. I-Heng Meng


國 立 交 通 大 學

電機資訊學院 資訊學程

碩 士 論 文

A Thesis

Submitted to Degree Program of Electrical Engineering and Computer Science

College of Electrical Engineering and Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Computer Science

January 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年一月

# 基於SAO結構之中文專利文件自動摘要技術研究

Design and Study of Automated Text Summarization for Extracting SAO Structures from Chinese Patent Documents

研究生：劉翰卿　　指導教授：楊維邦博士，蒙以亨博士

國立交通大學電機資訊學院資訊學程

## 摘要

自動文摘的基本精神乃是將原始文件的內容經由電子計算機的演算處理後，自動萃鍊出足資代表全文內容的精華出來，以便縮短研讀的時間，進而提升工作的效率。

本研究試圖藉由英文的主詞、動詞與受詞(Subject-Action-Object；簡稱 SAO)結構句型為基礎，藉由一系列的分析、運算、處理等過程，自動判讀出專利文獻的全文內容並且取其精髓後將之匯集成為一簡明扼要的摘要內容，讓企業研發部門、專利工程師、產業分析師或智權人員毋需詳閱艱澀難懂的專利全文，便可快速掌握到專利文獻所欲描述之概念，以加速取得目標資訊。

在雛型系統實驗中，我們以十六篇攸關電子商務領域的專利文獻為實驗素材，將 SAO 結構句的概念應用於中文專利文獻摘要的擷取上。經效益評估後的結果顯示，我們所設計的概念(Concepts)及 SAO 結構句的擷取演算都有還不錯的表現。以整體平均來說，概念(Concepts) 擷取方面的召回率為 95.34%，準確率為 92.13%；而 SAO 結構句組擷取方面的召回率則為 92.45 %，準確率為 93.79%。

**關鍵字：自動摘要技術、中文專利文獻、SAO 結構、經驗法則**

# Design and Study of Automated Text Summarization for Extracting SAO Structures from Chinese Patent Documents

Student : Han-Ching Liu Advisor : Dr. Wei-Pang Yang, Dr. I-Heng Meng

Program of Electrical Engineering Computer Science

College of Electrical Engineering and Computer Science

National Chiao Tung University

## ABSTRACT

The basic idea of automated text summarization is that distilling the most important information from a source to produce an abridged version, in order to shorten the time to understand the original source and then improve the efficiency of the work.In this thesis, the research attempts to extract SAO Structures from a Chinese patent document based on the basic sentence patterns of English, one of which, for example, is Subject, Verb, and Object (namely, Subject-Action-Object; abbreviated as SAO). With a series of analysis, operation, complicated process,… etc., we could create a brief and concise summary for the document. In the experiment of the prototype, we use sixteen Chinese patent documents of e-commerce -related field as the experiment material, and apply the concept of one of SAO structures to the picking and fetching of the Chinese text summarization. The results which were evaluated seems to be satisfactory. On average, it is 95.34% recalling rate of extracting the aspects of the concepts and the rate of accuracy is 92.13%. In addition, average recalls of 92.45% and average precision of 93.79% were achieved respectively in SAOs.

**Keyword：Automated Text Summarization, Chinese Patent Documents, SAO, Subject-Action-Object, Heuristic Rules.**

# 誌謝

近代極富盛名的學者王國維先生在其《人間詞話》一書中曾云：古今之成大事業、大學問者，必經過三種之境界：『*昨夜西風凋碧樹，獨上高樓，望盡天涯路*』，此第一境也。『*衣帶漸寬終不悔， 為伊消得人憔悴*』，此第二境也。『*眾裏尋他千百度，回頭驀見，那人正在燈火闌珊處*』，此第三境也。

近一千個日子的步步為營，不寒不暑，不沉不淪，在交大電機資訊學院資訊學程碩士專班修習過程當中，也深歷了這三種奇妙神境。箇中滋味，正如人飲水，冷暖自知。工作與學業交融的辛酸血淚以及成長的歡樂喜悅，其實只有自己最能夠體會，此等刻劃非過來人不能道矣。從資策會長官們的提攜、核准進修的那一刻起，心中就懷抱著『望盡天涯路』那樣地雄心壯志，欲窺交大這一學術大觀園的堂奧及其點點滴滴。默默地告訴自己要靜下心來忍住那『昨夜西風凋碧樹』的淒涼以及『獨上高樓』的孤寂，要耐得住世間一切的誘惑，以真功夫、苦功夫、細功夫埋首通讀、努力付出、盡心鑽研；以百折不撓的精神，在浩瀚的學術宇宙中努力不懈地“尋它千百度”，即便是“衣帶漸寬”也無怨無悔！“為此消得人憔悴”也心甘情願！只希望在“燈火闌珊處”、不經意地“驀然回首”的當下能夠頓悟出做學術研究的科學真諦。

“在職進修”——一個令自己深感疲累與鼻酸的代名詞，不容否認，的的確確是蠻辛苦的一種蛻變歷程。在每週兩、三回〝台北－新竹〞兩頭往返穿流通勤的奔波中，動輒來回一趟車程就要蹉跎我近四到六個小時的光陰歲月。在事業與課業、體力與耐力以及親情、友情與愛情等諸多考驗不斷地矛盾衝突與相互衝擊的掙扎中，也粹鍊了我一身的堅強與果斷，潛移默化中造就了我“凡事捨得”、“凡事盼望”、“凡事相信”、“凡事包容”、“凡事忍耐”的堅定信仰。

從小到大，上圖書館的次數其實是屈指可數的。儘管如此，我仍是認真向學的，只

因『交大浩然數位圖書館』的存在，總能讓我隨心所欲"*Any Time*、*Any Where*"地〝予取予求〞。有了它，真好！感謝交大一級名師楊維邦教授的愛心與體恤，讓不才的我有個學術研究的避風港得以依靠，並惠賜一位優越的學長蒙以亨博士來就近指導、督促我，使我免於舟車奔波之苦，並給予我無上的啟蒙。從初探的『無所不在的運算環境(*Pervasive Computing/Ubiquitous Computing Environments*)』開始，以至於現今的『中文專利文獻自動摘要技術研究』，兩位指導老師〝*降龍十八掌*〞的學術功力，自不在話下；清新的思路，一絲不苟、絕對嚴謹的學術研究態度與精神，總能讓喜歡恣意遐想、迷迷糊糊、非相關科系的我甘拜下風，耳濡目染下著實也領受了不少的啟發。同時，也非常謝謝資料庫實驗室柯皓仁教授、葉鎮源學長友情的客串指導以及校外口試委員黃明居老師細心的指點迷津，還有資策會電子商務研究所資源的鼎力協助。此外，也深深感謝父、母親的劬勞與關懷，讓我在對課業與工作心灰意冷時有了向上的動力與衝勁；還有岳父、岳母的大膽假設，願意將『*我的野蠻女友*』——*秀卿*在進修期間交託給一個一事無成的我來加以〝*馴服*〞，讓她變成一個溫柔婉約又可愛的『*美麗吾妻*』，只可惜這段期間無法履行對她的承諾：帶她到處遊山玩水、周遊列國而深感歉意。

總是喜歡在上課前到交大竹湖邊坐著小憩一番，一邊聆聽著音樂、啜著一抹綠茶，一邊欣賞著湖波盪漾、大小魚兒自由自在悠游其中、以及無憂無慮的綠頭雁鴨肆無忌憚的追逐與嬉戲，感覺這一刻，彷如置身在愛麗絲夢遊仙境般，令人心曠神怡，忘卻了人世間一切的紛紛擾擾，再多的煩惱與憂愁也隨著此情此景而蒸發人間！套用好友阿吉常說的：『交大，人傑地靈，真的是一個非常不錯唸書做學問的好地方。』

雖然這輩子註定永遠與奧運金牌絕緣，亦無法成為諾貝爾獎的得主；但，對我而言，須感謝的人、事、物仍實實在在有許許多多，無法一語道盡，細說分明。只好*謝天*、*謝地*、*謝謝自己*。*感謝有你*，*感謝上帝*，*感謝主*。讓我喜歡真理，不自誇，對愛依然是永不止息。

2005 乙酉年 自由日

# 目錄

# 表目錄

# 圖目錄

# 方程式目錄



# 第一章　緒論

## 第一節　研究背景

對一個企業來說，『專利』乃是一種無形的資產，一家公司擁有的專利數愈多，意謂著該公司的智慧財產也愈多。運用專利為籌碼，可迫使同業無法進入相同之領域與之相抗衡。在這知識經濟與智慧財產權掛帥的今日，『專利』儼然已成為一種生存競爭的遊戲，只要符合參賽資格且遵循遊戲規則者，即使起步較緩或是資源有限，依然有機會可以成為最後的贏家。但申請專利過程的第一步就是得要去查證你的創意、心血結晶是否早已是他人專屬的權利，以免誤觸了法網，侵權而不自知、徒勞而無功。

根據世界智慧財產權組織(WIPO)的調查指出：在專利文獻中可以查考全世界每年90-95%的發明成果；不僅如此，在研究工作中若先行查閱專利文獻不但可以縮短60%的研發時間，還可以節省高達40%的研究經費。因此，閱讀與分析專利文獻勢必就成為極其重要且不可或缺的一項工作！若能善加利用專利文獻，不但可以吸收滿滿具創新前瞻性的技術資訊，也可以從中獲取不少極具商業價值的競爭情報[32]。所幸，拜由網際網路之賜，目前已通過的專利文獻在各國相關專利管理機構皆有提供完整的網際網路檢索功能；可惜的是，透過關鍵字詞查詢專利後的結果往往還需要以人工的方式再去過濾這些專利文獻，以查證目前的創意點子是否與已通過的專利有所衝突。

目前，全世界所發行的專利仍不斷地以驚人的速度持續的成長中，每個領域需要監控的專利數量也因此而大幅增加。由於專利文獻獨特的文法結構以及特定的遣辭用語與一般的文章大相逕庭，其晦澀拗口、難以閱讀乃是不爭的事實。因此，當找到的專利文獻資料篇幅太過於冗長或是專利分析師不想逐字閱讀專利的全文內容時，如何透過精簡的方式迅速且經濟的來掌握這篇專利文獻之精華，以減少專利分析師或研發工程師的閱

讀時間，在這個分秒必爭的時代，便成為一個不容忽視的重要課題。

## 第二節　研究動機

『科技始終來自於人性』。科技的研發，相關機制的創新與設計，都不能自外於人類的基本需求，尤其是在這資訊急速爆炸的時代更是如此。在分秒必爭繁忙的工商業社會中，若能藉由簡明扼要的摘要內容來縮短研讀的時間，免於閱讀篇幅很長的文章內容，讓讀者不費吹灰之力即可知曉全文之意涵，快速而又有效率地吸取原始文件內容之菁華，不曉得該有多麼地好呀？想法很好，但問題是：該要如何透過電腦機器來幫我們自動判讀相關知識庫中的原始文件內容，以去蕪存菁的方式來產生摘要呢？而產出的摘要內容又當如何確知已涵蓋原始文件所要表達的意涵而不至南轅北轍呢？

學者 Pinto Molina(1995)認為人類產生摘要的歷程可以分為四大階段(如圖 1所示)，分別是：解譯(Interpretation)、挑選(Selection)、再詮譯(Re-interpretation)以及綜合(Synthesis)等 [13]。

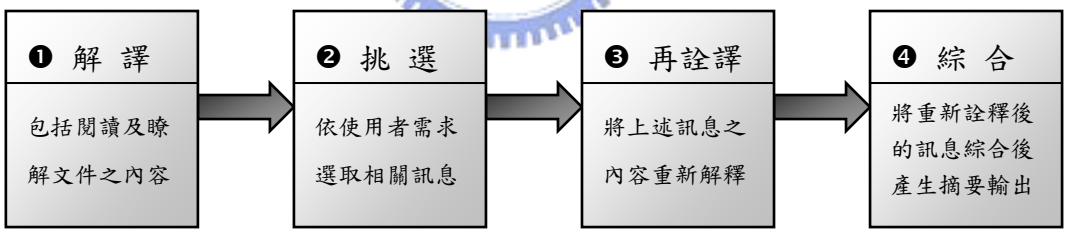


圖 1：人類產生摘要的四大階段

也就是說，對一般人而言，我們在撰寫摘要的過程當中，通常必須先對原始文件的內容進行所謂全文式的閱讀、分析並且在瞭解文章的語意內容後，才能夠對原文的內容重新給予詮釋進而產生所謂的摘要(Abstract)結果輸出。換言之，撰寫摘要的過程就是要想辦法透過一套邏輯的思維去除較為瑣碎、不重要的訊息，而保留比較重要的訊息內容，以提供相關使用者來閱讀，以快速吸收原文之意涵，見微知著。而這種人為產生摘要的方式，如無太大之意外，通常都是採取重新編寫(Re-writing)的方式來進行，也就是由摘要

撰寫者依據其個人主觀的價值經驗以及判斷，將理解過後的文件內容重新予以詮釋並加以組織後，以不同於原始文章內容的方式來加以改寫。在自動化摘要方法的研究當中，依據摘要產出的形式不同，通常我們可以將自動化摘要的方法概分為兩大類：第一種方法稱為重新詮釋法：從瞭解文章內容開始，依照個人本身的背景、經驗以及主、客觀的價值判斷來進行詮釋，進而重新編寫摘要，最後將原文的內容以較精簡的模式來產出，此種方法所產生的摘要我們謂之為『摘述(Abstraction)』；而另一種較為簡易的方法則稱之為複製剪貼法，其概念乃是基於某一種演算法來衡量不同段落、語句或是關鍵詞彙所映射出來的重要性差異並據此來決定其順序，然後再依序抽取一定比例的句子作為摘要，這種摘要我們謂之為『節錄(Extraction)』[4][5][10][11][12][18][19][23]。

有鑑於此，我們在此提出一個自動摘要的方法，讓簡短的摘要內容可以用來代表某一篇專利文獻的內容，甚而取代原來的專利全文。透過這種見微知著的方式，以減少專利分析師瀏覽以及閱讀的時間。本研究企圖透過前人研究的智慧結晶，嘗試以英文語言SAO(Subject-Action-Object)結構的特性，應用於中文專利文獻的摘要擷取上，最後並將之運用於本實驗的雛型系統中。

## 第三節　研究目的

文件自動化摘要的基本精神乃是將原始文件的內容透過電腦計算機的運算處理後，自動萃鍊出足資代表全文內容的精華出來。因此，本研究乃從解讀專利說明書中最重要的權利宣告部份(即“申請專利範圍”；Claims)開始，分解其構成要件，並釐清各要件之間的組合關係，將晦澀拗口、難以閱讀的專利文獻，轉換成簡明易讀的摘要內容，以提供企業研發部門、專利工程師、產業分析師或智權人員一目了然的專利資訊，縮短研讀的時間，進而提升其工作效率。

然而，無庸置疑的，一個言簡意賅的自動文件摘要技術應該是要能夠在理解原始文

件的內容後，建構出足以代表該原始文件所要表達的知識意涵模型，以便透過該知識模型來生成最後的摘要結果才是 [27]。因此，本研究為了要迅速且正確地萃取出中文專利文獻當中的精華內容，我們需要藉助外在工具以便於在短時間內可以理解該專利所隱含的意義。首先，先擷取出申請專利範圍(Claims) 的資料，然後透過概念(Concepts) 擷取技術去萃取出此申請專利範圍(Claims) 中重要的概念(Concepts)，然後利用SAO(Subject-Action-Object) 的句型結構設法將概念(Concept)、以及概念與概念之間的關聯(Relation)串接起來。之後，利用概念(Concepts)與概念(Concepts)之間的統計共現矩陣來判斷概念(Concepts)間的語意關聯強度。此後，我們再根據一些組合規則(Rules) 將之合成，便可以完成代表此篇專利文獻的摘要內容(Summarization)出來 (如圖 2所示)。

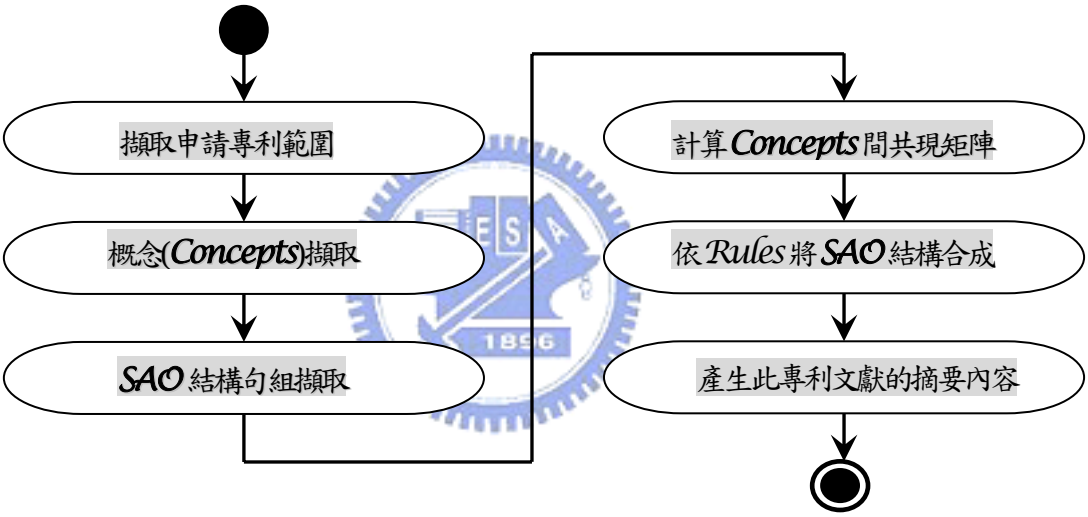


圖 2：本研究摘要產出之主要流程

## 第四節　研究範圍與限制

本研究係與資策會共同合作之創新前瞻技術之研究，所有據以實驗用之專利文獻之取得，皆係由資策會統籌提供，以作為本研究實驗素材之來源。因不同領域的專利文獻，其技術語句、專業文法及語意剖析可能會有所差異；故本研究之實驗範圍乃是以電子商務(e-Commerce) 相關領域(包含軟體)之專利文獻為主要實驗對象。

而關於中文斷詞切字方面，將直接採用中央研究院中文詞庫小組所研發的『CKIP

中文自動斷詞系統1.0版』來處理。該工具除了具有中文自動斷詞的功能外，更可以標示每個字或詞的中文詞類；同時，CKIP 也允許使用者根據自己的需求選擇不同的詞典，作為斷詞與及詞性標記的參考。

## 第五節　研究流程及論文架構

本論文的研究架構如圖 3所示，共分為六章。首先在第一章透過簡介形式來概略描述本研究之背景、動機、目的以及研究的範圍與限制，隨後在第二章深入探討自動文件摘要的相關研究及文獻資料（包括自動化資訊摘要、淺層摘要研究取向、深層摘要研究取向等），並將相關論述加以歸納並整合之。第三章及第四章則詳細描述了本研究的系統架構，經由相關工具及方法的使用，構建模式，並實作一雛型系統；之後在第五章處，進行實驗結果的解釋、剖析與評估，探討此模式在實務上應用的機會與限制，並藉自動化資訊摘要評估模式的探討來驗證本研究方法的可行性。最後，第六章針對研究的過程與實驗的結果做一總結及心得分享，並進一步描述未來可行的研究方向。

