Web Searching Using Term Entropy on Virtual Document and Query Independent Importance in NTCIR-4 Web Task

Yinghui Xu Kyoji Umemura
Information and Computer Science Dept.
Toyohashi University of Technology
xyh@ss.ics.tut.ac.jp umemura@tutics.tut.ac.jp

Abstract

In this paper, we report our experiments on the NTCIR-4 Web Task. We submit results for the information retrieval task. Our goal is to evaluate the effectiveness of some important Web search functions, such as anchor text and link analysis, as well as explore the impact of combining link and content information. The distinguished characteristic of our experiment system lies in:

1. *Constructing the virtual document collections of Web pages based on DOM tree structure.*
2. *Introducing query term weighting through term entropy in virtual document collection space into general Okapi model for relevance ranking function.*
3. *Proposing a literal matching aided link analysis model for ranking Web resources.*

The experiment results show: Our proposed relevance ranking function which make use of term entropy on virtual document collection to weight query term can perform better than general Okapi model, especially for searching on anchor data. In addtion, our proposed link analysis model has the potential ability on improving searching results.

Keywords: *Information Retrieval (IR), Virtual Document (VD)*

1 Introduction

This is the first year that our group participates in the Web task of the NTCIR-4. Here we report our system and methods on information retrieval task. The main goal of ours is to investigate the effectiveness of some important Web search function, such as link analysis and anchor text. All of our experiments are conduced on a Web search platform that we designed and developed from scratch.

It has often been observed that anchor text, title information and meta data play an important role on Web searching. Accordingly, in our system, to explore the effectiveness of such kind of information, we introduce the virtual document concept that are mainly organized by those information in Web pages. Note that for the particular characteristic of Web search, usually, user tend to submit short query and terms in query are seldom repeated. Hence, traditional query term frequency based weighting scheme may fail to capture the major motivation of user. In our experiment system, we assume that a virtual document that we extracted from Web pages shares similar term distribution of user queries. Therefore, in our experiment system, query term is weighted through term entropy on virtual document collection space and then we combine it with Okapi relevance ranking function [4].

Intuitively, the link information may provide some clues as to whether a page is a key resource or not. It is thus interesting to investigate how we may combine the link information with the content information to improve the accuracy in finding key resources. We propose a novel literal matching aided link analysis model, which is used to calculate the query independent score for Web pages. How to make use of query independent evidence for adjusting the query dependent ranking sequence is also an interesting and difficult issue for Web searching task. In our experiment system, two simple ranking adjusting scheme were adopted for gauging the possibility of improvement through our proposed link analysis model.

The experiment results show that our proposed relevance ranking function with query term importance consideration and our proposed link analysis model has the potential ability on improving searching results, though the amount of improvement is modest, sensitive to the document collection and tuning parameters.

The rest of this paper is organized as follows. In section 2, we will describe our system and architecture in detail. Our proposed model will be introduced in its correspondent module of our system. In Section

3, we give the statistical information of the Web data that are processed in our system. In section 4, we will present our experiment results. In section 5, we give the conclusion.

2 Architecture and system description

According to the Overview of the Web Retrieval Task NTCIR-3 [5], we designed and developed the Web searching system from scratch. Our experiments were performed on the Web repository with EUC encoding. Dom tree based parsing scheme [3] was adopted for extracting the corresponding tag information. Morphological analyzer, Chasen [10], was used for obtaining segmented terms from Japanese Web pages after parsing. To reduce the size of dictionary of Web corpus and indexing file size, only the segmented words that belong to noun group are used for the Web page representation. The architecture of our system shows in Fig. 1. Hereafter, we would like to introduce several important modules in our system respectively.

