클라이언트 서버 환경에서 한글텍스트 검색을 위한 베스트매치 알고리즘의 구현

An Implementation of Best Match Algorithm for Korean Text Retrieval in the Client/Server Environment

이 효 숙 (Hyo-Sook Lee)*

〈Content〉

초 록

본 논문에서는 웹 기반의 데이터베이스에 대한 자연어 검색을 목적으로 베스트매치 탐색알고리즘을 구현하였다. 알고리즘의 효과를 평가하기 위해 확률적 베스트 매칭함수를 사용하는 시스템에서 적합성 정보가 문헌의 검색순위 결정에 미치는 영향을 테스트하였다. 실험결과, 스태밍된 한글 텍스트 및 질의어에 대한 베스트매치 탐색에서는 소수의 적합성 정보도 검색효율을 개선하는데 영향을 주는 것으로 밝혀졌다.

주제어 : 베스트매치, 탐색알고리즘, 스태머, 서버 스크립트, 한글텍스트

Abstract

This paper presents the application of best match search algorithm in the client/server system for natural language access to Web-based database. For this purpose, the procedures to process Korean word variants as well as to execute probabilistic weighting scheme have been implemented in the client/server system. The experimental runs have been done using a Korean test set which includes documents, queries and relevance judgements. The experimental results demonstrate that best match retrieval with relevance information is better than the retrieval without it.

Key Words : best match, search algorithm, stemer, server script, Korean texts.

* R & D division, N-Technology; part-time lecturer, Myungji University(lcsljw@chollian.net)
· 접수일 : 2001. 2. 13 · 최초심사일 : 2001. 2. 21 · 최종심사일 : 2001. 2. 21

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 stemming procedures 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 search 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 relevance judgements through the inspection of the initially retrieved output can be used for more enhanced retrieval performance.

For a natural language access for Korean texts, the probabilistic best match retrieval model has been experimented. For this purpose, before applying the matching function, the automatic text processor (Lee, 2000) has been operated for processing stopwords as well as word variants. After these procedures have been executed, the stemmed keywords of each query are matched with the documents which are also processed.

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 weighting function to calculate the index terms has more influences upon the retrieval performance 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, stopwording and stemming procedures has been operated to process both the documents and queries in Korean language. So the Korean text processor has been executed on the server system for words conflation. It permits a natural language query on the client page.

Secondly, active server pages which can process best match search routines which compute the retrieval status value are coded and runned.

Thirdly, SQL statements are used for searching the Korean test collection as Open Database Connectivity (ODBC) data source in the server system.

Fourthly, the comparative retrieval tests by search type have been carried out. Each run of the search scheme has been based on two different search types: one is the initial searches without relevance judgements; the other is the searches with retrospective relevance 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, automatic text processor calls stemming procedures which run iteratively to get a stem of a word. The algorithm of the Korean stemmer which is based on context-sensitivity is as follows:

```
Initial processing routines
1. Check the number of syllables
    if syllable_number>=1 go to 2; else go to 7;
2. 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;
3. 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;
                    }
               }
4. Search stem dictionary
    if syllable_number>2 compare string with a stem dictionary;
          if found go to 7; else go to 6;
5. Apply rule
    if context_sensitive
          {apply rule;
           remove suffix;
           go to 7;
          }
    else {remove suffix;
           go to 7;
        }
6. 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;
        }
7. Return
    Get word and terminate
```

3.2 Weighting function

The main components of the best match search scheme are:

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

where

Q is a query, consisting of terms T

w 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

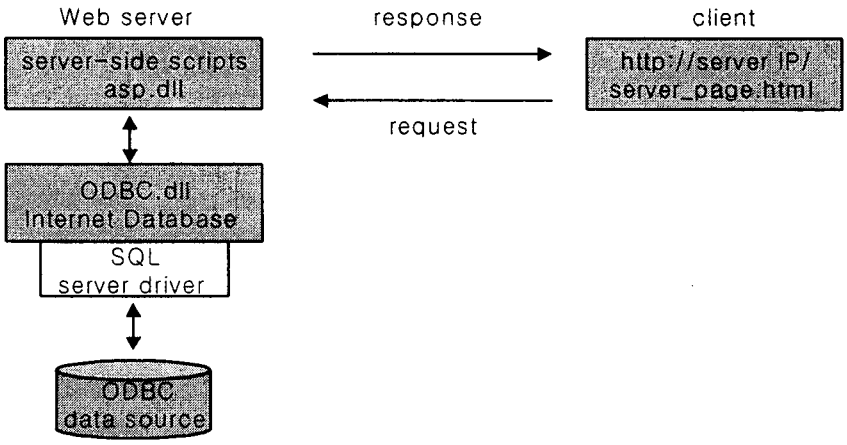
dl is the document length

$avdl$ is the average document length

3.3 Implementation

Best match search scheme has been implemented in the Internet Information Server(IIS) through Active Server Pages. The server initially read the scripts calling ODBC data source, SQL queries and the name of html file. The server opened the connected experimental database, operated both stopwording and stemming routines, executed the weighting scheme for documents as well as queries. After these routines were finished, it finally sent back an html response which contained the retrieval results. For feedback search, the server recognized the response information from the client page at an initial search stage and it continued to interact with the client until it could get the information to be added. It sent the final retrieval results.

The search scheme in the IIS application are shown in Figure 1 as below.



Active server pages which were created by VB scripts are as below. Each of them operated the following function:

dbinfo.asp	database connection
stemlist.asp	texts stemming
stemmer.asp	queries stemming
default.asp	searchtype selection
initval.asp	initialization of variables
querymodule.asp	searching records set using SQL queries; sorting the result set in descending order
computeweight.asp	computation of document retrieval value applying weighting function
query.asp	processing of input query strings; providing ranked output; enabling to make relevance judgements
doccontent.asp	displaying a full record when a user asks

3.4 Retrieval tests

The purpose of the test was to evaluate the retrieval effectiveness of best match search for Web-based database in Korean language. For these experimental runs, KT test collection (Kim et al., 1994) were used, which include research papers, queries and relevance judgements made by domain experts.

