Paper Reading Guide

The Anatomy of a Large-Scale Hypertextual Web Search Engine

by Sergey Brin and Lawrence Page

Some background on indexing for information retrieval e.g. see http://en.wikipedia.org/wiki/Search_engine_indexing, will be helpful for understanding this paper. In particular, look up the *forward index* and the *inverted index*.

Information retrieval can be considered as an application. However, scaling it to the web scale requires substantial system development. The initial part of the paper describes how properties of the web such as its scale and the presence of link structures changed how the information retrieval problem should be formulated.

1. What is the main design goal of Google?
2. Is high precision or high recall preferred by Google? Why?
3. What are the differences between the Web and typical information retrieval corpuses?

The link structure of the web is exploited by Google in two ways.

4. How is anchor text used in Google?
5. How does PageRank exploit the link structure? Does it make intuitive sense?
6. What is the 'random surfer' interpretation of PageRank?
7. What is the form of the PageRank equation? Note that it is defined in terms of PageRank of other pages, and may eventually depend on itself? Does that make sense, and under what conditions?

The last question appears difficult to answer. However, by removing the details, we can reduce the problem to that of a random walk on a graph (finite Markov chain). From the theory of finite Markov chains, we can deduce that the PageRank iterations will converge and its value can be interpreted as the stationary distribution of the Markov chain (probability of finding a random surfer on a web page). You can find a fairly readable description from the link analysis section of http://nlp.stanford.edu/IR-book/information-retrieval-book.html.

The next parts of the paper describe the system that allows Google to scale to the size of the web. Size is always an issue throughout this section.

8. What are the main components of Google's architecture?
9. Why does Google avoid disk seeks?
10. How are full HTML documents stored?
11. How are documents found given a URL?
12. What are the considerations in designing the encoding of the hit list?
13. How is the forward index encoded?

14. How is the inverted index encoded? Why is there a need for both a forward and inverted index?
15. What are the main challenges in crawling the web?

Validation is minimal (the authors states that it is beyond the scope of the paper).

16. How does the Google ranking system work?
17. How well does Google work?

What do you think of the paper?

18. What are the main contributions of the paper?
19. Can you do similar work?