Machine Learning Approach for Homepage Finding Task

Wensi Xi and Edward A. Fox
Department of Computer Science
Virginia Polytechnic Institute and State University

ABSTRACT

This paper describes new machine learning approaches to predict the correct homepage in response to a user's homepage finding query. This involves two phases. In the first phase, a decision tree is generated to predict whether a URL is a homepage URL or not. The decision tree then is used to filter out non-homepages from the webpages returned by a standard vector space IR system. In the second phase, a logistic regression analysis is used to combine multiple sources of evidence on the remaining webpages to predict which homepage is most relevant to a user's query. 100 queries are used to train the logistic regression model and another 145 testing queries are used to evaluate the model derived. Our results show that about 84% of the testing queries had the correct homepage returned within the top 10 pages. This shows that our machine learning approaches are effective since without any machine learning approaches, only 59% of the testing queries had their correct answers returned within the top 10 hits.

1. INTRODUCTION

With the fast development of the internet and World Wide Web, information from the Web has become one of the primary sources of knowledge for human beings. Although traditional information retrieval techniques have provided many methods to seek relevant information from the internet in response to a user's need, they are still far from sufficient in some cases, such as when a user is seeking information that is too broadly or vaguely specified for a traditional IR system to give a precise result. On the other hand the linking structure and various tagged fields of a Web page can be rich sources of information about the content of that page. Making use of this information can be very helpful in solving those information seeking problems that can not be satisfactorily solved by traditional IR techniques. Among this kind of user's information need are special information seeking tasks like "homepage finding" which involves trying to find the entry page to a website. This paper describes new methods of using machine learning approaches to consider extensive tagged field, URL, and other information to best predict the relevant homepage in response to a user's homepage finding query. The rest of the paper will be organized as follows: related work will be introduced in Section 2; research direction will be described in Section 3; the baseline IR system will be explained in Section 4; machine learning models and results will be reported in Sections 5 and 6; and research will be summarized and discussed in Section 7.

2. RELATED WORK

There are two major methods to make use of link information to identify the correct webpage in response to a user's query: the page rank algorithm and the HITS algorithm.

The page rank algorithm was first introduced by Page and Brin [1]. This algorithm was developed because using in-degree as the predictor of quality is weak. First, not all the back pages are of the same importance. Second, in-degree is spammable. In their page rank algorithm each page was first evaluated as to quality. Then each page allows all the page links to it to distribute their "value" of quality to it. The quality value of each page was divided by the out-degree before they could distribute their "authority" to other pages. The algorithm can be summarized as:

$$\text{PageRank}(P) = \beta/N + (1- \beta)\Sigma\text{PageRank}(B)/\text{outdegree}(B)$$

where β is the probability of a random jump to P and N is the total number of pages on the web.

The HITS algorithm was first introduced by Kleinberg [4]. He assumes that a topic can be roughly divided into pages with good coverage of the topic, called authorities, and directory-like pages with many hyperlinks to pages on the topic, called hubs. The algorithm aims to find good authorities and hubs for a topic. For a topic, the HITS algorithm first creates a neighborhood graph. The neighborhood contains the top 200 matched webpages retrieved from a content based web search engine; it also contains all the pages these 200 webpages link to and pages that linked to these 200 top pages. Then, an iterative calculation is performed on the value of authority and value of hub. Iteration proceeds on the neighborhood graph until the values converge. Kleinberg claimed that the small number of pages with the converged value should be the pages that had the best authorities for the topic. And the experimental results support the concept. Kleinberg also pointed out that there might be topic diffusion problems (with the answer shifting to a broader topic related to the query). There also might be multi-communities for a query, where each community is focused on one meaning of the topic. Sometimes the first-principal community is too broad for the topic and the 2nd and 3rd community might contain the right answer to the user's query.

Combining multiple sources of evidence from different IR systems to improve the retrieval results is a method applied by many researchers (e.g. [7] [9]), and had been proved to be effective. Using regression analysis to improve retrieval also had been studied, e.g. in [2].

