9 Conclusions & recommendations

9.1 Introduction

This chapter starts with a brief discussion of users' current expectations from interactive computer systems. Sections 9.2 - 9.4 bring together some of the results of the evaluation described in Chapter 8, for each of the three devices which were under test. We try to give answers to the evaluation questions listed in Section 8.1.

It is impossible to do meaningful research in the design of online catalogues, or other interactive computer systems for untrained users, without a testbed system which is finished to quite a high standard. The system cannot be used for the collection of realistic and representative data unless users perceive it as being a proper tool for the job - the *job*, in our case, being the location of books on given subjects in a library. Four or five years ago this would not have been true, but by now the great majority of users have had previous experience of interactive computer programs - computer games on home micros, fruit machines, cash dispensing machines, viewdata systems and online catalogues. Users expect these programs to reach certain standards of acceptability, suitability and performance.

A few years ago almost any online catalogue in a library was greeted with enthusiasm, and users tended to blame themselves for failures (see, for example, [1, Appendix 5]). This is no longer true. None of our interviewed users said anything which suggested they might think their failure was connected with the way they searched rather than the way the computer processed their search (admittedly we did not *ask* them this). Nor was there any comment to this effect in the suggestion books by the terminals.

The easiest way to evaluate our devices with respect to fairly crude, but "objective", measures of recall and precision would have been to implement them in a retrieval system specifically designed for the repetition by experimenters of searches collected from the use of a real system. This would have avoided all the complications of presenting the devices acceptably in an online catalogue, which has to be extremely simple to use.

Before the proposal was submitted we knew, from repetition of searches from Okapi '84 logs, that some degree of automatic stemming and the use of cross-reference tables would be beneficial. We also knew the extent of the problems

caused by miskeyings and misspellings (although we did not know whether attempted automatic correction would be worth the overheads). What we did not know was how the devices should be presented. The investigation of *presentation*, of methods of *incorporating the devices in an online catalogue*, was the most important part of the research.

About two person-years was spent on program design and programming during this project, although some of this was work towards a relevance feedback system (the subject of another project). We did not have as much time for the collection and analysis of data as we would have liked. We do have a wealth of transaction log data - from some 5000 sessions at the time of writing - mostly from use of the *EXP* system. There are several research projects using our data and systems which would make suitable topics for Masters' dissertations in library or information science. These include linguistic analyses of search statements, the investigation of different matching rules in spelling corrections and a study of weighting schemes and cut-off rules in ranked output searching. Further details will be given on request.

9.2 Stemming

9.2.1 Weak stemming

Table 8.6 shows that *weak* stemming caused the *CTL* system to find more records than the *OSTEM* system in 74 (48%) of 155 initial searches. In four of the 74 the extra records were all or mostly false drops, and in six the results were mixed. An example of erroneous conflation at the weak stem level is the case of "skiing" and "sky", which are both transformed to "ski". We have not had time to attempt an analysis of the *types* of search which are improved by weak stemming.

The proportion of the above searches which fail *completely* on *OSTEM* is, however, small. In other words, weak stemming rarely turned a search from a complete failure into a success. We know there are such searches (ACCOUNTS DICTIONARY), but they do not seem to be very common.

There appear to be two reasons for this - the combinatorial search, and the fact that both subject headings and titles are indexed, with many of the PCL records having both British (PRECIS) and American (LCSH) subject descriptors. The extensive indexing means that many records contain, say, the singular form of a word in the title and its plural form in a heading. There are also many records with both British and American spellings.

Because the combinatorial search often results in the retrieval of records which do not contain all the words of the search, it is difficult to estimate the proportion

where a real user would not have found any relevant records. This particularly applies to two-term searches where, if neither word is very common, all records containing either of the words will be retrieved. A good example is ABORTION ACTS. Because of the weak stemming this finds three records under "Abortion Act" on *CTL*. These are, of course, displayed first (followed by the remaining records under "abortion". *OSTEM* finds the 158 records indexed by one of "abortion" and "acts". The 18 records under "abortion" come out first, because "abortion" is less common than "acts", so the user might still come across the three "Abortion Act" records. (In a previous experiment [2] we used the rule - for comparison purposes - that a search is to be counted a failure if no relevant record appears in the first ten which are displayed.)

