從廿五史全文資料庫的經驗
談中文文件檢索系統設計的考量

Some comments on developing Chinese language full-text database — lessons learned from the 25 Dynasties full-text database project

謝清俊

中央研究院　資訊科學研究所　研究員

1992 10月26-28日，北京

Ching-chun Hsieh

Professor and research fellow,

Institute of Information Science, Academia Sinica, Taipei

一．前言

廿五史全文資料庫是史籍自動化計畫中一項重要的成果。史籍計畫肇始於1984年，其目的在探索資訊技術在文史工作上應用的可行性 [1] 。 在此之前， 中文文獻處理(document processing)的研究不多，在台灣只有國字整理小組做了些初步的試探，包括機讀字典之建構[2]、斷詞[3]、文件索引與查詢[4]等，其目的在發展一些基礎的應用工具[5]。

廿五史資料庫的發展可分爲四個時期。 我們用它的資料庫管理程式 CTP (Chinese Text Processor)的各版本來說明(請參閱表一)。在CTP1.0時段是學習期；CTP2.0是過渡期，也就是說， 在此期間經過了仔細的檢討， 注入了創意並做了些實驗， 從頭到尾徹底地改變了原有的系統結構，於是便產生了全新的CTP3.0。這也是整個計畫最主要的時段[6]。

CTP3.0到CTP4.0是實務期。從CTP3.0的成功，才能放心大膽地在CTP4.0時，將全部的廿五史資料庫完成。此時已是1990年，計畫已經歷了6年2個月。此後進入了第四期，即CTP4.1及CTP4.2兩個改良版本，這是調整期。在此期間吸收了使用者的意見後，對人機介面和檢索功能做了些改良[7.8]。

一個計畫做了八年，歷經上述之變遷，失敗的經驗自然不少。因此，本報告即將這八年來的一些心得，作野人獻曝，以就教於各位先進。以下即是依各時期之先後次序報告。

表一　中文文件處理機之各版本特性（以廿五史資料爲例）

	CTP 1.0	CTP 2.1	CTP 3.1	CTP 4.0	CTP 4.1	CTP 4.2
1. 機　件	IBM 5550 <16位元PC>	MICRO-VAX II,天龍570終端機	AT&T 3B15, 天龍570終端機	AT&T 3B4000 天龍530/570 終端機	386/486 PC, 天龍530/570 終端機,可接 CD ROM	386/486 PC, 天龍530/570 終端機
2.操作系統	PC-DOS	VMS 4.1	UNIX V BINIX 1.0	UNIX V, R.3.3	SCO UNIX System V /386 R.3.2	CCL/Ux R.4.0 V.2.0
3.程式語言	BASIC	C	C	C	C	C
4.查詢方式	交談式	交談式	交談式	交談式	交談式	交談式
5.文書元素	頁	頁	段落	段落	段落	段落
6.擷取方法	索引檔	索引檔	文件結構,排版結構,全文掃描	同左	同左	同左
7.廿五史之內　容	食貨志	食貨志	史記　漢書 後漢書 三國志	除表格外已完整	同左	同左
8.文書空間	400K	1200K	10M	84M bytes	84M bytes	84M bytes
9.附加空間	35K	105K	2M	15M bytes	15M bytes	15M bytes
10.使用人數	1	2-4	1-24	1-48	1-4(推荐數目)	1-4(推荐數目)
11.尋取速率	------	------	約每秒1.5萬字	約每秒2萬字	約每秒20萬字 (用486PC及 EISA Bus) 約每秒7萬字 (386 PC) [若用CD,則約每秒3萬字]	同左
12.發表日期	1985, 3月	1986, 3月	1987, 9月	1990, 9月	1992, 3月 (1992,7月發表 CD 版)	1992, 4月

二、系統設計之考量

這節中所談的，大都是在過渡期中得到的心得。首先，當然要檢討中西語文之差異；其次在設計通用的軟體原則下，檢視可資利用的訊息；而後，討論文件中有那些訊息要處理；最後，將綜合所知列爲系統設計之一些規範。茲分述如下：

語文的性質

由於中西語文的結構和性質不同，文件檢索技術因而有相當的差異，詳請參閱表二。以廿五史資料庫爲例，目前的版本中均無索引之建制，這個特色在國外是沒有先例的。這麼做的好處很多，其一是節省了相當於資料空間約1至3倍的索引空間，資料庫之空間甚小，使得小機器也很方便使用，並且使操作所需之資源和成本都跟著下降。其次，是消除了預設檢索詞彙的立場。能做到沒有索引，就是任何字串均可做索引。這個性質對使用者是相當重要的。其三，是資料庫之維護變得非常單純，因爲維護時最令人頭痛的索引沒有了。例如，要加新文件時，只須附在舊文件之後，無需費時費力地去更新全部的索引。廿五史資料庫之所以能這麼做，完全是由於中文的字串比對較英語的簡單太多的緣故，只要比對的速度能快到每秒數萬字，對一般的應用已綽綽有餘。

