Reuters test collection

Saturday, 11 June, 1994

Mark Sanderson

Abstract

This short paper presents the little known Reuters 22,173 test collection, which is significantly larger than most traditional test collections. In addition, Reuters has none of the recall calculation problems normally associated with some of the larger test collections now available. This paper explains the method (derived from Lewis *[Lewis 91]*) used to perform retrieval experiments on the Reuters collection. Then, to illustrate the use of Reuters, some simple retrieval experiments are also presented that compare the performance of stemming algorithms.

1 Introduction

To establish the retrieval performance of an IR system, it is necessary to use a test collection. Such a collection consists of: a set of documents; a set of standard queries; and for each query, a list of the documents relevant to that query. These relevant document lists are manually identified, a process which involves significant human effort.

Many test collections have been created, a large selection of these are listed in van Rijsbergen and Sparck-Jones *[Sparck-Jones 76]*. A few of these collections have become freely available and used by many IR researchers. However all of these collections are significantly smaller than the collection that a modern IR system may retrieve from. This difference is illustrated in the table of Figure 1 which lists a selection of traditional test collections and a collection that is available on CD-ROM (IND '91).

Collection name	No. of docs.	Bytes per doc.	Size (Mb)
ADI	82	466	0.04
MEDLINE	1,033	1,079	1.10
TIME	423	3,663	1.50
CRAN	1,400	1,203	1.60
CACM	3,204	717	2.20
CISI	1,460	1,526	2.20
NPL	11,429	283	3.10
LISA	5,872	610	3.40
IND '91	75,900	2,853	206.50

The work of Blair and Maron *[Blair 85]* has indicated that retrieval performance varies with collection size. Their findings suggest that results from experiments based on small test collections may not hold when applied to larger collections. Because of this, the traditional test collections are increasingly falling out of favour with IR researchers.

In recent years a set of much larger test collections have been created which are approximately 4Gb in size. These collections are collectively known as the TREC collection. The popularity of TREC has been demonstrated by a number of recent conferences *[Harman 93]* held soley for the purpose of presenting IR research using TREC. However, because the collection is so large, it has not been possible to generate a complete list of relevant documents for each standard query. This means that the recall figure calculated from any TREC retrieval experiment is unreliable, opinions vary as to the seriousness of this.

Given the short comings of the collections outlined so far, it is clear that testing the performance of IR systems is problematic. Recently, Lewis *[Lewis 91]* presented work using a test collection that goes some way to solving the problems outlined above, but at the same time introduces some new problems. The collection Lewis used was the Reuters text categorisation collection, originally created to test the Construe system *[Hayes 90]*, it was modified by Lewis to test document representation methods. The collection consists of 22,173 documents taken from the Reuters newswire. The two advantages of the Reuters collection are: that it is around 10 times larger than traditional test collections (~20Mb); and that recall values can be calculated from retrieval experiments performed on the collection. The main disadvantage of Reuters is that the retrieval performed on the collection is of a somewhat different type from retrievals run on normal test collections. This difference will become clear in the following chapter.

The rest of this paper will: explain the experimental method used for Reuters; present some simple retrieval experiments run on the collection; and then outline conclusions on the use of the Reuters collection.

2 The Reuters test collection

The main difference between Reuters and an IR test collection is that Reuters doesn’t have a set of standard queries with corresponding relevant documents. However each document in Reuters is tagged with a number of manually assigned subject codes. It is these codes that allow us to use Reuters as a test collection for comparing document representation methods. This use of Reuters was first described by Lewis and it is his method, with some modifications, that is described here.

First, **R** is defined as the set of all documents in the Reuters collection. This set is then partitioned into two subsets of equal size: **Q** (the query set) and **T** (the test set). The method used to partition **R** was chosen to be a random assignment of documents into one of the two subsets. This method ensured that groups of documents covering common themes would be evenly distributed to both **Q** and **T**[1].

Next, **S** is defined as the set of all subject codes that have been assigned to at least one document in **Q** and at least one document in **T**. If we pick one of the subject codes from **S**, we can now perform a retrieval. (The retrieval system used in these experiments was developed specifically for this work. It is based upon the probabilistic weighted term model as described in *[Robertson 76]*.)

