

FULL TEXT PROCESSING
OF
CHINESE LANGUAGE

— an experimental system for studying Chinese History Literatures†

*Ching-Chun Hsieh**

Introduction

Full-text databases and associated processing technology are becoming increasingly important for library automation, on-line information service systems, and office information systems. They also play a key-note in developing new computer applications where automation of the filing and search procedures will free workers from these time-consuming tasks.

Some major research and development topics related to full-text database systems now underway include the studies of document representation, access methods for text, text analysis (in natural language), and the new hardware for text retrieval. An excellent review of text access methods and retrieval hardware was given by Faloutsos.¹ And, an excellent position paper about full-text databases was given by Tenopir². Since full-text of document is usually in natural language form, the studies

† Paper presented at the SEAL Sub-Committee on Library Technology, AAS 38th Annual Conference, Chicago, 1986.

* Dr. Hsieh is the Research Fellow, Institute of Information Science, Academia Sinica, R O C.

mentioned above are unavoidably language dependent, ie they may not directly applicable to those full-text processing systems which are not in English. This fact becomes more obvious while Chinese full-text database systems are confronted, although at present, only quite limited and scattered effort has been devoted to the development of full-text database systems in Chinese language.

In this paper, an experimental Chinese text processor (CTP) for studying history literatures will be presented. The system provides both free-text and controlled vocabulary search for the text. The goal of this study is to find out the feasibility of using computers, especially small or microcomputers, as a tool for history studies. Our experiment shows that by applying existing technology, such a system is highly feasible.

The System

The project of CTP is a part of a long-term research plan, namely Computer Applications in Humanities(CAH) conducted by the Computing Center of Academia Sinica, Taipei, Taiwan, ROC. The project CTP was started in July, 1984. At present, the CTP is still an on-going project in its second phase. Therefore, as time goes, the system is unavoidably subjected to change. For the time being, we have two operational prototypes on a mini and a 16-bit micro, respectively.

A basic conceptual configuration of the CTP is shown in Figure-1. It can be considered basically as a workstation for humanity studies. As comparing the CTP to an information retrieval system, the major differences are: the abstract text-structure supporting mechanism, more supporting tools, and a reinforced application section which supports documentation generation functions, i.e. user's note-card system^{7,8}, word processor etc. The most important supporting tool, at present stage, is considered to be the auto-indexing program. These major

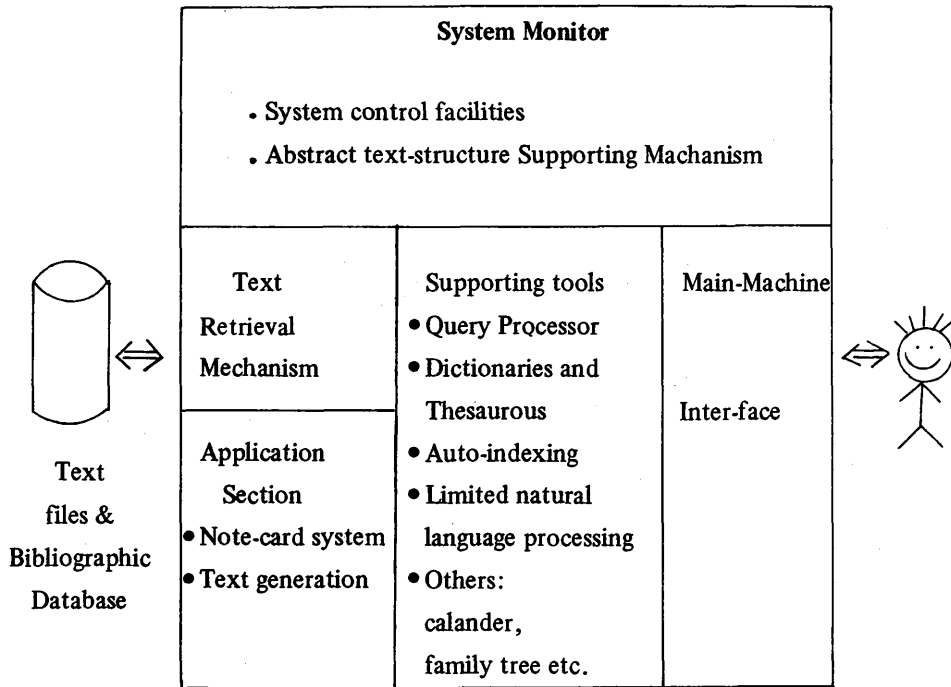


Figure-1: Base Configuration Concept of CTP

differences will be presented in the subsequent sections.