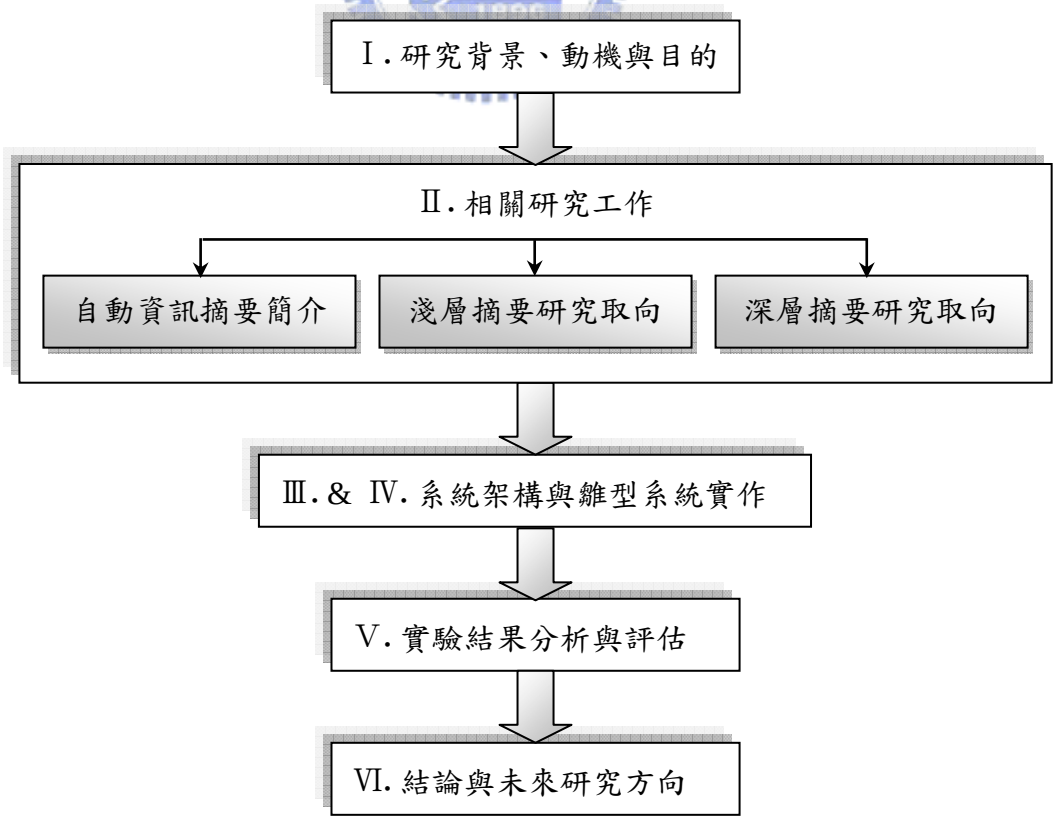


圖 3：研究流程示意圖

# 第二章　相關研究工作

## 第一節　自動化資訊摘要概述

自動化資訊摘要的研究已將近有50年的歷史了，是一種融合了自然語言處理、資訊檢索、圖書資訊學、統計學、認知心理學和人工智慧等多學門的綜合應用 [12]。近年來隨著計算語言學(Computational Linguistics)理論的興起，自動摘要又再度成為眾所矚目的熱門研究焦點。

英國劍橋大學的K. Spärck Jones (Jones 1995)率先把自動文摘的處理架構歸納成為三個階段( The framework for summarization in terms of the three-phase architecture )，分別是：解譯(Interpretation)、轉換(Transformation)、產生(Generation) (如圖 4所示)。


圖 4：自動文摘處理架構的三階段(Spärck Jones(1995))

也就是說，將原始文字的內容透過某一種演算法先把它**解譯**成為某一種形式的表達，再將此原始的表達透過另一種演算法**轉換**成摘要形式的表達，最後再將此摘要形式的表達透過某一種演算法**產生**出恰當的文字摘要 [21]。而 I. Mani 及 M. Maybury (1999)等人則依據上述Spärck Jones(1995)的摘要架構重新將自動摘要系統予以詮釋，將自動化文件摘要的處理過程概分為三大階段，依序為：分析(Analysis)、轉換(Transformation)、合成(Synthesis)等(如圖 5所示) [12][18]。


圖 5：自動化文件摘要處理過程的三大階段(I. Mani & M. Maybury (1999))

首先是依據某種重要的特徵(Salient Features)來『分析原始文件』(*Analysis*：Analyze the input and build an internal representation of it.)；接著將分析的結果轉換為系統內部的摘要表示法(*Transformation* 有時也稱之為 *Refinement*： Transform the internal representation into a representation of the summary.)；最後是透過相關的演算法權衡內部摘要表示法的相對重要性，挑選重要性較高的表示法來合成摘要的格式後輸出(*Synthesis*：The summary representation is rendered back into natural language.) (如圖 6所示) [12][18]。

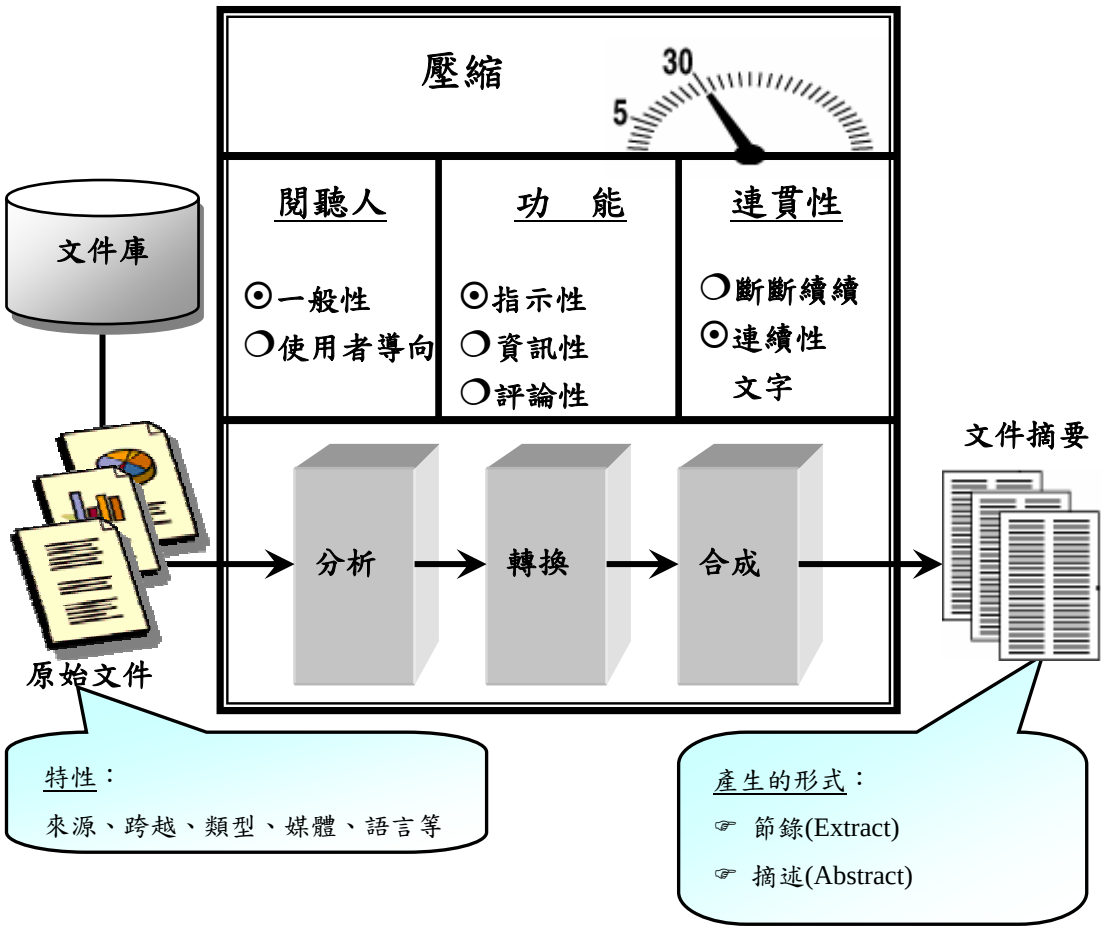


圖 6：自動摘要系統的架構概觀

在上圖 6中，我們可以看出有幾好個不同的重要參數會來影響到自動摘要系統的設計，如文件摘要的壓縮率(Compression rate)、原始資料的媒體(Media)性質、讀者(Audience)的角色、所欲達成之功能(Function)、語言(Language)、原始文件數量的多寡(Span)、摘

要產生的形式(Genre)或者是構成摘要的參考來源(Relation to source)等，Mani & Maybury (1999)；Sparck-Jones(1999)；Hovy(2001)等人都曾深入討論過。

Mani & Maybury(1999)曾替文件摘要(Text Summarization)作了如下之定義：

The process of distilling the most important information from a source (or sources) to produce an abridged version for a particular user (or users) and task (or tasks).

> 所謂的『文摘』就是從來源文件當中萃取出最重要資訊的一種過程，並依照特定使用者的特殊需求或者是應用系統所欲達成之功能要求來產生一個忠於原始文件內容的精緻版本。

根據上述之定義，我們可將摘要產出的歷程圖解如下(如圖 7 所示) [20]：

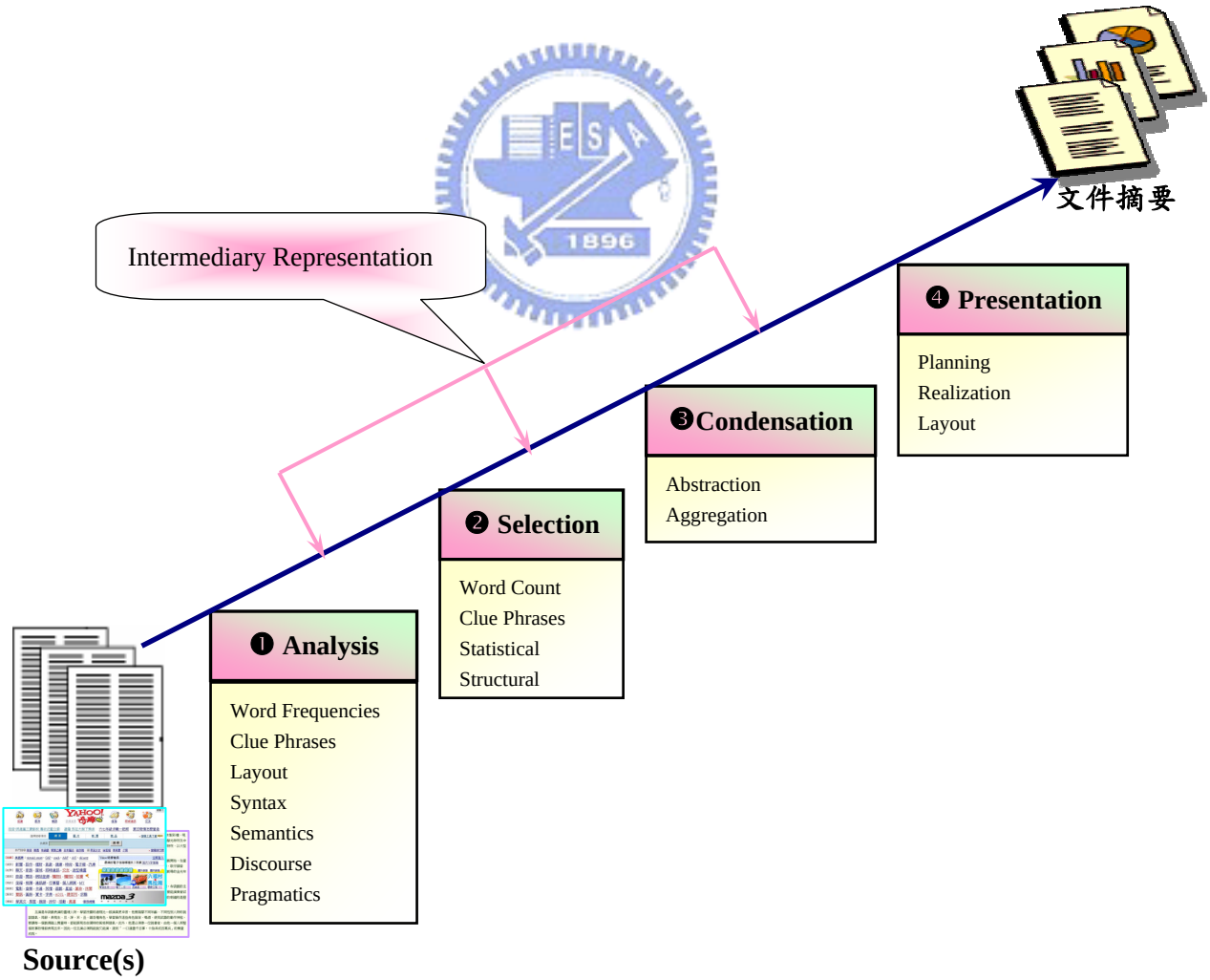


圖 7：文件之摘要產出流程(Summarization Processes)

也就是說，在自動文摘的處理過程中，我們可以透過Selection、Aggregation、Generalization等三大基本的操作處理(Condensation operations) 來使之實現(Mani & Maybury(1999)；Paice(1981)等人) [12]。其中，

- 揀選(*Selection*：filtering of elements)：

  依據某種顯著性的重要特徵來加以演算，權衡單位元素(例如：字、詞、句子、段落等)之間的相對重要性，選取重要性較高且未重複的資訊出來。

- 聚合(*Aggregation*：merging of elements)：

  將不同語言描述或者是來源文件當中的不同部份之資訊加以融合、組織起來。

- 泛化(*Generalization*：substitution of elements with more general/abstract ones)：

  用更一般化或是更抽象化的概念來替換單位元素；也就是說，以抽象、廣泛的上位概念來含括較為具體的下位概念。比如說：火車、轎車、摩托車、腳踏車等皆以“交通工具”來替代。

2.1.2 兩大類摘要研究取向(Summarization Approaches)

自動摘要方法的研究從1950年代迄今已將近有半個世紀的歷史了，無論是使用的方法、應用的領域、評估的方式等等皆有一定的研究水準與成果發表。藉著前人研究的智慧結晶，也促進了各式各樣、匠心獨具的精湛方法源源不斷地傾流而出。

我們可以把某一種語言文字想像成是一種由多維度構建而成的空間，稱之為『語言空間』(Linguistic Space)。透過元素(Elements)、層次(Levels)及位置(Position)等三個面向可以形塑這語言或文字的三維空間，如圖 8所示[12]。其中，

- 元素(Elements)：乃是以字詞(Word)、片語(Phrase)、子句(Clause)、句子(Sentence)、

段落(Paragraph)、文件(Document)等作為運算操作處理的基本單位。

- 層次(Levels)：可將上述之元素依照深淺程度不同之層次進行語言的分析。一般而言，依層次由淺到深的程度可將之區分為四種，分別是：

  ❶ 構詞(Morphological/Word)解析

  ❷ 句法(Syntactic)解析

  ❸ 語意(Semantic)解析

  ❹ 話語(Discourse/Pragmatic)解析

  等不同深度的層次。而在上述自動文摘處理過程的三大階段中，其中分析(Analysis)階段可以視作是一種由淺到深的處理過程(亦即，朝向更多的語意及語段分析)；而合成(Synthesis)階段則與分析階段的方向恰好相反。

- 位置(Position)：可以反映出元素在來源文件中的順序。

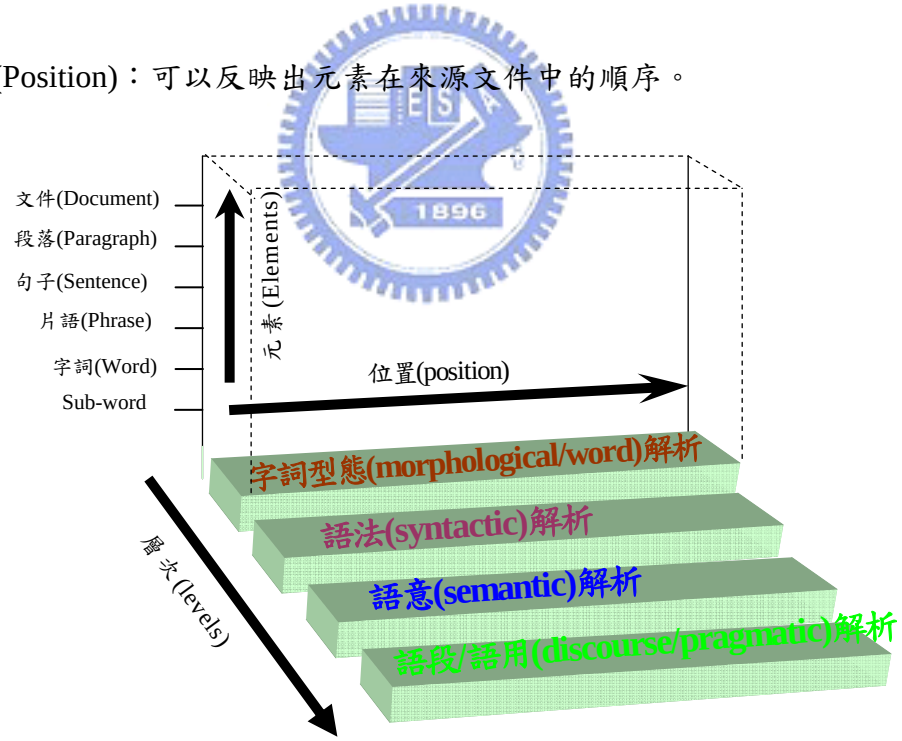


圖 8：語言空間(The linguistic space)

同一種語言現象，研究的出發點不同，往往會得出不同的結果。『語法學』研究句法結構關係，著重於描寫語法之特徵；『修辭學』研究表達之效果，著重分析倒裝等之

效用；而『語用學』研究的則是制約語言使用的各種現象。因此，我們可以透過這種『語言空間』(The linguistic space) 的層次(Level) 概念，把摘要研究的取向(Summarization Approaches)區分為兩大類：一為淺層的研究取向(Shallower Approaches)，另一則為深層的研究取向(Deeper Approaches) [12][19]。但此種劃分方式絕不是一種決然的楚河漢界，隨著技術不斷地推陳出新，現在已有愈來愈多的研究融合了這兩大取向所述的方法(Hybrid Approaches)以及原則運用於自動摘要的系統上。以下，我們依照年份的先後對於上述兩大類研究取向做了一些經典相關研究工作的整理(如圖 9所示) [1][2][4][5][6][8][9][11][18][19][20]。

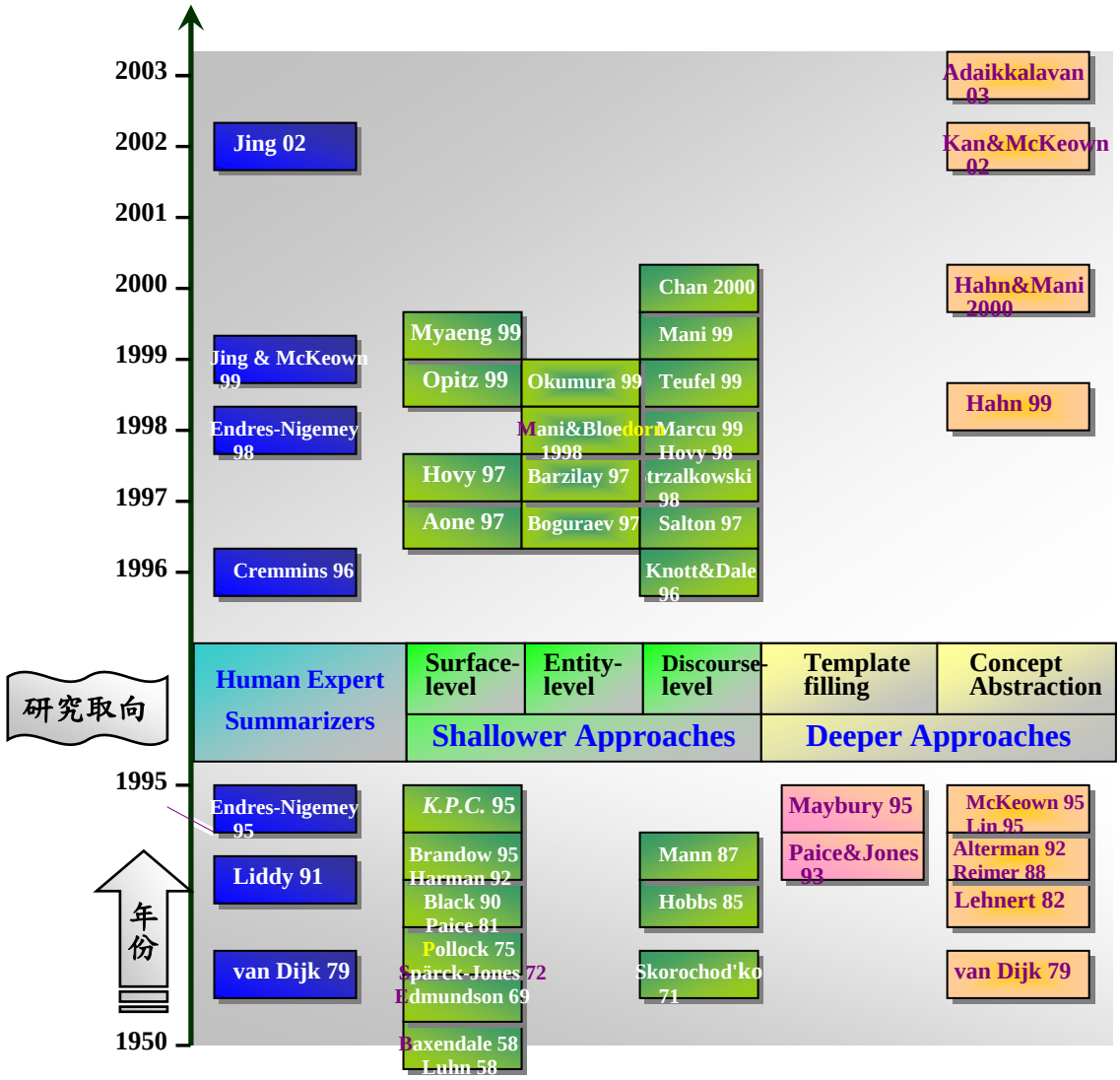


圖 9：一些經典的自動文摘相關研究整理

接下來，我們將針對兩大研究取向中的重要概念做選擇性的說明：首先，將在第二節處先來介紹淺層研究取向(Shallower Approaches)之方法及其運用，隨後在第三節的地方再來簡介一下深層研究取向(Deeper Approaches)的概念及其運用。

