Web Search Engines: Mining Right Information

Naveen

Department of Computer Science and Engineering
D.A.V. College of Engineering & Technology
Kanina, Mohindergarh – 123027, Haryana, India

Dharmender Kumar

Department of Computer Science and Engineering
Deenbandhu Chhotu Ram University of Scc and
Tech. Murthal, Sonipat – 131029, Haryana, India

ABSTRACT
A Web Search Engine maintains and catalogs the content of Web pages in order to make them easier to find and browse. There are many Search Engines which are similar, differentiates from the other by the methods for scouring, storing, and retrieving information from the Web. Usually Search Engines search through Web pages for specified keywords, in response they return a list of containing specified keywords documents. After finding the list of specified keywords documents, list is sorted by relevance criteria which try to put at the very first positions the documents that best match the user's query. The usefulness of a search engine to most people is based on the relevance of results it retrieves from the web. This paper tries to address some issues regarding some of the major challenges faced by Search Engines, since the size of the Web is rapidly growing.

Keywords
Web Search Engine, Clustering, Crawler, Hyper Text Transfer Protocol.

1. INTRODUCTION
For hundreds of years the mankind has organized information. In order to make it more accessible to the others, the last media born to globally provide information is the Internet. With the Web, in particular, the name of the Internet has spread all over the World. The scale of the WWW (World Wide Web) is immense, consisting of billions of publicly visible web documents, distributed on millions of servers world-wide. There is no central repository of the WWW's contents and documents are often in a constant state of flux.

Due to the explosion of the number of documents published on the Web, Search engines have become the main mean of initiating the interaction with the Internet. The document on internet is usually searched by querying a Web Search Engine. Now days Web Search Engines receive million searches per day over a collection of several billion web pages indexed. So can easily explain why in such environments the efficiency, as the effectiveness, of Search have become the issue.

Search Engines search through Web pages for specified keywords. After finding the list of specified keywords documents, list is sorted by relevance criteria. Documents appearing at the top of this ordering are considered to be more relevant. The two most accepted metrics to measure ranking effectiveness are: Precision (i.e. number of relevant documents retrieved over the total number of retrieved documents) and Recall (i.e. number of relevant documents retrieved over the total number of relevant documents in the collection) [1].

When the Web was new, a single entity could (and did) list and index all of the Web pages available, and searching was just an application of the Unix egrep command over an index of 110,000 documents [2]. Today, even though the larger search engines index billions of documents, any one engine is likely to see only a fraction of the content available to users [3]. Still search engines are the improving the search by implementing new techniques such as clustering.

In this paper the Section I is about the Introduction, Section II is about the web request mechanism, Section III is about the web search system, finally section IV with the Conclusion, followed by Section V future scope.

2. THE QUANDARY
A search engine has four components; first document processor indexes new documents. Indices are a mapping between words and what documents they appear in. Most engines are spider-based, so a crawl of the web for new documents and the updating of the index is automated, second query processor inspects a user's query and translates it into something internally meaningful, third matching function uses the above internally meaningful representation to extract documents from the index and last ranking scheme positions the more-relevant documents on top, using some relevance measure.

A. Web Request Mechanism
Web can be thought of as a set of Web clients such as Web browsers, or any other software used to make a request of a Web server. In order to retrieve a particular Web resource, the client attempts to communicate over the Internet to the origin Web server as shown in fig 2.1.

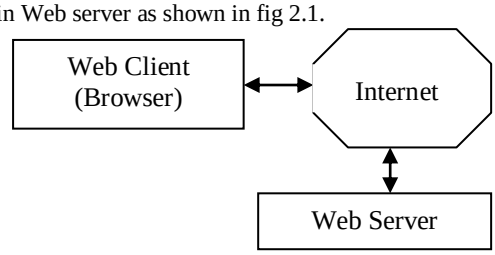

Fig 2.1: Web Request Mechanism

For example for a given URL such as http://www.dcrustm.org//, a client can retrieve the content (the home page) by establishing a connection with the

server and making a request. However, in order to connect to the server, the client needs the numerical identifier for the host. It queries the domain name system (DNS) to translate the hostname (www.dcrustm.org to its Internet Protocol (IP) address (174.36.139.100). Once the Webserver has received and examined the client's request, it can generate and transmit the response.

The Hypertext Transfer Protocol (HTTP) specifies the interaction between Web clients, servers, and intermediaries. The current version is HTTP/1.1 [4]. Requests and responses are encoded in headers that precede an optional body containing content. Figure 2.2 given below shows one set of request and response headers.

B. Http Request Header

Connect to 174.36.139.100 on port 80 ... ok

GET / HTTP/1.1**[CRLF]**

Host: www.dcrustm.org**[CRLF]**

Connection: close**[CRLF]**

User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; GTB6.6; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; MDDC; InfoPath.2)**[CRLF]**

Accept-Charset: ISO-8859-1,UTF-8;q=0.7,*;q=0.7**[CRLF]**

Cache-Control: no-cache**[CRLF]**

Accept-Language: de,en;q=0.7,en-us;q=0.3**[CRLF]**

Referrer: http://web-sniffer.net

C. HTTP Response Header

Name	Value
Date:	Thu, 24 Feb 2011 04:43:06 GMT
Server:	Apache/2.2.3 (Red Hat)
X-Powered-By:	PHP/5.1.6
Set-Cookie:	761b573241e1edac1f04c011468a1714 =lir6patmhnaveds52v42vqeg97; path=/
P3P:	CP="NOI ADM DEV PSAi COM NAV OUR OTRo STP IND DEM"
Expires:	Mon, 1 Jan 2001 00:00:00 GMT
Last-Modified:	Thu, 24 Feb 2011 04:43:07 GMT
Cache-Control:	no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma:	no-cache
Connection:	Close
Transfer-Encoding:	chunked
Content-	text/html; charset=utf-8

Name	Value
Type:	

3. WEB SEARCH SYSTEM

A web search engine typically consists of a document gatherer (usually a crawler), a document indexer, a query processor and a results presentation interface [5]. The document gatherer and document indexer need only be run when the underlying set of web documents has changed, which is likely to be continuous on the WWW, but perhaps discontinuous for other web corpora).