The CTP is designed to facilitate the access of full-text documents, creating user's own private index systems to both the original text and user's own note-cards, and the generations of new documents during/after studies.

Text Representation

The object document in the full-text data-base under testing consists of all volumes of 食貨志 (for food and goods in meaning) from the history books of 24 Dynasties (廿四史) of ancient China. An example of 食貨志 is shown in Figure-2. It is of the form of an ordinary text book. In this paper, we will use full-text or text interchangeably to indicate such a type of document. A complete list of the 食貨志 and associated number of characters in each book is shown in Table-1.³

One of the reasons to choose 食貨志 as object document is that there is no tables and figures in them. Therefore, its logical structure is simple and it is very nature to use a tree-structured representation scheme in computer to describe the text organization of 食貨志. The tree-structured representation of 食貨志 is shown in Figure-3. In Figure-3A, a logical structure of a generalized text-tree is presented. It is implemented as a part of the system monitor. In Figure-3B, an example of the physical structure of 食貨志 is shown. Each terminating node of this graph represents a paragraph of the text, respectively. Of course, for a more detailed representation, sentences may be considered to be the terminating nodes. But for the moment, we choose paragraph as the smallest unit used in the internal representation of text. If any address to the sentence level is necessary, it will be carried out by a text-scan and analysis program within a chosen paragraph.

The advantage of separating the logical structure and physical structure of the text is obvious: the system is generalized to have

宋史卷一百八十一

志第一百三十四

食貨下三

會子 鹽上

會子、交子之法，蓋行取於前之飛錢。既際時，陳詠鎮關，民間人餽錢真不便貿易，設
 質劑之法，一交一稱，以三年爲一限而換之。六十五從爲二十二界，謂之交子，實限十六戶
 主之。後富民私積錢不能償所負，爭訟不息。轉運使傅頤、限禁存留益州交子券，以楮
 共出入，私造者禁之。仁宗從其議，界以百二十五萬六千三百四十稱爲額。

神宗熙寧初，立爲鐵陌貫如官印文律法。河東運鐵錢勞費，公私苦之。二年，乃明置
 交子券于關州。轉運司以其法行則鑄、鑄不傳有害入中積弊，遂廢罷之。四年，復行於陝西，

志第一百三十四 食貨下三

四〇三

史記卷三十

平準書第八

漢初之時，興與，接之際，夫去從軍旅，老弱相講，作業劇而財匱，自天子不能具鉤陳，二而
 今，將伯政乘生事，薄民無城，三於是爲，陰發軍難用，已更令民鑄錢，已黃金一斤，已
 物價高漲，約法省禁，更不軌逐利之民，游積餘業以積市物，物踊騰貴，已米石百錢，馬一匹則百
 金，已

漢初之時
 今
 物價高漲

- (一) 國 天子不能具鉤陳，即作府庫，即更民向城一邑也。
- (二) 國 興與，即興與，即興與。
- (三) 國 夫去從軍旅，即去從軍旅，即去從軍旅。
- (四) 國 老弱相講，即老弱相講，即老弱相講。
- (五) 國 作業劇而財匱，即作業劇而財匱，即作業劇而財匱。
- (六) 國 自天子不能具鉤陳，即自天子不能具鉤陳，即自天子不能具鉤陳。
- (七) 國 今，即今，即今。
- (八) 國 將伯政乘生事，即將伯政乘生事，即將伯政乘生事。
- (九) 國 薄民無城，即薄民無城，即薄民無城。
- (十) 國 於是爲，即於是爲，即於是爲。
- (十一) 國 陰發軍難用，即陰發軍難用，即陰發軍難用。
- (十二) 國 已更令民鑄錢，即已更令民鑄錢，即已更令民鑄錢。
- (十三) 國 已黃金一斤，即已黃金一斤，即已黃金一斤。
- (十四) 國 已米石百錢，即已米石百錢，即已米石百錢。
- (十五) 國 馬一匹則百金，即馬一匹則百金，即馬一匹則百金。

平準書第八

一四七

Figure-2: An example of the text of 食貨志

Name of the books 書 名	Number of characters 字 數
漢書 (上、下)	3 6 4 1 1
史記	1 4 3 4 3
舊唐書 (上、下)	2 4 8 0 7
唐書 (一至五)	3 1 3 8 0
隋書	1 2 0 5 3
魏書	1 1 6 1 5
晉書	1 1 0 8 5
舊五代史	8 8 1 0
遼史 (上、下)	4 9 5 6
金史 (一至五)	5 2 3 2 0
宋史 (上、下)	2 1 4 9 9 4
元史 (一至五)	6 8 5 3 3
明史 (一至六)	6 8 5 1 3
Total number of characters 共 計	5 5 9 8 2 0

Table-1: All the volumes of 食貨志 and their number of characters

the capability of handling more tree-structured texts.

