Search Results Optimization & Summarization – Case study on Google

Shanmugasundaram Hariharan

Lecturer, Department of Information Technology, B.S.Abdur Rahman Crescent Engineering College, Chennai, Tamilnadu, India, shari1981@rediffmail.com

Abstract

Evolution of Internet has brought enormous changes in human life. With millions of information pouring online, the user has no time to surf the contents completely. As a result he has to skim most of the information. Moreover the information available is repeated or duplicated in nature. The above issues have created the necessity to build an effective search engine that could yield results in an optimized way; which can be even summarized further. This paper focus on the case study performed using the results retrieved by Google search engine. A framework for summarizing the optimized results is also presented. It is also shown the proposed system improves the efficiency of the system significantly compared to the existing system.

Keywords: *Web mining, clustering, summarization, link analysis, WWW.*

1. Introduction

Internet is the era. Today, Google accounts for more than 85 percent of all Internet searches on a daily basis[a]. Google now has many versions running in many different countries, including China, Japan, the U.K., Hong-Kong and many others. Commercial search engines are those, which retrieve pages based on the user request. Number of popular search engines exists like Google[b], AltaVista[c] and others. Search engines are those that crawl the web and gives the results in some indexed order based on some criteria. Finally results are displayed to the end user through the browser. Web crawler also known as a web spider or web robot assists these search engines in retreiving the results form the database. Other less frequently used names for web crawlers are ants, automatic indexers, bots, and worms[1].

A web crawler is one type of Internet bot or software agent. In general, it starts with a list of URLs to visit, called the seeds. As the crawler visits these URLs, it identifies all the hyperlinks in the page and adds them to the list of URLs to visit, called the crawl frontier. Some crawlers intend to download as many resources as possible from a particular Web site. Different types of crawling strategies available include path-ascending crawling [2] and focused crawling [3][4][5].

Google interprets a link from page A to page B as a vote, by page A for page B. But, Google looks at more than the sheer volume of votes, or links a page receives. It also analyzes the page that casts the vote. Votes cast by pages that are themselves important or are favorably viewed as "established firms" in the Web community weigh more heavily and help to make other pages look established too [12].

To keep the index current, Google continuously recrawls popular frequently changing web pages at a rate roughly proportional to how often the pages change. Such crawls keep an index current and are known as fresh crawls. Newspaper pages are downloaded daily, pages with stock quotes are downloaded much more frequently. Of course, fresh crawls return fewer pages than the deep crawl. The combination of the two types of crawls allows Google to both make efficient use of its resources and keep its index reasonably current.

[a] http://www.rankforsales.com/google-page-rank.htm
[b] www.google.com
[c] www.altavista.com

The importance of a page for a crawler can also be expressed as a function of the similarity of a page to a given query. Web crawlers that attempt to download pages that are similar to each other are called focused crawler or topical crawlers. The main problem in focused crawling is that in the context of a web crawler, we would like to be able to predict the similarity of the text of a given page to the query before actually downloading the page..

Good news about the Internet and its most visible component, the World Wide Web, is that there are hundreds of millions of pages available, waiting to present information on an amazing variety of topics. The bad news about the Internet is that there are hundreds of millions of pages available, most of them titled according to the whim of their author, almost all of them sitting on servers with cryptic names [12].

Internet search engines are special sites on the Web that are designed to help people find information stored on other sites. The difference by the way various search engines works and retreives information are:

- Search and selecting based on important words.
- Keeping an index of the words they find and where they are found.
- Allowing users to look for words or combinations found.

Early search engines held an index of a few hundred thousand pages and documents, and received maybe one or two thousand inquiries each day. Today, a top search engine will index hundreds of millions of pages, and respond to tens of millions of queries per day. Normally these in formations are stored in the database based on results, reranked, finally displayed in any of the following criteria's like keyword occurrence in the content, frequency of the URL visited, order of the links displayed by search engines etc. Keyword occurrence plays a vital role in the retrieval process. Hence the problem of duplicating the query terms inside the documents has become a major hurdle again. This has led to the unwanted information climbing up the hierarchy.