## 第二節　淺層摘要研究取向(Shallower Approaches)

淺層摘要的研究取向(Shallower Approaches)乃是依據某種淺顯易懂的實體特徵作為分析之依據。所謂的淺顯易懂的實體特徵(Shallow Physical Features) 可以是線索字(Cue Words)、關鍵詞(Keyword)、主題特徵(Thematic Features)、背景特徵(Background Features)、語句位於文件中的位置(Location)或是提示片語(Cue Phrases)等等。之後，藉由某種有效的演算法來權衡語句之相對重要性，以選出具關鍵性的語句(Sentence Extraction)，然後再利用剪貼(Cut-And-Paste)的技巧，將語句重新予以排列，進而組成摘錄(Extracts)後輸出(如圖 10所示) [6][8][12][14][15][18][19]。

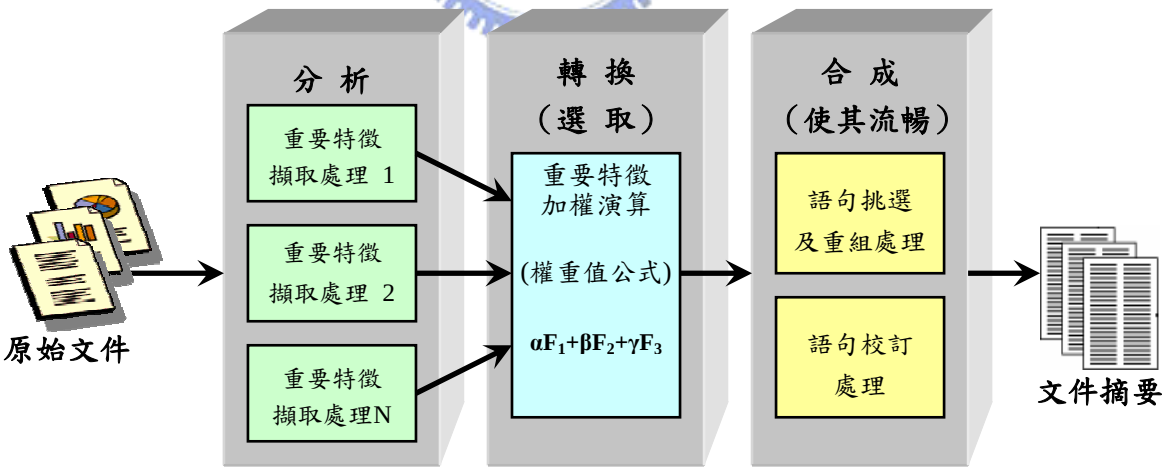


圖 10：淺層摘要的研究取向(Shallower Approaches)之架構

茲將此類作法的執行機制圖解如下(如圖 11所示)： [19][27]

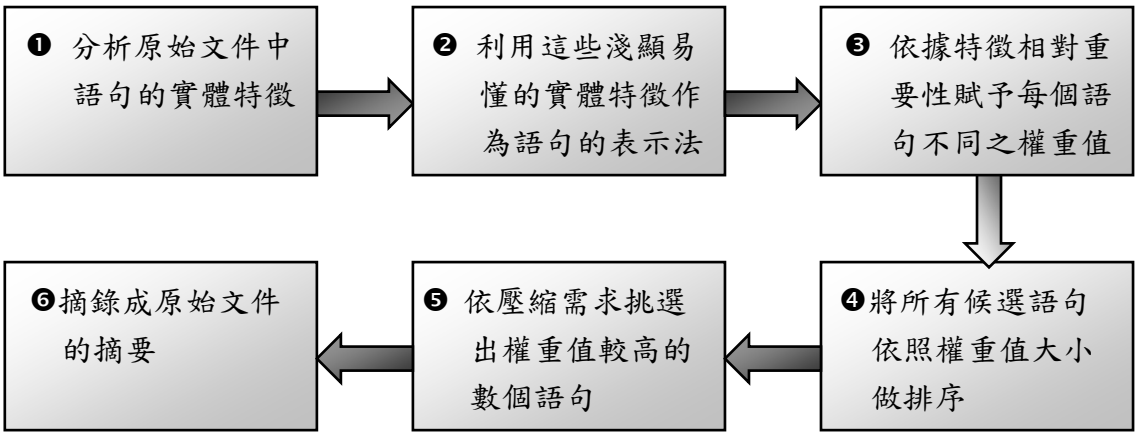


圖 11：淺層摘要的研究取向(Shallower Approaches)之執行歷程

通常這類的研究方法與所應用之領域較無關聯(Domain-Independent)，意即這是一種所謂的〝Knowledge-Poor〞(Very General-Purpose：通用性)的方法。語句(Sentences)通常至多也只解析至語法層(Syntactic Level)而已。由於透過這種節錄式節錄出的語句在銜接組合後極有可能會與原文的本意發生了脫節，產生了風馬牛不相及的文摘內容出來，所以我們依據 I. Mani 及 M. Maybury (1999)等人三階段的自動文摘處理架構來審視，可能需做一些適度的修正與調整：其中，在轉換(Transformation)階段需包含選取顯著而重要的單元(Salient Units)出來；而在合成(Synthesis)階段則需考量內容的流暢度(Smoothing)，修正不連貫的敘述，藉由語句的重新排列，使文摘更加地簡潔、易懂 [12][19]。

總而言之，此類取向之研究方法乃是採用與人類專業文摘作者極其類似的淺顯特徵(Shallow Features)作為編輯摘要的線索，是一種花費較為低廉的自動摘要解決方案，系統程式易於建構，所建構出的摘要系統也比較穩健，並且還可以使用語料庫(Corpus)等來加以訓練。只是這些方法所產生出的摘要內容較為貧乏，因它僅僅是透過某些特定且層次較低的特徵(Shallow Features)來加以分析、處理，進而建構出統計之模型來進行決策。然而，正因為未實際考慮到較高層次的語意分析，如知識概念(Knowledge Concepts)等課題，因此，節錄可能無法真實反映到文件內容的基本精髓[12][19][27]。在附錄二中，我們將淺層摘要研究取向(Shallower Approaches)依照實體參考特徵的深淺程度不同而區分為表層(Surface-Level)取向、個體元素層(Entity-Level)取向、以及語段層(Discourse-

Level) 取向等三種方法[18]。

### 2.2.1 重要的參考實體特徵(Shallow Heuristics)舉隅

目前關於華語文的研究其實都脫離不了對中文字詞方面的探索。其中攸關於字詞方面的相關性研究——“字與字間”、“字與詞間”以及“詞與詞間”等等之間的相關性可以歸結為現代華語文的Markov特徵，是揭示現代中文內在規律的重要途徑之一。語言本身可以說是一種習慣性的系統，也是一種少數服從多數的統計學原則。其中有許許多多是有章法可循的規律或是道理，比如說：詞語創造的原則及其構成方式等都是約定俗成的；然而這當中卻也充斥著不少既沒道理、亦無跡可循的例外——一些強制性的積非成是的習慣或是語言事實，例如：唾手可得 vs. 垂手可得。可是，如果我們從數理的角度運用統計學的方法出發，就會發現在這些語言事實中，不管是“規律”還是“例外”，都可能會符合一種統計學上的規律——藉由“字詞相關性”的統計，找出字與字、字與詞、詞與詞之間是否經常在一起出現的通則。若再將其推而廣之，就可以發現中文文件中的“詞法”、“句法”甚至“章法”的結構與組織規律了[8][25]。底下，我們依據圖 9所列的參考文獻以及 [8][23][26][27][30]，嘗試歸納出這近五十年來學者們對於自動摘要研究此課題中攸關語句重要性判斷的關鍵特徵，分別闡述如下：

- 主題特徵(Thematic features)：[4][9][18][19]

所謂的『主題特徵』乃是在文件當中具有重要作用的專業詞彙，可用以表達某種明確概念的關鍵字詞(或稱主題詞)。所以，主題詞乃是在組成一篇文章的單字當中，最能夠用來表達該文章意義的重要詞語。而在文件當中若包含了相對多數主題詞的語句，我們就稱該語句為主題句——可用以代表一個段落或是文章的最重要句子之一。一般來說，計算句子權重的方法大部分皆採用了詞頻統計(Term-frequency Statistics)的方法來做分析，若在一篇文件當中某個關鍵字或詞重複(Repetition)出現許多次，超過了某一閾限值

或門檻值(Threshold)，達到了統計上顯著的差異水準，那麼這個關鍵字、詞極可能就是這篇文件的主題。因此，若某個語句擁有愈多的主題特徵，那麼此語句越有可能入選而成為摘要內容之一。

一個極具代表性、最常用於計算字彙頻率與文件重要性的演算法為 TF*IDF演算法，將關鍵詞彙與文件相關程度作內積相乘，以此來評判相關程度之高低。TF*IDF的公式後來也有了許許多多的變種，其基本量測演算想法如下(如方程式 1 所示)，每個向量分量TF*IDF(i)對應到某一個關鍵詞 $W_i$：

$$TF * IDF(i) = TF(W_i, D_j) * IDF(W_i) = TF(W_i, D_j) * \log(D / DF(W_i))$$

方程式 1：TF*IDF 向量形式之定義

其中，$TF(W_i,D_j)$表示詞 $W_i$ 在文檔 $D_j$ 中的出現頻率；D 為總文件數；$DF(W_i)$表示包含詞$W_i$ 的文件數。

所謂的TF (Term Frequency；關鍵詞頻率/詞頻)乃為關鍵詞彙位於文件中所出現的頻率，意謂著該詞彙於個別文件中的重要性，TF 值愈高代表該詞彙是文件主題詞的可能性愈高；而 IDF(Inverse Document Frequency：文件頻率倒數)則是表示詞彙於同一個領域（Domain）文件集合中的重要性，若在一文件集中如果一個關鍵詞彙出現次數很高的話，那麼表示該關鍵詞彙的字義很廣，不應給予太高的加權；反之，如果一個關鍵詞彙出現在文件集的次數很低，那麼就代表該關鍵詞彙很重要。因此，IDF 愈高代表該詞彙用來鑑別主題的能力愈高。 TF 與 IDF 乃是計算文件符合程度的重要指標，由於 TF 指標對文件長度不一的情形誤差較大，所以必須藉由兩者來“共同決定”哪些文件與關鍵詞彙的相關度最高。其中 TF 關係到召回率(Recall Rate)，而 IDF 則關係到精確度(Precision Rate)。

■ 位置特徵(Location Features)：[1][4][9][13][18]

一份文件當中重要的語句通常都會出現在某幾個特定的位置上而有跡可循。因此，位置的資訊(*Positional Information*：Position in text, position in paragraph, section depth, particular sections)依據經驗法則通常也可以成為一種判斷語句重要性的線索之一。舉例來說：以整篇文章為例，若我們將之區分為數個段落，那麼通常在第一段可能會說明全篇的主旨、最後一段會總結出摘要而與主題有高度的相關。而以每一個段落(Paragraph)為例，通常在第一句和最後一句這兩個語句，往往會帶有較高的可能性包含與主題高度相關或是總結主題的資訊而成為候選的摘要內容，所以，落於這兩個部份的語句相對地來說就具有較高的重要性。因此，依據語句位置的不同應該要賦予其不同的重要性。換言之，我們可以透過每個語句不同的期望權重值來計算該語句所具有的相對位置特徵值，以此來權衡語句之相對重要性。

■ 背景特徵(Background Features/Add Term)：[4][18][19]

從文章的標題(The title or headings in the text)、簡介(Introduction) 或前言(The initial part of the text)等部分，甚至是使用者的查詢(A user's query)等線索詞彙(Lexical cues)，通常都可以用來代表文件中所要描述的主題。因此，假如文件中語句的詞彙出現在上述背景當中越多，則代表該語句與文件主題的相關程度也越高。但是，這種方法的最大缺點在於必須依賴特定的寫作格式以及使用特定的字詞才能有效篩選出有用的資訊；一旦寫作的模式改變，透過這種文章背景結構分析技術所選取出來的摘要品質也會大受影響。

■ 語句長度(Sentence Length)：[4][9]

語句的長度往往會左右語句所涵蓋資訊量的多寡。也就是說，較長的語句所包含的資訊量通常會比較短的語句所含的資訊量來得更加豐富，語意也會更加地完整，也比較能夠用來代表原始文件所欲表達的意涵。因此，我們可以依據實際的經驗法則來定義一

個閾限值或門檻值(Threshold)，比如說：7 個中文字，也就是說一個語句的長度必須至少要具有7 個中文字才有可能候選而成為摘要的一部分。

■ 線索字詞/提示片語(Cue words and phrases/Fixed phrases)：[1][4][9][13][18]

在文件當中往往會使用一些提示片語或轉折語來介紹或總結主題之敘述，如：『首先』、『總之』、『總而言之』(“In summary”) 等等，或是與特定領域相關的特定詞彙(“bonus” or “stigma” term)，例如：在專利文獻申請專利範圍(Claims)當中常用『如申請專利範圍第2項所述之系統，.....』(A system as set forth in claim 2,...)、『其中，.....』(wherein said)、『更包括.....』(further comprising...)等。因此，文件中的語句如果包含這些常用的提示性片語或轉折語，那麼該語句便有極高的可能性是屬於摘要。

■ 相似度(Similarity)：[18]

所謂的『相似度』乃是指語句間語彙的重複性(Vocabulary Overlap)，亦即兩個詞語在不同的上下文當中可以互相替換使用而不會影響到文本中的句法語意結構。如果兩個語彙在不同的上下文之間可以相互的替換而不會影響到原文之句法語意結構的可能性愈大的話，那麼此二者的相似度就愈高；反之，相似度就越低。詞語相似度是一個主觀性相當強的概念，迄今尚無明確的標準可以用來客觀地衡量。目前常用的方法主要有兩種：一種是利用句子的表層資訊(如：組成句子的詞之語法、語意資訊等)，但不包含任何結構上的分析，也未考慮到句子整體結構的相似性；另一種方法則是對語句進行完全的句法分析，並將分析的結果以結構樹(Parse Tree)的形式來加以呈現，依此基礎來進行相似度的計算。

■ 鄰近度(Proximity)：[18]

所謂的『鄰近度』乃是指文字單元(如關鍵詞、概念等)在文件當中的距離，是一種

位置運算子--布林邏輯運算子“AND”的延伸。它描繪了語意空間中語彙基本要素(Text Units)之間的出現順序和相對距離，是語意空間分析的一個重要方法，可以經由距離或位置向量的關係來加以度量。

■ 同時並列出現(Co-occurrence)：[18]

詞語同時並列出現(Co-occurrence)是指相關的詞彙在文件中常常一起出現且在統計上具有顯著意義的線索，通常存有類似、相關和同義等等關係。這種關係是一種存在於所有人類語言的普遍現象，表示詞語與詞語之間的語意和語法的關係，但卻又是一種隨意性的語言現象，鮮有規律可尋。自動化的方法，大抵都倚賴此種共現型態，來建構索引典。以關鍵詞與關鍵詞關聯的假設為基礎，透過詞頻等屬性來計算出詞彙與詞彙之間的相關性，並以其相關性的值來分類詞彙，此種方式將關鍵詞分成為數個類別，且將同一類別中的成員視之為是擁有相同概念的。因此，利用這種同步出現之分析技術可用以描述概念之空間(Concept Space)。

■ 同指涉/共同參照關係(Coreference)：[1][2][18][27]

同指涉(Coreference) 是達到自然語言“理解”中幾個特殊而困難的問題之一，其較廣義地來說乃是指重複語法中前向對映詞的解析 (Anaphora Resolution) ——亦即，設法找出文件當中指向同一真實世界裡的共同事物(Entity) 之語詞。 例如：『好巧喔！早上搭你的車來，現在回去也是搭你的。』其中，第二個“你的”所指的乃是前一子句中的“你的車”之意。再舉一例：『侯佩岑是當今年代新聞台超人氣主播，她笑容可掬，擁有迷人的甜美臉蛋與明眸皓齒，即使在播報新聞時不小心吃了個螺絲，這也常讓喜歡她的粉絲為之瘋狂，她永遠是最可愛的甜姐兒主播。』在這個範例中，三個“她”所指稱的都是“侯佩岑”。其中三個代名詞“她”為 Anaphor(s)，而被參照的對象“侯佩岑”則為Antecedent。一般的代名詞(包括反身代名詞) 都可當成 Anaphor；而找出文件中所有的 Anaphor 參照到的 Antecedent 這些詞的過程，就稱為 Anaphora Resolution (指示解

析/前向對映詞的解析) 或是 Coreference Resolution (同指涉的解析)。同指涉(Coreference)的現象亦可用來協助解決文摘(Text Summarization) 的問題，是達到自然語言“理解”的一個重要步驟。

■ 句法關聯(Syntactic relations)：[1][18]

句子是由詞和片語按照語法規則構築而成的，是表示一個完整意思的語言單位。而一個句子的表意主要是透過詞義、句法結構、語義、層次、語氣等五大因素的交叉作用而來。當代形式句法理論已將句法關係歸納為有限的幾種，分別由特定的結構關係來表達。所謂的『句法關聯』指的是經由一定的語法形式將詞和詞組合後所表現出來的各種語法關係，通常可以分為陳述、修飾、支配、平行、補充等關係以及主謂、偏正、動賓、聯合、後補等五種結構模式。所有的關聯，可藉由結構樹(Parse Trees)中的附加語來表示句法的問題，實質上就是語法的邏輯問題，利用固定的詞序來表現各種句法關係——亦即，一個相同的字詞在語句中擺放的位置不同，那麼它的句法角色也會因此而跟著不同，例如：“我愛紅娘”和“紅娘愛我”，又如：

Ⅰ．他已經破解了這個密碼。
Ⅱ．這個密碼他已經破解了。
Ⅲ．他已經把這個密碼破解了。
Ⅳ．這個密碼已經被他破解了。

利用文法剖析器與辭典來解析本文句法，可將每個詞彙皆產生一個自身的關聯詞彙串列，透過這些串列就可以用來計算詞彙間之相關性。

### 2.2.2 Edmundsonian 典範

絕大多數以語料為基礎(Corpus-based)的自動摘要研究其實都是濫觴於Edmundson (1969)，後續類似這種節錄式摘要(Extraction)的研究可以說都是以他的研究為主要的典

