An implementation of best match algorithm for Korean text retrieval in the client/server enviornment.

Hyo Sook Lee (R & D division, N-Technology; part-time lecturer, Sangmyung University)

〈Content〉

1. Introduction

While the interactive text retrieval system has been mainly based on partial match or Boolean logic, it has been continuously suggested (Robertson, 1977; Perry and Willett, 1983; Salton and Buckley, 1988) that the best match retrieval algorithms should be applied for free-text retrieval. The system employing this algorithm retrieves documents in descending order of the matching function that reflects the degree of the similarity between the query and a document. Among the experimental or operational systems on the Unix platform, some efforts (Walker,1997; Larson et al.,1996) applying this algorithm for natural language access to textual databases in English has been already reported.

This paper presents the applicability of the best match algorithm for Web-based retrieval for the Korean texts. For natural language access, stopwording and stem ming has been done before applying the matching algorithm. Best match retrieval for Web-based database in Korean language has been implemented.

2. Experimental Background

2.1 Best match search

The main concept of the best match searching algorithm includes both nearest neighbor and ranked output (Willett, 1988). The earlier researches of best match algorithm have been focused on English textual databases (Smeaton and van Rijsbergen, 1981; van Rijsbergen and et al., 1981), meanwhile recently the research for Chinese text retrieval using TREC Chinese text collection has been reported. (Huang and Robertson, 2000). Offering the various match functions option on a low-level operation and providing an interactive interface for naïve user, the experimental retrieval system e.g. OKAPI used the best match retrieval model. The system with best match search algorithm computes the similarity between a query and a document after matching stems, and makes the retrieved document sets ordered depending the document retrieval value.

In a best match retrieval model, the first document expecting to be examined by the searcher has the most similarity for a query. A range of matching functions is available to measure the similarities between search queries and documents. So when the probabilistic similarity measure is used for a best match search, the system can output the document in the order of the greatest probability which is relevant to a query (Robertson, 1977). And the weights derived from users relevance judgements through the inspection of the initially retrieved output can be used for more enhanced retrieval performance.

In this experiment, the probabilistic best match retrieval model has been applied. And the stemmed keywords of each query are matched with the texts which are also processed by the text processor.

2.2 main components of match function

The match function of this scheme includes two major parts: one is index term weighting which computes the numerical values of the terms in view of their relative importance for a certain query; the other is similarity coefficient which calculates retrieval status value for each document matched with search terms. Previous researches reported that the retrieval performance depends on the weighting functions to calculate the index terms than the choice of similarity coefficients for matching of documents and queries (Willett, 1988; Huang & Robertson, 2000).

For the term weighting scheme without relevance information, an approximation to inverse collection frequency (Sparck Jones, 1979) has been introduced and used for a best match function. Under the assumption that the occurrences of index terms in documents are statistically independent, new sets of weights which reflect the importance to each of the query terms have been also used for this retrieval model (Robertson & Sparck Jones, 1976).

3. Experiments

For this experimental work, the major work has been done as below:

Firstly, the Korean text processor (Lee, 2000) for words conflation has been runned on the server system to permit a natural language query on the client page.

Secondly, best match search routines which compute the retrieval status value were coded and the active server pages have been runned.

Thirdly, the retrieval tests by search type have been carried out using the test collection in Korean language and the results have been evaluated with the average recall/precision measures.

The following procedures between a server and a client were operated during each retrieval run:

Recognizing the natural language query which has been submitted from the client; Activating the Korean text processing routines for stopwording and stemming of texts; Searching textual database using a database engine; Weighting search terms and computing estimated similarity values between queries and documents; Producing ranked output in descending order; showing the initial retrieval results on the client to permit the relevance judgements; Further searching with relevance feedback information.

3.1 documents and queries processing

All the evidence of previous researches suggest that the indexing language needs to be a natural language rather than the controlled language oriented and that selective text content characterization is needed but it should be derived from the text. For the end-user searching, search terms extracted from texts can be directly accessible by the user for a query formulation without knowing highly artificial language (Lewis & Sparck Jones, 1996).

To process documents as well as queries in a natural language, stemming procedures have been done. The algorithm of the Korean stemmer for word conflation is as follows:

```
1. Initial processing routines
2. Check the number of syllables
    if syllable_number>=1 go to 2; else go to 7;
3. Select rule table
    case a: {compare string with rule table3;
             if found go to 7; else get current_word and terminate;
             }
    case b: go to 3;
    case c: go to 4;
4. Search relevant rule
    compare string with a stem dictionary;
            if found go to 7;
            else {compare string with both suffix list and rule table3;
                  if found go to 5;
                  else {compare string with rule table4;
                        if found go to 5;
                        else go to 7;
                        }
                  }
5. Search stem dictionary
    if syllable_number>2 compare string with a stem dictionary;
            if found go to 7; else go to 6;
6. Apply rule
    if context_sensitive
            {apply rule;
             remove suffix;
             go to 7;
            }
    else {remove suffix;
             go to 7;
         }
7. Partial match
    assign syllable_number+1 to number;
    compare string with strlen(entry)+2;
    if found go to 7;
    else {assign word(syllable_number-1) to word;
             go to 7;
         }
8. Return
    Get word and terminate
```

3.2 Weighting function

The main components of the best match search scheme are:

$\sum_{T \in Q} w_{(1)} \, [(k_1+1)tf \,/\, (K+tf)][((k_3+1)qtf)/(k_3+qtf)] \; + \; k_2 \cdot |Q| \cdot \; [(avdl-dl) \,/\, (avdl+dl)]$

where

Q is a query, consisting of terms T

$w^{(1)}$ is the Robertson-Sparck Jones weight

$\log [(r+0.5)/R-r+0.5)]/[(n-r+0.5)/(N-n-R+r+0.5)]$

N is the number of items (documents) in the collection

n is the number of documents containing the term

R is the number of documents known to be relevant to a specific topic

r is the number of relevant documents containing the term

K is $k1((1-b) + b \cdot dl /avdl)$

k_1,b, k_2 and k_3 are parameters which depend on the database and possibly on the nature of the topics

tf is the frequency of occurrence of the term within a specific document

qtf is the frequency of the term within the topic from which Q was derived

dl is the document length

$avdl$ is the average document length

3.3 Retrieval tests

The purpose of the test was to compare the retrieval effectiveness of best match search between with relevance information and without it in the system. As a test collection, KT set and 26 queries with relevance information have been used.

