# The Anatomy of Adolescent Google

April 3, 2006

Based on [BrinPage98 section 1, 2, and 4.2]

Rasmus Pagh
Advanced Database Technology, Spring 2006
IT University of Copenhagen

# Google – a web search engine

Google is one of the most successful search engines on the web. The main ideas behind Google (as of 1998):

1. Use the structure of hypertext to produce better search results.
2. Use efficient crawling to enable frequent updates.
3. Use efficient indexing to produce results quickly, while not using too much space.

The first of these ideas is probably the key to Google's success. This is one example of *data mining*.

This lecture will say a little about 1 and 3. You may read [BrinPage98] for more information. Beware that the current Google may use more sophisticated techniques.

# Data mining hypertext

Simple observation: The text in a hyperlink often describes the content of the target page. It may thus be used to index the target page.

**Example:** Search for "Evil Empire" used to give `www.microsoft.com` as the top hit, though it is not used on the page itself (!).

Another observation: A hyperlink is a kind of recommendation. However, you probably trust hyperlinks from some places more than hyperlinks from other places.

**Idea:** Try to assess the credibility of web pages in a consistent way such that the "combined credibility" of pages linking to page $p$ is proportional to the credibility of $p$.

- This has a mathematical formulation that we will not discuss.
- Putting the most credible pages first in a result gives **PageRank**.

# Google data structures

Google uses a number of data structures to answer queries, most notably:

- A repository containing the (compressed) HTML for every web page.
- A lexicon containing all words occurring on the web pages. In 1998 the 14 million different words could fit in the memory of a large PC.
- An inverted index that stores, for each word in the lexicon, the "document ID" of the pages containing this word.

Searching for a single word amounts to a lookup in the inverted index (suppose that the highest ranked pages appear first).

# Searching for several words

When searching for pages containing several words, one considers the list of pages for each word.

- Suppose that the lists are sorted according to rank (and in case of equal rank according to document ID).
- Finding pages with all (or some) words corresponds to merging all (or some) lists.
- Note that the lists may be long, but it (usually) suffices to find the first 10 results.
- Merging lists of very different sizes can be done more efficiently than usual symmetric merging (How?).