範，我們稱之為『Edmundsonian paradigm』[14]。

Edmundson(1969) 在其研究中所認為的特徵主要有四，分別是：Cue words (線索字詞)、Title words(標題字)、Key words(關鍵字)以及Sentence Location(語句位置)，其中前三者屬於字詞(Word-Level)特徵。而為了權衡語句的重要性，我們可以透過一個線性函式(如方程式 2所示)來對這四個特徵分別評分而後再予以加權之 [4][14]。

$$W(s) = \alpha\ C(s) + \beta\ K(s) + \gamma\ L(s) + \delta\ T(s)$$

方程式 2：Edmundsonian paradigm的語句四特徵線性函式(Linear function)

其中，C=Cue word，K=Key word，L=Sentence Location，T=Title word，C(s)代表某語句s在Cue word特徵上的權重值，其餘依此類推。而$\alpha$、$\beta$、$\gamma$、δ則是依訓練回饋結果比較後動態調整的參數。

然而，此類的研究未必是以語句(Sentence)為基本單位，所以我們可以將上述方程式 2 所述的幾個特徵更進一步地重新予以詮釋，得到如下之線性特徵(Linear Feature)組合公式(如方程式 3所示) [19]：

$$Weight(U) = \alpha * FixedPhrase(U) + \beta * ThematicTerm(U) + \gamma * Location(U) + \delta * AddTerm(U)$$

方程式 3：線性特徵(Linear Feature)組合公式

其中，***U*** 所代表的是文字單元(*Text Unit*，例如：字、詞、片語、子句、句子、節、段)、***FixedPhrase*** 代表指標性片語(*Indicator Phrases* or *Cue Phrases*，例如：“總而言之”、“*In summary*”)、***ThematicTerm*** 代表主題詞(*Thematic terms*，例如： *tf.idf* 相對權重值較高者)、***Location*** 代表該文字單元的位置(*Position*，例如：整篇文章中的首段、中段、末段；段落中的首句、末句等；或是科技論文中的簡介(*Introduction)*或是結論(*Conclusion*)

部份等等。)、***AddTerm*** 則代表了文字單元中的關鍵語彙亦出現在“書目”、“題目”、“文章標題”、“第一段”、“使用者偏好” 、“使用者查詢”等背景資訊者，而$\alpha$、$\beta$、$\gamma$、$\delta$等希臘字則是依實際研究情境動態調整的參數(*Tuning Parameters*) [14][19]。

2.2.3 以全文整體結構特徵(Discourse Features)來剖析之自動文摘

根據文件的起、承、轉、合，文件依其所使用的功能、所欲達成的目標而有各式各樣、形形色色之格式產生(*Layout in terms of sections, chapters, etc.*)。比如說：『歷史』書寫時常是以某一種特定的形式、類型與詮釋策略來編撰雜亂無章的各種原始資料。而一則『新聞』可能就是由『人』(Persons)、『事』(Events)、『時』(Time)、『地』(Places)、『物』(Things)、『觀念』(Concepts)等基本要素所構築而成的。也就是說，不同類型、目的的文件，可能因著寫作方式以及用字遣詞等等特性的不同，造成了文件格式亦有所不同(*Narrative structure*)。由於文件格式的不同，最後所產生的摘要形式也可能會因此而有所差異。以科技論文為例，科技論文的摘要可以著重於緒論(Introduction)以及結論(Conclusion)這兩部分；而以專利文獻為例，我們則可以把重心擺在描述較為抽象的『申請專利範圍』(Claims)部份，再輔以陳述較為具體的『發明說明』(Detailed Description of the Invention)來加以闡釋，即可完成代表此篇專利文獻的摘要內容(Summarization)。而新聞文件的摘要在本質上可能須著重於給讀者一個全面而概觀的描述——講重點、說明白。然而，無庸置疑的，若文件之間的格式類型相同或是相似的話，其所產生的文件摘要就有可能具有某些共通的特性。從認知心理學的角度來看，一份文件的誕生，乃是由作者本身依其所認知的概念空間來進行詮釋並加以組織後所得之結果。所以，我們可以將一份完整的文件予以解構，用一些共通的特性將字詞共聚為某一集合，用以代表某一種概念或認知。因此，語段層的方法(Discourse-level Approaches)主要是傾向於剖析出全文內容的整體結構原型及其各組成要件或稱為「命題」(Proposition) (例如：關聯詞彙)之間的關聯性(Cohesive ties)，以這些命題之間的語意關係，來明白作者的思路，進而建構出自動文摘出來。此類的語意鏈結關係，一般來說可以分成兩大類，如表 1所示 [19]。

表 1：文件內涵中的兩大類語意關聯性(ties)

| 文法關聯性(Grammatical cohesion) | 語彙關聯性(Lexical cohesion) |
|---|---|
| ❶代名詞前向指涉名詞關係(anaphora)<br>❷省略(ellipsis)與取代(Substitution)<br>❸連接關係(conjunction) | ❶同義詞(synonymy)<br>❷上位語(hypernymy)<br>❸重複語(repetition) |

## 第三節　深層摘要研究取向(Deeper Approaches)

深層摘要的研究取向(Deeper Approaches)乃是將原始文件透過語意層或語段層的內部解構來表達其原文意涵的知識體系，然後再利用自然語言的經驗法則來加以組織後產生文摘(Abstracts)輸出。因此，透過這種方法所產生的文摘內容有可能不是直接取自於原始文件當中的內容。

一般而言，這類的研究方法通常與所應用之領域是息息相關的(Domain-dependent / Background knowledge-dependent)，意即這是一種所謂的〝Knowledge-rich〞(Knowledge-intensive)的方法。語句(Sentences)通常至少要被解析至語意層(Semantic Level)。由於透過這種重述法可能事先需建構出與應用領域知識高度相關的資源(如：知識本體(Ontologies))來產生摘要(Abstracts)，因此，隨著應用領域的不同，往往需要額外之編碼(Coding)。除此之外，可能還需借助語言學的專門知識來分析語句之意涵或是用以協助產生文本。所以我們依據 I. Mani 及 M. Maybury (1999)等人三階段的自動文摘處理架構(如圖 5所示) 來審視，可能需做一些適度的修正與調整：尤其是在合成(Synthesis)階段通常需從語意層或語段層的表達當中透過自然語言的處理(NLP)以及一些探索性的經驗法則(Heuristic Rules)方式來加以試探，以產生出具連貫性的文摘輸出 [12][16][19]。

總而言之，此類取向之研究方法乃是以文件全文內容意義為根基來做簡化並且可以產生出具語意連貫性、資訊更為豐富的摘要內容出來，所產生的摘要也比較符合人類的閱讀習慣。不但如此，它還允許非常高的壓縮(極低的壓縮率)，以提供更為一般化的文

摘內容。可惜，此類系統建構成本昂貴、代價亦高，且實作較為不易。稍有不慎，可能會由於摘要生成過程的瑕疵而導致文摘的內容有誤，間接誤導了讀者們的判斷。

以下，分別簡介此類研究取向常用的兩種方法：樣板摘錄(Template Extraction)法以及概念擷取(Concept Abstraction)法 [16]。

### 2.3.1 樣板摘錄(Template Extraction)法

所謂樣板(Template)擷取文章摘要的方法，簡單來說乃是利用語意的架構(例如：語篇中的人物關係、時間關係、空間關係或情節發展所用之關聯詞等)來進行重要資訊的選取。由於同一類型的文章(例如：政治新聞)其表達資訊的型式較為固定，因此若有某些句型雷同且不斷地重複出現，則此類句型便極有可能是用來記載重要資訊的句型。所以，我們可以透過事先已定義好的語意標籤來分析文章中的內容，將內容轉換成各式語意標籤的組合，然後再利用標籤組合的重複性找出此文中較為重要的句型出來，以這些重要句型作為摘要的範本(如圖 12所示) [16][19][20][23]。

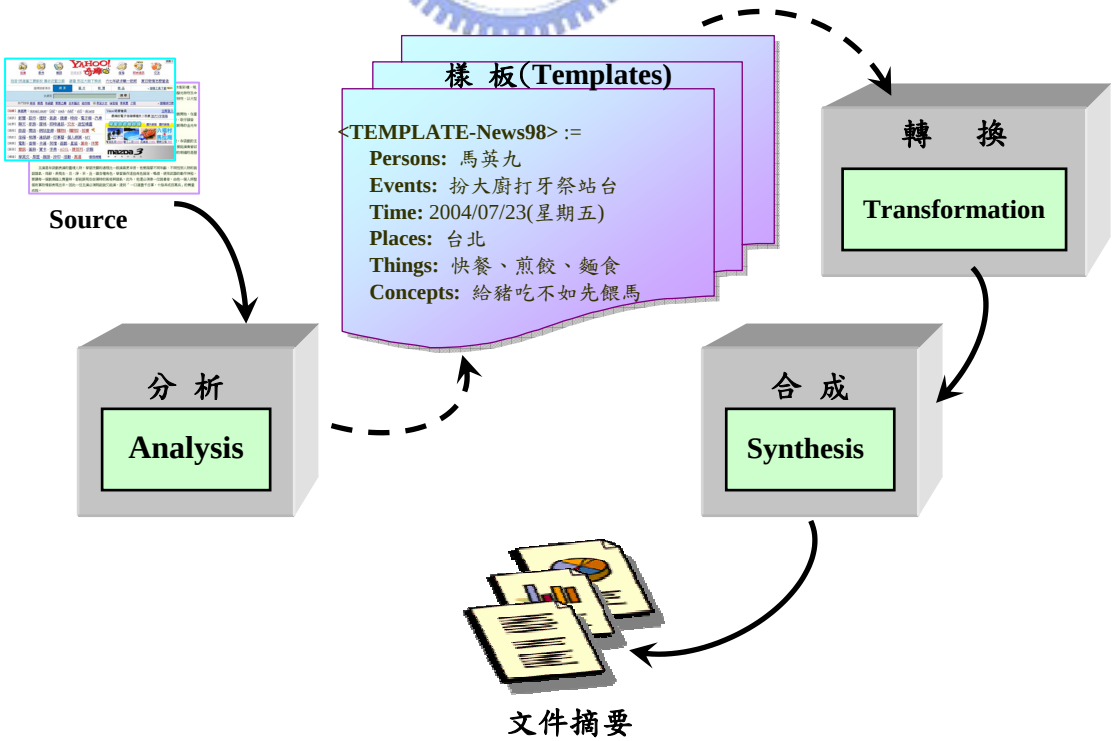


圖 12：樣板摘錄(Template Extraction)法的技術架構概觀

其演算步驟概述如下：

❶ 首先，透過自然語言處理的系統(Natural Language Processing)分析文章中的內容，利用人工建立的語料資料庫(e.g. Ontology)由機器標上語意標籤。此標籤的目的在於將此文中有意義的詞彙轉變成具有語意訊息的處理模式。

❷ 利用這些已標記好的語意標籤找出在文章中出現頻率較高的標籤組合，作為摘要樣板(Template)的句型。

❸ 將原文(Source)中句子的句型與摘要樣板(Template)中的句型進行逐一比對，若發現摘要樣板(Template)的標籤組合為文章句子標籤組合的子集(Subset)且順序相同的話，就選取該語句成為候選摘要內容之一。

❹ 將候選出的摘要內容依照原文內容的順序加以排序。

❺ 將選出的每一個語句去除未標上語意標籤的字彙，只將含有語意標籤的詞彙留下，最後所餘之結果便是『摘要』——樣板法所產生的摘錄(Template Extraction)。

使用此法的最大優點是可從大型語料庫(Large Corpora) 中透過機器學習並以穩定漸進的方式來擷取出部分具語意內容的最重要資訊；而其缺點則是以有限的語意標籤作為摘要的輸出方式，可能會因刪除了原語句中部分的字詞而使得新語句之閱讀變得更令人難以理解。此外，透過語意標籤組合所形塑出的新語句，可能會因某個原始語句比較長及其語意標籤與語意標籤之間的間隔字、詞稍多，致使語意標籤的順序恰好吻合了摘要樣板(Template)中的句型而已，但卻間接造成了選出的語句是較無意義、不具代表性的摘要內容結果輸出[16]。

### 2.3.2 概念擷取(Concept Abstraction)法

所謂的『概念』(Concepts)乃是一種概念性的術語，簡單來說就是指一些字彙或是詞彙等相關名詞(Term)，而透過這些字或詞可以用來描述相同領域中普遍存在的共通基

本知識或是實體。

我們可以將某一領域中的知識拆解成好幾個樹狀結構的方式來加以呈現，其中所呈現的每一樹節點(Node)就代表了某一種獨立之概念。藉由這一棵棵的樹，就可以明白得知該領域內有哪些重要的觀念，所定義的名詞之間又是以怎樣子的關係連結而成的；也是藉由這樣的分享機制，使得每一個人都可以更清楚地知道屬於這個領域中的關鍵字、詞的組織脈絡而有跡可循。不同的樹狀結構所呈現的概念(Concepts)也意謂著各式各樣不同的文章主題；透過特定領域下一致性之概念(Concepts)，不但可用以描述對於特定文件中的知識，有效釐清因觀念或是用詞所產生的認知上的混淆；更能夠提昇語意關聯搜尋的準確性，達到有效的名詞分享。我們透過這種階層式的架構即可將文章內容整合成為各種主題資訊，以利摘要的抽取及分析。

目前，階層式(Hierarchy)的架構仍是建構『概念』與『概念』之間最常見的關聯。概念階層(Concept Hierarchy)定義了從下位概念(即較具體、特殊化之概念)集合到上位概念(即較抽象、一般化之概念) 之間一連串的對應關係，用以描述概念之間的種種語意關係。而對於不同概念之間的語意關係，主要可將之區分為三種：

❶ 一般化關係(Generalization Relationship)：一般化關係主要用以描述概念與概念之間的上、下位關係，亦即子概念必須無條件地繼承父概念之屬性與關連性，並可衍生出新的屬性和關連性。比如說：動物(上位) vs. 老虎(下位)。

❷ 屬性關係(Attribute Relationship)：屬性關係主要用以描述概念與概念或屬性值間基於某個特徵之關連性。

❸ 包含關係(Inclusion Relationship)：包含關係主要用以描述概念之間的整體-部份關係，代表特定概念與一般概念之間的對應(Mapping)，而利用概念階層方式來加以呈現，舉例來說：台北市包含了大安區、信義區、士林區等更特定之概念；反過來說，大

安區、信義區、士林區亦可對應至較為整體之概念『台北市』。

運用這種概念階層(Concept Hierarchy)的方法，其最大的好處是可以將真實世界當中的資源知識內容及可能的資訊架構描述方式予以統一並加以簡化，同時也清楚地定義出概念之間的關係和推理的邏輯規則，以期建構出一個共通的知識背景平台，進而提高了機器對資訊處理之能力以及語意之理解，大幅降低了機器交換訊息的困難度。然而其最大的缺點就是概念階層(Concept Hierarchy) 的建構是一項極為艱鉅之工作，尤其是需要建立一個龐大架構的領域知識的時候，不管是採用人工的方式抑或是透過機器學習的完全自動化處理技術，將會耗費非常大量的時間以及金錢的投入。因此，儘管在一個概念階層(Concept Hierarchy)中可以包含許許多多的應用領域，但其所含括的領域知識愈廣，則其複雜度也將會隨之而增加[16]。

## 第四節　基於 SAO 結構之相關研究探討

以下簡單說明使用SAO的理由及其相關之作法：

### 2.4.1 從英文句型剖析為何要 SAO：

一個合理完整的句子必須文法、句型結構和語意三者兼顧，才能使之言之成理、言之有物。對於英文語句來說，我們可以將其常用的句型結構歸納整理成為所謂的『五大基本句型』(Five Basic Sentence Patterns)(如表 2 所示)。

也就是說，不管英文句子再怎麼樣地千變萬化與複雜多變，它的基本結構和句型卻可以建立在亙古不變的──〝主詞（Ｓubject）與動詞（Ｖerb）〞的架構上，而句子的基本結構就由動詞來開始啟動，並由此向外擴張，進而衍生出五大基本的動詞句型，形成簡單句的『內在主要基本結構』。透過這五大基本句型結構之脈絡，任何外在擴張、複

雜橫生的句子，皆可信手拈來、藉收立竿見影之效而有跡可循。

表 2：英文句子的五大基本句型結構 [整理自 http://cc.vit.edu.tw/~cfs/9301/CD.htm]

| 英文的五大基本句型(FIVE BASIC SENTENCE PATTERNS) | | |
|---|---|---|
| Ⅰ. | **S. + Vi.** | 主詞 + 完全不及物動詞. |
| Ⅱ. | **S. + Vi. + S.C.** | 主詞 + 不完全不及物動詞 + 主詞補語. |
| Ⅲ. | **S. + Vt. + O.** | 主詞 + 完全及物動詞 + 受詞. |
| Ⅳ. | **S. + Vt. + O. + O.C.** | 主詞 + 不完全及物動詞 + 受詞補語. |
| Ⅴ. | **S. + Vt. + I.O. + D.O.** | 主詞 + 授與動詞 + 間接受詞 + 直接受詞. |
| 其中， S. = Subject (主詞)、 O. = Object (受詞)、 C. = Complement (補語)、<br>Vi. = Intransitive Verb (不及物動詞)、 Vt. = Transitive Verb (及物動詞)、<br>I.O. = Indirect Object(間接受詞)、 D.O. = Direct Object(直接受詞) | | |

一篇文章乃是由許許多多的『命題』(Proposition) 所組織而成的，而一個命題之意義以傳統簡單的語言邏輯來說就是透過了〝主詞(Subject Term)〞與〝述詞(Predicate Term)〞此類的基本結構所構築而成的主賓式陳述句，其中的『述詞』乃是用以描述主詞之狀態，作為主詞的性質或是屬性，但屬性本身是無法獨立存在的，它必須附屬在某些事物如 Subject 或是 Object 之下。因此，透過此一觀點，我們可將上述英文的『五大基本句型』(Five Basic Sentence Patterns) 約化成為『主詞(S)-動詞(V)-受詞(O)』或是『Subject(S)-Action(A)-Object(O)』的結構形式，其中 Subject(S)與 Object(O)依被動式或主動式的呈現方式的不同未必要同時存在。亦即，對於每個語句來說，可單由『Subject-Action-Object』(簡稱 **SAO**)、『Action-Object』(簡稱-**AO**)、『Subject(S)-Action(A)』(簡稱 **SA-**)三種形式之一來加以呈現。所以，由此觀之，『主詞-動詞-受詞』(Subject-Action-Object，簡稱 **SAO**』的語句結構最能保證較好的理解效果。

儘管中文的語言結構和英文的情形並無法相提並論、完全等同，但我們仍舊可以仿照這種 **SAO** 的結構句型作為參考，透過“名詞”和“動詞”的關係來嘗試理解其語意。[32]

2.4.2 透過 SAO 結構模式的文件摘要(美國專利第 6,167,370 號文件探討)

