DATA MINING - 1DL105, 1DL111

Fall 2007

An introductory class in data mining

http://user.it.uu.se/~udbl/dut-ht2007/
alt. http://www.it.uu.se/edu/course/homepage/infoutv/ht07

Kjell Orsborn
Uppsala Database Laboratory
Department of Information Technology, Uppsala University,
Uppsala, Sweden

Introduction to Data Mining: Search Engines

Technology and Algorithms for Efficient Searching of Information on the Web

(slides + supplemental articles)

Kjell Orsborn

Department of Information Technology
Uppsala University, Uppsala, Sweden

Lecture's Outline

- The Web and its Search Engines
- Heuristics-based Ranking
- Page rank (Google)
  - for discovering the most "important" web pages
- HITS: hubs and authorities (Clever project)
  - more detailed evaluation of pages' importance

The Web in 2001: Some Facts

- More than 3 billion pages; several terabytes
- Highly dynamic
  - More than 1 million new pages every day!
  - Over 600 GB of pages change per month
  - Average page changes in a few weeks
- Largest crawlers
  - Refresh less than 18% in a few weeks
  - Cover less than 50% ever (invisible Web)
- Average page has 7–10 links
  - Links form content-based communities

Chaos on the Web

- Internet lacks organization and structure:
  - pages written in any language, dialect or style;
  - different cultures, interests and motivation;
  - mixes truth, falsehood, wisdom, propaganda…
- Challenge:
  - Quickly extract from this digital morass, high-quality, relevant, up-to-date pages in response to specific information needs
  - No precise mathematical measure of “best” results

Search Products and Services (in 2000)

- Verity
- Fulcrum
- PLS
- Oracle text extender
- DB2 text extender
- Infoseek Intranet
- SMART (academic)
- Glimpse (academic)

- *Inktomi (HotBot)*
- Alta Vista
- Raging Search
- Google
- Dmoz.org
- Yahoo!
- *Infoseek Internet*
- *Lycos*
- *Excite*

* *heuristics-based*
* humanly-selected

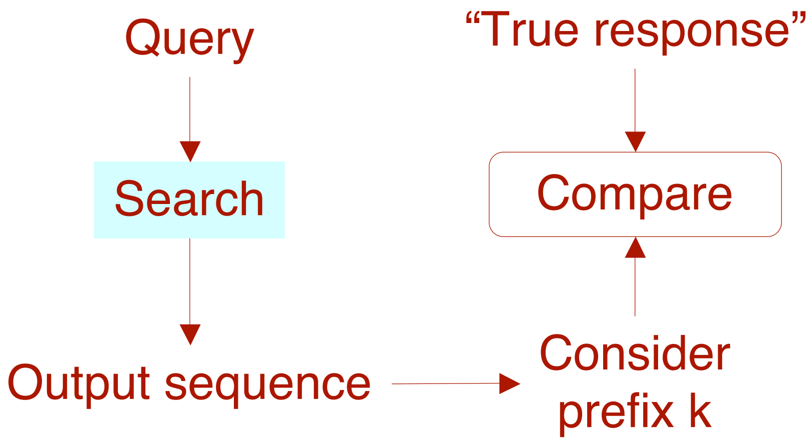
UPPSALA UNIVERSITET

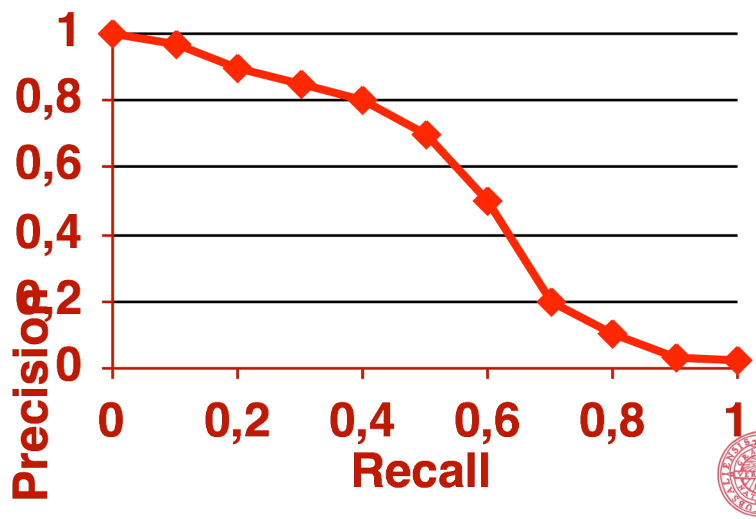
Web Search Queries

- Web search queries are short:
  - ~2.4 words on average (Aug. 2000)
  - Has increased, was 1.7 (~1997)
- User expectations:
  - “The first item shown should be what I want to see!”
  - This works if the user has the most popular or most common notion in mind; not otherwise

