CHAPTER 1

Establishing the NPL test collection at Cambridge

The original task of data preparing the Vaswani and Cameron test collection (hereafter simply NPL) is described on pp 9-13 of their report [1]. It was done in two stages. The first 1600 abstracts were keypunched on primitive hand punches, with an error rate of one incorrect word in 25, by Vaswani and Cameron's estimate. The remaining 10,000 or so abstracts were keypunched on machines with proper typewriter keyboards, and the error rate fell to about one incorrect word in 250. Since there are about a quarter of a million words in the collection as a whole, this suggests a total of

$$\frac{1600}{11600} \times \frac{250000}{25} + \frac{10000}{11600} \times \frac{250000}{250}$$

or about 2500 errors. These errors of original typing were not removed during the period of Vaswani and Cameron's work on the data. After receiving the NPL data, we adopted the following strategy for clearing some of the errors out.

First, a word index was prepared, giving all the text words in alphabetical order, their frequency of occurrence, and the first four line numbers at which each word occurred. So for example the entry:

```
    VALVESARE  1     155
```

would mean that the word VALVESARE occurred once, on line 155. Next a correction list was prepared, in which each word thought to have been mistyped was followed by a solidus and then its correct form, e.g.

```
    VALVESARE/VALVES ARE
```

Then this correction list was converted into a set of edits for the phoenix text editor, each of the form

```
    df/VALVESARE/b/ /:VALVES ARE¦/n
```

which can be read "delete from the current line to the line beginning VALVESARE, insert the string :VALVES ARE¦ before the first space in the line, and go on to the next line." (This conversion can itself be done by the text editor.)

The edits were then run on the word index, to produce a list of records for the faulty words only, having the general form

```
VALVESARE:VALVES ARE¦  1    155
```

and a one-off program converted this list to a sequence of edits, in the form

```
m155
e/VALVESARE/VALVES ARE/
```

which means "move to line 155 and exchange VALVESARE with VALVES ARE". The sequence of edits was then sorted into line number order, and was then run on the original file of abstracts. The result of each edit was checked and many of the edits were modified accordingly. The final number of edits obtained was about 1900, which compares favourably with Vaswani and Cameron's estimate of 2500 errors. This method of editing is of course very quick, and one would notice rapidly diminishing returns if one tried to correct the data further.

The initial tidying up of the data revealed an oddity apparently overlooked by Vaswani and Cameron, which was that many document abstracts appeared to be duplicated. Possible duplicates could be found very quickly by producing an index of first lines of document abstracts. To check that they were indeed duplicates, the entire document collection was printed out for inspection. In this way 142 duplicates were found. These were deleted from the document collection, and the document numbers in both the main document file and the file of relevance assessments were reduced accordingly.

Finally, Vaswani and Cameron's document 0 was put at the end of the document collection as document 11429 so that the counting would begin from 1.

The process of deriving the test collection from this corrected text file is described in the account of the programs for doing this work, in Chapter 4. Briefly, the common text words (stopwords) are removed, and each word is replaced by a term number which is unique to the stem which remains after a suffix stripping process has been applied to the word. Thus CONNECTION, CONNECTIONS, CONNECTING ... will share the same term number. The suffix stripping algorithm is described in Appendix - . The numbers are chosen so that term 1 is the most common term, term 2 the next most common, and so on. Typically, each document will be represented by a vector of integers

$$d\ t_1\ f_1\ t_2\ f_2\ \dots\ t_k\ f_k$$

which means that document d contains f_1 occurrences of t_1, f_2 occurrences of t_2 and so on. The terms are usually arranged so that $t_1 < t_2 < \dots t_k$. Similarly, there is an inverted document file, or term file, containing records of the form

$$t\ d_1\ f_1\ d_2\ f_2\ \dots\ d_k\ f_k$$

which means that t occurs f_1 times in d_1, f_2 times in d_2, and so on, and $d_1 < d_2 < \ldots d_k$.

REFERENCES

1. VASWANI, P.K.T. and CAMERON, J.B. [1970] The National Physical Laboratory experiments in statistical word associations and their use in document indexing and retrieval. National Physical Laboratory, Teddington. 1970.