Reading Report - Week 8 **Varad Meru**
CS 221 - Information Retrieval Student # 26648958
Prof. Cristina Lopes Due Date: 02/28/2015

The Anatomy of a Large-Scale Hypertextual Web Search Engine* - Summary

By Sergey Brin and Lawrence Page. (Stanford University/ Google).

Summary

The authors start with an historical overview of the work done by the researchers and the industry on the problem of finding relevant information by the means of queries put by the users. The mechanisms could be using carefully curated set of directories, such as the one pioneered by Yahoo! or more automated systems like WWWW or Altavista. The average number of queries handled by the engines per day was 20 million (Altavista) and the expectation was to grow till 100 million by the end of the millennium. They then introduce "Google", a search engine which leverages the link structure and the anchor text.

The discussion then heads to the system features, starting with PageRank and later describing anchor text, Proximity of words, font size and shapes of words, and special indicators. Google uses the idea of counting inlinks but with the extension of (a) not counting links from all pages equally, and by (b) normalizing by the number of links on a page, to determine the importance of the web page, which is called PageRank. Let u be a web page. Then let i be a webpage, $i \in F_u$, where F_u be the set of pages u points to and B_u be the set of pages that point to u. Let d be the damping factor (between 0 and 1). Then,

$$PR(u) = (1 - d) + d \sum_{i=1}^{N_u} \frac{PR(i)}{|F_i|} \tag{1}$$

Where $PR(u)$ is the Pagerank of u and $i \in B_u$. The Pagerank(Equation 1) is computed as a simple iterative algorithm.

The paper then describes the architecture of the Google search engine which contains the crawlers, URLserver, storeserver, indexer and sorter. The crawling is done by several distributed crawlers, which fetch the list of URLs to be fetched from the URLServer. The storage of the crawled webpages is managed by the sotreserver which store in a repository. A docID is assigned to a new URL whenever it is parsed out of a webpage. Indexing performs multiple tasks - It reads the repository, uncompresses the documents, parses them. A document is converted into a set of word occurrences called hits. The hits record the word, position in document, an approximation of font size, and capitalization, and then are distributed in "Barrels", creating a partially sorted forward index. It also does anchor text analysis to fetch enough information about the links it points to, by checking the text. The major data structures used and maintained by Google were BigFiles, Repository, Document Index, Lexicon, Hit Lists, Forward Index and Inverted index.

Google's Query Evaluation is a seven-step process - (1) Parse the query, (2) Convert words into Word IDs, (3) Go to the start of docList for every word, (4) Scan through the docList to find the first document containing the word, (5) Compute the rank of the documents for that query (6) If we don't find any docId from the short barrel, seek to the full barrel and go to (4), (7) Go to step (4) if we are at the end of the short barrel. The ranking mechanism considers various parameters and has particular settings, known as type-weights. The IR score is computed for the results generated by each parameter and then combined with PageRank to give the final score.

Commentary: Reading this paper helped me to understand the structure of a highly successful search engine with academic acumen of both the authors helping to solve the age-old problem of large scale IR.

*Brin, Sergey and Page, Lawrence. *The Anatomy of a Large-Scale Hypertextual Web Search Engine.* Computer networks and ISDN systems 30.1 (1998): 107-117.