2.1 Document generator

One of the distinguished feature in our system is the virtual document generator. The concept of virtual document (VD) is introduced by Glover [7]. We cite it in our system for investigating the functionality of some special tag information of Web pages. What is the virtual document? The virtual document of a given page is comprised of the expanded anchor text from pages that point to him and some important words on the page itself. There may several possible way to define VD. In our system, we define VD as following equations:

$$
\begin{aligned}
&AnchorText\,(i,j):\ set\ of\ terms\ appears\ in\ and\\
&around\ anchor\ of\ the\ link\ from\ page\ i\ to\ j\\
&BodyText\,(j):\\
&\begin{cases} Set\ of\ terms\ that\ appear\ in\ the\ "title"\ tag.\\ Set\ of\ terms\ appear\ in\ meta\ tag.\\ Set\ of\ terms\ that\ appear\ in\ the\ "H1, H2"\ tag. \end{cases}\\
&VD\,(j):\ Set\ of\ terms\ in\ the\ virtual\ document\ j\\
&VD\,(j)=\bigcup_{i}\left(AnchorText\,(i,j)\,,BodyText\,(j)\right)
\end{aligned}
$$

The concrete building method refers to two steps:

- At first, we create the link text table which includes triple elements $\langle URL_i, URL_j, DT\rangle$. It represents that the page with URL_i to the page with URL_j has the description text DT. The DT is extracted based on DOM tree structure. The left and right sibling node with text properties of the anchor tag "a" node and the text information under it are all extracted as description data. Considering the case that structure neighboring text node around anchor tag may be over several lines and deviate from the main anchor motivation due to the bad page structure, only the text information around anchor tag within one line are kept for description data of anchor link in our system. The description data can be regarded as the objective impression of author of Page i on the page j. Thus the collective description of what the page is about does a useful implicit resource for representing the page characteristic.
- Next, we also extract some important words from the page itself. In our system, we simply extract the following data: Text information under "title" tag, Meta "description" and "keyword" tag and "H1 H2" tag. such kind of information can be looked at as subjective presentation of page author about his motivations.

It has often been observed that users of Web search engines tend to submit very short queries, consisting of very few terms on average. In many ways, the anchor text shares this characteristic, since anchor text is typically very short and provides a summarization of the target document within the context of the source document being viewed. The process of creating anchor text for document might be a good approximation of the type of summarization presented by users to search system in most queries. In addition, it was observed by Jin, Haupmann, and Zhai [11] that document titles also bear a close resemblance to queries, and that they are produced by a similar mental process. Thus the functionality of virtual document in our system lies in:

1. Allowing set up different weighting from the actual document text information and investigating whether virtual document based searching process can improve the final ranking results.
2. Predicting the query term importance and providing different weight into Okapi ranking function.
3. Providing the representative summarization of Web pages for deciding the transition probability in our proposed link analysis model.

As for the actual document (AD), it is extracted based on the actual visible text information in the DOM tree of Web pages, denoted by:

$AD\,(j):$ set of terms in actual document of page j.

2.2 Indexer

Two kinds of indexing module are built in our system. The inverted index file [6] is used for providing the document list through direct term hitting. The forward index file, which stores a list of wordID with its term frequency of every Web page, is used for representing the bag of words structure of Web pages. To