D. The Document Gatherer

Web-based documents are normally gathered using a crawler [6]. Crawlers traverse a web page by recursively following hyperlinks, storing each document encountered, and parsing stored documents for URLs to crawl. Crawlers typically maintain a frontier, the queue of pages which remain to be downloaded. The frontier may be a FIFO queue, or sorted by some other attribute, such as perceived authority or frequency of change [7]. Crawlers also typically maintain a list of all downloaded or detected duplicate pages so pages are not fetched more than once. The crawler frontier is initialized with a set of seed pages from which the crawl starts (these are specified manually). Crawling ceases when the frontier is empty, or some time resource limit is reached. Once crawling is complete, the downloaded documents are indexed.

E. The Indexer

The indexer distils information contained within corpus documents into a format which is amenable to quick access by the query processor. Typically this involves extracting document features by breaking-down documents into their constituent terms, extracting statistics relating to term presence within the documents and corpus, and calculating any query-independent evidence. After the index is built, the system is ready to process queries.

F. The Query Processor

The query processor serves user queries by matching and ranking documents from the index according to user input. As the query processor interacts directly with the document index created by the indexer, they are often considered in tandem. A comprehensive overview of efficient document query processing and indexing methods is provided in [8].

G. The Results Presentation Interface

The results presentation interface displays and links to the documents matched by the query processor in response to the user query. Current popular WWW and web search systems present a linear list of ranked results, sometimes with the degree of match and/or summaries and abstracts for the matching documents.

H. Clustering Documents for Search Engines

Current WSEs retrieve too many documents, of which only a small fraction are relevant to the user query, the most relevant documents do not necessarily appear at the top of the query output order. For improving the performance of WSEs, clustering techniques are now being used to give a meaningful search result on web. Clustering techniques are used for grouping similar documents together in order to facilitate presentation of results in more compact form and

gives a meaningful search result on web. The four main criteria for creating cluster categories are making the titles concise, accurate, distinctive, and "humanlike" (not something that looks like it was generated by a machine). Key Requirements for Web Document Clustering are defined in [9]. With clustering search engines gather results into groups around a certain theme, or in some cases just provide you with related keywords that perhaps you wouldn't have thought of yourself. Clustering of results is the next step up from ranking of documents Web search Engines.

4. CONCLUSION

Web search services have proliferated in the last years because here is no central repository of the WWW's contents and documents are often in a constant state of flux i.e. distributed on millions of servers world-wide. Users have to deal with different formats for inputting queries, different results presentation formats, and especially differences in the quality of retrieved information. Clustering improves the search engines a bit.

5. FUTURE SCOPE

Still Web Search engines aren't as smart as we'd like them to be. Sure, Google's and Yahoo comes in real handy sometimes, but sometimes your search terms just aren't finding what you're looking for. The performance (i.e. search and retrieval time plus communication delays) is also a problem that has to be faced while developing such a type of application which may receive thousands of requests at the same time. This can be improved by preloading a local cache with resources likely to be accessed but the difficulty then, is to determine what content to prefetch into the cache.

6. REFERENCES

[1] C.J. Van Rijsbergen. Information Retrieval. Butterworths. Available at http://www.dcs.gla.ac.uk/Keith/Preface.html

[2] Oliver A. McBryan. GENVL and WWWW: Tools for taming the Web. In Proceedings of the First International World Wide Web Conference, Geneva, Switzerland, May 1994.

[3] Steve Lawrence and C. Lee Giles. Accessibility of information on the Web Nature, 400:107-109, July 1999.

[4] Roy T. Fielding, Jim Gettys, Je_rey C. Mogul, Henrik Frystyk, L. Masinter, P. Leach, and Tim Berners-Lee. Hypertext Transfer Protocol HTTP/1.1. RFC 2616, http://ftp.isi.edu/in-notes/rfc2616.txt, June 1999.

[5] BRIN, S., AND PAGE, L. The anatomy of a large-scale hypertextual web search engine. In Proceedings of WWW7 (Brisbane, Australia, May 1998). http://www7.scu.edu.au/programme/fullpapers/1921/com1921.htm.

[6] HEYDON, A., AND NAJORK, M. Mercator: A Scalable, Extensible Web Crawler. World Wide Web Journal (December 1999), 219 – 229. http://www.research.digital.com/SRC/mercator/.

[7] CHO, J., GARC´I A-MOLINA, H., AND PAGE, L. Efficient crawling through URL ordering. Computer Networks and ISDN Systems 30, 1–7 (1998), 161–172.

[8] WITTEN, I. H., BELL, T. C., AND MOFFAT, A. Managing Gigabytes: Compressing and Indexing Documents and Images. John Wiley & Sons, Inc., 1999.

[9] Zamir, O., Etzioni, O. 1998. Web document clustering: a feasibility demonstration. Proc. of SIGIR '98, Melbourne, Appendix-Questionnaire, pp.46-54