It is clear, though, that with weak stemming a high proportion of searches find more relevant records than they do without stemming. Of the 64 cases where *CTL* found more than *OSTEM*, 33 (2 of which were all false drops) contained more records of *maximum possible weight* (i.e. records which would have been retrieved using if the words were combined using AND).

9.2.2 Spelling standardisation

In the searches which behaved better on *CTL* than on *OSTEM* there is only one word ("advertising") which is affected by spelling standardisation. A random sample of 68 words extracted from the much larger set of logged searches collected up to mid-February 1987 contained three examples: "behaviour", "advertising" and "color". This suggests that there is an appreciable proportion of such words in real searches. The words "(re)organis(z)ation(s)" occurred 48 times in 7700 searches. The bibliographic file contains 675 books under "organisation" and 648 under "organization"; in 50 of them both forms occur.

Despite the examples in 6.2.11 (*shoe* --> *she* etc) we found no real search where the effect of spelling standardisation might have been detrimental.

The mapping of "oe" to "e" should be conditional on the weak stem being at least five letters long (to avoid *poet* --> *pet*). "Poetry" (which occurred five times in 7700 searches) should be treated as exceptional if there is strong stemming which might conflate it with "petri(fied)". *Amme* --> *am* is occasionally contentious - "program" is a homograph in American English, and in only one of its meanings is it synonymous with the British or French "programme". This mapping might better be done with strong stemming than with weak. *Ism* --> *ist*, which is really stemming not spelling standardisation, should be incorporated with weak stemming, as almost all "ism/ist" pairs are very closely related in meaning, but there are a few

words (like "organism/ist") which must be treated as exceptional.

9.2.3 *Strong stemming*

Table 8.6 shows that the *EXP* system found more records than *CTL* in 53 (34%) of 155 initial searches. In nine of these cases the extra records were false drops, and in six the results were mixed. In 17 of the searches (13 good), *EXP* showed an "absolute" improvement over *OSTEM* - i.e. *CTL* obtained the same result as *OSTEM*, but *EXP* retrieved more records. These results are partly due to the effect of the *go/see* list (9.4). *Strong stemming* affected the result in 37 of the 53 searches, beneficially in 22 and with mixed or bad results in the other 15. Summarising, strong stemming led to better results than weak stemming alone in 14% of the 155 initial searches, and to mixed or worse results in 10%. Table 8.7 shows a few of these searches.

Clearly, strong stemming is not always safe. Strong stemming alone, applied to all searches, would be disastrous. On the other hand, it behaves well more often than it behaves badly. We would guess that it is rarely detrimental when applied to searches of three or more terms. When combined with weak stemming we would tentatively recommend its use in a combinatorial system like Okapi, provided strong stems are always given lower weight than corresponding weak stems. This would ensure that records retrieved on strong stems would usually be displayed after records retrieved on weak stems. Such records would certainly not be offered as "matching your search exactly", as can happen in the present system. (Okapi '86 is supposed to ensure that strong stems have lower weight than weak stems (see 6.5.3), but the procedure which assigns weights was never properly finished.)

The question of improved stemming procedures must be considered. The most ambitious of the schemes mentioned in Chapter 3 is the MARS project (3.3.8), but we have not seen detailed enough material to make an assessment of its suitability for this type of application.

Practically all of the schemes referred to in Chapter 3 would, for example, conflate "organisation" and "organism". Even "industry" and "industrialisation" should not be blindly conflated. One possibility would be to limit possibly contentious stemming to words which produce stems which occur only rarely in the bibliographic file. Although there would still be false drops the user could be warned that not all the records match very well, and there will not be many books indexed under the contentious stems. Such a system would also need a transparent way of showing the user why it found the records. Highlighting of the relevant stem is the obvious answer, but, as pointed out in 7.6.1, this is not particularly easy to achieve. To make

an improved system for strong stemming it would be necessary to use a considerable number of conditional rules and a dictionary of words (not stems) to which the rules apply (cf UNITED in our weak stemming procedure). The dictionary would be consulted before applying "blind" stemming. Two examples of such rules are is "If word is ORGANISATION or ORGANISER or ORGANISABILITY [etc] stem it to ORGANIS" and "If word is ORGANISM or ORGANIST or ORGANIC leave it unchanged".