For example suppose we perform a retrieval for the subject code ‘crude’. First, all documents in **Q** tagged with ‘crude’ are selected. Then by performing relevance feedback using the selected documents, word/weight pairs are generated to form a query. This query is used to retrieve from the **T** set. The resulting ranked document list is examined to see where in the ranking, documents tagged with ‘crude’ appear. The position of the tagged documents is used to produce precision/recall figures. A conservative interpolation technique (outlined in *[van Rijsbergen 79]*) is used to transform these figures into precision values at ten standard recall levels (0.1, 0.2, …, 1.0).

This process is repeated for each subject code in **S**, each time producing another set of precision

[1] This differs from Lewis who partitioned the collection based on the document’s creation date. Such a partitioning was necessary for testing a newswire categorisation system, however this was not a factor of the experiments presented here.

values. These precision values are then averaged to give an overall set of values for each of the ten standard recall levels.

So by partitioning Reuters and using the subject codes, all the components of a classic IR test collection are created.

- the collection to be searched **- T**
- a set of queries - generated from **Q**, for each element of **S**
- a set of relevant documents for each query - documents in **T** tagged with the respective element of **S**.

The use of relevance feedback to generate the queries in place of verbose user generated queries means that the form of retrieval can be likened to an iteration of relevance feedback during a retrieval session.

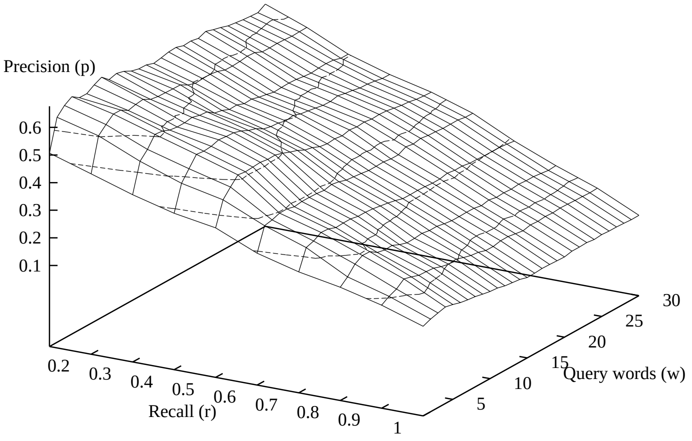

figure 2

2.1 Data reduction

When performing a retrieval experiment using Reuters the question arises, how many query words should be generated by the relevance feedback process? It is clear from the work of Hughes *[Hughes 68]* and Harman *[Harman 92]* that in a given situation there is an optimum number of words to use. As it was thought that such optimum numbers may be dependent on the amount of introduced ambiguity, the number of query words added was made a variable of the retrieval experiments. Therefore, the experimental results are expressed in three variables: precision (*p*), recall (*r*) and the number of query words added (*w*). The results can be plotted on a three-dimensional graph as shown in Figure 2. From the graph we can see that for all recall levels, the precision is low at *w*=1, with a rapid rise peaking at around *w*=5, before falling away as *w* increases.

Recall (*r*)	Precision (*p*)	F-measure (*f*)
0.1	0.592995	0.171140
0.2	0.544545	0.292552

0.3	0.472835	0.367091
0.4	0.432949	0.415823
0.5	0.398068	0.443249
0.6	0.326031	0.422488
0.7	0.278630	0.398600
0.8	0.224293	0.350358
0.9	0.165700	0.279872
1.0	0.107376	0.193929

figure 3

Unfortunately it was found that three-dimensional graphs become difficult to read when the results of several retrieval experiments were plotted together. What was needed was a two dimensional plot of *w* against a variable expressing retrieval performance, in other words reduce the *p*/*r* figures to a single number. The method used to calculate this number is illustrated with the following example. To calculate the retrieval performance of the *p*/*r* figures tabulated in Figure 3, for each of the ten pairs of *p*/*r* numbers a corresponding *f* measure is calculated. The formula for *f* is,

$$f = \frac{1}{1/2(1/p) + 1/2(1/r)}$$

(the measure is discussed in detail in *[van Rijsbergen 79]*[2]). Of the ten *f* measures calculated (Figure 3), the maximum (f_{max}) is selected as the retrieval performance figure. Applying this data reduction method, the graph in Figure 2 can now be plotted in two dimensions (Figure 4).