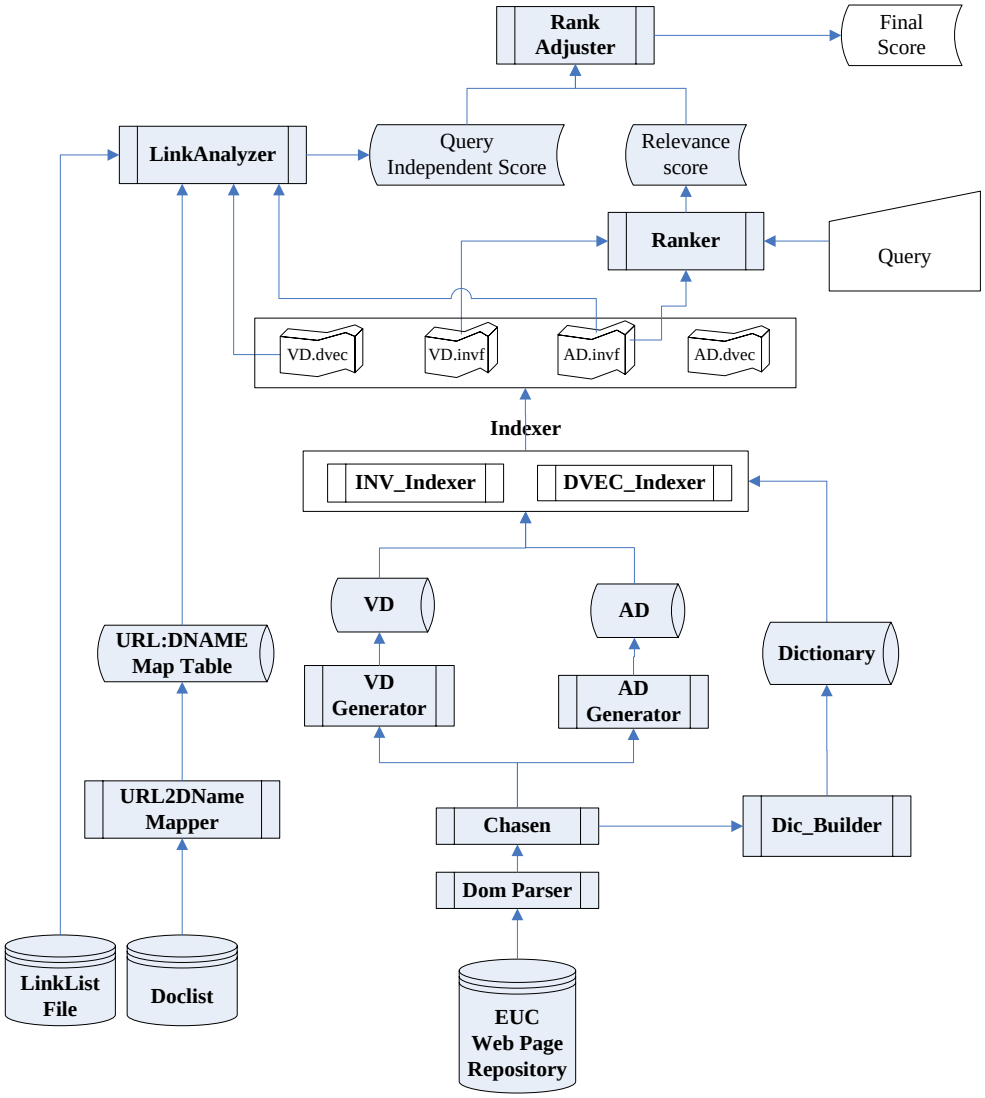


Figure 1. Implementation architecture

reduce the memory requirements, no format, type and position information of terms in a particular Web page are recorded in the forward index file. In our system, The Inverted Index file and the Forward Indexing file are created for both virtual document collection and actual document respectively.

2.3 Ranker

2.3.1 Baseline funking function (BASE)

According to some reports about Web Information Retrieval task, Okapi model are proved to be efficiency on content based Web searching. Accordingly, we use Okapi's BM25 as our baseline for comparison. The equation that we used in our system is:

$$SIM\,(Q,d) = \sum_{w \in Q} \frac{tf}{\left(tf + 0.5 + \frac{1.5 * dl}{ave\_dl}\right)} \times \frac{\log_2\left(0.5 + \frac{N}{df}\right)}{\log_2(1.0 + \log_2(N))}$$

2.3.2 Query term importance based ranking function (QTIBRF)

In the general information retrieval system, especially for a long topic, query term frequency is used to indicate the term importance for relevance ranking function. However, in the practical Web searching, usually, the input information of user is tend to short and seldom repeated. Query term frequency based ranking function may fail to capture the main purpose of user request. For example, for the query "google, pagerank", "PageRank" should be the term which reflect the

main purpose of user request. How to set up a reasonable term weighting for each query term? In our experiment system, query term are weighted by its entropy which are calculated based on virtual document collection space. The entropy based term weighting [8] in virtual document space will reflect how user think about whether it is important for his purpose or not. The computing equation is:

$$VDTF(w,j) = \#\{w|w \in VD(j)\}$$
$$P(w,j) = VDTF(w,j) \Big/ \sum_{k=1}^{N} VDTF(w,k)$$
$$VDET(w) = -\sum_{j=1}^{N} P(w,j) \log_N P(w,j)$$
$$VDTW(w) = 1 - VDET(w)$$
$$where:$$
$$N : number\ of\ virtual\ documents\ in\ virtual\ document\ collection$$