9.2.4 Answers to the questions on stemming

1 Does it [stemming] significantly increase recall? If so, for what types of search? In particular, how often do stemmed searches succeed where they would fail without stemming?

It *does* significantly increase recall; we have made no investigation of the second question, and the answer to the third is "not very often".

2 Does stemming significantly decrease precision or lead to false drops?

Weak stemming does not lead to a marked decrease in precision, but strong stemming does.

3 How does the use of both strong and weak stemming (*EXP* system) compare with weak stemming only (*CTL* system)? For example one might find that there are, on average, fewer rephrasings of searches on *EXP* than on *CTL*.

There was no significant difference in the mean number (just under two) of searches per session between *EXP* and *CTL*.

4 Does the *EXP* system's two-level merge (6.5) make any difference (except to decrease search speed)?

The two-level merge avoided the need for the construction of search sequencing rules (of the form "repeat the search using strong stemming if it fails with weak stemming). It enables the user interaction to appear pleasantly simple. More generally, it is a technique which allows the use of implicit OR relationships in ranked-output searching. An informal description of the merge procedure can be read between the lines of Section 6.5. The actual merge algorithm will be published elsewhere. It is available on request.

5 Is there a case for using strong stemming only? If so, should this apply to all searches, or only to those containing more than a certain number (two, say) of terms?

Strong stemming only would be almost insupportable in a general catalogue. It may be acceptable for systems which

access small collections of specialised material, but we are not concerned with such systems here.

9.2.5 Recommendations on stemming

Weak stemming is undoubtedly beneficial. In fact, it is inexcusable for it not to be provided in a keyword catalogue. Even weak stemming procedures should be improved by using a rather small dictionary of exceptions (ours consists of the single word "united"). Alternatively, searches could be processed using two levels - no stemming and weak stemming, with the weak stems given lower weights than the "raw" words. The former makes lighter demands on computing resources, but someone has to invest a good deal of intellectual effort in constructing an exception table (which might need contextual information).

Although *spelling standardisation* only affects a small proportion of searches (in our subject areas) it costs almost nothing to incorporate it with weak stemming, and its effect should be almost entirely beneficial. With the possible exception of "amme" it should be used at any level of search.

It is doubtful whether really good results can be obtained with *strong stemming* unless it does use a fairly large set of word-specific rules. However, it is on balance better than nothing, until we have better indexing (9.7) and better linguistic processing.

9.3 Spelling correction

6 How effective is *EXP*'s semi-automatic correction procedure? How does it compare with users' response to *CTL*'s "CAN'T FIND" message? (Figs 7.5 and 7.6).

This was answered in 8.7.4, where Table 8.9 shows that users' treatment of words which are not known to the system is almost certainly better if spelling correction is applied than if it is not. On the *EXP* system 78% were handled "well", against 64% on the *CTL* system where the user has to type a replacement.

Nevertheless there is scope for improving the correction procedure, which appears to be able to correct only about half the misspellings.

There does not seem to be a serious rival, using current hardware, for a two-stage process comprising soundex or n-gram similarity matching followed by a string similarity check of the user's word against the list selected at the first stage. For systems like online catalogues where it is undesirable to present the user with a choice of replacements, soundex is probably preferable to an n-gram

technique (5.4). (We did not have time to experiment with n-grams, but the research done by Willett and others, and the SPEEDCOP team (5.4.2) probably renders further experiments unnecessary.)

9.3.1 Recommendations and discussion

Semi-automatic spelling correction should be used in online catalogues. The procedure described in 6.4 is not unsatisfactory. It should be improved by

1 Weakening the selection of candidate replacements so that the correct replacement is more likely to appear in the output from this stage. We suggest a procedure very similar to the original Soundex: truncate at four or five characters and ignore vowels (other than initial vowels). If this gives rise to some very long lists, then it can be tightened by retaining the first two letters unchanged (see Section 8.7.1 for some evidence that this would not markedly decrease recall). The treatment of misspellings (as opposed to miskeyings) would be somewhat improved if some attention were given, in coding the consonant structure, to the treatment of consonant groups such as "ng" (treat it as belonging to the same class as "n") and to "dg" (treat it like "g").