表二　中西語文在文件檢索工作上的差異

項　　目	西　　文	中　　文
一般		
1.文章長短	較長	較短
2.語文之統計結構	較成熟	需再研究
3.語言學知識	較成熟	需再研究
語言結構		
4.大寫字母	有,可利用	無
5.索引層次	詞,片語	字,詞,片語
6.異體	異體詞不多	異體字嚴重
7.標點間字串長度	較長	較短
8.人名集合	較封閉	無限制
9.外語	無大問題	嚴重,須特別處理
技術類		
10.接詞(hyphentation)	要,且複雜	無
11.斷詞	單純	複雜
12.剔除詞清單(stop list)	已有現成可用	視斷詞能力而定
13.字串比對	複雜,要去詞尾來產生詞幹(stems)才能比對,速度較慢	直接了當,快速
14.文章簽名之表達	可用	尚須研議

可用的訊息

設計一個通用的文件處理系統時， 對文件的內容是事先無法預知的。 如果先選定了文件， 再依其特有的內容來設計程式， 那麼就無法達到「通用」 的目的。 換言之， 這些程式必須加工修改後才能對另一個文件作有效的處理。 我們知道， 天下的文件多如牛毛， 若是不能設計出通用的文件處理程式， 則無法眞正做到大家都能享受用電腦來處理文件的目的。 文件檢索系統也正是如此。 因此， 在設計文件檢索系統之初， 來檢查一下究竟有多少與文件特有內容有關的訊息可資利用，是非常重要的事情。

在設計之初， 可資利用的訊息約可分爲四類。 其一， 是語文的統計資料 (statistical structure of language)。對語文統計資料應用得最精彩的是仙農(Claude Shannom)爲設計電報系統而發展出的消息理論(Information Theory)。馬可夫(Markoff)的Stocastic model則是另一個成就非凡的例子，這些都是大家耳熟能詳的。

其次， 是文件結構的訊息。 文件結構的訊息包括了文章的結構資料， 如卷、 篇、 章、 節、 段之類，和版面結構的資料，如頁次、行次等。這些訊息和文件在計算機中表達的方式， 以及文件檢索和列印的功能等， 都息息相關。 所幸這些訊息的抽象結構都可以用樹狀的資料結構表達，所以它亦可據以建立通用的模式。ISO 8879 SGML，就是爲此而設之標準，除上述之目的外，還可作文件分享之用。

第三，是語言學上的知識。語文的訊息，如文字、構詞、語法、語意等等，若妥爲利用，無不可用之於文件之檢索。中文須要斷詞，便是一個例子。

最後， 是一般的背景知識。 使用者在做檢索時， 常會用到許多我們已有的知識，然而文獻的檢查目前對背景知識的利用尙在初步嚐試階段， 相信以後人之智能在檢索方面亦會有其貢獻。

以上談到的四類訊息， 都和某文件特定內容無關。 如何獲得這些訊息， 是做好文件檢索的基礎；如何利用這些訊息來設計一個好的檢索系統，則是工程設計的本事。

在設計廿五史資料庫之初， 語文統計的訊息非常有限， 可資利用者不多， 然而稍後各種析詞的研究卻利用了不少統計或隨機程序的技術。 廿五史資料庫的成功， 主要是靠文件結構資料之充分利用， 捨此無他。 然而其餘的訊息均可利用來改善現有的廿五史資料庫所用的CTP。以後的發展空間，仍然很大。

文件中的訊息

以一本書爲例， 試問一本書中有多少種訊息？ 這個問題不容易回答， 但務必要一試。以廿五史爲例，則有：正文、註解、表格、序、跋、目錄、版權頁的訊息、頁碼、.....等等。這些是有形的訊息。無形的訊息則包括它的屬性，如作者、成書年代、記載年代、售價....等等，又如凡是在書目記錄中之資料均屬之。復次，上述各種訊息之間的關係如何？與此書以外之文件(如字典，其他書籍等)之關係又是如何？這些關係，也是一種訊息。這些訊息，不難想像，它們和文件的表達和檢索都有關係。

弄清楚了這些訊息，下一個課題就是要決定：在自動化以後，要留下那些訊息？正確到什麼程度？並且要將它們抽象的結構找出來，作爲檢索系統中設計資料結構的主要依據。

有些訊息是難以存留在計算機中的，例如原書中的字樣，或是紙張的性質等。若是不必要的，如紙張的性質，則可揚棄；若是必要的，則必須設法犧牲一些正確程度而保留其精華，如原書字樣必須轉換爲計算機中之文字字樣。在做廿五史之初，認爲保存原來文字之筆劃結構甚爲重要，於是，凡是計算機中之字樣與原書(鼎文版)中筆劃不同者，則造一新字。於是，在廿五史總共使用的13966字中，共造了4015字[9]。花了這麼大的工夫，只爲存其眞(而且不是百分之百的眞)。這只是許許多多此類檢討中的一個例子。在做完了全面此類檢討後，所得的決定便已定下了此文件檢索系統的品質、功能、甚至於資料和檔案的結構。