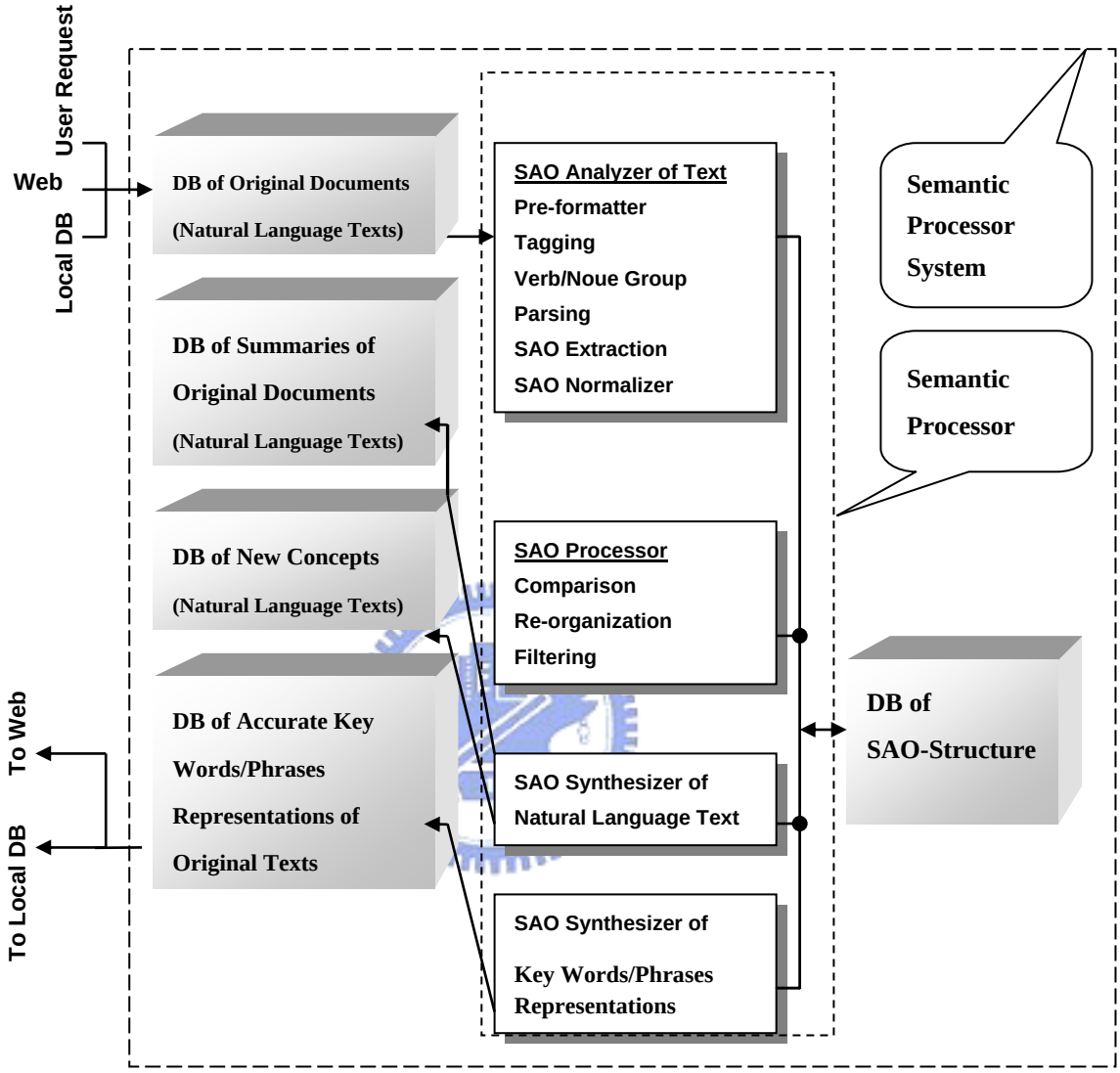


圖 13：以SAO結構模式之文件摘要架構(取自：Valery M. Tsourikov等，美國專利第6,167,370號)

經由上述之探究，我們可以得知對於一個完整的英文語句來說，可能需要同時包含主詞(Subject)、動詞(Verb) 與受詞(Object)。以美國專利第 6,167,370 號文件 (專利名稱：Document Semantic Analysis / Selection with Knowledge Creativity Capability Utilizing Subject-Action-Object (SAO) Structures)為例，該篇專利主要描述透過某一種電腦系統可將各種文件先行轉換成一組一組的SAO(Subject-Action-Object) 結構句，並且將這些SAO

的結構句儲存至資料庫中，用以代表該篇文件之語意內容。之後，當使用者輸入了自然語言的查詢需求(Request)後，此時，系統亦會將該查詢需求轉換為SAO的結構句，接著再將代表此使用者需求(Request)的SAO 結構作為一種關鍵詞彙(Key Words/Phrases)，拿來跟代表各文件語意內容片段之SAO 結構作匹配(Match) 的處理，以協助使用者找出所需求之文件出來，並下載之。最後，將這些相關文件的SAO 結構句加以分析其關係，以此創造出新的SAO 結構句以及新的知識概念，並根據這些相關文件的SAO 結構依照一些規則將之串連後，表達出自然語言的摘要(Summaries)出來(如圖 13所示)。

上述專利所述之系統乃是一種自然語言文件分析及揀選的電腦化系統，其中，由圖13來看，此系統之核心——語意處理部份，主要是由關鍵性的四大模組來運作達成的。

- SAO分析器(SAO Text Analyzer)：

  包含了許許多多的規則在裡頭，如：文字格式規則、編碼規則、字詞標記規則(例如：Markov chain theory code)、SAO 動詞(Verb)及名詞(Noun)辨識規則(註：透過建立動詞(Verb)、名詞(Noun)群組)、解析規則、SAO 擷取規則、SAO 正規原則等等，以便將候選文件的資料以及使用者自然語言的查詢需求轉換為 SAO 結構句組的表達。其中，在這個系統語意處理的過程當中，會將此查詢需求之 SAO 結構句組的表達予以合成，以作為查詢用之關鍵詞彙，然後再透過 WEB 或是本機資料庫的文件搜尋引擎下載候選文件的資料至系統的 CPU 裡，以便做後續之處理。

- SAO處理器(SAO Processor)：

  主要是將上述使用者自然語言查詢需求之 SAO 結構句組的表達拿來跟候選文件之 SAO 結構句組的表達做匹配處理，以比較是否至少有一 SAO 結構句相符合，以便做過濾篩選，將完全無法匹配的候選文件及其相對映已儲存之 SAO 結構句組逕予淘汰、刪除。

- 自然語言之SAO合成器(SAO Synthesizer of Natural Language Text)：

將上述完成匹配處理過後符合查詢條件之相關文件，取其所相對映之 SAO 結構句組中的至少某一些部份，透過一些演算步驟將之組織合成為一自然語言的形式(如：句子)後，使之成為可以展示在螢幕上的自然語言摘要輸出，並且將此摘要以及合成處理後所產生之新的 SAO 結構句組儲存至系統中。

- 關鍵詞彙之SAO合成器(SAO Synthesizer of Key Words/Phrases Representations)：
  從 SAO 的結構句組中，擷取重要的關鍵詞彙(Key Words/Phrases)以作為同義字或詞(Synonyms)，然後透過一些演算步驟將之銜接後，使之成為另一新的關鍵詞彙(Key words/phrases)，以形成使用者的查詢需求條件送至搜尋引擎來做查詢。

其中，上述 SAO 合成器的演算規則為：若儲存至系統中的任兩個 SAO 結構句組(S1-A1-O1)及(S2-A2-O2)，經系統辨識後發現其中 O1 同義於 S2，則可將之合成處理為(S1-A1-S2-A2-O2)的語句，使成為摘要的一部份或是作為查詢用之關鍵詞彙。此外，若 S1 與 A2 也有關聯關係存在的話，也可將之合成處理為(S1-A1/A2-O1)的結構句作為查詢用之關鍵詞彙，以搜尋出想要之結果出來。

我們可以運用此篇專利發明的構想作為我們中文專利文獻SAO 結構擷取的研究指引。不過，透過這樣的方式極有可能會因此而衍生出為數眾多且分散的SAO 結構句組出來，如此的結果對於專利分析人員來說，反而會因著焦點的模糊而造成更大之困擾。基於此，我們希望能夠模擬人類閱讀專利的方式來挑選出極具重要性並且有意義的SAO結構句出來，並由此建構出它們彼此之間的階層關聯。最後，綜合此篇專利文獻所彙集之階層式SAO 結構句組以及使用者之需求，以重點式的形式來加以呈現出資訊量合宜、足資代表此篇專利文獻全文內容之摘要出來。

### 2.4.3 方法 $\mathcal{A}$ 之 Concepts(概念)、SAO 之相關擷取技術[31]

本研究係與資策會電子商務研究所共同合作之創新前瞻技術之研究。而所述之『方

法 $\mathcal{A}$』乃是發表於[31]2004 年第十五屆物件導向技術及應用研討會中之論文：『以 SAO 物件為基礎之中文專利文件摘要方法及架構』所提之演算方法(註：以下皆以『方法 $\mathcal{A}$』來代稱)。茲將其攸關於概念(Concepts)以及 SAO 方面的擷取技術概述如下。

■ 『方法 $\mathcal{A}$』之概念(Concepts) 擷取技術：[31]

如圖 14 之流程圖所示[31]。首先，在進行 Concepts 擷取之前，先行定義了 Concepts (概念)的擷取關鍵字，其擷取關鍵字分為兩個集合，分別為“第一次提及”和“第二次及之後提及”這兩個集合，如下所示：

第一次提及：{一(Neu)、一(D)、複數(Na)、兩(Neu)、之一(Nc)、…}
第二次及之後提及：{該(Nes)、上述(Na)、述(VE)、…}

緊接著，開始著手進行 Concepts(概念)擷取關鍵字的比對(Mapping)處理，若比對出的關鍵字係屬於“第一次提及”的集合，則擷取此關鍵字後面的字串存到 TempSet1 集合，其字串的範圍為關鍵字後面的第一個字元至句子的最後一個字元；之後，繼續對後面句子之內容進行 Concepts (概念)的比對(Mapping)處理，如果比對(Mapping)到的是諸如“上述”此等之類的關鍵字詞，因為它是屬於“第二次及之後提及”，所以如前述之方法將其後的字串擷取出來，並與 TempSet1 集合的字串作比對，其比對到的最大相同之子字串，此即為所擷取的 Concepts (概念) 。此外，也定義了“消除詞彙”(StopWord)之集合，以此去除掉 Concepts (概念)字串前、後可能之贅字。

消除詞彙(StopWord)：{在(P)、至少(Da)、與(P)、與(Caa)、以及(Caa)、以便(Cbb)、或(Caa)、包括(VK)、包含(VJ)、更(D)、是(SHI)、代表(Na)、用以(D)、為(VG)、主要(D)、根據(P)、中(Ng)、之(DE)、的(DE) …}

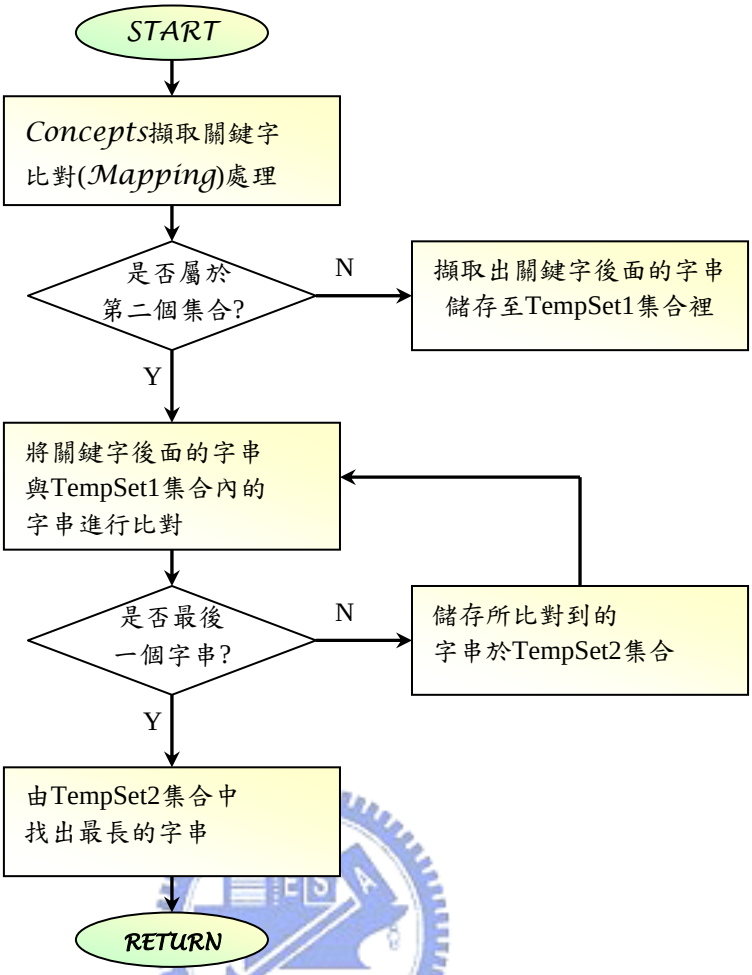


圖 14：『方法 $\mathcal{A}$』之概念(Concepts) 擷取流程示意圖[31]

■ 『方法 $\mathcal{A}$』之 SAO 擷取技術：[31]

如圖 15 之流程圖所示[31]。根據從 Claims 的句子擷取出來的 Concepts(概念) 及 Relations(關聯)，再進一步擷取出 SAO 的物件。其主要的擷取方法為判斷句子的主動式及被動式，包括句子中的主詞、動詞及受詞之架構。

首先，針對每個 Claim 中的每一子句來進行判斷，先行判斷是否有 S-A-O 之順序的物件存在於此子句中。若沒有的話，則再進一步地判斷是否存在 A-O 物件在這個子句中，若無則停止判斷，並進行下一個子句之判斷。如果存在 A-O 的結構句型的話，則以上一子句之最後一個 Concept(概念) 作為此句之主詞，作為此句 S-A-O 物件之表達。若句子存在 S-A-O 之順序的物件，則進一步判斷是否存在“被動式關鍵詞”，若發

現存在所定義的被動式關鍵詞的話，則以此被動式關鍵詞之句子呈現的規則來表示此 S-A-O 之物件；倘若句子不存在被動式關鍵詞的話，則以原始的順序作為 S-A-O 物件，並回傳給系統。

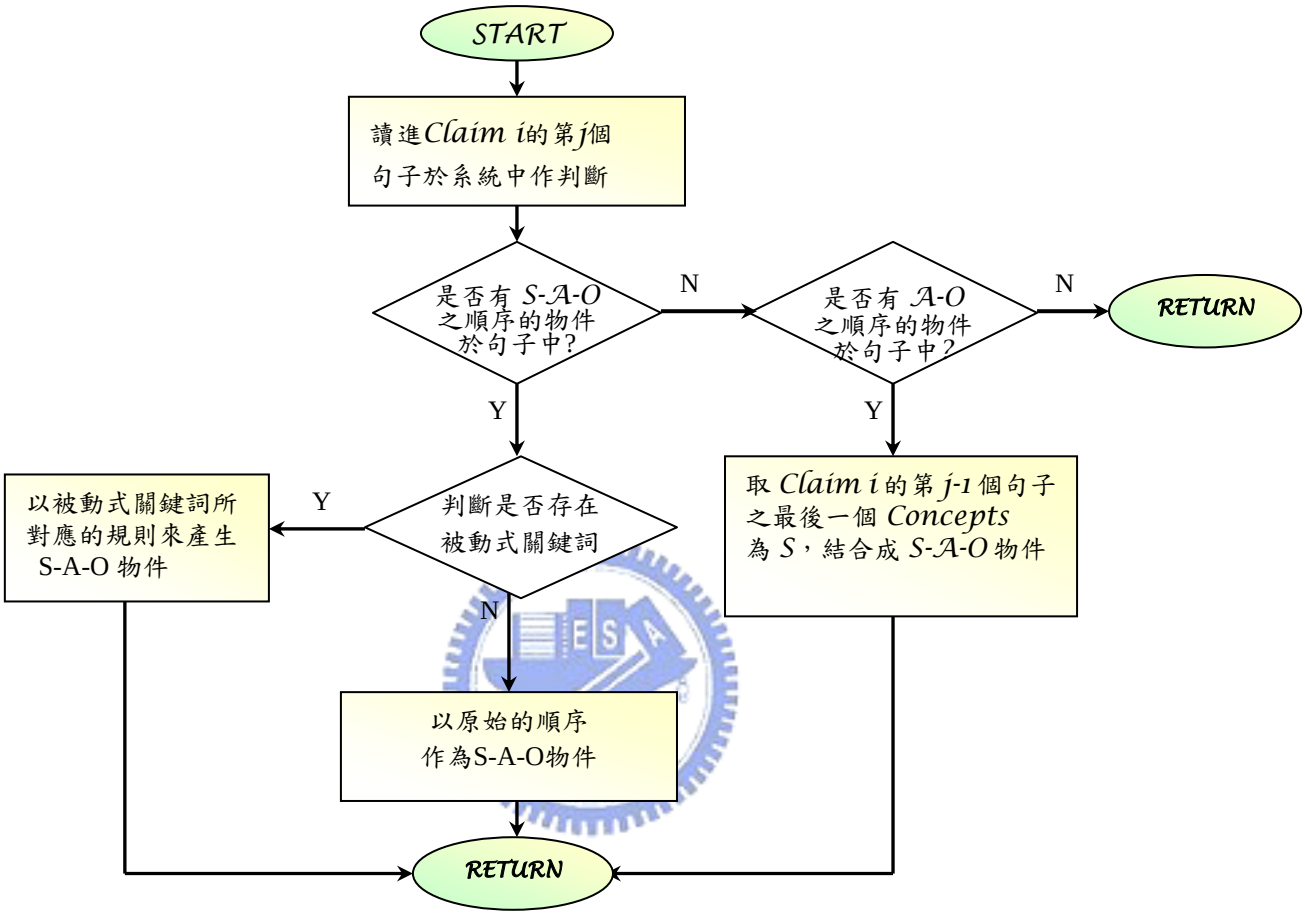


圖 15：『方法 $\mathcal{A}$』之 SAO 擷取流程示意圖[31]

# 第三章　系統架構剖析

## 第一節　系統雛型架構剖析

本研究所建置之系統雛型架構，如圖 16所示。

首先，從『中文專利』文件資料庫中讀取一篇特定專門領域之專利文獻說明書出來(註：本研究之特定專門領域暫時先鎖定為電子商務領域，作為本實驗之範圍)。之後，將此中文專利文獻做文件結構的剖析，先行擷取出“申請專利範圍”(Claims)、“發明說明”(Detailed Description of the Invention) 以及發明摘要(Abstract)此三部份的資料。

而為了要迅速且正確地萃取出此中文專利文獻當中的精華內容，我們需要藉助外在建構好的輔助工具——中央研究院中文詞庫小組所研發的『CKIP中文自動斷詞系統』，以便於我們可以在短時間內設法理解出該專利文獻中所隱含的意義。所以我們可將“申請專利範圍”(Claims)的部份送進 CKIP 中文自動斷詞系統做中文的斷詞以及詞性的標記等等，並針對斷詞以及詞性標記之部份謬誤之發生，觀察其現象後，嘗試歸納出一些通則出來，作為Heuristic Rules，以對此不甚合理之現象做一些適度之調校與修正。

接下來，再運用另外一些探索性的經驗法則( Heuristic Rules)透過概念擷取的技術去萃取出此“申請專利範圍”(Claims)中的重要概念(Concepts)出來，並運用SAO(Subject-Action-Object)的結構句型設法將概念(Concepts)以及概念與概念之間的關聯(Relation)串接起來，並將之暫儲為一組一組的SAO單元句。接著，利用概念與概念之間的統計共現(Statistical Co-occurrence)矩陣來衡量評判概念(Concept)與概念(Concept)彼此之間的語意關聯強度；然後，再將上述階段中從“申請專利範圍”(Claims) 所擷取出的“概念”(Concepts) 透過一些『下位用語指述關鍵詞』至“發明說明”或“實施方式”(Detailed Description of the Invention) 中尋找出足以具體闡述這些抽象化“概念”(Concepts) 的