2 Ensuring that the dictionary contains as many as possible of the words which are actually used, by incorporating words from a very large number of real searches. This means that the dictionary would contain words which the system will *recognise* but which do not occur in the bibliographic indexes. The catalogue must be able to report 'No books under "brimstone"' and give the user options of starting a new search, ignoring the word or entering a replacement (cf Fig 7.5). It must not offer "brainstem" (Table 8.8). This would show the user that the system recognises the word, but has nothing indexed under it. (Okapi '86 can do this, but only for *go/see* terms which have no postings.)

The preceding paragraph leads naturally to the suggestion that *all* the user words should be looked up in the dictionary. Since more than 80% of users' words are likely to occur in the dictionary (Table 6.1 shows 15.8% of a large sample were misspellings or "rubbish"), it is not efficient processing to do this if the *dictionary* is separate from the *index*. But an ordinary inverted index designed for the retrieval of postings lists cannot be searched in such a way as to retrieve lists of candidate corrections for a misspelt word. Hence the dictionary should be partially duplicated in the index: the index would contain all recognisable words even if they do not occur in the bibliographic source data. This would not seriously increase the size of an inverted index, because most of the indexing storage is occupied by

postings lists rather than by terms.

Finally, the system should be augmented by the inclusion of a small set of common and unambiguous misspellings, which should be directly mapped to their corrected versions. This is better implemented by putting such words into the cross-reference table (9.4) than into the spelling subsystem.

3 Our procedure for measuring similarity (Appendix 2) should be improved. Much work has been done on this, under such headings as "string similarity measures", but we could not find any procedure which looked outstandingly good without being computationally complex. It looks as if increasing sophistication leads to diminishing returns. We chose the "anagram" method because it is easy to implement, but such cases as the *thacher --> teacher* example show that it is not good enough (8.7.1). We have not given any further consideration to this. It is one of the minor research topics suggested in 9.1.

9.4 Cross-reference tables - the *go/see* list

This contains some 230 sets of terms. Some of the sets consist of a single phrase which is to be treated as if it were a word. Others contain more than one item, and have the effect of causing a search for any one of the items to retrieve records indexed under any of them. There is an extended discussion of the types of term in the list in 6.3. The list itself, designed for our particular user population, is given in Appendix 5.

A summary of results combined with answers to the questions of 8.1 is given in the next section.

9.4.1 Answers to the questions on cross-reference tables

7 How often does it [the table] make any difference? Does our list contain appropriate entries? How should one compile such a list for a given environment?

About a quarter of 1087 searches contained a member of the list (8.8). The terms which were actually used are given in Table 8.10. An examination of searches suggests a number of additional entries, such as *contract law = law of contract*, because "contract" AND "law" gives about 100 postings, a considerable proportion of which are false drops due to false coordination. We drew up the list after a study of past searches by users of the same library. This appears to be a good way of doing it. It could be expanded greatly by the use of search data from other disciplines.

8 Does the list lead to false drops ('us' [pronoun] = 'United States')?

We have found no example of a false drop arising from the use of a *go/see* term. One of the largest groups of *go/see* entries is that consisting of abbreviations and acronyms linked to the spelt out forms in which they are more likely to be given by cataloguers. People like to make acronyms which are homographic and suggestive, such as Okapi (elusive, long gestation period). This is all right in ordinary written language, because of context and (decreasingly) the use of upper case. It is a serious problem for information retrieval systems. We can put "US" in the list because the pronoun "us" is rare in bibliographic (subject) data and very unlikely in search statements. But we cannot put "AIDS" in the list. "IT" might be considered for inclusion. The pronoun "it" occurs some 200 times in titles, but this can be stopped. In 7,700 searches of Okapi '86 there were seven occurrences of "it" or "IT", and three of "information technology". All the occurrences of "it/IT" in searches intend the pronoun (e.g. INDUSTRIAL REVOLUTION WAS IT A REVOLUTION). To cope with words like AIDS and IT, there has to be a new type of object in the list (see below).

9 Should there be more than one type of object in the list (e.g. *see alsos* as well as *sees*)?

Clearly a catalogue should have ways of offering *see also* references. This is rather outside the scope of the present project. The AIDS and IT examples show that there should be a third type of object - sets of homographs. These are like multi-valued *see* references: *aids - see aids (role 1)* or *aids (role 2)*.