The calculated term weighting is regarded as query term importance factor which is integrated into Okapi ranking functions. The augmented Okapi Model is:

$$SIM(Q,d) = \sum_{w \in Q} VDTW(w) \times \left(\frac{tf}{\left(tf + 0.5 + \frac{1.5 * dl}{ave\_dl}\right)} \times \frac{\log_2\left(0.5 + \frac{N}{df}\right)}{\log_2(1.0 + \log_2 N)} \right)$$

Here, to indicate the effectiveness of our query term weighting scheme, we give some samples which shows in Fig. 2. From the observation, term entropy based weighting does somewhat reflect the user motivation.

Q ID	Term 1	Ent.	Term 2	Ent.	Term 3	Ent.
0003	コペルニクス	0.258	地動説	0.221	キリスト教	0.397
0012	湘南	0.437	海岸	0.482		
0032	風邪	0.375	薬	0.52	薬局	0.478
0038	憲法	0.379	問題	0.626		
0087	google	0.266	pagerank	0.121		

Figure 2. Query term and entropy

2.3.3 Score merging ranking function (SMRF)

In our system, for a given Web page, it has two kinds of information resources, virtual document and actual document. It is natural to think about merging the ranking score of the two searching process which performed on both virtual document collection and actual document collection respectively. Simple linear merging scheme was adopted:

$$FinalScore(p_i) = SIM(Q, VD(p_i)) + \lambda SIM(Q, AD(p_i))$$

Where the parameter λ is set to 0.112 after tuning.

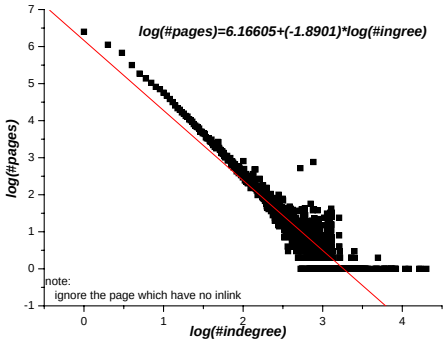


Figure 3. In-degree distribution of Web pages

2.3.4 Literal matching aided link analysis (LMALA)

Link analysis which make use of hyperlink structure for ranking Web resources is an important Web searching function. The two best-known algorithms that perform link analysis are HITS [9] and PageRank [1]. The latter, used in Google, has been proved its efficiency in the practical World Wide Web searching. In our system, the LinkAnalyzer module has the same purpose on bringing order to the Web through query independent ranking as Google's PageRank but different calculation mechanism. We propose an unified model of literal mining and link analysis. We aim at assigning a reasonable link weights through the literal information between a page contents and the virtual document contents of its target pages. The calculation mechanism is defined as: Given a page P and its outgoing sets $Q = \{q_1, q_2, \cdots, q_m\}$, the transition odds from p to q_k are determined by:

$$TranOdds(p \to q_k) = prob(VD(q_k)|p) + \sum_{w \in (VD(q_k) \cap VD(p))} prob\left(\frac{w}{p}\right)$$
$$where:$$
$$prob(w|p) = \left(\begin{array}{l} (1 - VDTW(w)) \times \log_2(tf(w,p) + 1) \\ \times \log_N({}^{N}/_{df(w)}) \end{array} \right)$$
$$prob(VD(q_k)|p) = \sum_{w \in VD(q_k)} prob(w|p)$$

Based on the calculated values that indicate transition likelihood for all possible connections on a page, we can assign the transition probability to them and regard them as the link weight in the Markov chain. Then we can use the same processing way as original PageRankt to calculate the principle eigen-vector of transition Matrix. The link allocation method is shown in the following Equations. Parameter γ is used for adjusting the probability that the surfer tends to follow those links with literal matching information. In this

paper, γ is set 0.7.

$$PR(j) = (1-\lambda)\,1/N + \lambda \sum_{i \in B_j} PR(i)\, prob(i \to j)$$

$$prob(i \to j) = \begin{cases} \gamma \times \frac{TranOdds(i \to j)}{\sum_{k \in F(i)} TranOdds(i \to k)}, \\ \quad Lit(link(i,k)) = 1 \\ (1-\gamma) \times \left(\frac{1}{(\#F(i) - LitLink(i))} \right), \\ \quad otherwise \end{cases}$$

