Catalog Records Retrieved by Personal Author Using Derived Search Keys

Alan L. LANDGRAF and Frederick G. KILGOUR: The Ohio College Library Center

This investigation shows that search keys derived from personal author names possess a sufficient degree of distinctness to be employed in an efficient computerized interactive index to a file of MARC II catalog records having 167,745 personal author entries.

Previous papers in this series and experience at the Ohio College Library Center have established that truncated derived search keys are efficient for retrieval of entries by name-title and title from large on-line computerized files of catalog records.[1-4] Experiments reported in the earlier papers were ". . . based on the assumption that each key had a probable use equal to all other keys."[5] However, Guthrie and Slifko have shown that random selection of entries, rather than keys, yields results closer to actual experience but with a higher number of entries per reply.[6] For example, they found on retrieving from a file of 857,725 records using a 4, 5 (four characters of main entry, five characters of title) key that when the basis of the search was random keys there was one entry per reply 81.3 percent of the time, but when the basis was random records, there was one entry per reply 55.7 percent of the time.

This paper presents the results of experimentation with search keys to be used in constructing an author index to a large file of on-line catalog records. An interactive environment is assumed, with the interrogator employing a remote terminal. A companion paper describes the findings of an investigation into retrieval efficiency of search keys derived from corporate author names.[7]

MATERIALS AND METHODS

The investigation employed a MARC II file containing approximately 200,000 monographic records from which a computer program extracted 167,745 personal-name keys. The program extracted these keys from main entry, series statement, added entry, and series added entry fields. The basic key structure consisted of sixteen characters—the first eight from the surname, the first seven from the forename, and the first character from the middle name (8,7,1). If the surname and forename contained fewer char-

NO. OF CHARACTERS EXTRACTED FROM THE FORENAME WITHOUT AND WITH THE MIDDLE INITIAL		LIKELIHOOD	NO. OF CHARACTERS EXTRACTED FROM THE SURNAME: 3	4	5	6
0		90.00% 99.00% 99.50%	(>200)	(>200)	(>200)	171 (>200)
		90.00% 99.00% 99.50%				
1	w/o	" " "	67 172 (>200)	25 90 105	18 71 102	16 63 87
	with	" " "	16 55 67	8 25 36	6 23 32	6 11 30
2	w/o	" " "	26 87 106	12 44 62	9 38 57	
	with	" " "	8 29 37	5 21 30	5 21 30	
3	w/o	" " "	17 50 78			
	with	" " "	5 23 31			

Fig. 1. Number of Names Retrieved 90, 99, and 99.5 Percent of the Time for Different Key Structures

acters than the key segment to be derived, the segment was left-justified and padded out with blanks. If there was no middle name or middle initial, a blank was used.

Another program derived shorter keys from the 8,7,1 structure ranging from 3,0 to 5,2,1. Next, a sort program arranged the shorter keys in alphabetical order. A statistics collection program then processed the alphabetical file. This program counted the number of distinct keys, built a frequency distribution of names per distinct key and cumulative frequency distributions of names per distinct key in percentile groups.

RESULTS

Figure 1 presents the findings at three levels of likelihood for retrieving *n*

Table 1. Number of Names Retrieved With 90 Percent Likelihood

No. of Characters	*No. of Names Retrieved*	*Key Structure*
3	(>200)	3,0
4	(>200)	4,0
	(>200)	3,1
5	(>200)	5,0
	26	3,2
	25	4,1
	16	3,1,1
6	171	6,0
	18	5,1
	17	3,3
	12	4,2
	8	3,2,1
	8	4,1,1
7	16	6,1
	9	5,2
	6	5,1,1
	5	3,3,1
	5	4,2,1

or fewer names when a variety of search key combinations were employed ranging from three to six characters from the surname, zero to three characters from the first name, and with or without the middle initial. Table 1 is an extraction from Figure 1 and contains the number of names retrieved at a level of 90 percent likelihood for the various search keys employed.

Figure 2 has the same structure as Figure 1 but contains the degree of distinctness as percentages,

$$\left(\frac{\text{no. of distinct keys}}{\text{no. of entries}}\right) \times 100 \text{ percent.}$$

Table 2 records distinctness arranged by number of characters per key. Figure 3 is a graphical representation of the degrees of distinctness of the various keys. In this figure, different types of lines connect points representing key structures that contain an equal number of characters.