Recently, Craswell and Hawking [3] used anchor text to retrieve documents in response to a homepage finding task, and compared their result with full-text retrieval. They found anchor text retrieval is far more effective than full-text retrieval.

3. RESEARCH DIRECTION

Our research makes use of the WT10g web collection provided by the TREC staff. The WT10g collection is about 10GByte in size and contains 1,692,096 webpages crawled in 1997. The average size of a webpage in the collection is 6.3 KBytes.

The TREC Conference provided 100 sample homepage finding queries and their corresponding correct answers (homepages). These sample queries can be used to train the homepage finding system developed. TREC also provided another 145 testing queries without corresponding answers. These queries can be used to evaluate the system developed.

The 100 sample homepage finding queries are very short queries. Most of them only contain 2 to 3 words. They include the name of an institute (e.g., UVA English department), organization (e.g., Chicago Computer Society), or a person's name (e.g., Jim Edwards). Some of the queries also contain descriptive information (e.g., Unofficial Memphis Home Page). After a close analysis of the 100 training queries and URLs of their corresponding homepages, we found these clues:

- A homepage usually ends with a "/"
- Most homepages contain at most 2 other "/", beyond the 2 in http://
- The last word in the homepage URL (if the URL is not ending with a "/") is usually: index.html; index1.html; homepage.html; home.html; main.html; etc.

Most of the 100 sample homepages confirm these rules. However there are exceptions, for example:

McSportlight Media This Week ->
http://www.mcspotlight.org:80/media/thisweek/
LAB MOVIE REVIEW SITE –>
http://www.ucls.uchicago.edu:80/projects/MovieMetropolis/
The Boats and Planes Store ->
http://www.psrc.usm.edu:80/macrog/boats.html

The basic rationale for UR analysis is to filter out non-homepages that rank at the top of the rank list returned by the content based information retrieval system, so that the correct hompages can be ranked higher.

The TREC also provided two mapping files:

in_links: which maps the incoming links to each collection page

out_links: which maps outgoing links from each collection page

4. BASELINE IR SYSTEM

At the beginning of this research, a vector space model IR system was developed to retrieve relevant webpages for each of the 100 training homepage finding queries. The vector space model IR system uses a stop word list to filter out high frequency words. Each word left is stemmed using Porter's algorithm [4]. The IR system uses the *ntf*idf* [6] weighting scheme with cosine normalization to construct the query vectors and the *tf*idf* weighting scheme with cosine normalization to construct the document vectors. *ntf* refers to *normalized term frequency* and is given by the formula:

$$ntf = 0.5 + 0.5 * tf / max\_tf$$

where *max_tf* is the highest term frequency obtained by terms in the vector. The retrieval score for the document is calculated by taking the inner product of the document and query vectors.

The WT10g Web collection contains 3,353,427 unique keywords (after filtering out stopwords, and stemming). The inverted file developed from this collection is about 3 Gbytes in size.

Tagged fields:

In order to investigate the importance of tagged fields in HTML files during the retrieval, several tagged fields were extracted from the WT10g collection. The tagged fields extracted were <title>, <meta>, and <h1>.

Anchor texts:

Anchor texts are the text description of a hyperlink in a webpage. Previous research [3] had found that anchor text retrieval could help improve retrieval performance. In this research work, we extracted and combined the anchor texts with the destination webpage it links to and built a separate anchor text collection, in which each page only contains all the anchor text of other pages describing it.

Abstracts:

Some researchers [5] had found that text summary and abstract retrieval can yield comparable or even better retrieval results than full-text retrieval. Retrieval using abstracts also can save substantial time and space. In this research work, we extracted text to approximate an abstract for each webpage. The abstract contains the URL of the webpage, the <title> tagged field of that page, and the first 40 words following that field in that page. The extracted abstract collection is about 7% of the size of the WT10g collection. The rationale for the abstract collection is that we believe a homepage is very likely to repeat its name in its URL, title, or at the beginning of its homepage, and so this is more likely to achieve better results than full-text retrieval. On the other hand, the abstract contains more information than would the title field, and is not likely to lose the correct answer to queries; thus we should obtain higher recall.