$where:$
$B(i):\ set\ of\ pages\ which\ link\ to\ page\ i;$
$F(i):\ set\ of\ pages\ which\ page\ i\ links\ to;$
$r:\ transition\ probability\ follows\ literal\ link;$
$LitLink(i) =$
$\#\{k | k \in F(i) \wedge Lit(Link(k,i)) = 1\}$
where:

$$Liter: Link(i,j) \to \begin{cases} 1, if: A \wedge B \wedge C \\ 0, otherwise \end{cases}$$

$$\begin{cases} A: VD(j) \neq \emptyset;\ \text{B: Page (i)} \neq \emptyset; \\ C: \{w | w \in (VD(j) \cap Page(i))\} \neq \emptyset \end{cases}$$

2.4 Rank adjuster

Rank adjuster are performed on the top 1000 return results which obtained through relevance ranking funciton and the adjusting method is based on the query independent page importance score computed by our proposed link analysis model. Two kinds of rank adjusting scheme are attempted in our experiment. The first is based on simple linear score combination method, denoted as RA1; The paramter λ used in RA1 is set 0.1 after tuning.

$$FScore(P_i) = SMRF(P_i) + \lambda \times PR(P_i)$$
$$PR(P_i) = \frac{\log(LMALA(P_i) * N)}{\log(1.8)}$$
$$SMRF(P_i):\ score\ of\ P_i\ based\ on\ SMRF$$
$$LMALA(P_i):\ score\ of\ P_i\ based\ on\ LMALA$$

In the rank adjuster model 1, page rank score which is calculated by our proposed link analysis model is processed according to the distribution of in-degree of NCTIR-4 Web pages. The distribution characteristic, plotted in Fig 3, follows the power law distirbution [2], . The power value is 1.89 for NTCIR-4 Web corpus. We use this value in our ranking adjusting model 1 to normalize the PageRank score value and make it quantitatively comparable with relevance score.

In the rank adjuster model 2, we will make use of the rank information of both query dependent sequence and query independent sequence. There are two rank lists, one is ranked by SMRF score and the other is ranked by LMALA score. We assume that the higher summation of the two rank value of a page is, the lower score of the page will has. The higher difference between the two rank of a page is, the lower adjusting degree of the page score will has. The rank adjusting model 2 denoted as RA2. The paramter λ used in RA2 is set 0.08 after tuning

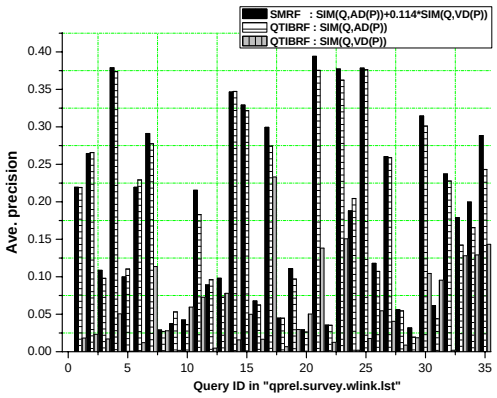


Figure 5. Topic by topic Ave. precision comparison among SMRF, QTIBRF on AD and QTIBRF on VD

$$FScore(P_i) = SMRF(P_i) - \lambda \times \frac{\tau_1(P_i) + \tau_2(P_i)}{|\tau_1(P_i) - \tau_2(P_i) + 1|}$$
$$R:\ return\ document\ sets\ for\ a\ given\ query$$
$$\tau_1:\ document\ in\ R\ sort\ by\ SMRF\ score$$
$$\tau_2:\ document\ in\ R\ sort\ by\ LMALA\ score$$
$$\tau_k(i):\ rank\ of\ i\ in\ \tau_k$$