Two different types of searches have been done: one is an initial search without relevance judgement; the other is to use retrospective relevance information for query expansion. For the former one, an approximation to inverse collection frequency has been used for 'w' component in formula of the best match weighting function described as above, since it was assumed that relevance judgements at an initial search stage was not known.

4. Experimental Results and Discussion

Presicion as well as recall averaged at certain cut-off have been used as the evaluation measure. So they are examined at three different cut-off points, 5, 10 and 15 to investigate whether a cut-off point influences the retrieval performance. The results are summarized in the Table 1.

Table 1. 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 search with retrospective relevance information is more effective at all three cut-offs than the searches without it.

Figure 2 and Figure 3 shows the pattern of the behavior of two different search types at each cut-off point. When it is examined, it should be noted that using retrospective relevance judgements for 'w' in the best match weighting function is more effective at

all cut-offs than simply using an approximation to inverse collection frequency for this component.

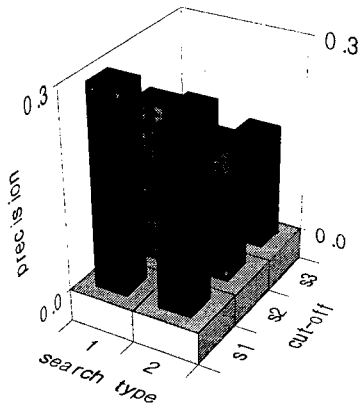


Figure 2. Precision by search type at three cut-off points

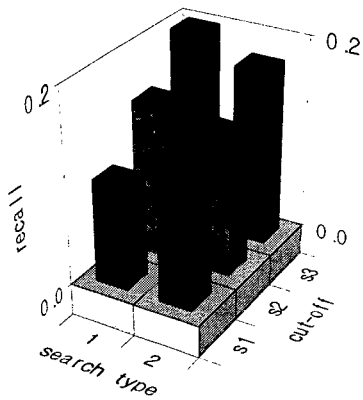


Figure 3. Recall by search type at three cut-offs

To see the statistical significance in pairs of observation, a sign test was used. For this test, the significance level was defined at 0.05 and one-tailed test was employed, 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 a)	4	4	4
Positive differences b)	9	13	13
Ties c)	13	9	9
Total	26	26	26

a) feedback > no feedback
b) feedback = no feedback
c) feedback < no feedback

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 a)	.087	.018	.018

a) a point estimate

Non-parametric sign test showed that probabilistic best match retrieval with retrospective relevance information is significantly effective at cut-off 10 and cut-off 15. So the observed differences mean a significant difference in performance at these cut-offs.

But further work with heterogeneous collections is needed to prove that the improvements of retrieval effectiveness are not affected by cut-off point.

5. Conclusion

The application of best match search scheme for Korean texts in the client/server environment has been presented and evaluated. The experimental results demonstrate that best match retrieval using retrospective relevance judgement is better than the retrieval with approximation information. It should be suggested that the implementation of this search scheme could serve as a starting point for a solution to enhance the precision ratio for Web-based database search. On the other hand, more experimental work is needed in future to support the findings which are given in this research. Another minor challenge of this scheme is to find a way to reduce the computational expense for interactive searching. The effort to resolve what is pointed out in this paper will be needed as a next research project.

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,* Vol. 50. pp. 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,* Vol. 6. pp. 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.* Las Vegas: University of Nevada. pp. 55-79.

Huang, X. and S. Robertson (2000). "A Probabilistic Approach to Chinese Information Retrieval: Theory and Experiments", *22nd Annual Colloquium on Information Retrieval Research.* Cambridge: Sidney Sussex College. pp. 178-193.

Kim, S. H. et al. (1994). "A Development of the Test Set for Estimating the Retrieval Performance of an Automatic Indexer", *Journal of Korea Information Management Society,* Vol. 11. pp. 81-102. (in Korean).

Larson, R. R. et al. (1996). "Cheshire II: Designing a Next-generation Online Catalog", *Journal of the American Society for Information Science,* Vol. 47. pp. 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 Conflation" (In submission to *Information Processing & Management*).

Lewis, D. D. and Karen Sparck Jones (1996). "Natural Language Processing for Information Retrieval", *Communications of the ACM,* Vol. 39. pp. 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,* Vol. 6. pp. 59-66.

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

Robertson, S. E. & Karen Sparck Jones (1976). "Relevance Weighting of Search Terms", *Journal of the American Society for Information Science,* Vol. 27. pp. 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,* Vol. 8. pp. 143-152.

Salton, G and C. Buckley (1988). "Term Weighting Approaches in Automatic Text Retrieval", *Information Processing & Management,* Vol. pp. 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,* Vol. 16. pp. 83-87.

Sparck Jones, K. (1979). "Search Term Relevance Weighting Given Little Relevance Information", *Journal of Documentation,* Vol. 35. pp. 30-48.

van Rijsbergen, C. J. (1981). D. J. Harper and M. F. Porter (1981). "The Selection of Good Search Terms", *Information Processing & Management,* Vol. 17. pp. 77-91.

Walker, S. (1997). The OKAPI Online Catalogue Research Projects, *In:* K. Sparck Jones and P. Willett ed. *Readings in Information Retrieval.* San Francisco : Morgan and Kaufmann. pp. 424-435.

Willett, P. ed. (1988). *Document Retrieval System.* London: Taylor Graham.