Relevance ranking in text retrieval

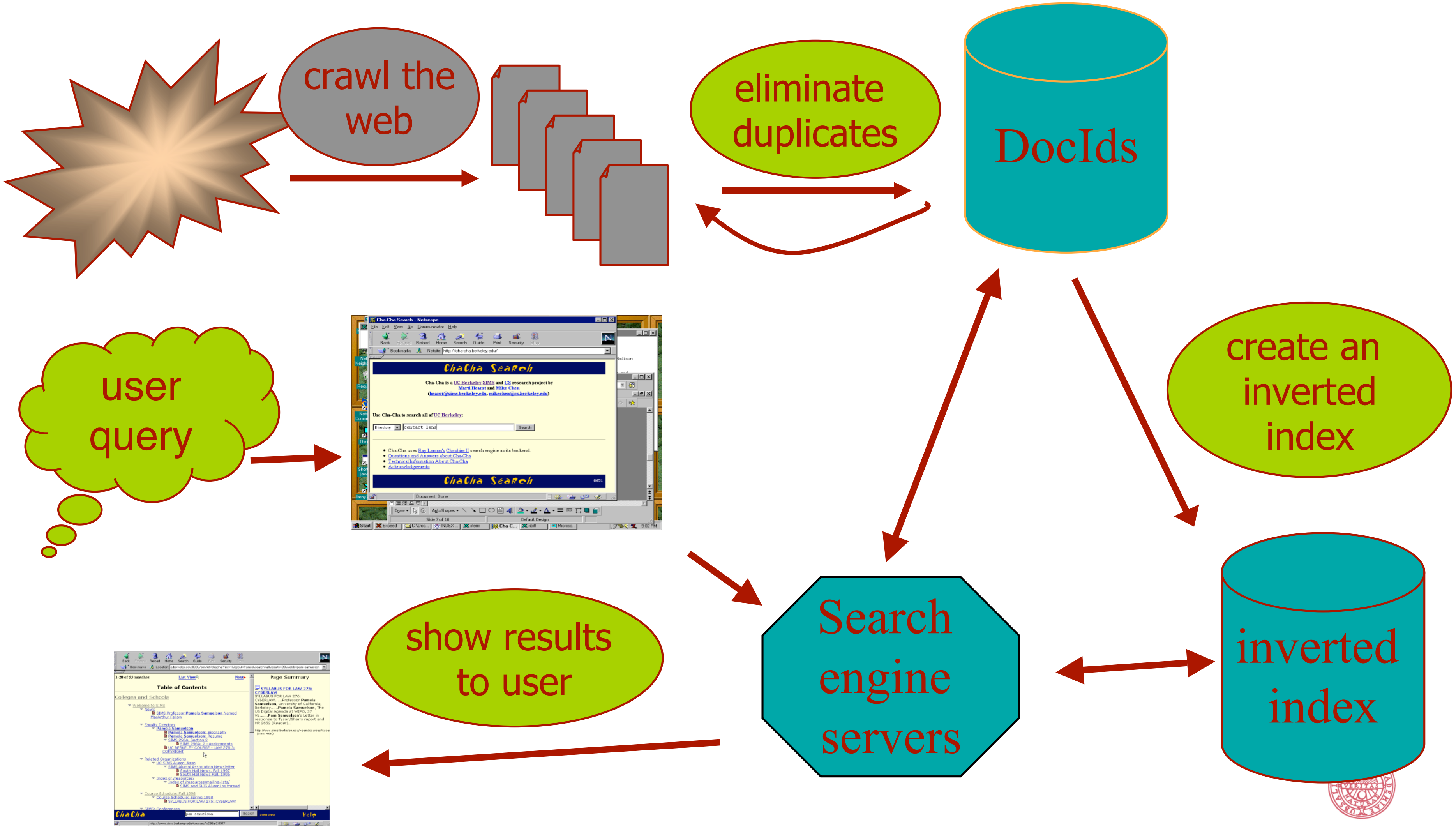
- Recall = coverage
  – What fraction of relevant documents were reported
- Precision = accuracy
  – What fraction of reported documents were relevant
- Trade-off between recall and precision
- Query generalizes to ‘topic’