3 Statistical information of Web corpus

The statistical information of Web corpus which is processed in our experiment system shows in Fig.4. There are 10,894,819 pages in the crawled Web corpus, where 325,277 pages have no page contents due to the page parsing errors or some other reason. Compared with the actual docuemnt collection, the size of virtual document collection is much smaller and make it easy to handle. As for the link structure [2] is used for describing its graph feature. Over 15 milion nodes to expose the large-scale structure of the Web graph as having a central, strongly connected core (SCC); a sub-graph (IN) with directed paths leading into the SCC, a component (OUT) leading away from the SCC. and relative isolated tendrils attached to one of the three large sub-graphs. In our proposed link analysis model, transition probability will be based on literal information which exist in Web pages, therefore, documents with "NX" prefix which have no page contents are ignored in our link analysis module. From the information in the Fig 4, it shows clearly that not all page entities in the Web corpus have their own page rank value. There are around 3 million pages without page rank value in our system.

4 Experiment results and analysis

In this section, we will present our experiment results which are done based on the modules that we

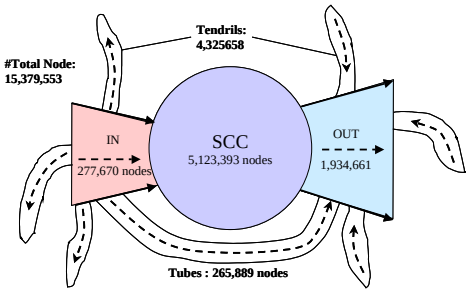
Actual document collection of Web corpus			
Number of document	10,569,542	Size of inverted index file	9.69G
Number of unique word	1,468,591	Size of forward index file	9.67G
Maximum document length	593,863	Average document length	197
Virtual document collection of Web corpus			
Number of document	9,693,268	Size of inverted index file	738M
Number of unique word	488,259	Size of forward index file	839M
Maximum document length	119,771	Average document length	19
Statistical information of Web graph			
Bow tie structure			
Total nodes	15,379,553		
Number of nodes in SCC	5,123,393		
Number of nodes of “IN”	277,670		
Number of nodes of “OUT”	1,934,661		
Number of tendrils nodes	4,325,658		

Figure 4. Statistical information of processed NTCIR-4 Web data

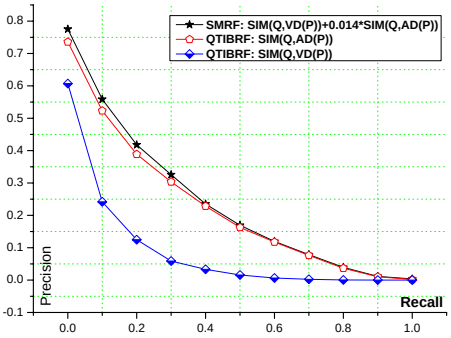


Figure 6. comparison of 11ppt precision recall between SMRF and BASE on AD

introduced in section 2. Without special explanation, experiment results that we reported in this paper are all under the condition that relevance ranking score is calculated for topic ”title” and relevant judgment file is the ”qprel.target.wlink.1st”, which contain 80 queries and 9244 relevant files. We think that the title information in topic is similar to the query which input by user in the practical Web searching.

At first, the comparison experiments of the BASE and QTIBRF on both virtual document collection and actual document collection are performed to evaluate the effectiveness of our proposed augmented Okapi model, results shows in table 1. It shows that our proposed QTIBRF model got improvements of average precision on both collection space. At the same time, we note that QTIBRF module did not get improvements on R-precision at the top 10 and 20 documents for actual document collection while obtained improvements on both Average precision and R-precision in the virtual document collection searching. It indicates that QTIBRF module may be more adaptable for improving anchor text based searching. To show the comparison clearly, the results based on all topic runs are also showed in table 1

Next, we continue to investigate whether the combination of the two score obtained through QTIBRF model for both virtual document and actual document collection can get more improvements than any individual searching using OTIBRF model. From another point, we are try to find whether the anchor text and some other special tag information can boost the precision of normal full-text searching. The comparison experiments were done and results shows in Table 2. It shows that page score after merging got 3.8 percent improvement over QTIBRF on AD only and great improvement over QTIBRF on VD only. To show comparison at each recall level, the 11PPT. precision recall results are plotted in Fig 5. It shows that the distinct improvements obtained through combination came out at recall less than 0.3 and there are no clear difference at higher recall. Such phenomenon can be explained that virtual document based searching get worse precision which is near 0 at the recall over 0.3, accordingly, the contribution on improving merging score is much smaller or nothing. To indi-