We have discussed works done earlier pertaining to search result optimization in section 2, while section 3 and 4 discuss about system description and experimental results & analysis correspondingly.

2. Previous work

Dragomir et al. [6] in their work have presented a open domain multi-document summarization in the context of web search. As a first step the authors have designed a personalized web search. During this step the downloaded URL is parsed, stemmed using Porter Stemming algorithm, stop words are removed and finally the frequency and position information are noted and indexed in a database. Secondly, the documents are clustered using CIDR -Columbia Intelligent Document Relater (Radev 1999). The last step is the selection of sentences based on the sentence score obtained by combining centroid score, position score and sentence overlap score.

Monica [7] in her paper has performed some hyperlink analysis over the web to retrieve the documents related to user query. She performed the analysis based on two assumptions. First if a hyperlink from page A to page B is a recommendation of page B by the author of page A. Second assumption is that if page A and page B are connected by a hyperlink, they might be on the same topic. At last all the documents are arranged based on the best answers retrieved at the top using connectivity based ranking.

Thomas [8] in his work has developed a quality based web search engine based on human judgments. He analyzed the features for characterization of the web using machine-learning approaches. The author has developed a meta search service called AQUAINT where all result pages are evaluated according to their quality and re-ranked accordingly.

Behanak et al. [9] proposed new method based on co-citation and network analysis.A set of 21 measures based on these methods were examined. The author has tried to see how well search engine rankings can be improved using a combination of co- citation and network analysis under ideal conditions. It is shown that results were significantly improved on the Google selection of the top 20 hits (for the specific sample of queries and human judges used in his study).

Yitong Wang et al. [10] have propose a new approach to cluster search results returned from Web search engine using link analysis. our approach is base on common links shared by pages using co-citation and coupling analysis. They also extended the standard clustering algorithm K-means to make it more natural to handle noises and have applied it to web search results. The experiment results show that clustering on web search results via link analysis is potentially beneficial by filtering some irrelevant pages.

Delort et al. [11] in their work addresses the issue of web document summarization. The authors have considered the context of a web document by the textual content of all the documents linking to it. To summarize a target two new summarization by context algorithms. The first one uses both the content and the context of the document and the second one is based only on the elements of the context.

3. System description

The system designed for optimizing the search results and summarizing the contents is discussed in section 3.1 and section 3.2. The architectural system is given in Figure 1. Initially the search query is given through the search engine (for our process it is Google), retrieved results are obtained in html form. The steps involved in our process is modularized as Link miner, Optimizer and summarizer. Each of these modules is discussed in detail shortly.

3.1 Link miner

The first step in our design is to mine the links structure. The steps carried out in this phase are:

a. Parsing the html file.
b. Extracting the desired links.
c. Analysis of links structure.
d. Redundancy removal through link optimization.
e. URL extraction after eliminating unwanted links.
f. Downloading the contents to document database.

We eliminate duplicates under the following categories.

Case 1: Analyzing the URL for matching anchor text

Case 2: Analyzing the root of the URL for matching anchor text.

3.2 Optimizer

Links that are identifies as useful after link mining are dumped into the documents database. The contents are compared to find the commonality exiting between the documents. We have estimated the commonality between the documents by comparing the words occurring in corresponding sets of documents.

3.3 Summarizer

Once the contents have been optimized, they can be summarized into clusters so that the end user will get an optimized summary. We have not focused in depth on summary generation, which is left for future extensions. But a framework has been provided to summarize the contents. The sequence of steps to be adopted is:

a. Converted of html pages to text format using html to text converter (if needed).
b. Extraction of anchor text for assigning special weights[*].
c. Sentence scoring for the sentences in each document.
d. Eliminating redundancy by subsumption[**].
e. Summarization depending the compression ratio.

[*] Special weights to anchor text terms and sentence scoring are assigned only to terms after we remove stop words.