The statistical facts of the full-text, tagged field, anchor, and abstract collections are listed in Table 1 below:

Table 1. Statistical facts of the various collections

Name	Size (Mbytes)	No. of Docs	Avg Doc Length (Kbytes)	Inverted File Size (Mbytes)	No. of Unique Terms
Full text	10000	1692096	6.3	3000	3353427
Title tag	100	1602137	62.5	59	158254
Meta tag	50	306930	167	28	59122
H1 tag	29	517132	56	15	82597
Anchor	180	1296548	138	53	219213
Abstract	710	1692096	420	400	646371

Retrieval results

Table2 and Figure1 report the retrieval result for the 100 testing queries on different collections. From the table we find that the <meta> tag and <h1> tag each performs poorly. This shows that the text in these fields is not a good indication of the main topic of the webpage. Full text retrieval doesn't work very well either. Abstract retrieval works much better than the full-text retrieval as we expected. Anchor text retrieval performs slightly better than abstract retrieval in terms of MRR (Mean Reciprocal Rank). Title tag retrieval performs best of all.

Table 2. Baseline system retrieval results for training queries

Relevant doc found in	full-text	title tag	meta tag	h1 tag	anchor text	abstract
Top1	5	34	4	7	22	23
Top5	18	62	8	11	47	40
Top10	25	68	11	14	54	49
Top20	34	73	14	14	57	59
Top100	48	85	18	15	65	73
Not in list	0	5	73	84	18	2
MRR	0.12	0.46	0.06	0.09	0.33	0.31

$MRR = \Sigma(1/rank)/N$

N: Number of queries

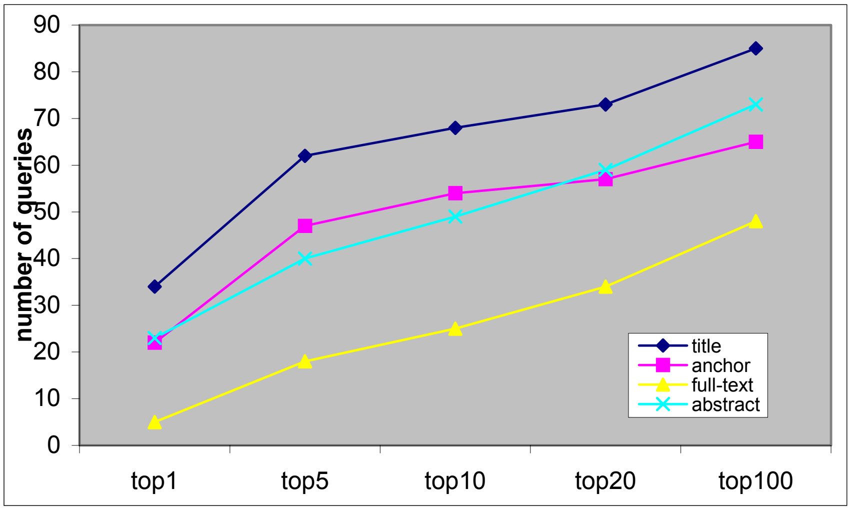

Figure 1. Baseline retrieval results comparison chart for training queries

5. DECISION TREE MODEL

In the second phase of our research work, a decision tree was generated to predict whether a URL of webpage is a homepage URL or not. The detailed steps are:

1. Manually select 91 non-homepages from the WT10g collection. These pages are identified not only by the content but also by the in-links and out-links of the pages and by the structure of the URL.

