4 Tables and dictionaries

4.1 Introduction

Automatic cross-reference tables can be used in both spelling correction and recall improvement. They are particularly useful for automatically linking synonyms, abbreviations, alternative spellings and other related terms to their equivalents.

4.2 Methods and techniques

4.2.1 Dictionaries in spelling correction

Pollock and Zamora [1] used a dictionary for spelling error detection in the SPEEDCOP project, although this technique is only one of several. SPEEDCOP uses a dictionary of common misspellings as a supplement to a similarity key/ reverse error algorithm. The common misspelling approach is based on the assumption that spelling errors which have occurred in the past will occur again. The incorrect form can be mapped to the correct spelling (for example, "teh" might be mapped to "the"). There are several limitations inherent in this approach. Firstly, most spelling errors do not recur at all frequently. Secondly, the list is always incomplete as new misspellings will constantly occur. Thirdly, up to 15% of misspellings are ambiguous (for example "hoise" could be a miskeying of "noise", "house" or "hoist").

Experiments conducted by the SPEEDCOP team indicate that the effectiveness of a dictionary of common misspellings depends on the size of the sample from which it is created; a dictionary generated from a sample of about ten million text words will probably correct about 10% of misspellings. Although its application is limited it is almost entirely accurate provided that it only contains unambiguous misspellings.

The SPEEDCOP project also used a general dictionary of around 40,000 correct terms. This is generally adequate even for a technical database as most of the text of a technical database does not consist of specialised vocabulary (Pollock and Zamora point out that 30% of Chemical Abstracts consists of function words but only 2% of chemical substance words).

Dictionary look-up does have limitations; a comprehensive dictionary can recognise a large *proportion* of the text but

the *number* of words unmatched will still be large when the volume of words processed is in millions. The situation is not improved simply by increasing the size of the dictionary. The proportion of words recognised will certainly increase but some misspellings are likely to be identified as "correct". For example, "ion" will probably be a high frequency word in a chemical database, but is more likely to be a miskeying of "icon" in a decorative arts database. "Ion" should not be included in a decorative arts dictionary as it is more likely to be wrong than right. A small special dictionary of specialised vocabulary can be used to supplement the standard dictionary of common words. In most applications this specialised vocabulary will be disproportionately represented in the flagged words. Pollock and Zamora have written that when the Chemical Abstracts dictionary was applied to Chemical Industry Notes (a more specialised database dealing with the chemical industry) more than 98% of the words flagged as possible misspellings were valid. Most of these words were eliminated by a small specialised dictionary.

The SPEEDCOP procedures include a suffix normalisation technique similar to that used by Galli and Yamada [2], which serves to increase the capacity of the dictionary for matching terms without increasing its size. If a word is not in the dictionary then the suffix algorithm would stem the word to its root and look for this in the dictionary. Although stemming can reduce the number of terms in the dictionary (by 15% in this instance) this saving in storage is offset by the computer time taken to identify the variants.

The SPEEDCOP algorithm attempts to bypass specialised classes of words, such as acronyms, trivial and trade terms for substances, systematic chemical nomenclature or proper names. Words which fall into these categories can often be treated by incorporating a document-level frequency threshold. More specifically, acronyms and systematic nomenclature can be recognised algorithmically. Acronyms can be detected with reasonable reliability if they are in upper-case letters.

4.2.2 Proof-reading methods

Galli and Yamada [2] describe an automatic dictionary which has been used for checking machine readable text in proof-reading. The dictionary verifies every text word and produces an output document in which all the words are hyphenated if necessary, corrected if misspelt and standardised in the case of spelling variations. British spellings are transformed into American spellings and non-preferred spellings are transformed into preferred forms.

The dictionary contains about 56,000 entries including word stems, word endings, whole words, prefix-combining forms,

suffix-combining forms, spelling standardisation entries, spelling-error correction entries and control entries. The size of the dictionary was reduced by including the prefix and stem information. Unlike the uses of stemming discussed in Chapter 3 stemming is here used as a compaction technique and not as a recall improvement device. The dictionary is used to identify compound words; some endings are listed with a code which allows tentative compounding and then invokes corrective measures at a later stage.