For comparison, two different types of searches have been runned, i.e. one is the initial search without relevance judgement and the other is to use retrospective relevance information for query expansion. For the initial search, an approximation to inverse collection frequency has been used in formula of the best match weighting function.

4. Experimental Results and Discussion

The average precision and recall measures for tested queries have been used to evaluate the effectiveness. And they are examined at three different cut-off points, i.e. 5, 10 and 15 to investigate whether the cut-offs impact the retrieval effectiveness. The results are summarized in the Table1.

Table1. Average precision-recall ratio at three cut-off points.

search type	variable	cut-off 5	cut-off 10	cut-off 15
feedback	precision	0.3308	0.2577	0.2062
	recall	0.1162	0.1712	0.2204
no feedback	precision	0.2923	0.2231	0.1869
	recall	0.1112	0.1558	0.1962

As shown in Table 1, best match searches with relevance information are more effective at all three cut-offs in terms of aveage precision and recall ratio than the searches without relevance information.

To see the statistcal significance in pairs of observation, a sign test has been used. For this test, the significance level a was defined at 0.05 and one-tailed test was used, since the test is to predict the direction of the difference.

Frequencies and test statistics are shown in Table 2 and Table 3.

Table 2. Frequencies of the pairs of observations

	cut-off 5	cut-off 10	cut-off 15
Negative differences	4	4	4
Positive differences	9	13	13
Ties	13	9	9
Total	26	26	26

Table 3. Test statistics

	cut-off 5	cut-off 10	cut-off 15
Exact Sign(2-tailed)	.267	.049	.049
Exact Sign(1-tailed)	.133	.025	.025
Point Probability	.087	.018	.018

Non-parametric sign test showed that probabilistic best match retrieval with relevance information is significantly effective at cut-off 10 and cut-off 15. But further work with heterogeneous collections is needed, since it has not been proved that the improvements of retrieval effectiveness completely are not affected by cut-off points.

5. Conclusion

The application of best match search scheme for Korean texts in the client-server environment have been presented and evaluated. The experimental results demonstrate that best match retrieval using relevance judgement on the client is better than the retrieval without it. More experimental work is needed in future to support the findings which are given in this research and finally to use the search scheme in the practical operational systems.

References

Abu-Salem, H., M. Al-Omari and M. W. Evens (1999). Stemming methodologies over individual query words for an Arabic information retrieval system. *Journal of the American Society for Information Science,* 50:524-529.

Ekmekcioglu, F. C., et al. (1996). Comparison of n-gram matching and stemming for term conflation in English, Malay and Turkish Texts. *The Journal of Computer Text Processing,* 6: 1-14.

Fujisawa, H. and K. Marukawa (1995). Full-text search and document recognition of Japanese text. *Fourth Annual Symposium on Document Analysis and Information Retrieval.* 55-79. Las Vegas: University of Nevada.

Huang, X. and S. Robertson (2000). A Probabilistic approach to Chinese information retrieval: theory and experiments. *22nd Annual Colloquium on Information Retrieval Research* 178-193. Cambridge: Sidney Sussex College.

Larson, R. R. et al. (1996). Cheshire II: Designing a next-generation online catalog. *Journal of the American Society for Information Science.* 47: 555-567.

Lee, H. S. (2000). Automatic Text Processing for Korean Language Free Text Retrieval. PhD Thesis. University of Sheffield.

Lee, H. S. and P. Willett (2000). Effectiveness of the Korean stemmer for word confation (In submission).

Lewis, D. D. and Karen Sparck Jones (1996). Natural Language Processing for information retrieval. Communications of the ACM. 39: 92-101.

Perry, S. A. and P. Willett (1983). A review of the use of inverted files for best match searching in information retrieval system. *Journal of Information Science.* 6: 59-66.

Robertson, S. E. (1977). The probability ranking principle in information retrieval. *Journal of Documentation.* 33:294-304.

Robertson, S. E. & Karen Sparck Jones (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science.* 27:129-146.

Robertson, A. M. and P. Willett (1993). A comparison of spelling-correction methods for the identification of word forms in historical text databases. *Literary and Linguistic Computing,* 8: 143-152.

Salton, G and C. Buckley (1988). Term weighting approaches in automatic Text retrieval. *Information Processing and Management.* 24: 513-523.

Smeaton, A. F. and C. J. van Rijsbergen (1981). The nearest neighbour problem in Information retrieval: an algorithm using upperbounds. *ACM SIGIR Forum* 16: 83-87.

Sparck Jones, K. (1979). Search term relevance weighting given little relevance information. *Journal of Documentation.* 35:30-48.

van Rijsbergen, C. J. (1981). D. J. Harper and M. F. Porter (1981). The selection of good search terms. *Information Processing Management.* 17: 77-91.

Walker, S. (1997). The OKAPI online catalogue research projects. *In:* K. Sparck Jones and P. Willett (eds.). *Readings in Information Retrieval.* San Francisco: Morgan and Kaufmann, 424-435.

Willett, P. (ed.)(1988). *Document retrieval system.* London: Taylor Graham.