Access Method

The system provides both the controlled vocabulary search and free-text search of the text database. At present, for controlled vocabulary search, only single key-word search is available. The keyword sets that the system provides include: the name of person (人名, 字號等), the name of place, the name of official titles (職官名), the name of books appeared in the text, the expressions of time in terms of an era, a dynasty or an emperor (朝代, 時代, 帝號, 年號, 名號名等) and 10 classes of special terminologies such as 賦稅類 (the class of tax terms), 鹽鐵類 (the class of salt and metals terms) etc. A more detailed example can be found in Table 2.³

For each set of keyword, an inverted index file which maps each keyword onto the related paragraphs of the text is generated in the system generation phase. A program that automatically generates these inverted index files will be described in the next section. At present, only a manual-driven man-machine interface is available for users to retrieve text by these system keywords. No logical operators are available at this stage.

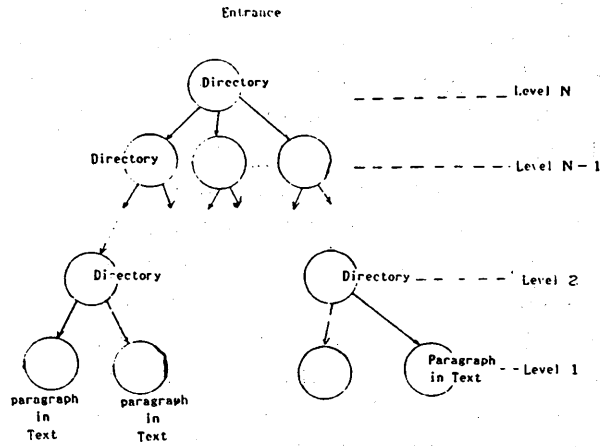
Besides these keywords provided by the system, one important theme of designing this system is to support user defined keyword searching facilities through the application section. In other words, this system supports two categories of keyword; one is in public domain to be used by all users and provided by the system, and another is merely created and used by a user in private manner.

The text-tree also provides a way of retrieval. It behaves like a content table of a book. Besides, it also provides proximity functions that will limit the search of text within a reasonably restricted scope.

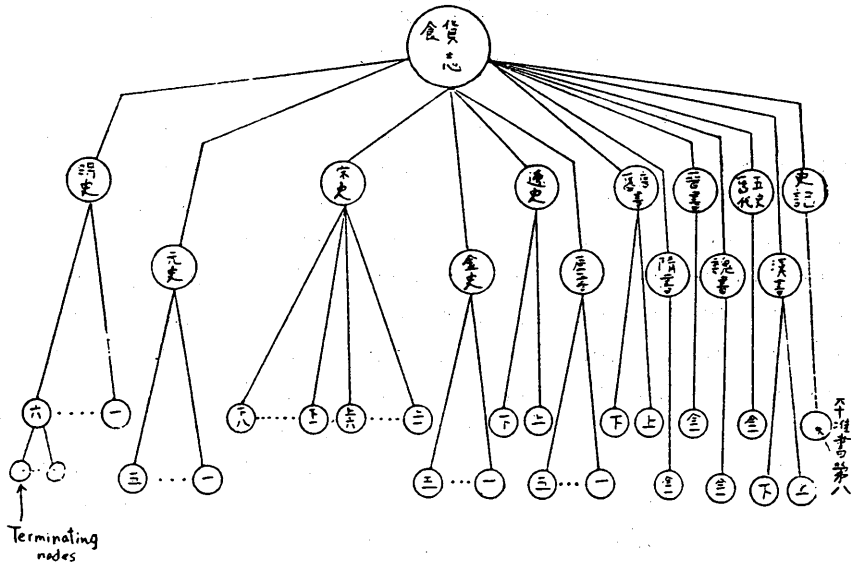
For the free-text search, users are always been asked to