The dictionary contains a list of almost 2000 words which can be spelt in different ways. In this way, a non-preferred spelling is transformed into the preferred form ("moveable" is altered to "movable"). The system also transforms British English into American English. Correct spellings of archaic words which could be misspellings of more common words are not altered but are flagged (Galli and Yamada give the example of "calender", the archaic spelling of "calendar"). Equally acceptable variants are both listed ("sirup" and "syrup"). The system lists about 2000 common misspellings which are mapped to the correct form (and flagged as corrected for possible manual checking later). A test of the system verified 89% of the documents.

4.2.3 *Spelling correction using a dictionary together with a word representation technique*

The designers of LEXICON [3, 4] suggest that a good error correction method would be a two-step process consisting of a moderately low threshold modified soundex system followed by a high threshold similarity check. Tests revealed that this combination should automatically correct 60-70% of errors.

The medical free text system at Massachusetts General Hospital [5] uses a dictionary as a pre-stage to soundex spelling correction. The dictionary includes some variant spellings, terms which are (medically) non-preferred (such as "womb") and expletives. The latter are ignored by the computer - they are in effect on a stop list - in order to discourage their use.

4.2.4 *Using tables to match related words*

Possibly the greatest utility of tables lies in their matching potential for semantically related words, as stemming and word representation devices are only useful for words which are morphologically similar. Some words which are similar in meaning are also orthographically similar, but many are not. Even when they are orthographically related, a dictionary link will ensure a unique match whereas a link through a stemming algorithm might include other irrelevant words.

Medical information retrieval systems seem particularly well-suited to the application of these techniques. Wong and others [4] have described a natural language dictionary (LEXICON) of anatomic pathology. Text is scanned and separated into keyword types: authoritative types, non-preferred keywords and optional synonyms. LEXICON is not organised as a hierarchical thesaurus with complicated cross-linking. More than half of the words (52.2%) are cross-referenced to a supplementary word. These supplementary words fall into several categories:

a preferred synonym or alternative spelling ("edema" for "oedema");

an optional synonym ("jaundice" for "icterus");

a disinflection ("kidney" for "kidneys");

a related word in a hierarchical scale ("enteritis" for "enterocolitis");

a medical term for a lay term ("pregnancy" for "gestation").

Doszkocs [6] describes AID, an "Associative Interactive Dictionary", for automatically generating and displaying related terms, synonyms, and broader/narrower terms. CITE (2.5) uses a table look-up procedure. This table, for example, maps "treat", "treatment", "treating", "therapy", "therapies", "therapeutic", "care" and "regimen" to the subheading "therapy" [7]. Other suggestions include the automatic mapping of author entries to the National Library of Medicine name authority file and the use of an automatically generated thesaurus.

4.2.5 *Linking natural language terms with controlled language terms*

Tables have been used in medical information systems as a means of linking natural language search terms to controlled language MeSH headings. Doszkocs has written that this dependency on MeSH is particularly pronounced in Medline since less than 50% of the records contain abstracts [8]. If a retrieval system is to work effectively, then it is essential to develop links from the search terms to the appropriate subject headings. This can be done in several ways; one of the simplest ways is to use a dictionary for automatically mapping search terms to potentially useful subject headings. In some disciplines this dictionary would need to be constructed manually. In the "hard" sciences there are often existing thesauri. Doszkocs considered identifying chemical synonyms by matching query terms against the National Library of Medicine's chemical dictionary file, CHEMLINE. Although Doszkocs is as yet uncertain of the best method of attempting synonym control,

the current CITE does achieve a considerable degree of semantic expansion by using tables to map search terms to the MeSH file.

4.2.6 Compound words and homographs

The MORPHS system [9, 10, 11] developed by Bell and Jones at the Malaysian Rubber Producers' Research Association uses lists in its treatment of compound words and homographs.