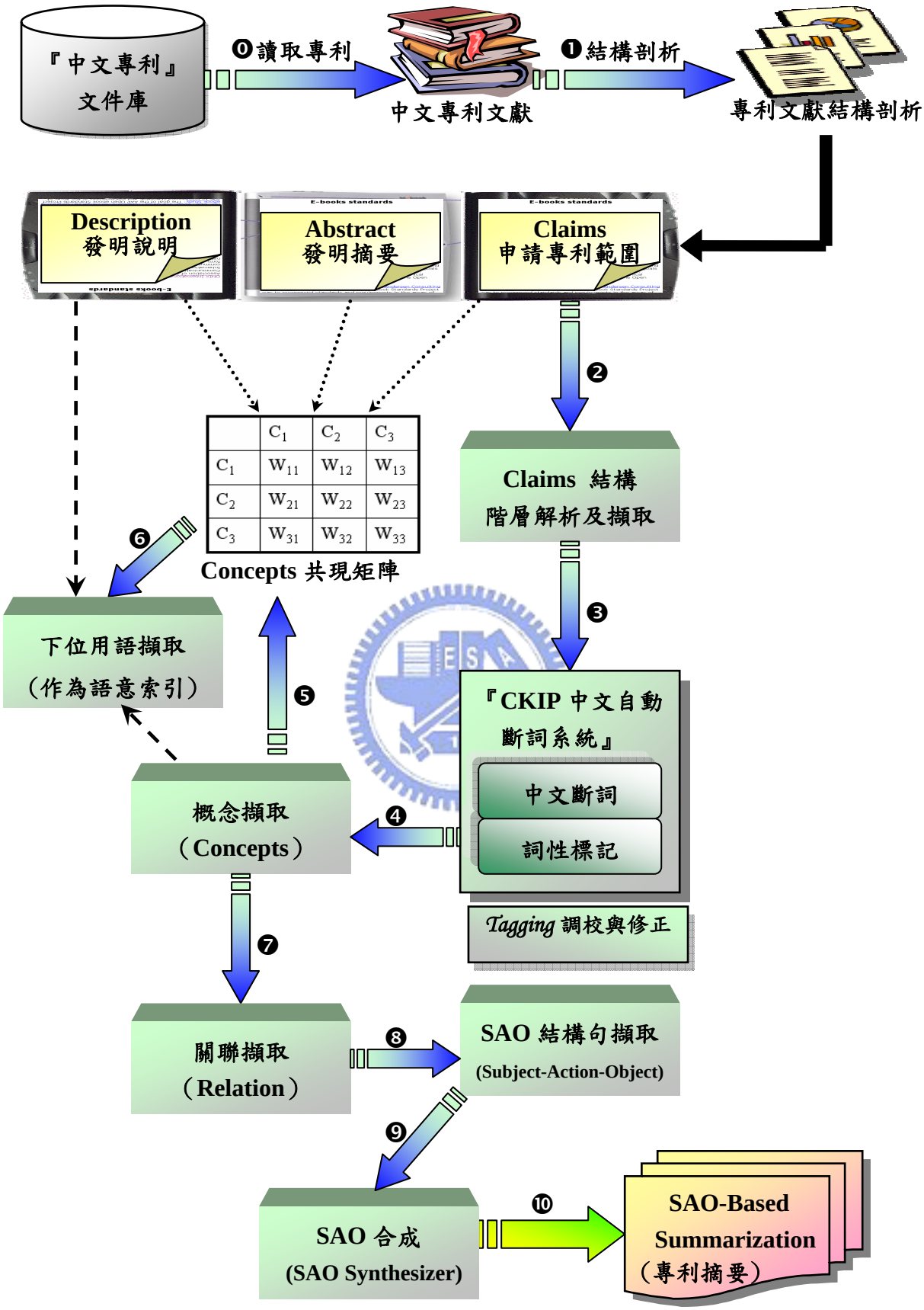


圖 16：本研究系統架構圖

下位用語出來，並參考先前已建立好的統計共現矩陣，據以建立出該概念群(Concepts)語意相符之下位用語之參考索引的部份，以更清楚的方式詮釋出這些語意較為抽象化之上位概念(Concepts)。

最後，再根據一些銜接規則(Rules)以及資訊量不同之需求，把一組一組的SAO結構句將之整理、綜合並加以組織後，即可產生出足資代表此篇中文專利文獻全文內容之摘要結果(Summarization)出來。

茲將上述的演算步驟歸納整理，說明如下：

❶ 從『中文專利』文件庫中讀取一篇特定專門領域之專利文獻說明書。
❷ 將此中文專利文獻做文件結構的剖析，先行擷取出“申請專利範圍”(Claims)“發明摘要”(Abstract)以及“發明說明”(Detailed Description of the Invention)等區段內容。
❸ 將“申請專利範圍”(Claims)的內容進行結構上的階層解析以及分段之擷取。
❹ 將“申請專利範圍”(Claims) 的部份送進 CKIP 中文自動斷詞系統做中文斷詞以及詞性標記；並針對斷詞、詞性標記上的一些謬誤現象，做調校及修正。
❺ 透過概念擷取技術去萃取出此“申請專利範圍”(Claims)當中重要的概念(Concepts)出來。
❻ 利用Statistical co-occurrence量測公式從“申請專利範圍”(Claims)、“發明摘要”(Abstract)以及“發明說明”(Detailed Description of the Invention) 等區段的內容中，計算出概念(Concept)與概念(Concept)之間的語意關聯矩陣，以此作為概念彼此之間的語意關聯強度。
❼ 以『下位用語指述關鍵詞』至“發明說明”(Detailed Description of the Invention)中尋找描述較為具體的下位用語，並搭配上述❺之語意關聯矩陣，以便在後續合成處理時可用以詮釋較為抽象化的上位概念，作為重要概念(Concepts)的語意參考索引。
❽ 以基本的『動詞』(Verbs)作為“候選關聯”(Candidate Relations)。
❾ 運用SAO(Subject-Action-Object)的結構句型，以『基本動詞』(Verb)為核心，將“概念”與“關聯”設法橋接成為一組組的SAO單元句，並暫存之。
❿ 透過一些合成的規則，將一組一組的SAO結構句加以整理、綜合並予以組織。
⓫ 產生資訊量適宜、足以代表此篇中文專利文獻全文內容的摘要結果輸出。

接下來，我們將逐一剖析本系統架構中幾個重要的組成元件，並對本實驗的摘要原理做詳細的說明及解析。

## 第二節　摘要系統之各組成元件及其運作之原理

本節將依序針對圖 16中所示的組成元件來加以剖析，並對其運作之原理詳加說明。

### 3.2.1 中文專利說明書的內容結構剖析

此部份元件可參考如圖 16所示之❼、❶等部份。一般而言，不管是中文或是英文的專利文獻，其內容大致上包含了：書目資料、發明/創作摘要、發明背景/先前技術、發明目的、技術內容、特點與功效、以及申請專利範圍等要項，就其資訊內容的結構來看(請參閱附錄一：專利說明書（Document Patent）的主要結構形式及其文件規範)，我們可以將之區分為三種形式：結構化的資訊內容、半結構化的資訊內容，以及自然語言的資訊內容(如表 3所示)。

表 3：專利說明書(Document Patent)的資訊內容結構

| Ⅰ.結構化資訊內容 | Ⅱ.半結構化資訊內容 | Ⅲ.自然語言資訊內容 |
|---|---|---|
| 這一類的內容在呈現的時候，會具有固定的樣式，使得各種資訊在表現的時候，比較具有一定之順序性以及規律性，並且資訊本身的類型也較為固定。 | 此一類的內容在呈現的時候，在語法上大概都會出現一些比較規律的樣式、脈絡得以依循，但是在順序性、種類上還有出現與否，則比較沒有一定之規律。 | 這一類的內容在呈現的時候，就比較沒有一定的限制範圍，並且在樣式上也沒有固定之型式，所以就資訊內容呈現來說，比較難有一定的規則形成。 |
| 例如：專利說明書上的書目資料(專利名稱、申請人名稱……等)。 | 例如：專利說明書上的申請專利範圍(Claims)之內容。 | 例如：專利說明書上的實施方式(Detailed Description of the Invention)。 |

儘管一些專利分析的方法企圖經由統計量化的資料中取得有用的專利相關情報，例

如：從特定的領域中分析專利歷年的申請狀況、專利權人的分佈等等。然而，以統計數字為基礎的專利分析僅僅只能探測出趨勢性的概觀情報而已，對於擷取新的技術資訊或是更進一步的競爭情報其實幫助並不太大，真正重要而有用的資訊還是埋藏在龐大且複雜的文字描述中。在目前現行的分析作業中，仍是以人工的方式來填寫所謂的『專利摘要表』——通常包括了專利目的、達成功效、技術手段和專利要件等項目，之後再據以製作技術/功效矩陣圖、專利趨勢分析圖等分析圖表，以便更進一步地來進行專利迴避設計或是專利佈局的工作。

正由於專利文獻獨特的性質，使得它不僅僅是一種具有研發領先指標意義的技術文獻——詳細記載著架構與技術製程；它更是一種法律的文獻——受國家保護的專利排他權。因此，剖析專利文獻的策略應是『先分析"**申請專利範圍**"(Claims)的部份，再分析其它說明書裡的內容』，亦即以"**申請專利範圍**"(Claims)為主要骨幹，其它內容為輔助枝節。在此值得一提的是：專利文獻上所述之"**摘要**"(**Abstract**)，其資訊並不足以代表此篇專利的全文內容。亦即，此"**摘要**"(**Abstract**)可能埋有伏筆、暗藏法律上的陷阱，其敘述可能並非全然是發明者的真心話語、也非發明內容的真實縮影。以法律保護觀點來看，仍須以"**申請專利範圍**" (Claims)部份為主要的客體對象來剖析較為適宜。

因此，本研究試圖以語意分析的技術對專利文獻中的"申請專利範圍"(Claims)部份進行分析與剖析，將半結構化的資訊內容轉換成易於瞭解與分析的SAO結構句組及摘要，如此不但可以大幅縮減人工閱讀的文字數量，並且藉由結構化的方式呈現將可以大幅提高閱讀的效率。

### 3.2.2 申請專利範圍(Claims)之結構剖析

此部份元件可參考如圖 16所示之❷的部份。就語法而言，用於專利文獻(Patent)當中的語言描述會比一般用途的語言描述更顯得貧乏與狹隘，其中的句法結構、用字遣詞

也較為單一，而且某些句法結構形式的復現率相對地也較高。因此，我們可以透過此種特殊的文件特徵將申請專利範圍(Claims)的內容做結構的剖析，其演算步驟歸納如下：

(1) 首以，以“。”作為分隔符號取出專利保護範圍(Claims)的每一項。
(2) 針對上述的每一項再依序以“：”及“；”作為分隔符號，以取出其此句 *Claim* 中的主要項及其次要項，並且將之儲存至預先宣告好之資料結構中。
(3) 以“**申請專利範圍第**”或是“**申請專利第**”字串為起始對象循序找出每一申請專利範圍(Claims)中所描述之階層式關係。
(4) 將每一申請專利範圍(Claims)存於上述(2)資料結構中的所有項素取出加上不具特殊意義的鑑別詞如“，@@，”組合後，然後再透過 **CKIP** 工具的協助，將每一個項素中的內容進行中文自動斷詞以及詞性標記的動作。
(5) 針對上述斷詞、標記上的一些謬誤現象，透過探索性之經驗法則(Heuristic Rules)尋找其通則，進而將之做適度的調校與修正。

### 3.2.3 CKIP 中文自動斷詞以及詞性標記

此部份元件可參考如圖 16所示之❸的部份。語言是動態的、有生命力的，不但隨時會有新的意義出來，現有的語意及詞彙也可能因此而改變。也就是說，同一個詞，在不同的領域、不同的時代，就會有不同的用法，它的意義也就會有所不同[26]。因此，在進行比較高階層次的語言分析之前，譬如：句法分析、語意分析、.....等等，勢必得先將文章中的內容給予正確的斷詞或分詞，才有辦法再做更進一步的剖析及處理。如果不預做斷詞處理的話，那麼關於中文詞類的劃分、語法關係以及規則的描寫等等，就等於沒有了著落，語言的理解也就無從下手。對於曾經受過某一定程度教育的人們來說，當在進行自己所屬文化或是本身經驗背景的閱讀活動時，斷詞的動作似乎都是那麼地駕輕就熟、輕鬆自然而不自覺。然而，如眾所周知的，在現代中文書面文件的撰寫方式上幾乎都是字與字相連的，其間只存有字的界線而無詞的分界點，因此字、詞之間往往存在著界限不清的複雜關係，而語句的轉折或是停頓處通常是以所謂的『標點符號』來做

為分隔的。在自然語言處理的研究裡，中文語彙中的『詞』可說是一種公認的基本單位。以本質上來看，中文語句的構成應該是由“字”與“詞”這兩個基本元素經一定的組字成詞、遣詞造句之語法規則排列後所組合而成的。儘管有些“字”的本身即可獨立表意，但絕大多數仍需將兩個以上的“字”組合形成為一個“詞”後，才能夠完整地建構出某一概念所欲表達的整體意義。在現今的中文斷詞研究領域當中，一般是以『詞庫比對法』與『統計分析法』較為普遍；不過，也有學者嘗試將詞庫比對法輔以構詞之規則後，而發展出了『文法剖析法』出來(請參閱 附錄四)。因此，隨著斷詞規則使用的不同，使得同一語句極有可能會衍生出不同的斷詞結果出來。以家喻戶曉的「員外留客」為例，可能就會有許許多多代表不同語意之斷詞斷法。如下所示：

下雨天留客天天留我不留 ➔

『下雨天，留客天；天留，我不留！』 vs. 『下雨天，留客；天天留，我不留。』 vs. 『下雨天，留客；天天留我，不留。』 vs. 『下雨，天留客；天天留我，不留。』 vs. 『下雨天，留客天；天留我不？留！』 (如表 4所示)

表 4：『下雨天留客天天留我不留』的可能斷法

| 『下雨天留客天天留我不留』 | | | |
|---|---|---|---|
| ❶ 下雨天， | 留客天； | 天留， | 我不留！ |
| ❷ 下雨天， | 留客； | 天天留， | 我不留。 |
| ❸ 下雨天， | 留客； | 天天留我， | 不留。 |
| ❹ 下雨， | 天留客； | 天天留我， | 不留。 |
| ❺ 下雨天， | 留客天； | 天留我不？ | 留！ |
| ✍ CKIP自動斷詞： | 下雨天 | 留客 | 天天 | 留 | 我 | 不 | 留 |

■ 中文自動斷詞 (Tokenization)：

在自然語言處理(NLP)的研究領域中，所謂的『斷詞』(Word Segmentation)，其目的乃在於掃瞄一段文句後，將此文句斷開成各個可賦予詞類的詞彙、片語或單字，以做為機器翻譯或是瞭解語意時的基礎。如眾所周知的，中文文本並沒有類似像英文文本用空

格之類的呈現方式來作為標示詞的邊界標幟。因此，在找出關鍵詞之前，勢必得先做斷詞(Tokenization)之類的前置處理。而中文自動斷詞或分詞的任務，簡單地來說，就是要由電腦機器在中文文本中的字或詞之間自動地加上空格；即便只是單純地輸入單一個句子，亦必須將其構成句子的各個詞彙斷出來。例如：『輸入模組，用以輸入外部之影像資料。』此句在透過CKIP 執行“自動斷詞”後就變成為『 輸入 模組 ， 用以 輸入 外部 之 影像 資料 。』(如圖 17所示)。除此之外，在所斷出來的字彙或詞彙當中亦可冠上組成句子的各式詞類 (亦即 所謂的“詞性標記”)，如名詞、動詞、形容詞、代名詞、連接詞、介系詞等等以供後續運用之。續以上例為例：在經過 CKIP 執行“自動斷詞與標記”後就會變成『 輸入(VC) 模組(Na) ，(COMMACATEGORY) ，(COMMACATEGORY) 用以(D) 輸入(VC) 外部(Ncd) 之(DE) 影像(Na) 資料(Na) 。(PERIODCATEGORY)』 (如圖 18所示)。(*註： 代表一個空格或空白。)

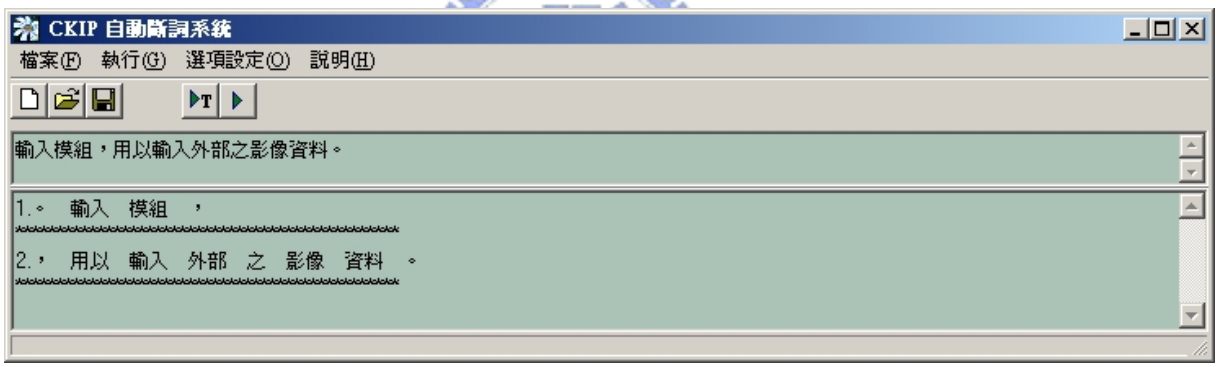


圖 17：經過CKIP 執行“自動斷詞”後的結果


圖 18：經過CKIP 執行“自動斷詞與標記”後的結果

■ 詞性標記 (Part-of-Speech Tagging)：

“詞類”是所有語言的基本架構單位，也是目前語言學家所公認建構語法的堅實基礎。簡單說來，倘若我們設法把擁有無窮多語句的現代華語文縮減為數萬個詞排列組合後的代表性樣本的話，即便如此，其規律性及相關訊息依然是相當地棘手複雜而不易於掌握；但若把其內部的關係化簡成至數十個詞類之間的關係，則其規律性相對來說將較為明顯而易於掌握，在統計上也比較容易來處理，有助於提高句子分析的正確性。因此，無論對語言欲從事何種研究都可以善用這些詞類的特性，詞類標記是有其存在必要的(如圖 18 所示)。在附錄五中，我們簡單列出了中央研究院 CKIP 現代漢語詞類標記及其所對應之意義作為研究參考。

3.2.4 概念(Concepts)擷取技術

此部份元件可參考如圖 16所示之❹的部份。由詞彙所驅動的『概念』可以說是領域知識的單位。也就是說，當我們透過語言來表達知識的時候，它的最小單位就是一個『概念』；而這個『概念』的表相就是一個詞彙[26]。照常理來看，並非所有文件當中的關鍵詞彙都具有同等的重要性。一般來說，名詞與動詞的重要性顯然就要比冠詞、副詞、形容詞或者是介系詞等詞類之重要性高出許多，『實詞』(Content word/Lexical word)的重要性也會優先於『虛詞』(Function word)。

而這裡所謂的『概念擷取技術』就是要從數位文件內容當中擷取出有意義且具代表性的詞彙(Keywords)、片語(Key Phrases)、字串(String)、或內容片段(Key Segments)等等，以這些關鍵詞彙來表示此文件內的重要概念(Concepts) ，以便於日後可以更進一步地來進行文件之探勘及各式剖析 [24]。基本上，每個中文語句都是由不定數目的基本字彙或是詞彙(包含所謂的片語)所組合而成的；因此，假如語句中的關鍵詞彙擷取不當的話，其結果便會大受影響、大相逕庭。正因這些『關鍵詞』是呈現文件主題意義的最小單位，所以對絕大部分非結構化文件的自動化處理來說，如自動化摘要、自動化詢答系統、知

識探勘、索引典自動建立、事件偵測與追蹤、知識本體(Ontologies)分析及建立等等，在在都須事先歷經關鍵詞的擷取過後，才能更進一步地進行其他方面的處理。然而，關鍵詞的認定卻牽涉到個人主觀價值經驗之判斷，且相同的詞彙在不同的主題、不同的專業領域下，也可能會有不同的解讀以及認定方式。