2. Develop attribute vectors for the 198 cases (107 positive cases provided from TREC and 91 negative cases derived manually); the attribute vectors contain these factors:

- URL length: the number of slashes in the URL;
- In link: the total number of in links;
- In link normalized by homepage: total number of in links divided by the length of the webpage;
- In link from outer domain: the number of in links from outer domains;
- In link from same domain: the number of in links from the same domain;
- Out link: total number of out links of a webpage;
- Out link normalized by homepage: the total number of out links divided by the length of the Web page.
- Out link to outer domain: the number of out links pointing to other domains,
- Out link to same domain: the number of out links pointing to the same domain;
- Keyword: whether the URL ends with a keyword; these keywords are “home”, “homepage”, “index”, “default”, “main”;
- Slash: whether the URL ends with “/”;

- Result: whether it is a homepage or not.

3. The 198 training vectors were provided to the data mining tool C5 or See5 (available at http://www.rulequest.com/see5-info.html). A decision tree was developed by the rule generator based on these training vectors. It can be seen in Figure 2. The correctness of the decision tree against the training cases is 97%.

4 Another 102 test Web pages were manually selected from the TREC collection. Among them, 27 are homepages. The decision tree was evaluated on the test cases and the results were 92% correct. This indicates that the decision tree model is fairly reliable.

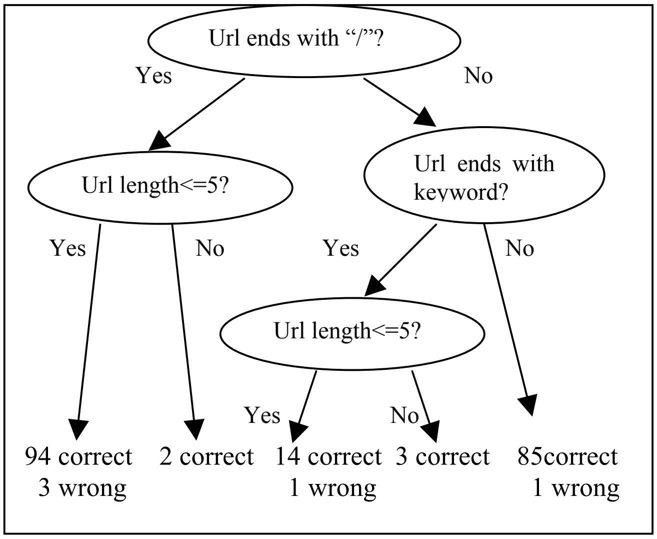

Figure 2. Decision Tree Model

5. The decision tree then was applied to the results returned by the baseline IR system, in hopes that we can filter out most of the non-webpages in these returned webpage lists. The decision tree model was only applied to anchor, title field, and abstract retrieval. Results of the decision tree applied on the title and anchor text retrieval can be found in Table 3.

6. LOGISTIC REGRESSION MODEL

In the third stage of this research work, a logistic regression analysis model was developed to combine link information with the various scores returned by the standard IR system, in order to improve the rank of the correct homepages in response to the query. The detailed steps are:

1. Two training queries (No. 5 and No. 51) were taken out of consideration because their correct answer was already filtered out as non-homepages by the decision tree model. The top 1000 pages for each rank-list file of the remaining 98 training queries were taken into the logistic regression analysis. Thus, there were 67855 pages in the training set; among them 104 pages were relevant to a specific query.

2. A logistic regression analysis was made using SAS software, version 8.02. The evidence thrown into the logistic regression analysis included IR scores from title, anchor, and abstract retrieval. (All scores are pre-normalized by the maximum score for each query, thus, the score ranges from 0 to 1.) Various types of linking information and the URL length information also were

considered. The logs of all these factors were thrown into the logistic regression analysis. The predicted factor is whether a page is relevant to a query (1) or not (0). The system showed that the log of title retrieval score, title retrieval score, anchor retrieval score, abstract retrieval score, and the reciprocal of the URL length can be used to predict the relevance of a webpage to a query. The correlation is 98%.