[**] Subsumption method of eliminating duplicates is done by setting a threshold level, so that documents that achieves this value are considered duplicates

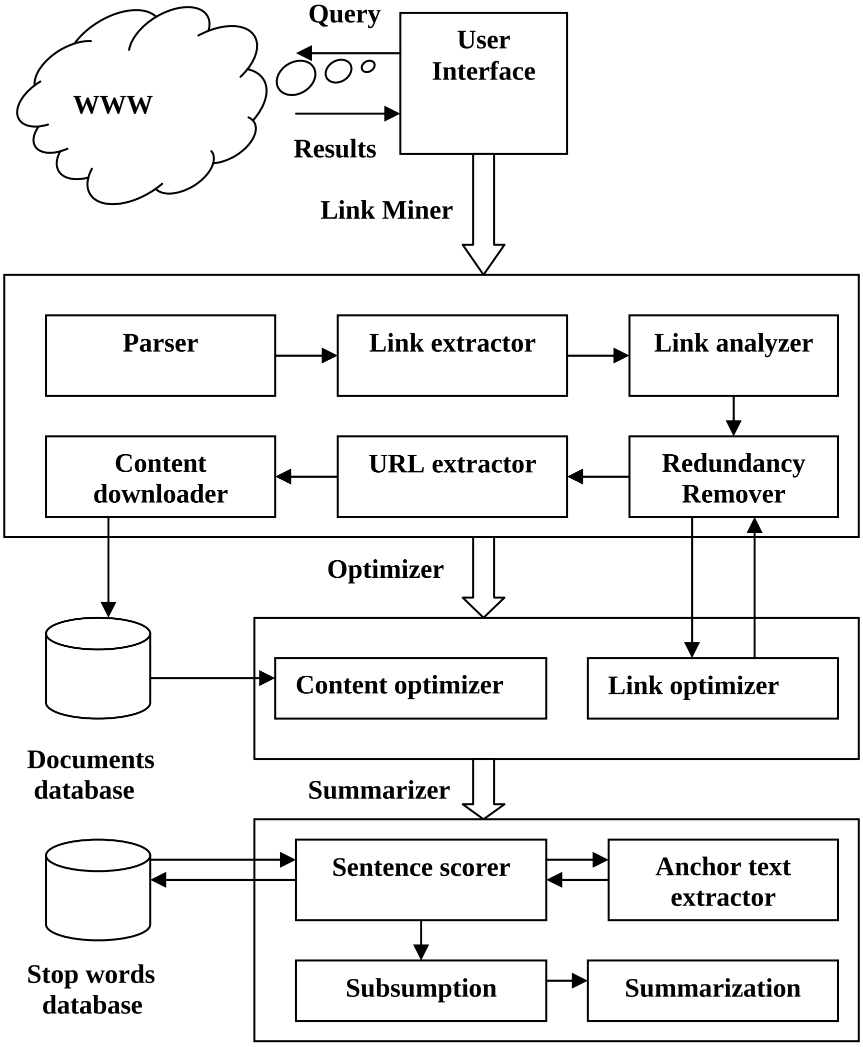

Fig 1: Architecture of our system

4. Experimental results & analyis

The experiments were carried out for 5 test cases, where we took 10 links form the search result retrieved by Google. Figure 2 shows the relevance of each link with the other links. Table 1 shows the cummulative relevancy of each link with the other links existing in the retrieved result. Using Table 1 we draw Fig 3, which shows that there is variations in the relevancy between the contents when we move down the order. This is due to the problem of keyword spamming technique. Hence unwanted documents climb up the order. Fig 4 shows the re-ranked result by our method.