9.4.2 *Recommendations on cross-reference tables*

Our application has proved successful across a fairly wide range of subject areas. We suspect that compiling entries for the hard sciences would be relatively easy as there is, on the whole, less ambiguity. Existing thesauri are a rich source of material. On the other hand, the problem is certainly greater in the humanities. Any list requires constant maintenance to reflect language changes. Since lists of our type are far smaller than, say, subject authority files, such maintenance would not lead to the problem of scale which is one the reasons why indexing and classification languages tend to lack currency.

We recommend that an extended cross-reference list should be compiled, and that this be maintained by merging entries from contributing libraries.

The use of tables adds to complexity and computational demands both when indexing and when searching. It is trivial to extract either individual words or entire "headings" from source text or from users' input. But procedures for automatically checking all embedded sub-phrases as candidate index or search terms are much more complicated. Attention needs to be given to the design of efficient algorithms. The number of lookups is proportional to the number of words in the text being processed. It is certainly necessary to hold the table in quick access memory. (We avoided this problem at search time - the list is in effect embodied in the index (6.3). Okapi '86 has only restricted knowledge of the list when it is performing searches. It would not, for example, be able to explain to the user that when it is looking up "UK" it is also looking up "United Kingdom", "Great Britain", etc. It may be thought desirable that the system should be capable of explaining itself.)

HOMOGRAPHS

MORPHS (4.2.6) handles some homographs, albeit in a fairly narrow subject area.

The searcher for AIDS could be asked

> Please explain "aids"
>
> Do you mean 1 : "aids" = devices for helping
> or 2 : Acquired Immune Deficiency Syndrome
>
> Type a 1 or a 2 .

The problem here (apart from that of compiling the list) is one of identifying the meaning of the word in the bibliographic record. In most cases, e.g. "China", this has to be done manually. Programs would have to be written to enable indexers to run a MARC file against a homograph list. It would pick out candidate words and show them in their context, and prompt the indexer to select the appropriate role.

9.5 Users' perception of and behaviour with the system

10 What sort of conceptual models do users have of the catalogue? How do they think it works? Is it *comfortable* to use? Is it *exciting* or *boring* or *silly*?

A study of user behaviour is outside the scope of the present project. However, as pointed out in 9.1, it is essential to do research of this type on a catalogue which does not behave in such a way as seriously to confuse,

surprise or irritate users. From the comments given in 8.4.3 - particularly from the fact that most users did not offer *any* comments - and the (remarkably few) entries in the suggestion books, it is fairly obvious that Okapi '86 behaves in such a way that it is taken for granted by most users. Most users regard it as being neutral. A substantial proportion seemed to think it rather or very good compared with other manual or computer catalogues which they had used.

Some users certainly notice that it "does words separately", and there was even one favourable comment about this. So long as a search succeeds, word searching is doubtless acceptable - catalogue users do not mind how the system works if it seems to find the right books. When searches are unsuccessful, the system is criticised as being "unintelligent" ("it only looks for keywords - doesn't analyse the search"). There were at least three interview comments, and several more in the suggestion books, to this effect.

Most users do not expect a catalogue to be exciting, or even interesting. Catalogues are taken for granted and regarded purely as tools which are to be used without the necessity of applying much in the way of forethought or initiative. We think that most users see Okapi '86 as a tool which is at least as effective as other catalogues.

However, we believe that online catalogues may come to be regarded as multi-function power tools rather than as spades.

11 Does it give a dangerous impression of cleverness or of infallibility?

A significant minority of searches were of the type (which we classified as *Q* - see 8.3.6) exemplified by the search BY WHAT MEANS ARE WE EDUCATED FOR SEXUAL INEQUALITY IN WORK. It is unlikely that users would try to look for such phrases as headings in a conventional catalogue. Okapi invites the user to "Type a word or phrase which describes the books you want". Many *Q*-type searches *do* satisfy this description. They *are* descriptive of the books the user wants. But they do not describe the books in a way which concords with the way books are described in bibliographic records. It is very difficult to think of any concise prompt which would inhibit people from entering this type of search.

There are two ways of tackling this problem. By far the simplest is to use a stop list which includes a wide range of function words and pronouns. Some *Q* searches then work quite well. But many will still fail because they tend to be far too specific (neither SEXUAL INEQUALITY nor INEQUALITY AT WORK finds more than a handful of books in

the catalogue, and SEXUAL INEQUALITY AT WORK, surprisingly, finds nothing but false drops). A better approach may be to use some simple linguistic analysis to try to identify "inappropriate" searches, and suggest to the user that he or she might try something rather less specific.