3. The formula derived from the logistic regression analysis was then applied to the 98 training queries. 70 queries found the correct answer on top of the list, 96 queries found the correct answer within the top10. The MRR is 0.802, which is 13% better than the title retrieval after non-homepage removal by using the decision tree model (the best of all the runs in the previous stage).

Results of the model can be found in Table 3 and Figure 3.

Table 3. Machine learning model results for training queries

Relevant Found in	**Anchor + Tree**	**Title + Tree**	**Abstract + Tree**	**Logistic Regression**
Top1	43	62	50	70
Top5	61	83	67	94
Top10	63	84	75	96
Top20	65	86	79	96
Top100	72	92	92	97
Not in list	19	7	4	3
MRR	0.504	0.710	0.597	0.802
Improve-ment	50% over Anchor	55.7% over Title	90.7% over Abstract	13% over Title + Tree

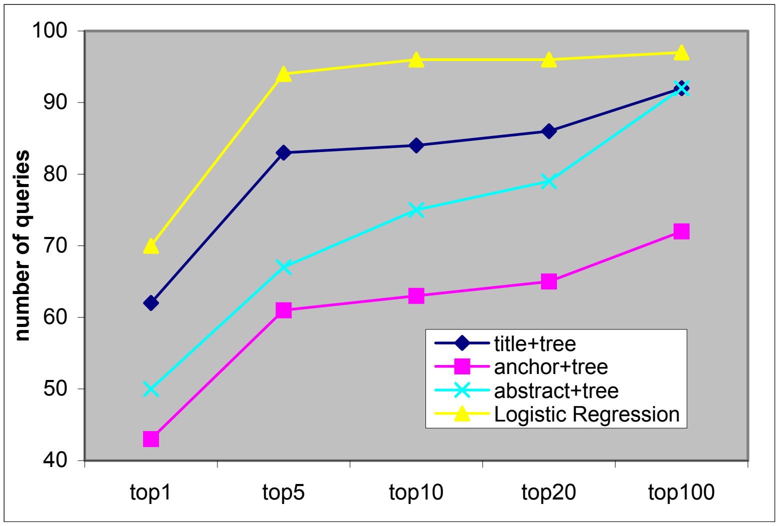

Figure 3. Machine learning model retrieval results comparison chart for training queries

7. TESTING RESULTS AND DISCUSSION

Finally, 145 testing queries provided by TREC were used to evaluate our system. Table 4 and Figure 4 report the retrieval results for the testing queries on title field retrieval with the baseline IR system. From the table we find that testing queries perform substantially worse than training queries. However, on anchor retrieval they perform much better than training queries.

Table 4. Baseline system retrieval results for training queries

Relevant Found in	Title Tag	Anchor Text	Abstract
Top1	38	46	30
Top5	74	71	58
Top10	85	76	66
Top20	92	79	70
Top100	109	90	97
Not in list	17	33	2
MRR	0.378	0.401	0.295

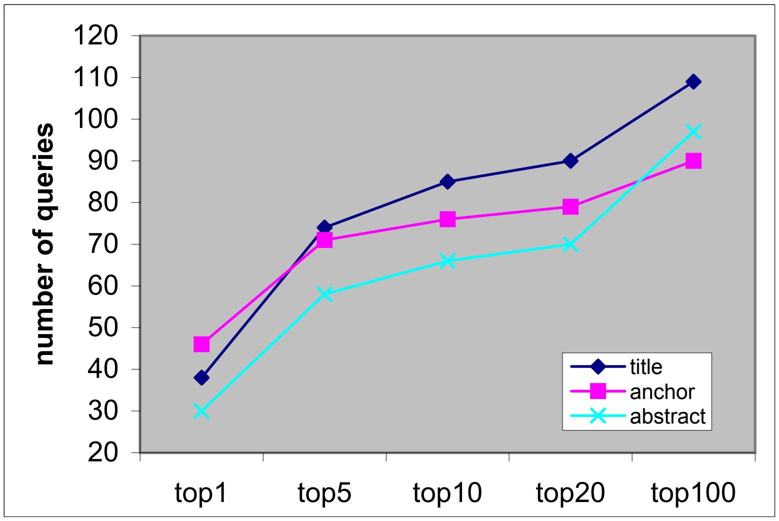

Figure 4. Baseline system retrieval results comparison chart for testing queries

Then, the decision tree model and logistic regression model were applied to the rank lists of the 145 testing queries from the baseline IR system. The results are shown in Table 5 and Figure 5.