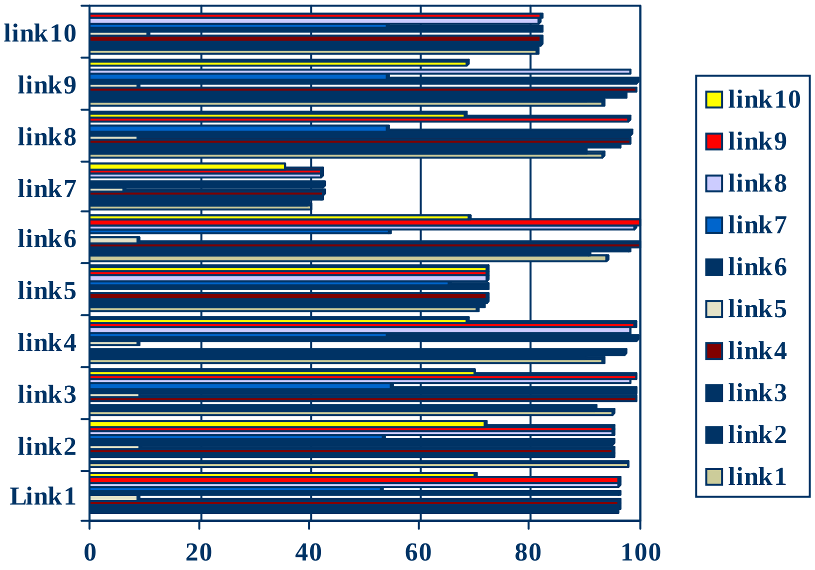

Fig 2: Relevance of each link with the other links

We calculate the efficiency of the system using the formulas given in Eq (1) and (2). Figure 5 shows the efficiency of our system compared to the existing system.

$$Original\ Efficiency = \frac{Total\ relevance - Duplicate\ relevance}{90} \rightarrow \quad (1)$$

$$\text{Im}\ proved\ efficiency = \frac{Total\ relevance - Duplicate\ relevance + average relevance}{90} \rightarrow \quad (2)$$

Link Number	Case 1	Case 2	Case 3	Case 4	Case 5
	Relevance Score	Relevance Score	Relevance Score	Relevance Score	Relevance Score
1	710	374	448	513	585
2	708	359	455	214	584
3	716	327	352	35	180
4	708	371	448	364	534
5	640	370	339	437	394
6	713	47	372	448	537
7	335	358	334	247	381
8	706	186	263	420	346
9	709	161	174	423	212
10	636	45	273	210	38