UPPSALA UNIVERSITET

Standard Web Search Engine Architecture



Heuristics-based Ranking

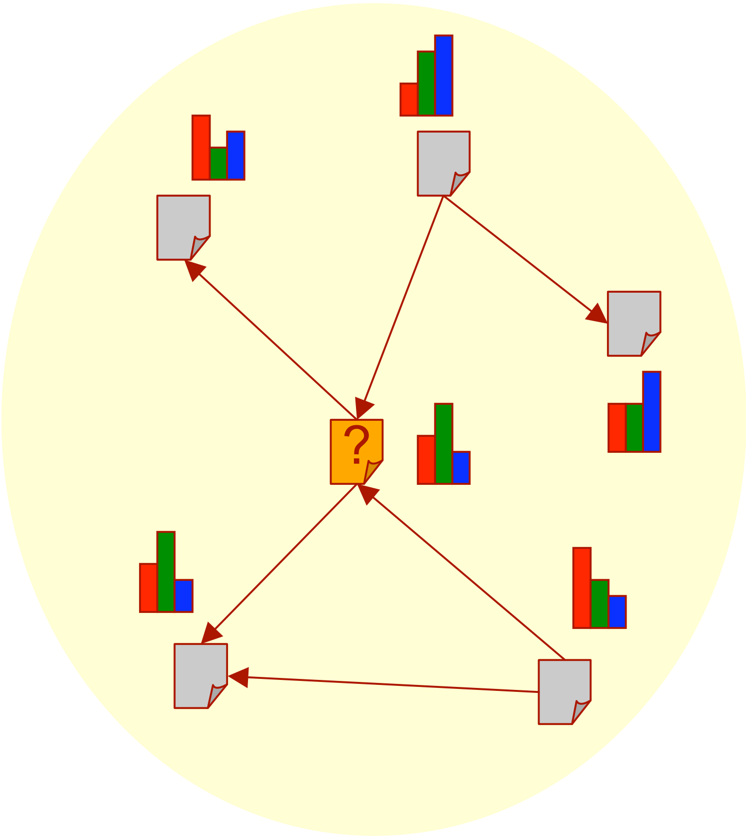
- Naive attempt used by many search engines
- Heuristics employed:
  - number of times a page contains the query term
  - favor instances where the term appears early
  - give weight to word appearing in a special place or form
    - e.g. in a title or in bold.
- All heuristics fail miserably due to:
  - spamming, or
  - synonymy and polysemy of natural language words

Hyperlink Graph Analysis

- Hypermedia is a social network
  - Telephoned, advised, co-authored, paid
- Social network theory (cf. Wasserman & Faust)
  - Extensive research applying graph notions
  - Centrality and prestige
  - Co-citation (relevance judgment)
- Applications
  - Web search: HITS, Google
  - Classification and topic distillation

Hypertext models for classification

- c = class, t = text, N = neighbors
- Text-only model: Pr[t | c]
- Using neighbors' text to judge my topic: Pr[t, t(N) | c]
- Better model: Pr[t, c(N) | c]
- Recursive relationships



Exploiting the Web's Hyperlink Structure

Underlying assumption: *view each link as an implicit endorsement of the location it points to*

- Assumption: If the pages pointing to this page are good, then this is also a good page.
  - References: Kleinberg 98, Page et al. 98
- Draws upon earlier research in sociology and bibliometrics.
  - Kleinberg's model includes "authorities" (highly referenced pages) and "hubs" (pages containing good reference lists).
  - Google model is a version with no hubs, and is closely related to work on influence weights by Pinski-Narin (1976).

Link Analysis for Ranking Pages

- Why does this work?
  - The official Ferrari site will be linked to by lots of other official (or high-quality) sites
  - The best Ferrari fan-club sites probably also have many links pointing to it
  - Less high-quality sites do not have as many high-quality sites linking to them