Table 1. Comparison results of BASE and QTIBRF on virtual document collection and actual document collection

		Vitual document			Actual document		
	Topic	AveP	P@10	P@20	AveP	P@10	P@20
BASE	tt	0.0621	0.2738	0.2206	0.2052	0.4550	0.3931
QTIBRF	tt	0.0705	0.2850	0.2431	0.2127	0.4487	0.3850
BASE	desc	0.0579	0.2550	0.2038	0.1839	0.4300	0.3713
QTIBRF	desc	0.0641	0.2825	0.2306	0.1987	0.4225	0.3625
BASE	alt0	0.0537	0.2287	0.1975	0.1109	0.2775	0.2456
OTIBRF	alt0	0.0551	0.2338	0.2038	0.1110	0.2725	0.2431
BASE	alt1	0.0511	0.2458	0.1861	0.1872	0.4444	0.3819
OTIRBF	alt1	0.0594	0.2542	0.2083	0.1911	0.4111	0.3708
BASE	alt2	0.0437	0.2286	0.2036	0.1646	0.4268	0.3884
OTIRBF	alt2	0.0495	0.2571	0.2321	0.1650	0.4125	0.3768
BASE	alt3	0.0259	0.1393	0.1250	0.1571	0.3607	0.3018
OTIRBF	alt3	0.0317	0.1500	0.1536	0.1578	0.3571	0.3000

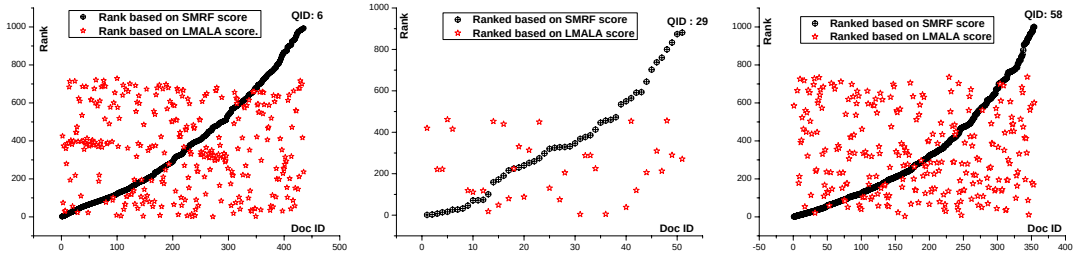


Figure 7. Rank comparison between query dependent relevance ranking and query independent link analysis ranking of relevant judgment files of some sample queries

cate the relationship between three ranking scheme for each query clearly, Average precision based on topic by topic are calculated and figured out in Fig 6. To reduce the number of topics in the graph, relevant judgment file "qprel.survey.wlink.1st", which contain 35 queries and 5674 relevant files, is used. Results shows in the Fig 5. We can find that Most of topics get the best Ave. precision in SMRF based method. What is more, for the case that topic get rather low Ave. precision, the Ave. precision based on SMRF will not exceed QTIBRF on AD or even worse than it. Therefore, we can conclude that virtual document based searching can boost the searching efficiency based on our augmented Okapi model.

At last, we will introduce the experiment results of our attempts on rank adjusting using query independent page importance score, described in section 2.4. As for the comparisons between Google's PageRank and our proposed approach are reported in our SIGIR-2004 paper [12]. As we pointed out in section 3, due to the incomplete link structure of Web corpus, one-fourth pages in the Web repository has no its correspondent page rank value. It leads to some relevant pages in the file "qprel.target.wlink.1st" and "qprel.survey.wlink.1st" have no correspondent page rank value, such issues bring some trouble for us on analyzing the effectiveness of rank adjusting methods. Therefore, to make the comparison experiment reasonable, we remove the pages that have no page rank value from the top 1000 return sets of IR baseline and then our proposed rank adjusting methods are performed on the left sets. The experiment results that shows in Table 3 are evaluated based on the "qprel.survey.wlink.1st" relevant judgment file. Though there is no clear difference between the re-ranking-before and re-ranking-after on IR results, we found some interesting point. Simple linear score combination get worse precision at top return sets. Such results tell us that content based IR score should be much more important than link based PageRank score at lower relevance rank and its importance will get weaker with the increasing of the relevant rank. That is to say, the linear combination may not be adaptable for score combination scheme. The results of the RA2 model which make use of rank information did not make the precision at top return sets