The bottom line in Table 1 may be read as saying that 90 percent of the time a 4,2,1 key will retrieve five or fewer names from a file of 167,745 personal name keys. The bottom line of Table 2 states that from the same file the 4,2,1 key yields a single name 64.1 percent of the time.

DISCUSSION

This experiment has shown the degree of distinctness—that is to say, the number of distinct keys divided by the total number of entries from which all keys were derived—to be a useful tool in determining what key structures may be efficiently used. As seen by comparing Figure 1 with Figure 2 and Table 1 with Table 2, there is a high degree of correlation between distinctness and the likelihood of retrieving a certain number of names 90,

NO. OF CHARACTERS EXTRACTED FROM THE SURNAME

NO. OF CHARACTERS EXTRACTED FROM THE FORENAME WITH AND WITHOUT THE MIDDLE INITIAL		3	4	5	6
0	0	2.271	9.934	19.220	24.587
	1	—	—	—	—
1	0	17.106	35.360	44.850	48.345
	1	44.551	57.148	61.449	62.891
2	0	34.676	49.870	55.803	
	1	56.979	64.155	66.186	
3	0	44.914	56.294		
	1	66.133	66.599		
4	0				
	1				

Fig. 2. Degree of Distinctness in Percent for Different Key Structures

Table 2. Distinctness by Number of Characters Per Key

No. of Characters	*Degree of Distinctness*	*Key Structure*
3	2.3	3,0
4	9.9	4,0
	17.1	3,1
5	19.2	5,0
	34.8	3,2
	35.7	4,1
	44.5	3,1,1
6	24.6	6,0
	44.9	5,1
	44.9	3,3
	49.9	4,2
	57.0	3,2,1
	57.1	4,1,1
7	48.3	6,1
	55.8	5,2
	56.3	4,3
	61.4	5,1,1
	62.1	3,3,1
	64.1	4,2,1

99, or 99.5 percent of the time. Thus, the investigator can eliminate many undesirable key structures on the merits of distinctness alone and pool his remaining resources toward studying in detail other structures.