Page Rank

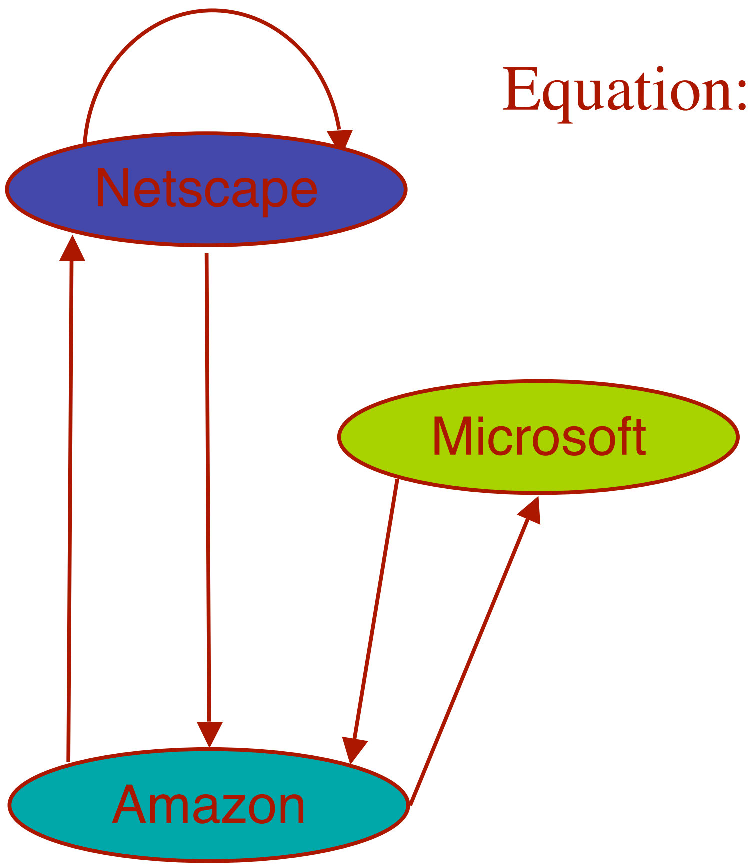
- Intuition: Recursive Definition of “importance”.
- A page is important if important pages link to it.

- Method: Create a stochastic matrix of the Web
  - each page corresponds to a matrix’s row and column
  - if page j has n successors, then the ij-th entry is
    - 1/n if page i is one of these n successors of page j
    - 0 otherwise

Page Rank intuition

- Initially, each page has one unit of importance.
- At each round, each page shares whatever importance it has with its successors, and receives new importance from its predecessors.
- Eventually, the importance reaches a limit, which happens to be its component of the principal eigenvector of this matrix.
- Importance = probability that a random Web surfer, starting from a random Web page, and following random links will be at the page in question after a long series of links.

Page Rank example: the web in 1689



Equation: $\begin{pmatrix} N' \\ M' \\ A' \end{pmatrix} = \begin{pmatrix} 1/2 & 0 & 1/2 \\ 0 & 0 & 1/2 \\ 1/2 & 1 & 0 \end{pmatrix} \begin{pmatrix} N \\ M \\ A \end{pmatrix}$

Solution by relaxation (iterative solution):

$$\begin{array}{lllllll} N & = 1 & 1 & 5/4 & 9/8 & 5/4 & \ldots\ 6/5 \\ M & = 1 & 1/2 & 3/4 & 1/2 & 11/16 & \ldots\ 3/5 \\ A & = 1 & 3/2 & 1 & 11/8 & 17/16 & \ldots\ 6/5 \end{array}$$

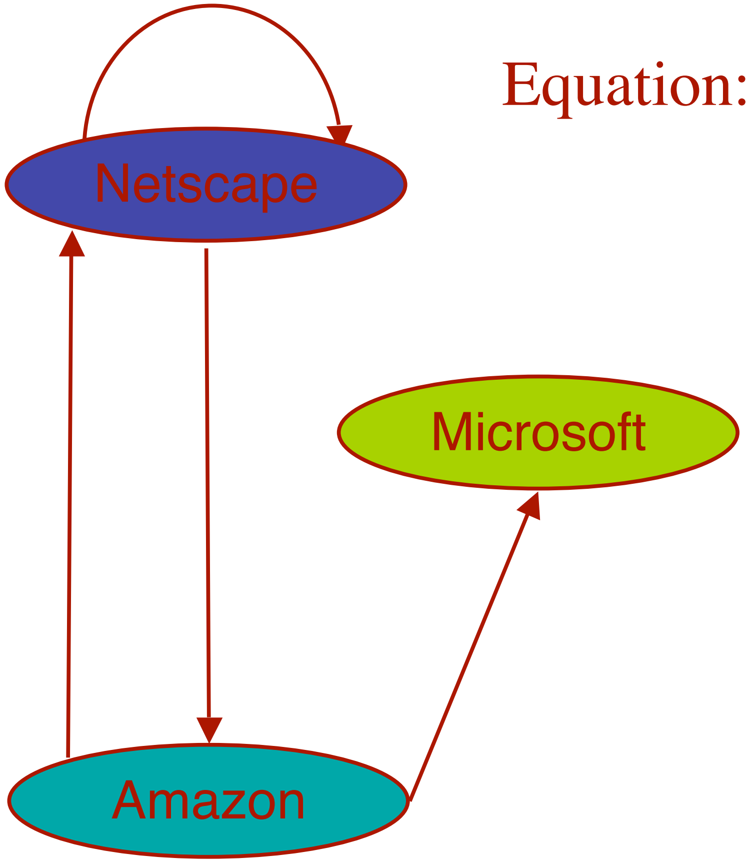
Problems with real web graphs

- Dead ends: a page that has no successors has nowhere to send its importance
  - Eventually, all importance will “leak out of” the Web