Table 1: Relevancy score between the links in the search results

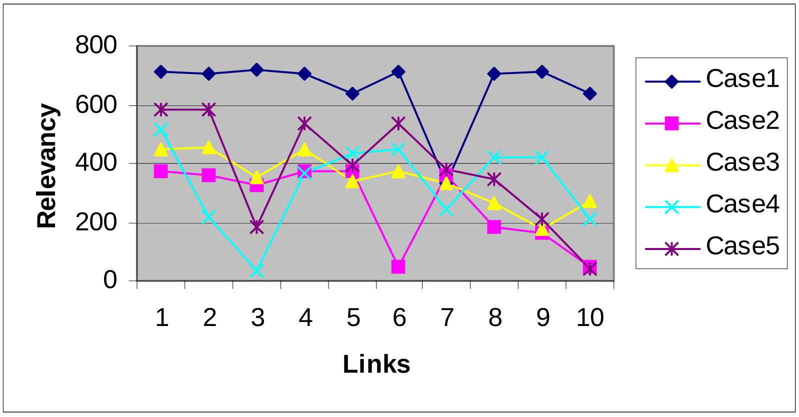

Fig 3: Existing system

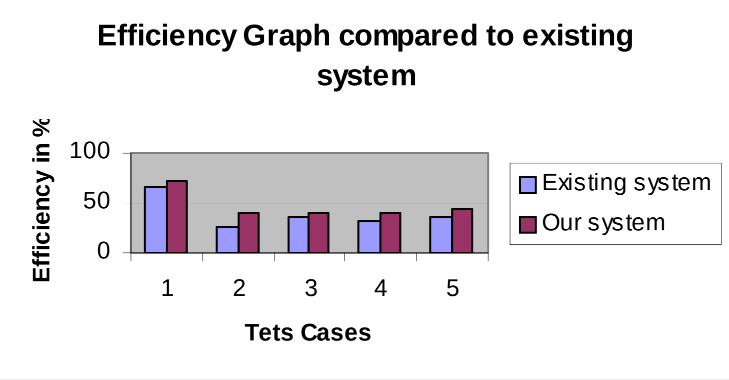

Fig 4: Our system

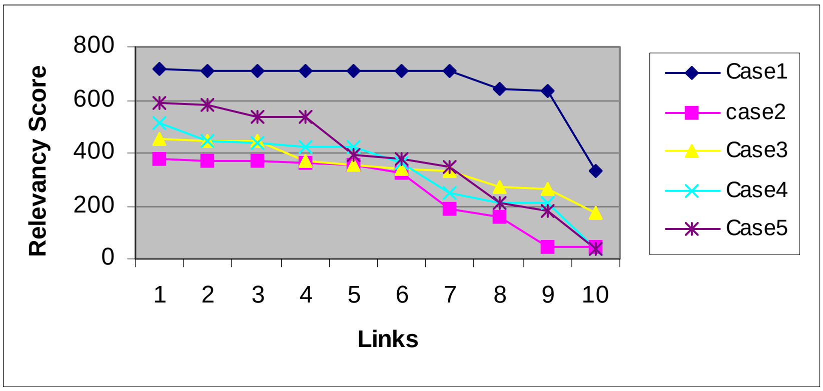

Fig 5: Efficiency of our system compared to existing system

5. Conclusion & future work

A case study has been carried out to improve the efficiency of the search results has been presented. We also provided a method to summarize the contents effectively, allowing the web surfer to skim through the unwanted articles or information and to get an organized content for reading. We have not focused on tuning the query to get efficient results, clustering the documents which would even categorize the results into a organized cluster so as to form a comprehensive or query focused summary.

Acknowledgements

The author would like to express his thanks to Mr. Abdul Qadir A. Rahman Buhari, the Correspondent, Dr. V.M.Periasamy, the Principal, Dr. T.R. Rangasamy, Dean & HOD, Department of Information Technology, B.S.A. Crescent Engineering College Chennai, for the environment provided.

References

[1] Kobayashi, M. and Takeda, K. (2000). "Information retrieval on the web". ACM Computing Surveys: 32 (2): 144–173. ACM Press. doi:10.1145/358923.358934.

[2] Cothey, V. (2004). "Web-crawling reliability". Journal of the American Society for Information Science and Technology 55 (14). doi:10.1002/asi.20078 Web-crawling reliability.

[3] Menczer, F. (1997). ARACHNID: Adaptive Retrieval Agents Choosing Heuristic Neighborhoods for Information Discovery. In D. Fisher, ed., Machine Learning: Proceedings of the 14th International Conference (ICML97). Morgan Kaufmann.

[4] Menczer, F. and Belew, R.K. (1998). Adaptive Information Agents in Distributed Textual Environments. In K. Sycara and M. Wooldridge (eds.) Proc. 2nd Intl. Conf. on Autonomous Agents (Agents '98). ACM Press.

[5] Chakrabarti, S., van den Berg, M., and Dom, B. (1999). Focused crawling: a new approach to topic-specific web resource discovery. Computer Networks, 31(11–16):1623–1640.

[6] Radev.D.R and Weiguo Fan (2000).“Automatic summarization of search engine hit lists” , Proceedings of the ACL-2000 workshop on Recent advances in natural language processing and information retrieval: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 11,pp 99-109.

[7] Monika R. Henzinger,2001. “Hyperlink Analysis for the Web”, IEEE INTERNET COMPUTING, pp 45-50.

[8] Thomas Mandl,(2006). “Implementation and Evaluation of a Quality-Based Search Engine”,ACM-HT’06, August 22–25.

[9] Behnak Yaltaghian and Mark Chignell. (2002) “Re-ranking Search Results using Network Analysis A Case Study with Google”, Proceedings of the 2002 conference of the Centre for Advanced Studies on Collaborative research, Toronto, Ontario, Canada.

[10] Yitong Wang and Masaru Kitsuregawa. (2001) “Clustering of Web Search Results with Link Analysis , Second International Conference on Advances in Web-Age Information Management (WAIM 2001).

[11] J.Y.Delort, B. BouchonMeunier and M. Rifqi. (2003). “Enhanced Web Document Summarization Using Hyperlinks” ACM HT’03, August 26–30.

[12] The Google Page Rank Algorithm. http://www.rankforsales.com/google-page-rank.html.

Article received: **2008-12-29**