When the 8,7,1 key was tested, it yielded a uniqueness percentage of

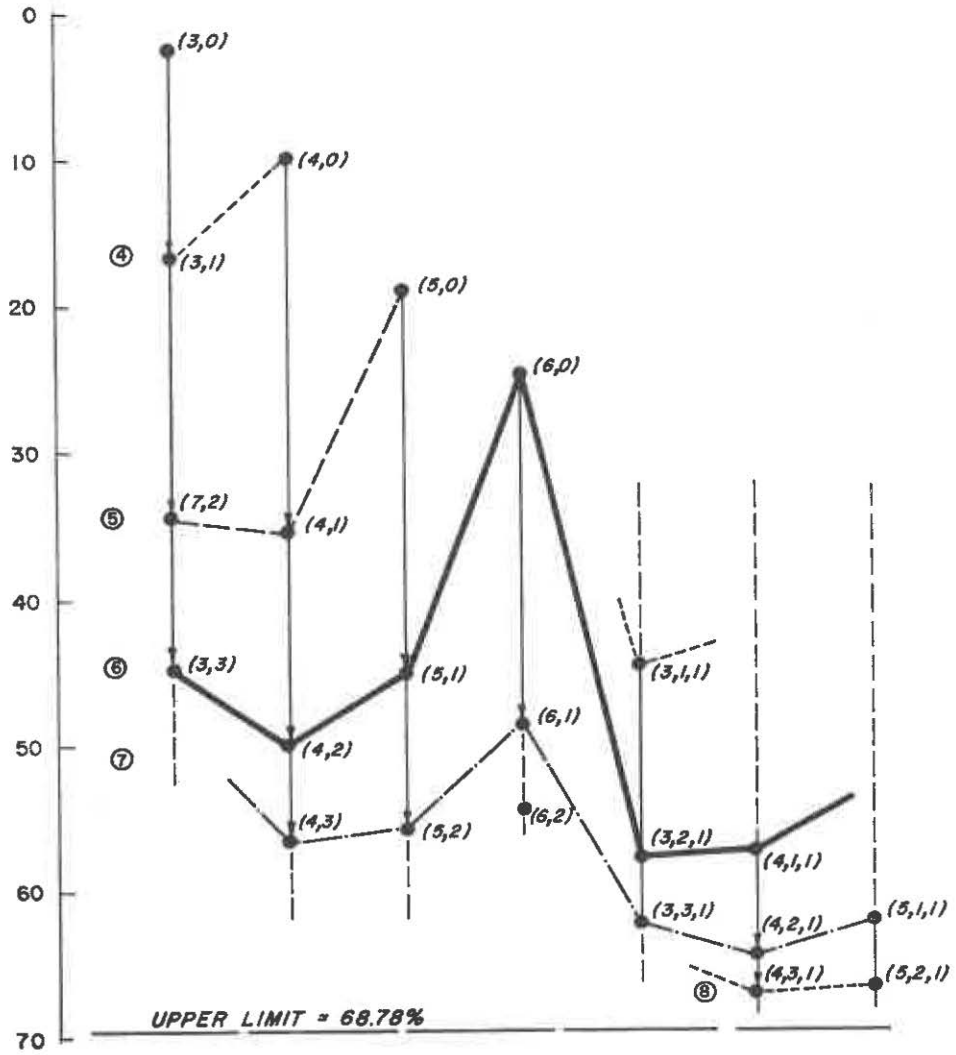

Fig. 3. Degree of Distinctness. Lines Connect Points Whose Key Structures Have an Equal Number of Characters

68.8 that represents the upper limit of uniqueness in this experiment. From Table 2 it is apparent that the bottom three keys yield a percentage of uniqueness near the upper limit.

Table 2 shows a distinct jump in percentage of uniqueness between the *n,0* and *n,1* key structures. Another sharp increase occurs between *n,m* and *n,m,1* structures. Each section of the key is derived from a Markov string, and it appears from the discontinuities between sections that the parts of personal names are not highly correlated.

As pointed out in previous papers, a key structure that possesses a rela-

tively high degree of distinctness also yields a small percentage of replies containing many entries. For the name-only search key, this effect could be reduced by performing the retrieval in two steps when necessary. First, the full names for each author whose name matches the entered search key would be displayed; names appearing with more than one work would be displayed only once. Next, the retriever would choose the name desired and request all of the titles associated with it. However, some title displays could be excessive—William Shakespeare's name appears with more than 500 works. A paper currently in preparation at OCLC describes an algorithm whose interactive use resolves this type of search problem.[8]

CONCLUSION

This investigation has yielded findings showing that there are several truncated search keys derived from personal names that are sufficiently specific to perform efficiently as an author index to a file of 167,745 personal names, thereby providing an on-line index that will make it possible for a terminal user to obtain a listing of all titles by a given author in an on-line catalog.

ACKNOWLEDGMENT

This study was supported in part by Office of Education contract OEC-0-72-2289 (506) and Council on Library Resources grant CLR-526.

REFERENCES

1. P. L. Long and F. G. Kilgour, "A Truncated Search Key Title Index," *Journal of Library Automation* 5:17–20 (March 1972).
2. F. G. Kilgour, P. L. Long, E. B. Leiderman, and A. L. Landgraf, "Title-Only Entries Retrieved by Use of Truncated Search Keys," *Journal of Library Automation* 4:207–310 (Dec. 1971).
3. F. G. Kilgour, P. L. Long, and E. B. Leiderman, "Retrieval of Bibliographic Entries from a Name-Title Catalog by Use of Truncated Search Keys," *Proceedings of the American Society for Information Science* 7:79–82 (1970).
4. F. G. Kilgour, P. L. Long, A. L. Landgraf, and J. A. Wyckoff, "The Shared Cataloging System of the Ohio College Library Center," *Journal of Library Automation* 5:157–183 (Sept. 1972).
5. Long and Kilgour, "A Truncated Search Key," p.18.
6. Gerry P. Guthrie and Steven D. Slifko, "Analysis of Search Key Retrieval on a Large Bibliographic File," *Journal of Library Automation* 5:96–100 (June 1972).
7. K. B. Rastogi, A. L. Landgraf, and P. L. Long, "Corporate Author Entry Records Retrieved by Use of Derived Truncated Search Keys," *Journal of Library Automation* in press.
8. J. A. Wyckoff, "A Technique for Extending Searches through Large Numbers of Duplicate Matches," in Preparation.