term classes	Example																
田制類	井田 阡陌 度世	田之制 之之法 之業	口世 之之 之荒	分業 之田 之荒	上下 之荒 之營	曰田 田田 田田	聖大 田田										
水利類	纏戊 屯井	田田 田田	占代 井王	田田 田田	湯糶 民名	沐田 田田	邑田 田田	歸公 私公	田田 田田	步畝 夫屋							
賦稅類	什初 履十	一稅 一之	賦稅 什租	斂租 五稅	而稅 一稅	三十一 什租	而稅 一稅	賦稅 租租	算民 租租	什賦 五租							
雜稅類	課租 兩田	調法 庸稅	秋賦 課庸	稅賦 課庸	調租 庸法	兩稅 錢賦	稅錢 錢賦	上留 送征	供州 使役	折賦 正貢	納役 賦賦	租地 庸租	調 庸租				
戶口類	六漏 逃雜	部戶 戶營	民戶 戶營	役口 夫夫	奴良 奴男	婢良 奴夫	老小 羸殘	樂男 三陰	遷女 長附	隣里 黨復	長長 長長	力里 三屯	役黨 長民	戶門 民			
錢財類	比四 沈比	論文 即輸	錢錢 錢錢	五大 當小	銖錢 千錢	銖泉 鑄府	錢刀 龍龜	鑄錯 朋元	白鑄 赤白	鹿錢 劍金	幣皮 劍金	半三 盜復	兩銖 鑄小	錢錢 錢錢	赤紫 赤白	劍劍 劍劍	錢錢 錢錢
鹽鐵類	鹽鹽 鹹兩	法勤 鹹池	穎私 末刮	鹽鹽 鹽官	煎煎 煎官	鹽鹽 鹽戶	鹽鹽 鹽法	煎青 青白	鹹白 鹽鹽	鹽食 大甜	鹽鹽 鹽次	鹽鹽 冷鹽					
市糶類	六除 輕平	輸貨 重糶	均常 贖五	輸平 茶均	粟泉 倉泉	人甘 太甘	穀金 輸準	人耐 均平	糶糶 常拜	犁犁 平拜	買贖 輕順	尉禁 重租	錮錮 錮錮				
漕運類	漕漕 漕合	運米 吏糶	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉	倉倉 倉倉
非財經	五西 胡漢	帝戎 都	三皇 三皇	長澤 六	城和 坊	百華 男寬	保人 士鄉	鮮卑 鮮卑	八十 軍侍	丁二 士官	兵丁 兵	兵丁 兵	武基 瑠官	偃隆 玳六	夏俗 突吐	人陽 宮谷	人最 宮文

Table-2: The 10 classes of special terminologies used as keyword set for text retrieval.



3A: A Logical Structure of the text tree



3B: The physical structure of 食貨志

Figure-3: Tree-Structured representation of text

define the scope of the search through the text-tree. The free-terms can be any length of consecutive characters. After a successful free-term search, the user may ask the system to update his/her private index file by adding the newly found entry.

As a final remark of this section, we like to point out that the nodes of text-tree are excellent entry points for inverted index files. Linking the text-tree to the indexing system can provide all information about the organization and associated attributes of the text to the retrieval system. And thus, greatly enhanced the performance of the system. For instance, a field in a node descriptor may be used to describe the length of the text covered by that node. And from this information, we may calculate precisely the time needed for a free-term search in advance. As another example, a broader-term in an thesaurus may be assigned to a node of the text-tree automatically if its aucessor nodes are indexed by most of the terms under that broader-term.

An auto-indexing method for Chinese text

The importance of auto-indexing ability to an information retrieval system is obvious and has been discussed in many papers^{1,2}. For Chinese text database systems, the auto-indexing ability is even more important and can be considered as a necessity for overcoming the feasibility requirements of implementing such a system. In this research, a fairly simple auto-indexing method has been proposed and, to our suprise, the results are especially encouraging. The method is described as follows.

In Figure-4, a block diagram of the auto-indexing system is presented. The inputs to the auto-indexing system are the text file and the keyword file. The first part of the auto-indexing system is a matching program that scans each characters in the text file and matches it against a tree of words^{4,5} in the keyword file. An example of a tree of keywords are shown in Figure-5.