- Spider traps: a group of one or more pages that have no links outside the group
  - Eventually, these pages will accumulate all the importance of the Web.

Dead ends - rank leaks:

Microsoft tries to duck monopoly charges...



Equation: $$\begin{pmatrix} N' \\ M' \\ A' \end{pmatrix} = \begin{pmatrix} 1/2 & 0 & 1/2 \\ 0 & 0 & 1/2 \\ 1/2 & 0 & 0 \end{pmatrix} \begin{pmatrix} N \\ M \\ A \end{pmatrix}$$

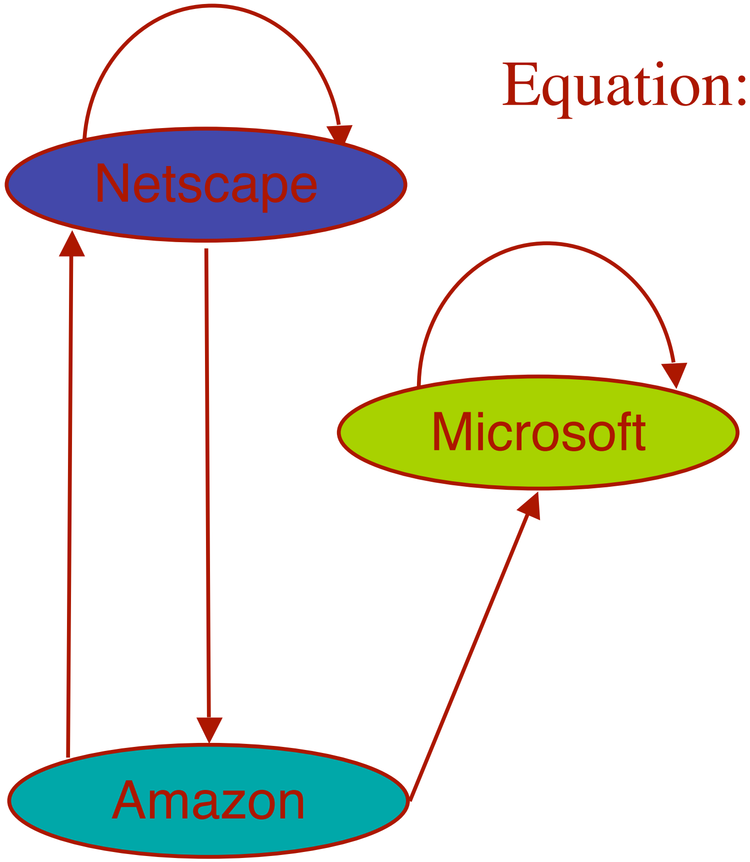
Solution by relaxation (iterative solution):

$$\begin{array}{lllllllll} N & = 1 & 1 & 3/4 & 5/8 & 1/2 & \ldots & 0 \\ M & = 1 & 1/2 & 1/4 & 1/4 & 3/16 & \ldots & 0 \\ A & = 1 & 1/2 & 1/2 & 1/2 & 5/16 & \ldots & 0 \end{array}$$

UPPSALA UNIVERSITET

Spider traps - rank sinks

Microsoft considers itself the center of the universe...



Equation:
$$\begin{pmatrix} N' \\ M' \\ A' \end{pmatrix} = \begin{pmatrix} 1/2 & 0 & 1/2 \\ 0 & 1 & 1/2 \\ 1/2 & 0 & 0 \end{pmatrix} \begin{pmatrix} N \\ M \\ A \end{pmatrix}$$

Solution by relaxation (iterative solution):

$$\begin{array}{lllllllll} N & = 1 & 1 & 3/4 & 5/8 & 1/2 & \ldots & 0 \\ M & = 1 & 3/2 & 7/4 & 2 & 35/16 & \ldots & 3 \\ A & = 1 & 1/2 & 1/2 & 3/8 & 5/16 & \ldots & 0 \end{array}$$