worse. It indicates that in the rank adjusting model, the rank information of both list (document list based on content IR score and PageRank score) may also be an important factor. In addition, to help us judging whether query independent ranking based on our proposed link analysis model can improve information retrieval results or not, the comparison of relevance rank and page importance rank (page rank) for relevant files of some queries are conducted. Some example plots are figured out in the Fig 7. The star points which represent the pages ranked by the LMALA score locate at both sides of the circle points which represent the pages ranked by the SMRF score. It indicates that there is possibility to get improvement through rank adjustment based on link analysis. Therefore, it is reasonable for us to believe that our proposed link analysis model has the potential ability on ranking Web resources and improving IR results.

Table 2. Comparison results of QTIBRF and SMRF

	Rank Fun.	Ave. P	p@10	p@20
VD only.	QTIBRF	0.0705	0.2850	0.2431
AD only	QTIBRF	0.2127	0.4437	0.3750
(VD+AD)	SMRF	0.2208	0.4767	0.4184

Table 3. Comparison results among SMRF, RA1 and RA2

		SMRF	RA1	RA2
Ave. P		0.1203	0.1212	0.1204
Recall	0.0	0.7036	0.7116	0.7226
	0.1	0.4157	0.4246	0.4143
	0.2	0.2576	0.2577	0.2557
	0.3	0.1751	0.1759	0.1740
Prec.	@5	0.4629	0.4457	0.4629
	@10	0.4000	0.3943	0.4057
	@20	0.3529	0.3514	0.3543
	@30	0.3314	0.3286	0.3343

5 Conclusions

In this paper, we report our experiment system for NTCIR-4 Web task. We also proposed query term weighting scheme for augmented Okapi model, score merging mechinism and literal matching aided link analysis model in our system. The experiment results shows: Our proposed relevance ranking function which make use of term entropy on virtual document collection to weight query term can perform better than general Okapi model, especially for anchor text based searching. In addtion, our proposed link analysis model has the potential ability on improving searching results.

References

[1] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30:107–117, 1998.

[2] A. Broder and r. Kumar etc. Graph structure in the web: Experiments and models. In 9th Inc. World Wide Web Conference, 2000.

[3] S. Chakrabarti. *Mining the web : Discovering Knowledge from Hypertext Data*. MORGAN KAUFMANN PUBLISHER, San Francisco, CA 94104-3205, 2003.

[4] W. B. Croft. *Language modeling for information retrieval*. KLUWER ACADEMIC PUBLISHERS, BOSTON, 2003.

[5] K. Eguchi and K. O. etc. Overview of the web retrieval task at the third ntcir workshop. In *Proceedings of the Third NTCIR Workshop on Research in Information Retrieval, Automatic Text Summarization and Question Answering*. NII, 2003.

[6] W. B. Frankes and R. Baeza-Yates. *Information retrieval - Data Structure Algorithms*. Prentice Hall, Englewood Cliffs, New Jersey 07632, 1992.

[7] E. J. Glover and K. T. etc. Using web structure for classifying and describing web pages. In *Proc. 11th WWW*, pages 562–569, 2002.

[8] H.-Y. Kao and S.-H. Lin. Mining web information structure and content based on entropy analysis. *IEEE transactions on Knowledge and Data engineeding*, 16, 2004.

[9] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[10] Y. Matsumoto and A. K. etc. Japanese morphological analysis system chasen version 2.2.1. 2000.

[11] A. G. H. Rong Jin and C. Zhai. Title language model for information retrieval. In *In Proc. Of the 25th annual international ACM SIGIR conference on research and development in information retrieval*, pages 42–48. ACM SIGIR, 2002.

[12] Y. Xu and K. Umemura. A unified model of literal mining and link analysis for ranking web resources. to be published in SIGIR 2004.