■ 關於中文關鍵詞或是概念(Concepts)之擷取技術：

由於用途的差別，不同的研究，對此問題的定義、採用的方法、運用的條件與擷取的成效也各有所差異。例如，在自然語言處理(NLP) 的領域中將此問題定義為所謂的『斷詞』問題。而關於中文關鍵詞彙的擷取技術方面，其主流的方法大致上可以區分為『詞庫比對法』、『文法剖析法』與『統計分析法』等三種[24]，目前仍以『詞庫比對法』最為普遍(詳請參閱　附錄四：三種主要的關鍵詞自動擷取技術比較一覽表 [24])。

然而，對大部分的現代漢語文句來說，可以是"**單字**"**詞**、或是"**多字**"**詞**的一部分，而絕大多數的白話文中的關鍵詞則是屬於**雙字詞**居多。以實際的經驗來說，一個句子其實是可以有相當多可能的斷法的 (如以表 4所示的員外留客：『下雨天留客天天留我不留』為例，就有許許多多可能的斷法)。儘管在上述所提之例中有這麼多的斷詞斷法，語句之意感覺似乎也通；但，當中可能只有一種情境的描述才是比較合乎人之常情的。那麼，究竟我們該採取什麼樣的演算準則才足以讓機器能夠自動判斷出那一種的中文斷詞之斷法才是較為精確的呢？

針對這些中文詞界不明的歧義現象，基本上我們可以採用『辭典比對』的方法，以CKIP工具快速地協助我們先行做好斷詞(Word Segmentation)及詞性標記(Part-of-Speech Tagging) 的初步工作，然後再輔以『長詞優先』、『平均詞長』等等經驗法則或是採用一種機率模型：HMM (Hidden Markov Model) [6]來協助我們找出有意義的概念(Concepts)出來。以我們使用中文詞典的斷詞方法來說，通常可能會採用的原則是 "長詞優先" 的經驗法則。一般來說，透過 "長詞優先" 會比 "短詞優先" 有較高的機率可將句子裡的

詞給正確地斷出。譬如說：『電子商務平台』這個詞，若是採用 "短詞優先" 法則的話，經過拆解後之結果乃為——『電子』、『商務』、『平台』這三個獨立的普通名詞，而這結果也正好與經由 CKIP 中文自動斷詞系統1.0 版執行後之結果——『 電子 商務 平台』相同；然而，若是採用 "長詞優先" 法則的話，其結果就是如眾所周知的專有名詞——『電子商務平台』一詞了。所以，從上述之例我們可以稍稍體會出：透過 "長詞優先" 法則所斷出來的詞彙比較能夠正確地將一個概念(Concept) 之意涵加以呈現出來。這裡所謂的 "比較.....正確" 乃是指它的語意會比較完整，也比較能夠反映、表達出我們原本想表達的意思[30]。

有一種概念極其簡單、準確率又高的方法──名之為『最大匹配法』(Maximum Matching Algorithm) 的演算法，即可套用於 "長詞優先" 的經驗法則中。它的基本想法如下：若某個語句 $S$ 是由N個中文字($C_1C_2C_3...C_k...C_n$)所構成之語句，那麼在理論上就共計有 $(2^{n-1})$ 種分詞或是斷詞的可能。首先，我們可以將句中第k 個字起的字串與一個事先建立好、擁有極豐富詞項的詞彙庫(Lexicon)來進行比對，藉以找出所有可能之斷法。假如$C_k$、$C_k\_C_{k+1}$、$C_k\_C_{k+1}\_C_{k+2}$ 都是詞庫中的詞項，那麼透過『最大匹配法』將會選取詞長最長的部份($C_k\_C_{k+1}\_C_{k+2}$)來輸出，然後再從$C_{k+3}$開始重覆同樣的流程，依此類推。這個演算法非常的簡單，只是得需事先建構好一個夠大的詞庫才能夠順利地來進行。『最大匹配法』(Maximum Matching Algorithm) 後來也衍生了許許多多不同的改良版本，以下表 5為例，以$C_k$為首的三個詞的組合假設共有四種可能之斷詞斷法：

表 5：假定 Maximum Matching Algorithm 三個詞的可能組合

| | 第一詞 | 第二詞 | 第三詞 |
|---|---|---|---|
| ❶ | $C_k$ | $C_{k+1}$ | $C_{k+2}$ |
| ❷ | $C_k\_C_{k+1}$ | $C_{k+2}$ | $C_{k+3}$ |
| ❸ | $C_k\_C_{k+1}$ | $C_{k+2}\_C_{k+3}$ | $C_{k+4}$ |
| ❹ | $C_k\_C_{k+1}$ | $C_{k+2}\_C_{k+3}$ | $C_{k+4}\_ C_{k+5}$ |

那麼依據經驗法則來說，最後一種斷詞組合最有可能是正確的斷法，因為它的總詞長為

6個字最長。像這樣子簡單明瞭的演算法不但可以解決九成以上句子語意不明時之問題，而且其正確率幾乎也已趨近於百分之一百 [29]。

中文關鍵詞擷取技術除了上述所述之方法外，我們還可以將各式各樣在第二章文獻探討中的方法加以綜合運用，或者是加入一些變化。比如說，透過觀察來探究中文專利文獻當中的一些顯著的或是潛藏的線索：諸如標題項、條列項中的文字、排版規則、重要的片語、強調詞(首字語、引號內的文句)等等。其實，每一種方法都各有其優、缺點，只是在運用時需仔細地針對不同的文摘情境、目的來詳加地予以斟酌並推敲之。茲將本研究概念(Concepts) 擷取之流程圖解如下(如圖 19 所示)：

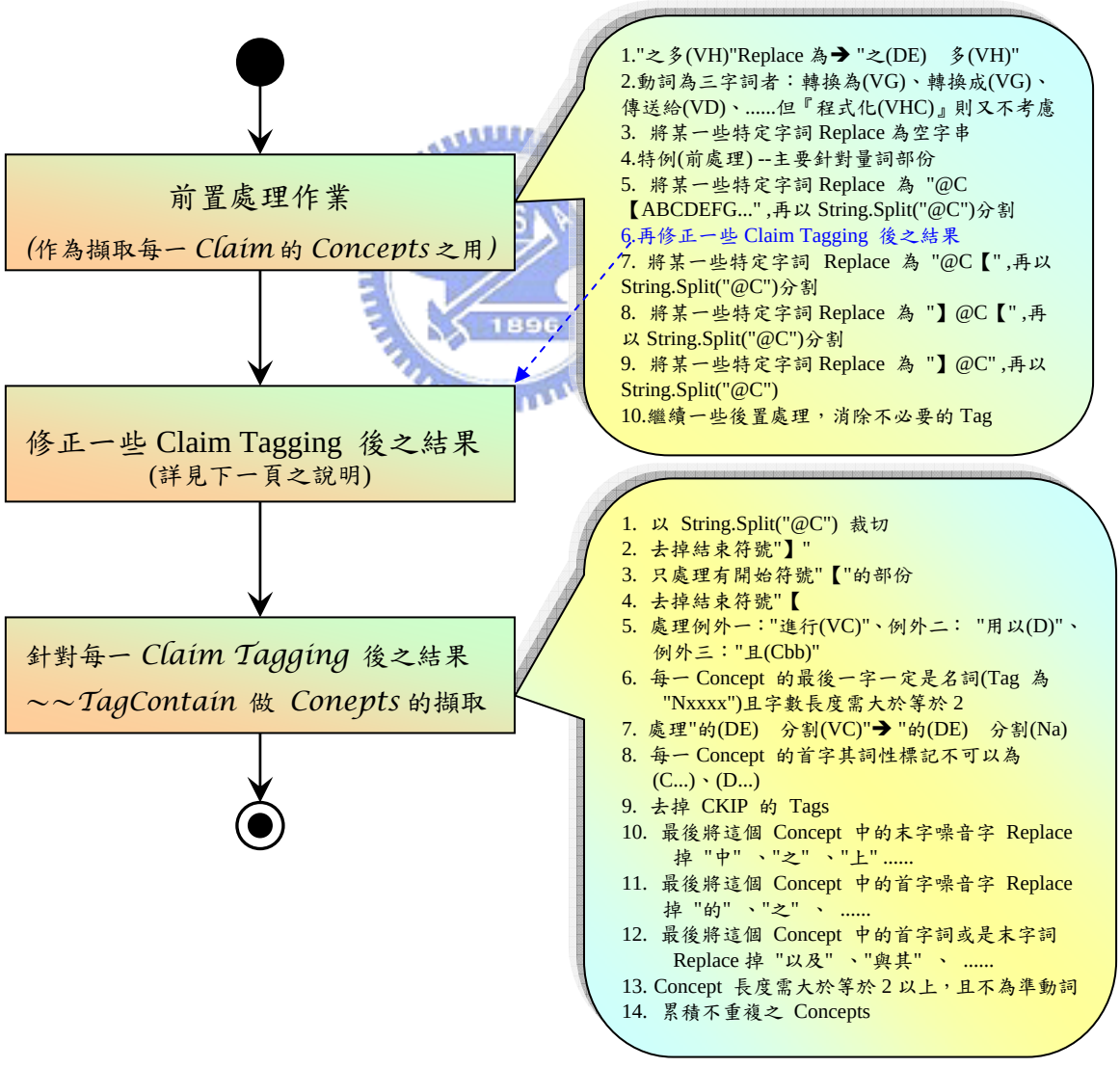


圖 19：本研究之概念(Concepts) 擷取流程示意圖

其中，第二個流程中關於『修正一些 Claim Tagging 後之結果』之部份，乃是針對斷詞及詞性標記部份的謬誤現象，嘗試用一些通則作為 Heuristic Rules 來調校與修正。其內部處理之邏輯，茲舉數例簡略描述如下：

```
假若 token(0)、 token(1)、...、 token(j-1)、 token(j)、 token(j+1)、 token(j+2)、...為經 CKIP 自動斷詞及詞性標記後之連續相鄰的字詞元素，則：
一、 token(j)、 token(j+1)兩項一起比：
  1. 若 j=0 And (token(0)中有 "(V"  Or  (token(0)中有 "以("  And token(1)中有"(V")) 則
       若 token(0)中有"(V"則令 k = 0
       反之，若 token(0)中有 "以 And  token(1)中有 "(V" 則令 k = 1；
     若 token(k + 1)中有 "(N" Or  token(k + 1)中有 "要(" Or token(k + 1)中有 "欲(" Or token(k + 1)
       中有 "(FW)" Or (token(k + 1)中有 "(V" And (token(k + 2)中有 "(DE)" Or token(k+2)中有
       "(N")) 則 token(k + 0) + "  " + token(k + 1)➔ token(k + 0) + "@C【" + "  " + token(k + 1)
  2. 若 token(j)中有 "並(" And token(j + 1)中有 "(V" 則
     token(j) + "  " + token(j + 1)轉換成➔token(j) + "  " + token(j + 1) + "@C【"
  3. 若 token(j)中有 "而(Cbb)" And token(j + 1)中有 "(V"  And 此區段內容中無 "因(Cbb)" 則
     token(j) + "  " + token(j + 1) 轉換成➔ token(j) + "  " + token(j + 1) + "@C【"
  4. 若 token(j) 之中文內容= "至" 則 token(j) 轉換成➔ "至" + "(VC)" + "@C【"
  5. 若 token(j)中有 "使(V" And token(j + 1)中有 "(V" 則
       token(j) + "  " + token(j + 1) 轉換成➔ token(j) + "  " + token(j + 1) + "@C【"
  6. 若 token(j)中有 "係(V" And token(j + 1)中有 "(V" 則
       token(j) + "  " + token(j + 1)轉換成➔"係" + token(j + 1)之中文內容 + "(VC)" + "@C【"
  7. 末尾之『學習』宜視作〝名詞〞，句中者之『學習』則宜視作〝動詞〞處理
  8. 若 token(j)中有 "別(D)" And token(j + 1)中有 "(VC)" 則
       token(j) + "  " + token(j + 1) 轉換成➔ token(j) + "  " + token(j + 1) + "@C【"
  9. 若 token(j)中有 "先(D)" And Len(token(j)) = Len("預先(D)") And token(j + 1)中有 "(VC)"  則
       token(j) + "  " + token(j + 1) 轉換成➔ token(j)之中文內容 +"(Na)" + "  " + token(j + 1)
 10. 若 (token(j)之中文內容= "要") Or (token(j)之中文內容="欲")  則
       若 token(j + 1)中有 "@C【進行(VC)" 0  則
         token(j) + "  " + token(j + 1) 轉換成➔ "@C【" + token(j)之中文內容 + "進行" + "(P)"
       否則
         token(j) + "  " + token(j + 1)➔ "@C【" +token(j)之中文內容 + token(j+1)之中文內容 + "(P)"
 11. 若 token(j)中有"(V" And (token(j + 1)中有 "成(" Or  "於(" Or "至("  則
     token(j) + "  " + token(j + 1) ➔ token(j)之中文內容 + token(j+1)之中文內容 + "(VC)" + "@C【"
 12. 若 (j - 1) >= 0 And token(j)中有 "(V" And (token(j + 1)中有 "為(" Or "於(" Or "成(") 則
       若 token(j - 1)中無 "該(", vbTextCompare)  則
         temp_CompositeVerb = token(j)之中文內容 + token(j + 1)之中文內容  + "(VC)"
         token(j) + "  " + token(j + 1) 轉換成➔ temp_CompositeVerb + "@C【"
         若 token(j + 1)之中文內容 ="為" 則
           token(j - 1) + "  " + token(j) 轉換成➔token(j - 1) + "  " + token(j)之中文內容 + "(Na)"
……餘，略！
二、 token(j)、 token(j+1)、 token(j+2)三項一起比：
  1. 若 token(j)中有"(Ng)" And token(j + 1)中有 "(V" And (token(j + 1)中無 "欲" 亦無 "要") And
       (token(j + 2)中無 "進行") And token(j + 2)中無 "的" 則
       token(j) + "  " + token(j + 1) 轉換成➔ token(j) + "  " + token(j + 1) + "@C【"
  2. 若 token(j)中有 "(V" And token(j + 1)中有 "及(Caa)" And token(j + 2)中有"(N"  則
       token(j) + "  " + token(j + 1) 轉換成➔token(j)之中文內容  +  token(j + 1)
  3.  令 Candidate_VC = "_用_成_為_得_做_作_出_入_供_與_有_應_達_遞_生_留_責_送_定_開_知_
                        取_立_"
……餘，略！
三、 token(j)、 token(j+1)、 token(j+2)、 token(j+3)四項一起比：
  1. 若 token(j)中有 "(N" And token(j + 1)中有 "(V" And token(j + 2)中有 "係(VG)" And token(j + 3)
       中有 "(P)"  則
     token(j) + "  " + token(j + 1)==> token(j)之中文內容 + "  " + token(j+1)之中文內容 + "(Na)"
……餘，略！
```

3.2.5 衡量概念間的語意關聯度

此部份元件可參考如圖 16所示之❺的部份。語意索引是利用語意矩陣為文件中的關鍵詞建構「概念空間(Concept Space)」或「知識空間(Knowledge Space)」。而這種「概念空間」乃是知識的一種表現形式，可以協助我們順利找到文件中關鍵詞與關鍵詞之間的相似性[28]。在傳統文件中，建置語意索引的方法乃是利用關鍵詞在文件中出現的頻率以及出現的文件總數做權重計算。有鑑於此，我們在本研究中參考了[17]中所述統計共現**(Statistical Co-occurrence)**的量測公式(Fano 1961；Church and Hanks 1990)的計算方法(如方程式 4所示)，並稍加做了微幅的調整，以作為本研究計算概念(Concept)與概念(Concept)之間語意關聯強度的依據。在[17]中所述的兩個字詞之間的共同資訊(Mutual Information)計算的公式摘述如下：

$$I(i,j) = \ln\left(\frac{N \cdot tf_{ij}}{tf_i \cdot tf_j}\right)$$

方程式 4：在[17]中所述的 mutual information measure for Statistical co-occurrence 計算公式

其中，$tf_{ij}$：表示 $i$、$j$ 兩關鍵字在*(cmp-lg)*語料庫中成對出現的最大頻率，

$tf_i$：表示關鍵字 $i$ 在語料庫中出現的頻率，

$tf_j$：表示關鍵字 $j$ 在語料庫中出現的頻率，

$\mathcal{N}$：表示在語料庫中關鍵字的總數。

然而，根據我們實際觀察相關人員閱讀專利的習慣來看，一般都會先從『發明摘要』(Abstract)的部份開始讀起，然後再繼續閱讀內文的部份—如『申請專利範圍』(Claims) 以及『發明說明』(Description)等部份。然，就以單篇專利文獻的內容結構來說，從較抽象的上位概念到較為具體的下位描述依序分別是 “發明摘要”(Abstract) ➔ “申請專利範圍”(Claims) ➔ “發明說明”(Description) (註：請參考附錄一之說明)。因此，我們在

此假設：從主要『申請專利範圍』(Claims) 中所擷取出之概念群(Concepts)，若其中有兩個概念(Concepts)同時出現在某一 Claim 句中的話，則代表這兩個概念(Concepts)可能會有某一程度上的語意關聯；出現的頻率愈高，則其語意關聯的強度也隨之愈高。同樣的，若此二概念(Concepts)又成對地同時出現在『發明摘要』(Abstract)或是『發明說明』(Description) 的語句中的話，則更又代表了此二概念語意關聯的強度更高、更為密切。所以，我們可以針對不同區段的部份分別給予不同的權重值來強調其語意關聯的程度。也就是說，因『發明摘要』(Abstract)為作者對此專利權利伸張以及說明的縮影，所以我們可以給予更高之權重值(比如：1.5 倍來計算)；而出現在『發明說明』(Description)的內容之部份，一般是用以詮釋說明較為抽象的『申請專利範圍』(Claims)之用的，故而我們可以給予比如：1.2 倍的加權來計算其語意關聯；而同時出現在概念(Concepts)的產出來源——『申請專利範圍』(Claims)之部份，則不予加權。我們依據上述之想法，套用方程式 4 的方法，將兩兩概念(Concepts)共同出現在“申請專利範圍”(Claims)、“發明摘要”(Abstract)以及“發明說明”(Description)中的關聯強度做計算，並且分別乘以 1、1.5、1.2 的加權係數來強調其語意關聯，以此來定義所謂的『概念間的語意關聯強度』。計算方式如方程式 5 所示：

$$I(i,j) = \sum_{k=1}^{3} \alpha_k \cdot \ln\left(\frac{N \cdot cf_{kij}}{cf_{ki} \cdot cf_{kj}}\right)$$

方程式 5：本研究概念間(Concepts)的語意關聯強度計算公式

其中，$k$：表示專利文獻不同區段的部份，k=**1** 代表“申請專利範圍”(Claims)、k=**2** 代表“發明摘要”(Abstract)、k=**3** 則代表“發明說明”(Description)；

$\alpha_k$：表示加權係數，$\alpha_1$ =**1**、$\alpha_2$ =**1.5**、$\alpha_3$ =**1.2**；

$cf_{kij}$：表示概念(Concept) *i* 及 *j* 同時出現在區段 *k* 的語句中之最大頻率，

$cf_{ki}$：表示概念(Concept) *i* 在區段 *k* 出現的頻率，

$cf_{kj}$：表示概念(Concept) *j* 在區段 *k* 出現的頻率，

$\mathcal{N}$ ：表示從『申請專利範圍』(Claims) 中所擷取出之概念群(Concepts)的總數。

3.2.6 尋找抽象化概念之『下位用語』以建構語意參考索引

此部份元件可參考如圖 16所示之❻的部份。由於專利語言的特殊性，通常在“申請專利範圍”(Claims) 的部份會採用所謂的“上位用語”的表達形式來擴大專利保護的範圍；亦即，採用最模糊的字眼來涵蓋最大的範圍，例如：不直接稱作『腳踏車』，而改以『交通工具』這樣的抽象化概念來作為表達。因此，下位用語擷取之方法，就是要將從“申請專利範圍”(Claims) 中所擷取出來的“概念”(Concepts)，對映到“發明說明”或“實施方式”(Detailed Description of the Invention) 中的內容，以擷取出在“發明說明或“實施方式”中被此“概念”(Concepts) 所映射到的句子作為下位用語，以此作為詮釋“概念”之基礎。問題是在這個過程當中，很有可能會有兩個以上的“概念”(Concepts) 同時出現在同一個句子中，並且也對映到相同之『下位用語指述關鍵詞』，亦即不同的“概念”(Concepts)卻有同樣的下位用語；也有可能會有某些“概念”(Concepts) 在“發明說明”(Description)當中，對映不到任何的『下位用語指述關鍵詞』，致使無法順利地擷取其下位之用語。茲將在本實驗中下位用語擷取之過程，描述如下(如圖 20所示)：