UPPSALA UNIVERSITET

Google solution to dead ends and spider traps

- Instead of applying the matrix directly, “tax” each page with some fraction of its current importance, and distribute the taxed importance equally among all pages.

$$\begin{bmatrix} N' \\ M' \\ A' \end{bmatrix} = 0.8 \begin{pmatrix} 1/2 & 0 & 1/2 \\ 0 & 1 & 1/2 \\ 1/2 & 0 & 0 \end{pmatrix} \begin{bmatrix} N \\ M \\ A \end{bmatrix} + \begin{bmatrix} 0.2 \\ 0.2 \\ 0.2 \end{bmatrix}$$

- The solution to this equation is now
- N = 7/11; M = 21/11; A = 5/11

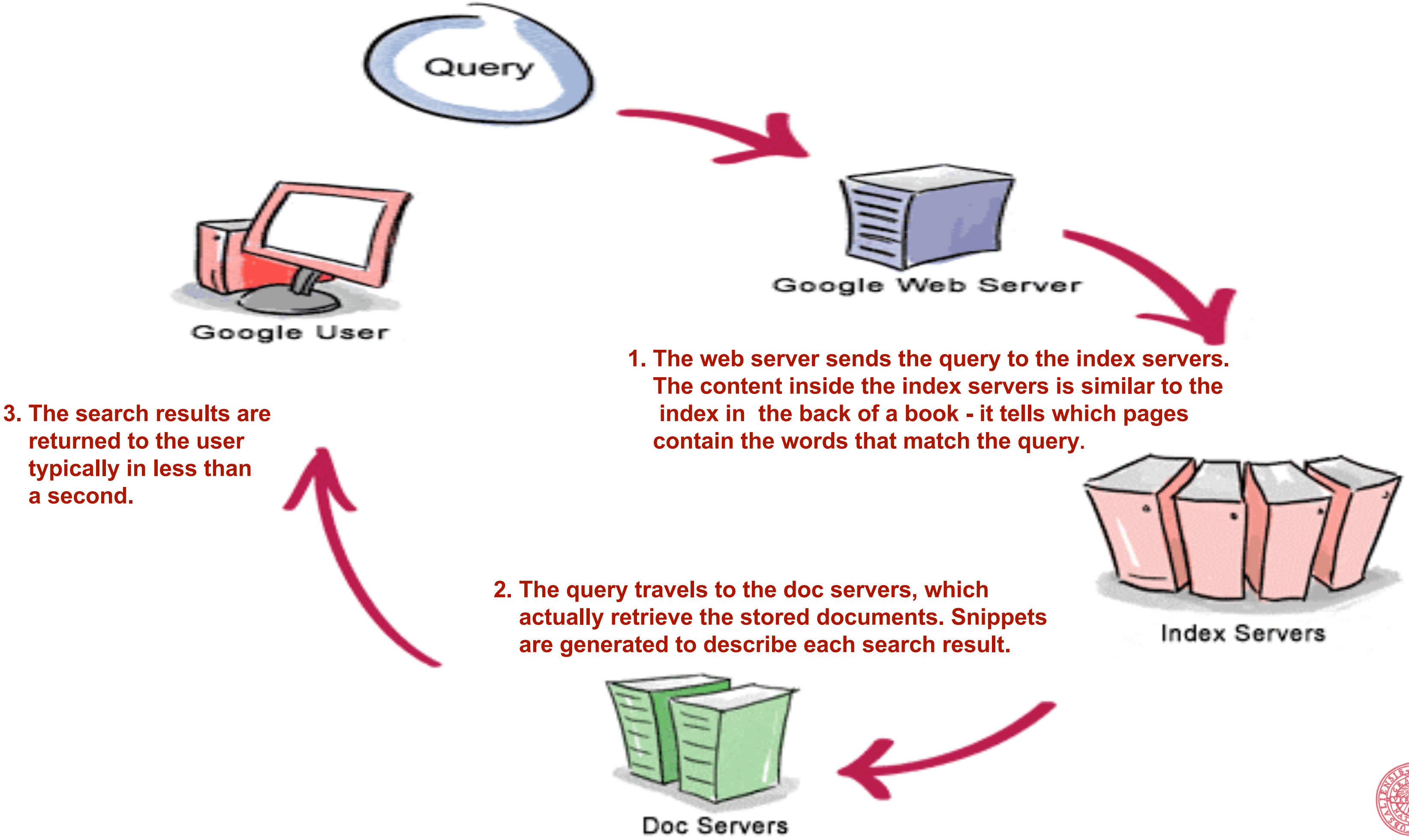
Google anti-spam devices

- *Spamming*: an attempt by many web sites to appear to be about a subject that will attract surfers, without truly being about the subject
- Early search engines relied on the words on a page to tell what it is about.
  - Led to "tricks" in which pages attracted attention by placing false words in the background color on their page.