9.6 Applicability of our findings

All the Okapi research has been aimed at investigating techniques which are possible now, using existing resources. That is, they could appear in commercial applications within five years or so. Some have already appeared - not necessarily as a result of the Okapi research (notably the use in keyword searching of combinatorial merging instead of implicit AND; this was first implemented - in a catalogue - in CITE).

The outcome of the present project is no exception. All the devices could be implemented now in a commercial system without demanding more in the way of hardware resources than what is normally available for integrated library systems.

However, although catalogue access is the most computationally demanding facet of an integrated system, from the design and programming point of view it is only a small portion of the whole. The design of catalogue access facilities has to be done so that it is compatible with the demands of cataloguing and acquisitions (which need rapid updating of files and indexes) and of circulation control. This makes catalogue access very much more difficult for the commercial designer than it is for us, who do not have to link to circulation status, and who update files only occasionally, and offline.

Another very important point is that to avoid extended development times commercial designers nearly always have to work within the constraints of languages, operating systems and database management systems which were designed long before the days of interactive computing for casual users. We do not use any existing system software. Okapi depends only on the pre-existence of four primitive input and output functions.

Much system software offers very tempting easy-to-program facilities (sorting and merging, the automatic extraction of words from text, automatic maintenance of indexed-sequential files). Of the system software which we have come across none is quite good enough to provide more than a just-acceptable compromise. Some system software will do the job, but will not do it efficiently enough. An example is index lookup. In Okapi we can be fairly prodigal with this. A search of four words may involve eight or ten lookup operations, including weak and strong stems and perhaps an attempt to match a possibly misspelt word. An

Okapi lookup rarely takes more than two disk accesses, because we can optimise the file structure to suit lookup rather than updating. A typical lookup in a commercial database system takes three to five or more accesses.

Of the devices treated in this report, it should be fairly easy to graft a single level of *stemming* onto most keyword systems. A few systems already allow the use of limited *automatic cross-referencing*. We have shown that this facility is worth using. *Spelling correction* systems are a little more ambitious, but they are not very demanding on storage, and they have the advantage that the dictionary, once constructed, does not need much maintenance.

We hope libraries will demand systems in which the search ACCOUNTS DICTIONARY does not fail when the library holds books with titles like "A dictionary of accounting" and subject headings like "Accountancy - dictionaries". If they do not make these demands on suppliers they are not meeting their responsibilities to users.

9.7 Concluding remarks

Although Okapi '86 is a relatively good subject search system given the content of bibliographic records, it is, by absolute standards, rather poor. Fourteen of 122 sessions reported in Table 8.1 and Section 8.4.2 failed although the library held probably-relevant material. Seven failed because, although the searches were quite comprehensible, the language did not match that of the catalogue well enough to be picked up by any of our devices. Four failed because they were too specific. Only two (STERLING when the user wanted "sterling shares and gold" and BRITAIN AS A DEVELOPING COUNTRY for "Economic development of Britain in the 18th century) failed because the search did not describe the user's needs.

Almost all these searches would have succeeded, and many more searches which did not completely fail would have given better recall and probably better precision if our records had proper analytical indexing using contents pages and added free language descriptors. The efficacy of such enhanced indexing was demonstrated long ago [3]. The time is long overdue for a large scale test of analytical indexing in a ranked-output system.

Much research effort has been put into the investigation of ways of making inadequately described records accessible. Is it not better to attack the problem by improving the quality and richness of the access points to bibliographic files?

References

1 MITEV N N, VENNER G M and WALKER S. *Designing an online public access catalogue : Okapi, a catalogue on a local area network.* (Library and Information Research Report 39). London : British Library, 1985.

2 JONES R M. Improving Okapi : transaction log analysis of failed searches in an online catalogue. *VINE 62*, May 1986, 3-13.

3 SYRACUSE UNIVERSITY. INFORMATION STUDIES DEPARTMENT. SUBJECT ACCESS PROJECT. *Books are for Use : final report of the Subject Access Project to the Council on Library Resources.* Pauline Atherton, Director. Syracuse University, 1978.