Table 5. Machine learning models results for testing queries

Relevant Found in	Anchor + Tree	Title + Tree	Abstract + Tree	Logistic Regression
Top1	68	77	61	96
Top5	81	97	84	118
Top10	84	102	96	121
Top20	88	108	105	128
Top100	98	114	124	130
Not in list	38	27	13	15
MRR	0.511	0.595	0.501	0.727
Improvement	27.4% over Anchor	57.4% over Title	69.8% over Abstract	22.2% over Title + Tree

From Table 5 and Figure 5 we find that the overall performance of the testing queries is much worse than the training queries. This is mainly because 11 testing queries' corresponding correct homepages do not confirm the decision tree model. Thus the correct homepage was filtered out of the rank list by the decision tree step. This greatly affects the final performance.

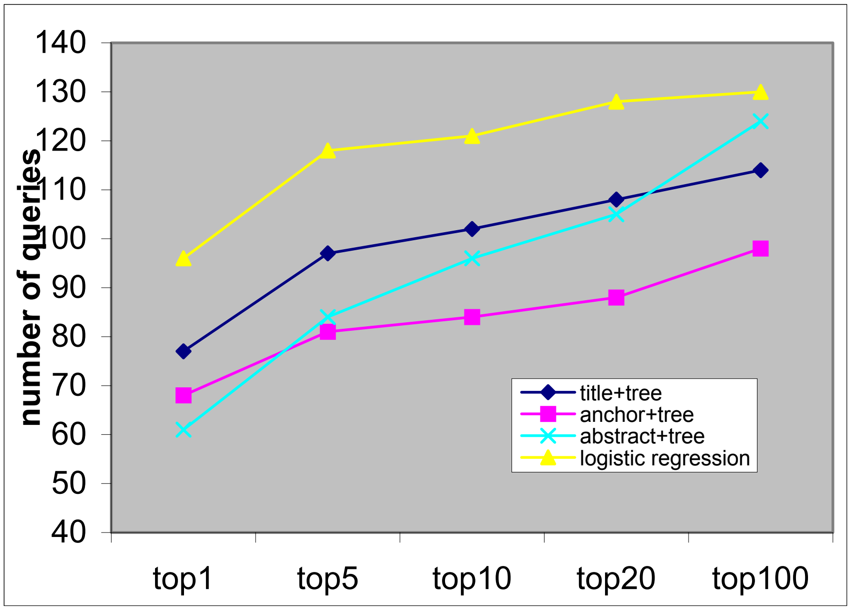

Figure 5. Machine Learning models retrieval results comparison chart for testing queries

After a close examination of these 11 queries, we find that in 3 cases, an argument could be made regarding what should be classified as homepages. For example for query No. 14 “Wah Yew Hotel”, the correct answer provided by TREC is

http://www.fastnet.com.au:80/hotels/zone4/my/my00198.htm

Another example: query No.16 “Hotel Grand, Thailand”, has correct answer:

http://www.fastnet.com.au:80/hotels/zone4/th/th00635.htm

When we go to the above locations we find each is only an introductory page to Wah Yew Hotel and Hotel Grand, Thailand, in an online hotel index website. It had no links to any other information about these hotels at all. Although this might be the only information about the two hotels on the internet, this may not guarantee itself to be the homepage of these hotels. Actually, common sense would suggest these two pages are not homepages at all.