2.2 Random case test

Before any retrieval experiments were performed, it was necessary to establish how well the Reuters subject codes indicated document content. In other words, are the documents in set **Q**, marked with the subject code ‘crude’, good sources of evidence for retrieving similarly marked documents in set **T**?

The experimental method used to test this was identical to the method outlined above except for an additional step: when documents tagged with a certain subject code were selected from the set **Q**, a random set of documents were selected (from **Q**) and used to form the query instead. Figure 4 shows the result of this ‘random case’ experiment along with the result of an experiment using the subject codes as normal. As can be seen the random case is significantly worse than the method using the subject codes. In addition to establishing the utility of the subject codes as document content indicators, this experiment provides a ‘baseline’ which gives a scale to compare the differences between subsequent experimental results.

[2] In fact van Rijsbergen defines a measure called E, however F is simply defined as 1-E.

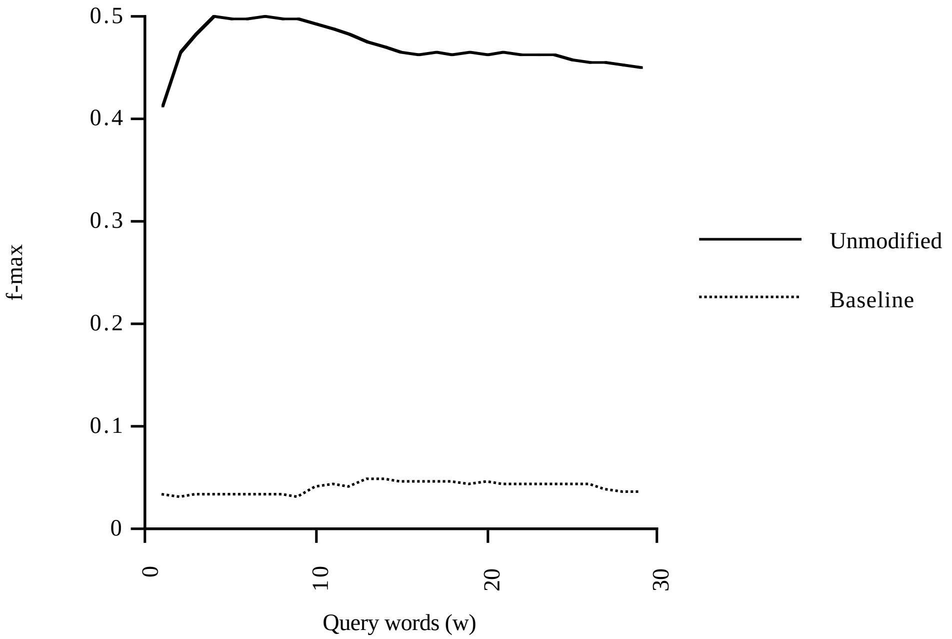

figure 4

2.3 Simple experiment

To illustrate the use of the collection, a set of experiments were performed which compared the effect on retrieval performance of two different stemming algorithms. (A more comprehensive set of experiments are outlined in *[Sanderson 94]*.) The stemming algorithms chosen were a modified version of the Porter stemmer *[Porter 80]* and the stemmer provided with the WordNet thesaurus *[Miller 90]*. In total, three experiments were run, one experiment for each stemmer and an experiment where no stemming was used.

Figure 5 shows the results of the three experiments. As can be seen the retrieval performances of all three configurations are similar. The WordNet stemmer is perhaps producing the best retrieval performance particularly for shorter queries. These results are similar to the results listed in the survey of stemming research by Frakes *[Frakes 92]*.