其他的考慮

在設計廿五史資料庫CTP3.0版本時，除上述之考量以外，還設定了下列的原則：

1.消除控制詞彙查詢(controlled vocabulary search)的缺點，作自由詞查詢(free-term search)。

2.使用者介面統一，跟著CTP一齊設計出來。

3.使文章結構和版面結構的訊息完全分開，各爲檢索點。

4.建立多人使用，及上網路之能力。

5.文件表達及標誌制式化。

6.應用程式的開發以電子卡系爲例，開始推廣。

以上這些決定在 [6] 中有詳細之說明，這些都是經過技術評估後認爲可行而做的決定。

三．系統實作之經驗談

這節裏，要談的是做廿五史時實際上的難題。

1. 正確性的問題

廿五史是經過「三人五校」之後才定稿的。也就是說，任何一篇稿子必須經過三個不同的人，至少五次校正才正式存入檔案。因此，校對比打字的工夫還要大許多。最近，我們開發成功了一套軟體，可以作兩篇文稿之比對，只要輸入兩次，便可由第三者以計算機輔助校對之方式校正錯誤，效果甚佳，可以省下不少人力和時間。

2. 造字的管理

造字一多，很多問題就出來了。例如，若有一字要造，怎知以前造過沒有？若造過了，如何輸入？等等問題都需要良好的規劃和管理。這些事情，對每一個新的全文

資料庫而言，都會發生。因此，如何設計一個這樣的管理系統給大家用是現在面臨的問題。這個問題還涉及異體字的整理，如果有一個自古至今完整的異體字表，那麼事情好辦得多了。否則，檢索會有問題，文字統計也做不準確[9]。

3. 表格的難題

在做廿五史之初，對廿五表格欠缺了解，總以爲可以用算式表(spread sheet)這種現成的程式就可解決。然而，事實卻不然。廿五史中表格十分複雜，到目前仍未輸入電腦。這也是留下來的習題。

4. 基本文件元素「段」之商榷

「段」的原先設想是內容完整的小單位。起先一個註便算一段，後來覺得這樣段會太多，於是一群註算一段，可是有時又太長。其次，有些段實在很小，小到只有一句話。目前廿五史共有22萬1千7百36段。若可將小段合併，將有效地減少段的數目。根據經驗，最好一段字數在200字至500字之間，這是一個可參考的數字。

5. 文件元素之自動識別值得研究

任何排版系統，對不同的文件元素，如章、節、小節、段落、註解、等等都有一定的版面排法。因此，若能將各元素版面以制式語言(formal language)表達，則可依據此表達，由打入的資料中認出各個文件元素，以及彼此之包含關係。若可做到這一點，將大幅減少標誌的工作。越基層的元素越多，其自動識別之價值越大。例如，若可自動識別段，則做廿五史時，可少打44萬3千多個識別段之標誌。

其實這是標準的句法導向識別(Syntax directed recognition)問題，要解決它應屬可行的。

6. 一個適合中文構詞特性的檢索指令

林晰 [7] 發現了一個很有用的指令，叫做排除字集查詢。即在查詢詞彙前後設有括號，而其中的字接在詞彙之前，或之後者都是要排除掉的。例如：{a}x{b}表示不要找ax或xb這樣的字串，這種檢索方式的精華是：先找x是提升其召回率(recall ratio)，而排除ax或bx則爲提高其精確率(precision ratio)。分階段查詢以提高召回率及精確率是檢索的必然趨勢[10]。而排除字集正是極佳的「相關回授」relevant feedback)，由於中文構詞的特徵，相關回授與查詢可置於同一查詢指令中。這是外國語言所沒有的方便。

7. 字串比對速率

前文已談及字串比對速率是極重要的因素。因此，如何增進其速率，亦是值得研究之課題。茲舉表三之例[7]，以說明目前最佳之成績，以爲參考。

表三　以史記 133 萬字爲例測試檢索詞數量對搜尋時間的影響（時間單位：秒）

詞數	時間	詞數	時間	詞數	時間	詞數	時間
1	5.7	5	6.7	40	12.5	200	23.2
2	6.0	10	7.1	60	12.6	300	29.5
3	6.2	15	8.4	80	13.5	400	36.7
4	6.3	20	9.6	100	15.2	500	41.4