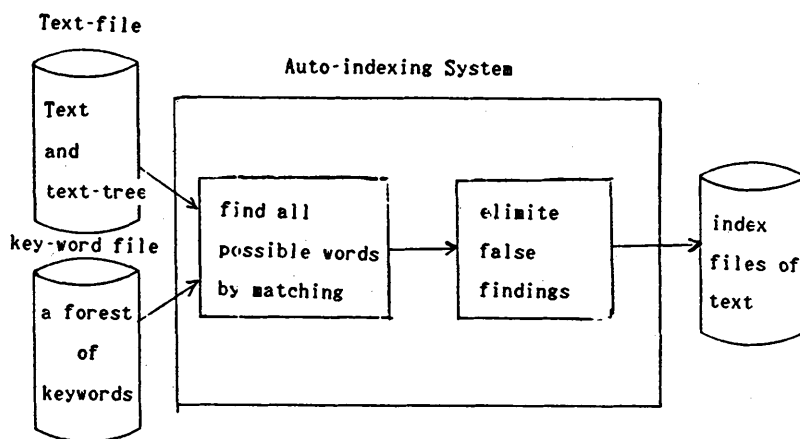


Figure-4: A block diagram of Auto-indexing method

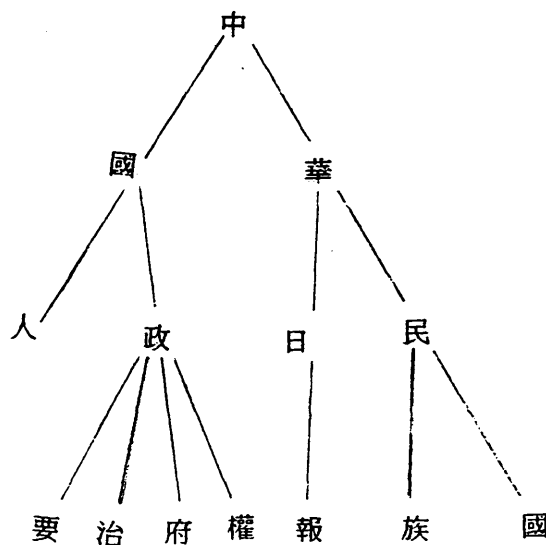


Figure-5: An example of a tree of words, that share the same leading characters. The node with a circle indicate a termination of a word.

As you see that all of the keywords that share the same leading characters will be organized in a tree. Therefore in the keyword file, there is a forest of keyword trees.

By this arrangement, the time needed for matching can be reduced. Let n denotes the number of character in text and T denotes the average time required to match a character against a tree of keywords, then, the total time required to build the index file is nT . Of course, the result is application dependent. It depends upon the content of the text and the complexity of the trees of keywords. For the text of ancient document, we found that each tree is rather simple, most of them are trieval, i.e. only one or two words in a tree. Since hashing technique can be easily applied to located the related tree for a specific character, the value of nT is much less than the time required by a straight-forward matching method whose matching time required is $n.m.l.t$, where l represents the average length of keywords, m the total number of keywords in the forest and t the time required to match two characters.

As for the recall and precision functions of the retrieval method are concerned, an example is given in Table-3. It can be easily proved that according to our indexing method, the index file made by human professionals is a sub-set of the index file found by the machine. According to Table-3, the percentage of the false indexes are very low. It is usually less than 5%.

The second part of the auto-indexing system is a text analysis program that will eliminate some possible false indexes. The text analysis program consists of a rule-based analyzer⁶ which will eliminate some false findings by examining the syntax information obtained during matching. An example is given in Figure-6. Since the number of false indexes are considerably small, this correcting process may be omitted in some applications.

Application section

In our system, an application section is designed for inter-

```

#---- 以下為 auto-index 所建立的資料 ----#
#
#      621 Mar 11 09:04 his2azzbk.ind      #
#      156 Mar 11 09:04 his2azzbk.iov     #
#     2622 Mar 11 09:04 his2azznm.ind     #
#      416 Mar 11 09:04 his2azznm.iov    #
#     5589 Mar 11 09:04 his2azznu.ind     #
#      104 Mar 11 09:04 his2azznu.iov    #
#     2070 Mar 11 09:04 his2azzof.ind     #
#         0 Mar 11 09:04 his2azzof.iov    #
#     6279 Mar 11 09:04 his2azzpl.ind     #
#       728 Mar 11 09:04 his2azzpl.iov   #
#     1587 Mar 11 09:04 his2azzyr.ind    #
#       572 Mar 11 09:04 his2azzyr.iov   #