- Google trusts the words in anchor text
  - Relies on others telling the truth about your page, rather than relying on you.
  - Makes it harder for a homepage to appear to be about something it is not.
- The use of page rank to measure importance, rather than the more naive "number of links into a page", also protects against spamming. E.g. Page Rank recognizes as unimportant 1000 pages that mutually link to one another.

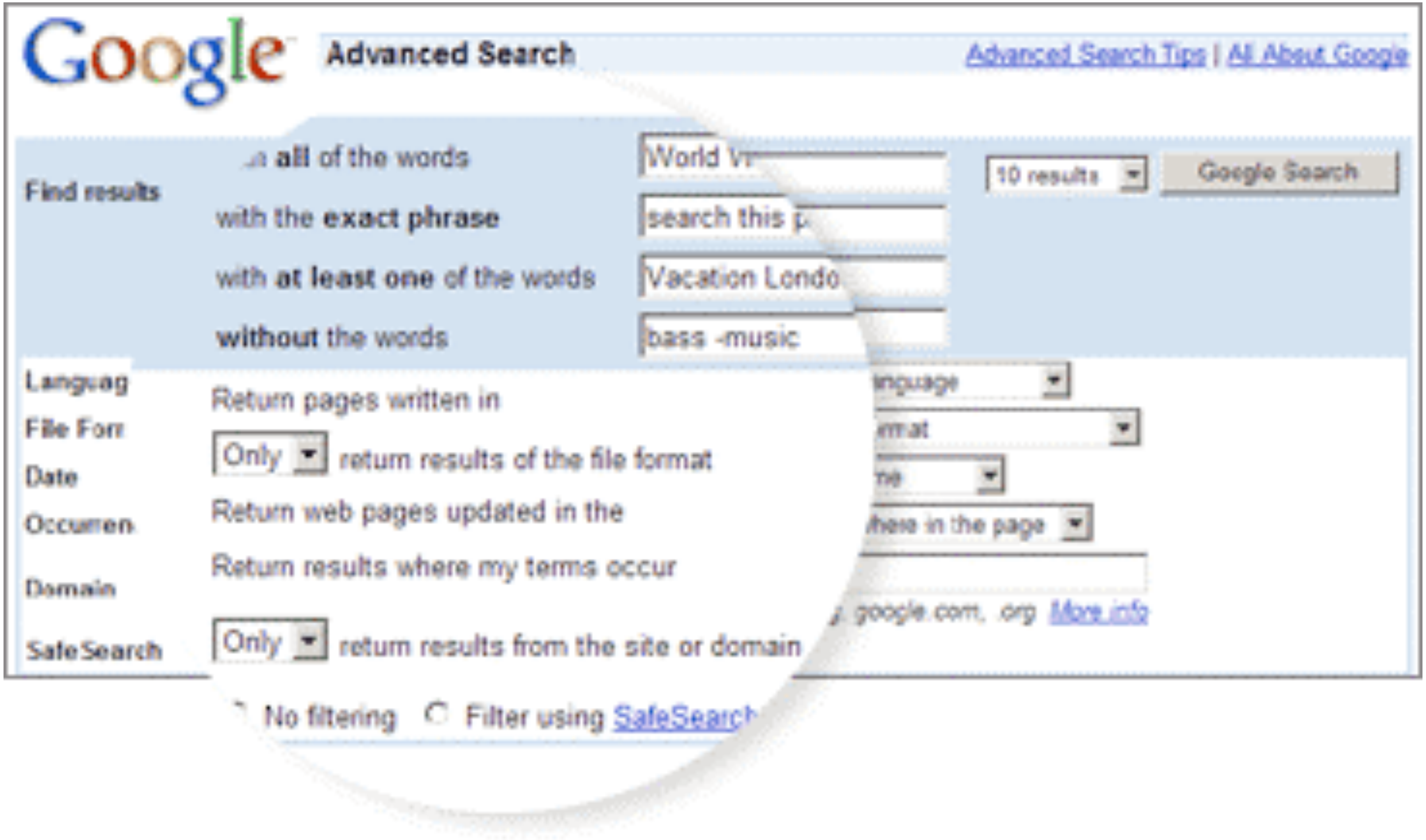
Google Facts (from end 2001)

- Indexes 3 billion web pages
  - If printed, they would result in a stack of paper 200 km high
  - If a person reads a page per minute (and does nothing else), (s)he would need 6000 years to read them all
- 200 million search queries a day
  - Approx. 80 billion searches a year!
- Most searches take less than half second
- Support for 35 non-English languages
- Searchable index contains 3 trillion items
  - Updated every 28 days

Google architecture (approx.)