(1).首先，先定義好所謂的『下位用語指述關鍵詞』，例如：“係”、“可以是”、“可以有”、“包括”、“包含”、“主要目的”、“表達”、“表示”等等。但，因同一個“概念”(Concepts) 有可能會出現在“發明說明”(Description) 中的許多句子裡，並且也可能會對映到許多的下位用語指述關鍵詞；所以，我們需選擇一個最合適、最能夠表達出下位用語意思的指述關鍵詞，來作為下位用語之擷取。以下為我們根據經驗法則(Heuristic Rules) 所採用之判斷優先順序：“係”➔ “可以是” ➔ “可以有” ➔ “包括” ➔ “包含” ➔ “主要目的” ➔ “表達” ➔ “表示”。

(2).然後，再從“發明說明”(Detailed Description of the Invention) 的內容當中去判斷其語

句裡是否在此抽象化“概念”(Concept)之後有出現我們事先所定義好的『下位用語指述關鍵詞』的存在。

(3).若有，那麼就將緊鄰此下位用語指述關鍵詞之後開始的字彙至句子結束點為止的敘述予以擷取出來，以此作為“概念”(Concepts) 的下位用語。


圖 20：“概念”(Concepts)之下位用語擷取方法示意圖

舉個例來說：假如某個專利文獻的相關內容節錄片段如下，

✂ Claims：如申請專利範圍第 5 項所述之該**主/從式架構**，其中，該第一多通道元件介面係經由一第一資料庫交易處理元件，對該資料層做資料存取。

✂ Description：再者，本發明尚提出一種**主/從式架構**，**包括**一伺服端與一用戶端。

✍ 預先定義好的指述下位用語的動詞(亦即，下位用語指述關鍵詞)如下：
“係”、“可以是”、“可以有”、“包括”、“包含”、“主要目的”、“表達”、“表示”等等。

☞☞那麼，我們可以在“申請專利範圍”(Claims) 裡面發現到“**主/從式架構**”這個“概念”(Concept)，然後再將它對映(Mapping) 到“發明說明”(Description) 中，再由其後的文字擷取到“**包括**”這個動詞，此動詞乃為指到下位用語的指述關鍵詞，進而擷取出它後面所指述的內容——一伺服端與一用戶端——此下位用語即為該抽象化“概念”(Concepts) 的候選替換語之一。

(4).之後，在後續合成處理階段，可以參考上述❺所得之語意關聯矩陣，將『下位用語』所欲闡述之主要“概念”與其它同時出現在此同一『下位用語』句中之相異“概念”(Concepts)，將其語意關聯強度之權重值予以累加，累計之相對參考強度之值愈高

者，即代表此概念(Concept)以及相對於此概念之下位用語之語意關聯強度愈高，也愈能夠用來詮釋較為抽象化的上位概念，作為重要概念(Concepts)的語意參考索引。

### 3.2.7 運用 SAO 句型之關聯(Relation)擷取

此部份元件可參考如圖 16所示之❼的部份。在擷取專利文獻中重要的概念(Concepts)後，接下來就要緊接著進行關聯擷取(Relation Extraction)的動作。因本研究所採用的乃是英文文法『主詞 + 動詞 + 受詞』(Subject-Action-Object，簡稱 SAO)結構句型的模組，其中 “S”和“O”的部份皆屬前項步驟所指稱的概念(Concepts)，故而我們可以將基本的『動詞群』(Verbs)視之為一種“候選的關聯”(Candidate Relations)，以此作為我們擷取“關聯(Relations)”的一種基本準則。亦即，將介於兩個概念(Concepts)之間的基本動詞(Verbs)或者是其他非含於概念(Concepts)裡頭的基本動詞(Verbs) 擷取出來，以作為候選之關聯(Relations)、SAO結構中的Action(如圖 21所示)。

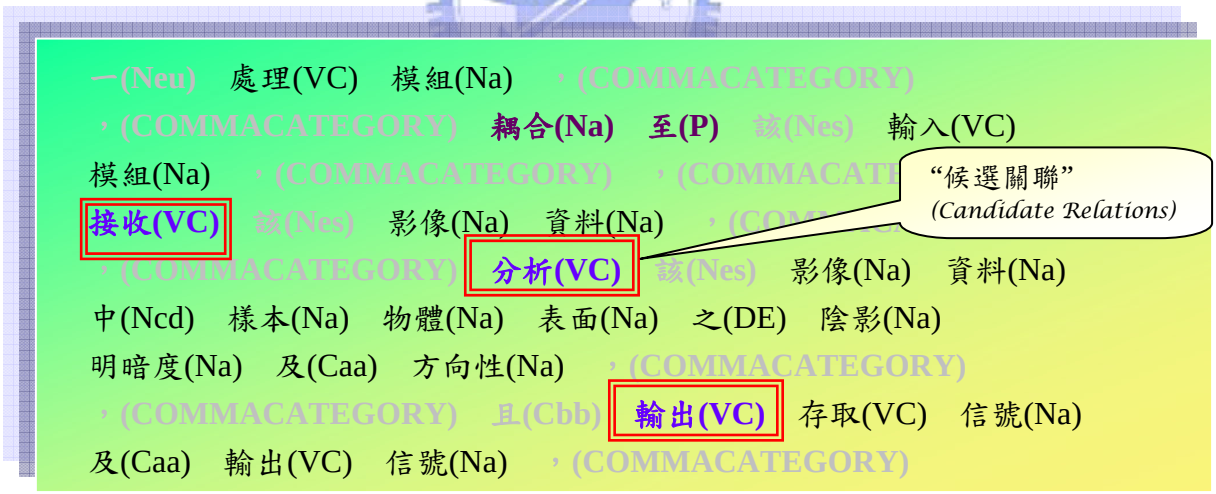


圖 21：“候選關聯”(Candidate Relations)擷取方法示意圖

### 3.2.8 SAO 單元句擷取

此部份元件可參考如圖 16所示之❽的部份。SAO單元句擷取之基本構想，主要來自於【第四章.第一節 中文專利摘要人工實驗解析】一節之說明。我們可以用 3.2.2 步驟

將句子截切後的資料結構為單位，然後再以“，”作為分隔符號將上述單位再做另一次之截切，使之成為SAO擷取來源對象的基本單位。接下來再以『基本動詞』(Verbs)為核心，將介於兩個概念(Concepts)之間的『基本動詞』(Verbs)視之為“關聯(Relations)”，或者是與其前、或後之概念(Concepts) 設法橋接，以判斷是否能夠有機會順利銜接而結合成為一SAO結構的單元句。之後，將這些順利擷取出之SAO單元句暫存之，如此即可完成本程序之運算處理。茲將關鍵性的擷取步驟三部曲解析如下，如圖 22➔ 圖 23➔ 圖 24之順序所演示。

【S自動影像置換重建系統S】，【V包括V】
【S輸入模組S】，【V輸入V】【S外部之影像資料S】
【S處理模組S】，【V耦合V】【S輸入模組S】，【V接收V】【S影像資料S】，
【V分析V】【S影像資料中樣本物體表面之陰影明暗度及方向性S】，【V輸出V】【S存取信號及輸出信號S】
【S儲存模組S】，【V耦合V】【S處理模組S】，【V接收V】【S存取信號S】，【V進行V】【S影像資料S】【V存取V】
【S輸出模組S】，【V耦合V】【S處理模組S】，【V接收V】【S輸出信號S】，【V進行V】【S影像資料S】【V輸出V】
【S輸入模組S】【V爲V】【S掃描器、電腦攝影機及數位相機S】
【S處理模組S】【V爲V】【S中央處理單元S】
【S儲存模組S】【V爲V】【S資料庫系統S】
【S輸出模組S】【V爲V】【S顯示器、印表機及繪圖機S】

圖 22：SAO結構句擷取處理過程三部曲之第一部

| | | | | | |
|---|---|---|---|---|---|
| V】【S | 置換成☞☞ | _O | 【S | 置換成☞☞ | S |
| S】【V | 置換成☞☞ | _V | 【V | 置換成☞☞ | V |
| V】【V | 置換成☞☞ | _ | S】 | 置換成☞☞ | "" |
| S】【S | 置換成☞☞ | _ | V】 | 置換成☞☞ | "" |

圖 23：SAO結構句擷取處理過程三部曲之第二部

S自動影像置換重建系統，V包括
S輸入模組，V輸入_O外部之影像資料
S處理模組，V耦合_O輸入模組，V接收_O影像資料，V分析_O影像資料中樣本物體表面之陰影明暗度及方向性，V輸出_O存取信號及輸出信號
S儲存模組，V耦合_O處理模組，V接收_O存取信號，V進行_O影像資料_V存取
S輸出模組，V耦合_O處理模組，V接收_O輸出信號，V進行_O影像資料_V輸出
S輸入模組_V爲_O掃描器、電腦攝影機及數位相機
S處理模組_V爲_O中央處理單元
S儲存模組_V爲_O資料庫系統
S輸出模組_V爲_O顯示器、印表機及繪圖機

圖 24：SAO結構句擷取處理過程三部曲之第三部

### 3.2.9 SAO 結構句之合成術

此部份元件可參考如圖 16所示之❾的部份。這裡所謂的合成術乃是將前項步驟的分項成果，進行綜合的歸納整理。其合成構想如下：

首先，將所有從“申請專利範圍”(Claims) 部份所擷取出的概念(Concepts)對應到從“發明說明”(Detailed Description of the Invention) 部份所擷取出來的下位用語，然後再參酌概念(Concepts)與概念(Concepts)之間的共現矩陣數值做一些決策，數值愈高者必然可以成為該概念(Concept)語意相符之下位用語的參考索引部份。而 SAO 結構句組的部份，則依其在“申請專利範圍”(Claims) 中的階層結構關係直接做合成之處理；除此之外，本研究也設計了下列之規則，以作為 SAO 結構句組橋接的依據，使摘要成為一可讀之自然語言之形式。銜接規則簡述如下：

- 若相鄰之兩個 SAO 結構句(S1-A1-**O1**)、(**S2**-A2-O2)中，若 O1 等同於 S2 的話，則可將之整併為一語句(**S1**-A1-O1-A2-O2。)，依此類推。
- 若相鄰之多個 SAO 結構句(**S1**，A1，O1)、(**S2**，A2，O2)、(**S3**，A3，O3)……中，若發現主詞 S1 等同於主詞 S2 等同於主詞 S3……的話，則可將之整併為一語句(**S1**-A1-O1，A2-O2，A3-O3，……。)，依此類推。

最後，再依我們對使用者所規劃的資訊量之需求，分別產生下列步驟所述之 Small、Medium、Large 的摘要出來。

### 3.2.10 基於 SAO 結構之中文專利文獻自動摘要

此部份元件可參考如圖 16所示之❿ 的部份。如果專利分析師或研發工程師想知道某件專利的詳細內容，就勢必需以如同以往的方式來仔細閱讀完此篇專利所有的全文內

容後方能準確得知。然而，再經過我們審慎的觀察這些專利文獻的撰寫特性後，我們將可發現到這之間的閱讀其實會有不少的時間、精神是花在不斷重覆的內容上，只因專利所有權人為了尋求法律上更多的權利保障，而將字句的詮釋不斷地向外擴展及延伸，透過模糊焦點的策略，以擴大專利保護之範圍。這也間接造成了因閱讀的資訊量過於龐大而讓專利分析師或研發工程師無形間降低了閱讀的品質。

針對這些現象，我們將專利全文中的“申請專利範圍”(Claims) 部份以不更動內容主體的情況下，將重覆的資訊內容透過資訊量大、中、小不等的安排方式來予以顯現(詳情請參閱【第四章.第一節 中文專利摘要人工實驗解析】一節之說明)，以方便閱讀者自行控制閱讀之篇幅，可選擇性地跳過重覆的資訊而不錯失重要的資訊。讓專利分析師或研發工程師優先閱讀以自然語言來描寫並帶有豐富資訊量的專利文摘內容，達到『用最短的時間，閱讀最精華的資訊』之目的。所以，我們依據使用者對資訊量需求的設想，分別產生出了Small、Medium、Large等資訊量不等的摘要出來，其方法描述如下：

- **Small(小)**：以“申請專利範圍”(Claims)中的**第一個獨立項**之 SAO 結構句組來產生資訊量最為精簡的專利摘要，用以代表專利全文。
- **Medium(中)**：以“申請專利範圍”(Claims)中的**各獨立項**之 SAO 結構句組來產生資訊量適中的專利摘要，用以代表專利全文。
- **Large(大)**：以“申請專利範圍”(Claims)中的**全體獨立項及其所屬之依附項**的 SAO 結構句組來產生資訊量較為豐沛的專利摘要，用以代表專利全文。

## 第三節　與方法 $\mathcal{A}$ 之擷取技術比較

本研究係與資策會電子商務研究所共同合作之創新前瞻技術之研究。而所述之『方法 $\mathcal{A}$』乃是發表於[31]2004 年第十五屆物件導向技術及應用研討會中之論文：『以 SAO 物件為基礎之中文專利文件摘要方法及架構』所提之演算方法。而其中，攸關於 Concepts

(概念)及 SAO 結構句擷取之技術，與本研究所提之演算方法是截然不同的。關於本研究相關之擷取技術，詳如本章第二節相關探討及說明；而關於『方法 $\mathcal{A}$』相關之擷取技術，則請參閱 2.4.3 乙節之探討。接著，僅就與『方法 $\mathcal{A}$』之相關擷取技術之差異做一比較。

■ 本研究與『方法 $\mathcal{A}$』之概念(Concepts) 擷取技術比較：

表 6：本研究與『方法 A』之概念(Concepts) 擷取技術比較一覽表

| | 方法 $\mathcal{A}$ | 本研究 |
|---|---|---|
| 擷取技術比較 | 1.假設：**Concepts(概念)為『重複出現有意義的最長詞彙(Longest Term)』。**<br>2.先宣告"第一次提及"和"第二次及之後提及"兩種擷取關鍵字集合，<br>3.定義"消除詞彙(StopWord)"以此去除掉 Concepts (概念)字串前、後可能之贅字。<br>4.依 2.之宣告對 Claims 中的語句進行比對(Mapping)處理。 | 1.假設：**透過"長詞優先"會比"短詞優先"有較高的機率可將句子裡的詞——Concepts(概念)給正確地斷出。**<br>2.本研究採用 CKIP 工具～～一種『辭典比對』的方法，再輔以『長詞優先』以及其它經驗法則(Heuristic Rule)。<br>3.訂定一些前置處理作業，以作為擷取每一 Claim 的 Concepts 之用，<br>4.修正一些 Claim Tagging 後之結果：針對斷詞及詞性標記部份的謬誤現象，嘗試用一些通則作為 Heuristic Rules 來調校與修正，<br>5.針對上述 4.修正後之結果(TagContain)做 Conepts 的擷取。 |
| 限制 | 1.容易造成意義不完整的 Concept(概念)出現。 | 1.數學公式中的特殊符號透過 CKIP 無法正確斷詞、詞性標記，故無法擷取出來。<br>2.擷取出 Concept(概念)有可能已涵蓋了一個完整的 SAO 在裡頭。 |

■ 本研究與『方法 $\mathcal{A}$』之 SAO 擷取技術比較：

表 7：本研究與『方法 A』之 SAO 擷取技術比較一覽表

| | 方法 $\mathcal{A}$ | 本研究 |
|---|---|---|
| 擷取技 | 1.原則：根據從 Claims 的句子擷取出來的 Concepts(概念) 及 Relations(關聯)，再進一步擷取出 SAO 的物件。其主要的擷取方法為判斷句子的主動式及被動式，包括句子中的主詞、動詞及受詞之架構。 | 1.原則一：以"，"作為分隔符號將 Claims 語句截切為數個子句，使之成為 SAO 擷取來源對象的基本單位。<br>2.原則二：本研究所採用的乃是英文文法『主詞 + 動詞 + 受詞』(Subject-Action-Object，簡稱 SAO)結構句型的 |

| | | |
|---|---|---|
| 術比較 | 2.針對每個 Claim 中的每一子句來進行判斷，先行判斷是否有 S-A-O 之順序的物件存在於此子句中。<br>3.若沒有的話，則再進一步地判斷是否存在 A-O 物件在這個子句中，若無則停止判斷，並進行下一個子句之判斷。<br>4.如果存在 A-O 的結構句型的話，則以上一子句之最後一個 Concept(概念) 作為此句之主詞，作為此句 S-A-O 物件之表達。<br>5.若句子存在 S-A-O 之順序的物件，則進一步判斷是否存在“被動式關鍵詞”，若發現存在所定義的被動式關鍵詞的話，則以此被動式關鍵詞之句子呈現的規則來表示此 S-A-O 之物件；<br>6.倘若句子不存在被動式關鍵詞的話，則以原始的順序作為 S-A-O 物件，並回傳給系統。 | 模組，其中 “S”和“O”的部份皆屬前項步驟所指稱的概念(Concepts)，故而可將基本的『動詞群』(Verbs)視之為一種“候選關聯”(Candidate Relations)，以此作為我們擷取“關聯(Relations)”的一種基本準則。<br>3.原則三：將介於兩個概念(Concepts)之間的基本動詞(Verbs)或者是其他非含於概念(Concepts)裡頭的基本動詞(Verbs) 擷取出來，以作為候選之關聯(Relations)、SAO 結構中的 Action。<br>4.以上述 3.之“候選的關聯”(Candidate Relations)為核心，將介於兩個概念(Concepts)之間的『基本動詞』(Verbs)視之為“關聯(Relations)”，或者是與其前、或後之概念(Concepts) 設法橋接，以判斷是否能夠有機會順利銜接而結合成為一 SAO 結構的單元句。<br>5.若句子存在 S-A-O 之順序的物件，則進一步判斷是否存在“被動式關鍵詞”，若發現存在所定義的被動式關鍵詞的話，則以此被動式關鍵詞之句子呈現的規則來表示此 S-A-O 之物件；<br>6.將這些順利擷取出之 SAO 單元句暫存之，如此即可完成本程序之運算處理。 |
| 限制及優、缺點 | 1.若一子句的動詞超過 2 個以上，則其擷取之 SAO 結構句組易發生錯誤配對排列組合之現象，致使 SAO 句組數膨脹。<br>2.『以上一子句之最後一個 Concept(概念) 作為 A-O 句型之主詞』未必是正確之作法。<br>3.構成 Concept(概念) 元素中的某個詞若其詞性為動詞(Vxxx)的話，易混淆而成為一種關聯(Relation)。<br>4.『上樑不正下樑歪』：由於此法之概念(Concept) 擷取之正確性不高，連帶會影響 SAO 擷取之正確性。 | 1.『以第一個子句無“關聯”(Relations)存在之獨立 Concept(概念)或者是前一個子句之最後一個 Concept(概念)作為 A-O 句型之主詞』未必是一種絕對正確的作法。<br>2.本研究這種循序擷取 SAO 之作法，雖其正確率無法達到 100%的正確，但其準確性會明顯地高於所述之方法 $\mathcal{A}$。<br>3.Concepts(概念) 擷取時，由於受到『長詞優先法則』以及『針對斷詞及詞性標記部份的謬誤現象，嘗試用一些通則作為 Heuristic Rules 來調校與修正』之影響，致使某些 Concepts(概念)會發生無“候選的關聯”(Candidate Relations)可用之窘境現象而無法順利地擷取出其正確之 SAO。 |