```

Totle memory space : 20744 bytes

```

#---- 以下為 partial auto-index 所建資料 ----#
#
#      621 Mar 11 09:54 his2mzzbk.ind    #
#      156 Mar 11 09:54 his2mzzbk.iov   #
#     2622 Mar 11 09:54 his2mzznm.ind    #
#      416 Mar 11 09:54 his2mzznm.iov   #
#     5589 Mar 11 09:54 his2mzznu.ind    #
#       52 Mar 11 09:54 his2mzznu.iov   #
#     2070 Mar 11 09:54 his2mzzof.ind    #
#         0 Mar 11 09:54 his2mzzof.iov   #
#     6279 Mar 11 09:54 his2mzzpl.ind    #
#       208 Mar 11 09:54 his2mzzpl.iov   #
#     1587 Mar 11 09:54 his2mzzyr.ind    #
#       624 Mar 11 09:54 his2mzzyr.iov   #

```

Totle memory space : 20197 bytes

The difference : 547 bytes (about 2.6%)

Table-3: A comparison of Auto-indexing and human corrected partial auto-indexing of all index files in 遼史食貨志 .

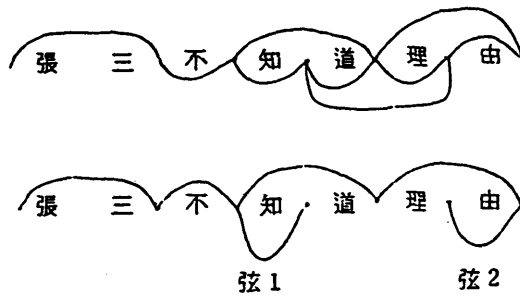


Figure 6 An example of eliminating false indexes by text analysis

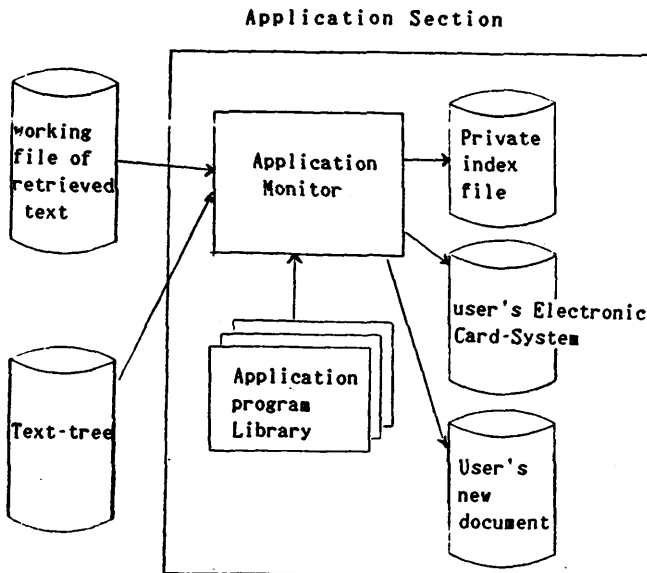


Figure-7 A block diagram of Application Section and its operational environment.

facing the retrieved text to user's applications. At present, the application section provides two major functions; to establish user's own private indexing system and to generate a card-system after user studied the text. A block diagram that shows the application section and its related system parts is shown in Figure-7.

The application section will be activated by user's request during retrieval. It provide users a manual-driven interface. Through the manual, the user can establish his/her own private index file, and generates a deck of cards on which a memo or a digest of the original text can be written. After a deck of cards has been established, the user can use them to create new document, such as essays, study reports, etc.

Conclusion

For illustration, two examples of search are given in Figure-8 and Figure-9, respectively.

By the experience of project CTP, we like to state the following comments as the conclusion of this paper:

1. There is a high demand of automating tool books usually referenced by humanist. Most of these tools are knowledge-based in nature. And thus, the automation involves the application of knowledge-based systems and AI technology. This is a field that open to us for future development.
2. Natural language processing capability will directly enhance the system performance and improve the friendliness of man-maching interface. Even with very limited natural language processing techniques, such as morphological analysis, do help a lot.
3. The representation o text is still worth doing research.
4. Application section needs further enhancement. Word processor, text editor spread-sheet, note-card system,

statistical package, etc, can greatly facilitate the power of text-processing.