The presence of compound words in textual information can substantially influence recall. A compound word or a phrase which has the same meaning can exist in several different forms: for example, "houseboat", "house boat", "house-boat", or even "a house which is a boat". These all have the same meaning but if the system makes no attempt to deal with this problem, the different forms will not be brought together in the index. The only totally automated method of separating compound words would necessitate checking all multi-syllable words against the system vocabulary in order to ensure that they do not contain embedded fracturable elements. A list would still be needed to detect compound words which are neither concatenated nor hyphenated. Bell and Jones discuss the use of what they call *links* as a solution (albeit partial) to the compound word problem. Using links, the compound "fountain pen" could be entered into an inverted index with a P-link added to the term "fountain" and an F-link added to the term "pen". A system of links has been incorporated into a recent version of MORPHS.

MORPHS also holds a list of homographs, such as "china". A user whose search contains "china" might be asked to choose between "ceramics" and "People's Republic".

4.2.7 Stop lists

Practically all information retrieval systems use stop lists of words which do not enter the index or which are automatically removed from searches submitted to the system. Stop lists range in size from a handful of words to many hundreds. Different lists may be used for different data fields or search types.

4.3 The use of tables in online catalogues

Several commercially available systems (see 2.4.2) allow user libraries to set up lists of groups of terms which will be treated as equivalent during indexing and searching. MELVYL uses an abbreviation table [12]. CITE's mapping of users' terms to MeSH headings has been mentioned above (4.2.5). Some of the phrase search systems can use authority files to map searches semi-automatically to preferred forms.

All the more recent systems allow the use of stop lists, although there may still be one or two "phrase searching" catalogues where initial definite and indefinite articles are not automatically removed from user input.

An intermediate version of Okapi (1.5) contained a single table which was used both during indexing and during searching. Entries in this table were of three types:

stop words and phrases

lists of terms to be treated as equivalent (*child, children*)

compound words or "go phrases" (*industrial revolution*)

It will be seen in Chapter 6 that the Okapi '86 *EXP* system uses more or less the same scheme. The most obvious difference in Okapi '86 is that everything (apart from some stop word processing) is done within the index to avoid the storage overhead of a large table when the search programs are running.

References

1 POLLOCK J J and ZAMORA A. System design for detection and correction of spelling errors in scientific and scholarly text. *Journal of the American Society for Information Science* 35 (2), 1984, 104-109.

2 GALLI E J and YAMADA H. An automatic dictionary and the verification of machine-readable text. *IBM Systems Journal* 6 (3), 1967, 192-207.

3 JOSEPH D M and WONG R L. Correction of misspellings and typographical errors in a free-text medical English information storage and retrieval system. *Methods of Information in Medicine 18* (4), 1979, 228-234.

4 WONG R L *and others*. Profile of a dictionary compiled from scanning over one million words of surgical pathology narrative text. *Computers and Biomedical Research 13*, 1980, 382-398.

5 FENICHEL R R and BARNETT G O. An application-independent subsystem for free-text analysis. *Computers and Biomedical Research 9*, 1976, 159-167.

6 DOSZKOCS T E. AID : an Associative Interactive Dictionary for online searching. *Online Review 2* (2), 1978, 163-173.

7 DOSZKOCS T E. CITE NLM : natural-language searching in an online catalog. *Information Technology and Libraries* 2 (4), December 1983, 364-380.

8 DOSZKOCS T E. From research to application : the CITE natural language information retrieval system. In: *Research and Development in Information Retrieval. Proceedings Berlin 1982.* Edited by Gerard Salton and Hans-Jochen Schneider. Berlin : Springer-Verlag, 1983, 251-262.

9 BELL C L M and JONES K P. A minicomputer retrieval system with automatic root finding and roling facilities. *Program 10* (1), Jan 1976, 14-27.

10 BELL C L M and JONES K P. Back-of-the-book indexing : a case for the application of artificial intelligence. In: *Informatics 5. The analysis of meaning. Proceedings of an Aslib/BCS Conference. Oxford, 1979.* London : Aslib, 1979. 155-61.

11 BELL C L M and JONES K P. The development of a highly interactive searching technique for MORPHS (Minicomputer Operated Retrieval (Partially Heuristic) System. *Information Processing and Management* 16, 1980, 37-47.

12 UNIVERSITY OF CALIFORNIA. OFFICE OF LIBRARY AUTOMATION. *MELVYL reference manual : University of California Online Catalog.* Berkeley : University of California, 1985.