LIS566

Lecture 6

Web search engines: google

Thomas Krichel

2000–11–04

Structure

1. Generalities
2. Finding relevant pages

Reading

Brin, Sergey and Page, Lawrence "The Anatomy of a Large-Scale Hypertextual Web Search Engine", availabable at http://dbpubs.stanford.edu:8090/pub/1998-8

Page, Lawrence, Sergey Brin, Rajeev Motwani and Terry Winograd, "The PageRank Citation Ranking: Bringing Order to the Web", available at http://citeseer.nj.nec.com/page98pagerank.html

These readings are not suitable for students of the class. A web site on search engines is http://www.searchenginewatch.com.

The history of search engines

1994–04 WWWW the WWW Worm was the first search engine of the web, with 110,000 web pages. 1500 queries per day

1995–11: Altavista has 20 million pages

1997–11: Altavista has 20 million queries per month

2000–08: both google and fast have about 500 Million pages. Google links to 500 Million more but they have not been indexed.

early search engine problems in 1997

As noted by Brin and Page, they include

- could not find themselves
- secret
- commercially oriented
- return pages of poor relevance

more general problem with web indexing

An initial idea is that the web is nothing else than a distributed hypertext system. However there are several qualifing points.

- distributed data with irregular bandwidth
- no overall topology
- high volatility of data
- weak structure of pages
- poor data quality
- heterogenous data
- large size

Indexing the web, a solvable problem?

In 1994, at the start of the web, people thought indexing it complete would be possible.

Today it appears that the fraction of the web that is indexed by search engines is falling.

However, with computer power improving, it may be the case that computers will eventually be able to catch up with human text production on the Internet.

Traditional IR and the web

Most search engines started with implementing the classic teaching of information retrieval.

A document is related to a query if the terms of the query appear often in the document.

Here is an elementary model.

Assume that a document has say W words. Say there is a query term i that appears n_i times in the document. Let the be q terms in the query.

Then the documnet in related to the query the more

$$\frac{n_1}{W} + \frac{n_2}{W} + \ldots + \frac{n_q}{W}$$

is high.

The "Clinton sucks" problem

Let the query be "Bill Clinton" and the page say

"Bill Clinton sucks" and have a picture.

Page highly related, but poor quality.

Page quality

If we index the whole of the web, and we search for terms on that huge collection, we need a way to select high-quality pages, and display those first.

How can that be done?

What is an important page

A page that many pages point to is important.

A page that is pointed to by other important pages is important.

Model of user behavior

A random surfer is given a page.

S/he keeps clicking on any link that she finds in that page with a probability d.

S/he gets bored and starts with a completely new random page with probability $1 - d$.

Does that surfer hit all pages on the web with the same probability?

Page Rank.

Let there be a page A. Let $l(A)$ be the number of links that go out of A. Let there be a number of pages P_1,...,P_n that have links to A. Then the of A, note $r(A)$ is given by

$$r(A) = \epsilon_P (1-d) + d\left(\frac{r(P_1)}{l(P_1)} + \frac{r(P_2)}{l(P_2)} + \ldots + \frac{r(P_n)}{l(P_n)}\right)$$

where $1-d<1$ is the probabilty that the user gets bored, and $\epsilon_A > 0$ is the probability, that if the user is bored, (s)he picks page A. Of course $\sum_{\forall P} \epsilon_P = 1$.

Interpret that formula.

Example

Imagine the web is composed of four pages A, B, C, D.

A links to B, C, D.

B links to A.

C links to B and D.

D links to B.

Calculate the page rank for each page, assuming that $1-d = 1/4$ and $\epsilon = 1/4$ for all pages.

Other features

- proximity is search terms
- uses visual representation data i.e. bigger print is more important than small print
- usage of link anchor data