One more example: query No. 134 “Kaye Bassman International” has correct answer provided by TREC:

http://www.kbic.com:80/toc.htm

However, when you look at the actual page, you will find this is only a table of contents. The homepage of Kaye Bassman International is clearly

http://www.kbic.com:80/index.htm, pointed to by the hyperlink at the table of contents page. These queries lead us to 2 basic questions: What is the definition of a homepage? Can a table of contents also be regarded as a homepage? However, these questions are not easily answered without further research on user behavior on the internet.

8. CONCLUSION AND FUTURE WORK

The conclusions from this research work are:

1. <Title> tagged field retrieval, Anchor text retrieval, and Abstract retrieval all perform substantially better than the full-text retrieval in the context of the homepage finding task. <Title> tagged field text retrieval performs best among these.

2. The decision tree model is an effective machine learning method to filter out homepages. This method can improve the retrieval performance by an average of 50% in terms of MRR.

3. Logistic regression analysis is another effective machine learning approach to combine multiple sources of evidences to improve the retrieval result. Our research results show this method can improve retrieval performance by 13% on training queries and 22% on testing queries.

4. By applying machine learning technologies to our system, our final testing results show 66% of the queries find the correct homepage on top of the return list and 84% of the queries find the correct homepage within the top 10 of the return list.

Future research may include:

1. Further looking into the homepages, finding more attributes that might indicate a homepage. For example, some homepages contain words such as: “welcome”, “homepage”, “website”, “home”, “page”, in the initial few lines of the text. Incorporating these new factors might help indicate whether a page is a homepage or not.

2. Making use of relevance feedback. The relevance feedback technique is found to be very successful at improving precision for very short queries. Since they are short, homepage finding queries might benefit from this approach.

3. Using a probabilistic rather than a binary decision tree, so likelihood of being a homepage becomes a factor in the logistic regression.

4. Experimenting with large collections to give more thorough and realistic testing of the methods, such as with the 1 terabyte crawling of text recently completed in collaboration with University of Waterloo.

ACKNOWLEDGMENTS

This research work was funded in part by America OnLine, Inc. Thanks go to Dr. A. Chowdhury from AOL for his kind help. Thanks also go to Fernando Das Neves, Hussein Suleman, and other colleagues in the Digital Library Research Laboratory at Virginia Tech for their very useful and important suggestions.

REFERENCES

[1] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Proceeding of the 7th International WWW Conference*, pp.107-117. 1998. http://www7.scu.edu.au/programme/fullpapers/1921/com1921.htm

[2] A. Chen. "A comparison of regression, neural net, and pattern recognition approaches to IR," in *Proceedings of the 1998 ACM 7th International Conference on Information and Knowledge Management (CIKM '98)* (pp.140-147). New York: ACM, 1998.

[3] N. Craswell; D. Hawking and S. Robertson. Effective Site Finding using Link Anchor Information. *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* pp.250-257. 2001.

[4] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms,* pp.668-677. 1998. http://**www.cs.cornell.edu/home/kleinber/auth.ps**

[5] M. F. Porter. An algorithm for suffix stripping. Program 14, 130-137. 1980.

[6] T. Sakai and K. Sparck-Jones. Generic Summaries for Indexing in Information Retrieval. *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* pp.190-198. 2001.

[7] J.A. Shaw & E.A. Fox, "Combination of multiple searches", in *Proceedings of the 3rd Text Retrieval Conference (TREC-3)* (p.105). Gaithersburg, MD: National Institute of Standards and Technology, 1995.

[8] G. Salton. and M. J. McGill. *Introduction to Modern Information Retrieval*. New York: McGraw Hill. 1983.

[9] C.C. Vogt & G.W. Cottrell, “Predicting the performance of linearly combined IR systems” in *Proceedings of the 21st annual international ACM SIGIR conference on Research and Development in information retrieval* (pp. 190 –196). New York: ACM, 1998.

[10] E. Voorhees and D.K. Harman. Overview of the Ninth Text Retrieval Conference (TREC-9). In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*, 1-28. NIST Special Publication pp.1-14. 2000.