四．現況檢討與結語

由以上之報告，我們知道廿五史全文資料庫，或CTP仍有極大改進的空間。就其主要者，分述如次：

1.文字之字形應該好好整理，這是重要的基礎工程。

2.勿忘文件分享之標準，像ODA,SGML,DSSSL,SPDL,等等，都應該參考。

3.做好文字和詞彙的統計工作，以便用以尋求文件檢索系統之最佳化。

4.語言學的知識對文件檢索關係至鉅,現在是該整合計算語言學和訊息檢索的時機[11]。

最後，讓我們談一談語意問題，這是較不易能解決的，然而它涉及系統架構之理念，其重要性不言可知。

在我們中國，自古以來一直認爲語文是人爲的產物，因此它無法百分之百的表達自然現象。不僅如此，由於每個人知識背景、經驗和所屬的環境均不一樣，對語文表達的內容之詮釋和感受也各個不同。文件檢索系統的統計，因爲和語文密切相關的緣故，也面臨上述的語意問題。

文件檢索的語意問題可用如下圖書館的例子說明。圖書系統的分類表，索引典(thesaurus)，主題詞表、同義詞表等等都由千中選一的專家定出來的，平常人對這些詞彙的了解當然沒有他們深刻。圖書登錄員(cataloger)做圖書歸類，登記書目資料的工作，也是專家，然而他們對於相關詞彙的詮釋並不見得能像前述的專家一模一樣。於是，此二者之間就已有差別存在了。再說，書本的作者對這些詞彙語意的掌握又和前兩種專家不盡相同。因此，在圖書館整理浩瀚的藏書工作中，已產生了上述語意差異所造成的不一致現象。

好了，現在要用計算機檢索，那麼，計算機內所用的機械化機制，對語意的詮釋又另有一套。如果再加上可憐的使用者，比方說是個中學生吧，他對相關詞彙的了解又是如何呢？我們知道，檢索非用到詞彙不可，如果使用者、計算機中的表達、作者、登錄員和各分類專家們對詞彙的理解都不盡一致的話，無論是人工系統也好或是自動

化的系統， 這個現象是足以影響到檢索系統的品質的， 相信大家在圖書館中都有些經驗。 我們能不能利用計算機來設計減少些此間的語意差異呢？ 我相信這是設計自動文件化檢索系統的一個重要課題。 若是各位從這個角度去檢討一下前人所發表的自動化檢索系統，你會很容易發現有許多品質問題都是由於疏忽了語意問題而產生的後果。

這個問題該如何解決呢？ 我以爲首先要釐清檢索系統中之語意關係。 譬如， 對於一個已存在的圖書系統，要自動化時，可化簡爲三個語意圈：

1.使用者

2.機器之制式機制

3.專家之分類機制（含上述之作者、編目員，以及各行業專家等構成之機制)

如果我們能在此三者間找到相互轉換之函數(mapping)，那麼就可消除彼此間語意之差異。以目前技術來看，這個系統應該是一個有學習能力的系統(learning system)，才能確實做到我們所期望的目標(mapping函數在學習中改變)。前所述及的相關回授，就是最簡單的學習形態。 此外， 類神經網路， 以及人工智慧中許多既有的工具， 亦是可考慮的嚐試對象。

如果語意差異的難關克服， 那麼眞眞正正友善易用的文獻檢索系統才會到來。 這麼說對不對呢？ 有賴您的指正。

參考資料

[1]毛漢光等，<<史籍自動化：食貨志輸入電腦，第一年總報告>>台北，中央研究院計算中心；1985,7月。

[2]曾士熊，<<國字資料庫的設計>>，台北；國立台灣工業技術學院碩士論文；1982,7月。

[3]何文雄，<<中文斷詞的研究>>，台北，國立台灣工業技術學院碩士論文；1983,7月。

[4]王義科，<<中文文件的處理>>，台北，國立台灣工業技術學院碩士論文；1983,7月。

[5]張仲陶等，<<The development of software tools for sinology>>，美國資訊學會(ASIS)1983年會發表；1983,10月3日。

[6]謝清俊等，<<中文全文處理系統的設計與製作>>，台北，中央研究院計算中心，技術報告T0003，1986年9月24日。

[7]林晰，<<文獻層級結構運用於全文處理的研究>>，台北，中央研究院計算中心，技術報告T0016，1990月12月22日。

[8]曾士熊等，<An Experimental Model of Chinese Textual Database>，台北，中國工程學報，13卷6期，1990年6月。

[9]謝清俊等，<廿五的文字統計與分析>，台北，第三屆中國文字學國際學術研討會，1992,3月21～22日。

[10] Gerard Salton，<<Autsmatic Text Processing>>，Addison-Wesley公司出版，1989。

[11]最近ACM的SIGTP 92'論文集，以及計算語言學會年會論文集CL92'中均出現不少此類之呼籲及研究的論文。