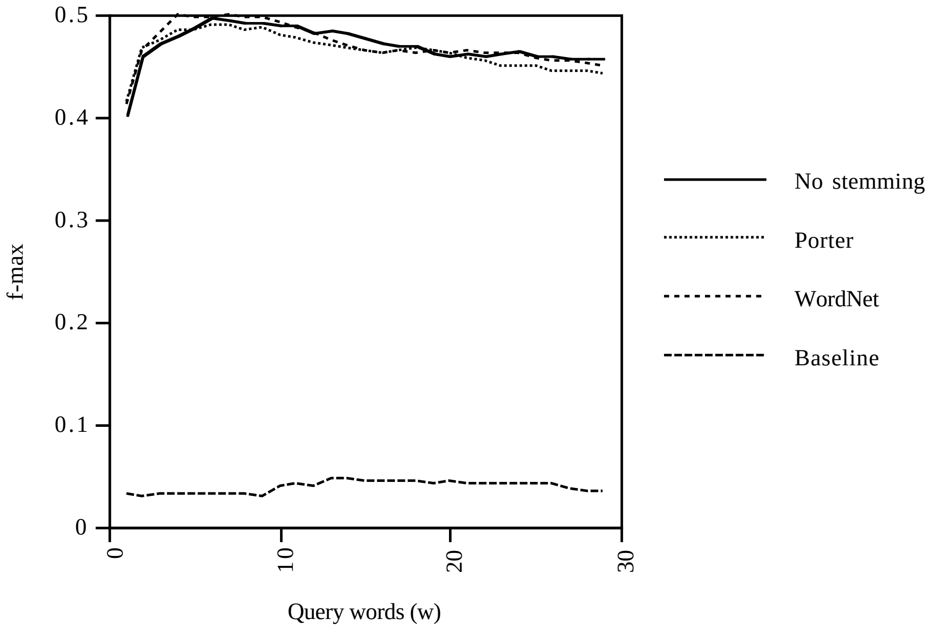

figure 5

3 Conclusions and future work

This paper presented a method for using the Reuters 22,173 collection as an IR test collection. Work on this collection is still at an early stage. It is intended that further retrieval experiments will be performed in an attempt to gain a greater understanding of how the collection performs with different retrieval configurations.

In using this collection it has been shown that IR systems can be tested using corpora other than standard test collections. Given the great amount of human effort required to create a test collection, this is good news. However, Reuters isn't unique and it is believed that there exist other text collections similar to Reuters. It is intended that these collections will be investigated to establish their usefulness for testing IR systems.

4 References

[Blair 85]
D.C. Blair & M.E. Maron (1985)
"An evaluation of retrieval effectiveness for a full text document retrieval system"
Communications of the ACM, Vol. 28, Num. 3, Pages 289-299

[Frakes 92]
W.B. Frakes & R. Baeza-Yates (1992)
"Information Retrieval - data structures and algorithms"
Prentice-Hall, Pages 131-160

[Harman 92]
D. Harman (1992)
"Relevance feedback revisited"
Proceedings of ACM SIGIR Conference, Pages 1-10

[Harman 93]
D. Harman (1993)
"Overview of the first TREC conference"
Proceedings of the ACM SIGIR Conference, Vol. 16, Pages 36-47

[Hayes 90]
P. J. Hayes (1990)
"Intelligent high volume text processing using shallow, domain specific techniques"
Working Notes, AAAI Spring Symposium on Text-Based Intelligent Systems, Pages 134-138

[Hughes 68]
G.F. Hughes (1968)
"On the mean accuracy of statistical pattern recognisers"
IEEE Transactions on Information Theory, Vol. 14, Num. 1, Pages 55-63

[Lewis 91]
D.D. Lewis (1991)
"Representation and learning in information retrieval"
PhD Thesis, COINS Technical Report 91-93
Department of Computer and Information Science, University of Massachusetts, Amherst, MA 01003

[Miller 90]
G. Miller. (1990)
"Special Issue, WordNet: An on-line lexical database"
International Journal of Lexicography, Vol. 3, Num. 4

[Sanderson 94]
M. Sanderson (1994)
"Word sense disambiguation and information retrieval"
Proceedings of the ACM SIGIR Conference, Vol. 17

[Sparck Jones 76]
K. Sparck Jones & C.J. van Rijsbergen (1976)
"Progress in documentation"
Journal of Documentation, Vol. 32, Num. 1, Pages 59-75

[van Rijsbergen 79]
C.J. van Rijsbergen (1979)
"Information retrieval (second edition)"
London: Butterworths