Acknowledgement

The author would like to express his sincerely thanks to the Council for Culture Planning and Development, Executive yan (行政院文化建設委員會) for supporting this project.

References

1. Christor Faloutsos, "Access methods for text", *Computing Surveys*, *ACM* volume 17. NO. 1, March 1985.
2. Carol Tenopir, "Full-text databases", *Annual Review of Information Science and Technology*, vol. 19, pp. 215-246, ASIS, 1984.
3. 毛漢光：史籍自動化第一年總報告，中央研究院，台北市，中華民國，1985。
4. C. C. Hsieh, "Experiments on the search and recognition of word from Chinese text", The first Asia-Pacific Conference on Library Science, Taipei, ROC, March, 1983.
5. 何文雄，中文斷詞的研究，國立台灣工業技術學院 工程技術研究所 碩士論文，July, 1983.
6. 陳正佳，一套中文語法分析系統的研究與設計，國立台灣大學 資訊工程學研究所碩士論文。June, 1985。
7. 林瑞華，卡系與資料整理，中央卡系統推廣中心，鳳山，高雄，台灣，中華民國，1978。
8. 梅棹忠夫原著，余阿勳、劉焜輝譯，知識誕生的奧秘，晨鐘出版公司，台北市，中華民國，1976。

代號	內容	書名	頁數	段數
17	尚書度支部中	欽定	2850	2
18	尚書度支部	欽定	2860	2
19	尚書令	欽定	2803	4
20	尚書令	尚書	880	2
21	尚書左丞	尚書	670	4
22	尚書左丞	尚書	2122	4
23	尚書左丞	尚書	1344	2
24	尚書省	尚書	2088	1
25	尚書省	尚書	2080	2
26	尚書省	尚書	2118	1
27	尚書省	尚書	1300	3
28	尚書省	尚書	1372	1
29	尚書省	尚書	1400	6
30	尚書省	尚書	1402	1
31	尚書右僕射	尚書	2118	4
32	尚書比部	尚書	1347	3

編號7(V/N)

1 史記 2 漢書 3 魏書 4 魏書 5 隋書 6 隋書 7 隋書 8 隋五代 9 0

(中文) (半形) 字根:

代號	內容	書名	頁數	段數
33	尚書省郎中	尚書	1404	1

編號7(V/N)

1 史記 2 漢書 3 魏書 4 魏書 5 隋書 6 隋書 7 隋書 8 隋五代 9 0

(中文) (半形) 字根:

請閱入款印代號: 20 後復換行單
 請閱入款印代號: 20 後復換行單
 請閱入款印代號: 1(0:真印, 1:模印)後復換行單
 請閱入款印代號: 0(0:顯示, 1:印或模)後復換行單

代號	內容	書名	頁數	段數
1	尚書	尚書	782	1
2	尚書	尚書	780	1
3	尚書	尚書	782	3
4	尚書	尚書	703	2
5	尚書	尚書	703	2
6	尚書	尚書	703	2
7	尚書	尚書	708	1
8	尚書	尚書	708	1
9	尚書	尚書	2852	2
10	尚書	尚書	2850	2
11	尚書	尚書	2802	2
12	尚書	尚書	2805	2
13	尚書	尚書	075	2
14	尚書	尚書	082	2
15	尚書	尚書	2003	1
16	尚書	尚書	1402	4

編號7(V/N)

1 史記 2 漢書 3 魏書 4 魏書 5 隋書 6 隋書 7 隋書 8 隋五代 9 0

(中文) (半形) 字根:

代號	內容	書名	頁數	段數
1	尚書	尚書	782	1
2	尚書	尚書	780	1
3	尚書	尚書	782	3
4	尚書	尚書	703	2
5	尚書	尚書	703	2
6	尚書	尚書	703	2
7	尚書	尚書	708	1
8	尚書	尚書	708	1
9	尚書	尚書	2852	2
10	尚書	尚書	2850	2
11	尚書	尚書	2802	2
12	尚書	尚書	2805	2
13	尚書	尚書	075	2
14	尚書	尚書	082	2
15	尚書	尚書	2003	1
16	尚書	尚書	1402	4

編號7(V/N)

1 史記 2 漢書 3 魏書 4 魏書 5 隋書 6 隋書 7 隋書 8 隋五代 9 0

(中文) (半形) 字根:

Figure-9 An example of search by “尚書”