Google Advanced Search



Hubs and Authorities

- HITS - hypertext-induced topic selection (IBM's Clever project)
- Defined in a mutually recursive way:
  - a hub links to many (valuable) authorities;
  - an authority is linked to by many (good) hubs.
- Authorities turn out to be pages that offer the best information about a topic.
- Hubs are pages that do not provide any information, but specify a collection of links on where to find the information.

Hubs and Authorities

- Use a matrix formulation similar to that of Page Rank, but without the stochastic restriction.
  - Rows and columns correspond to web pages;
  - A[i,j] = 1 if page i links to page j
  - A[i,j] = 0 otherwise
- The transpose of A looks like the matrix for Page Rank, but it has a 1 where the Page Rank matrix has a fraction
- An iterative solution where the matrix A is repeatedly applied will result in a diverging solution vector.
- However, by introducing scaling factors the computed values of "authority" and "hubbiness" for each page can be kept within finite bounds.

Computing Hubbiness and Authority of pages

- Let a and h be vectors
  – The i-th component of a and h correspond to the degrees of authority and hubbiness of the i-th page respectively.
- Let λ and μ be suitable scaling factors.
- Then:
  – the hubbiness of each page is the sum of authorities it links to, scaled by λ

$$\underline{h} = \lambda A \underline{a}$$

  – the authority of each page is the sum of the hubbiness of all pages that link to it, scaled by μ

$$\underline{a} = \mu A^T \underline{h}$$

Computing Hubbiness and Authority of pages

- By simple substitution, two equations that relate vectors a and h only to themselves:

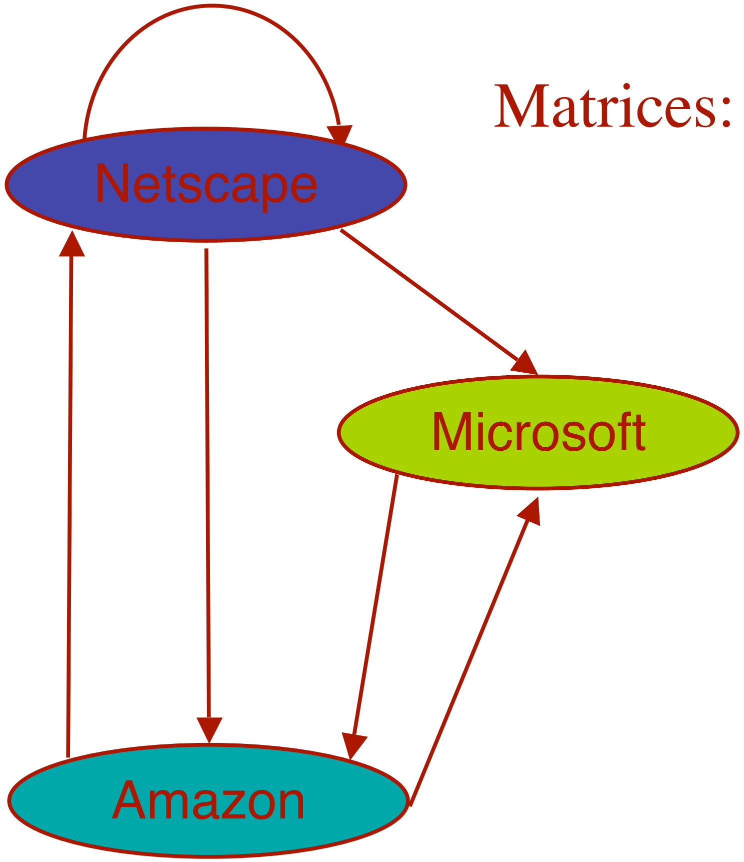
$$\underline{a} = \lambda\, \mu\, A^T A\, \underline{a}$$

$$\underline{h} = \lambda\, \mu\, A\, A^T \underline{h}$$

- Thus, we can compute a and h by iteratively solving the equations, giving us the principal eigenvectors of the matrices AA^T and A^TA, respectively.

Hubs and Authorities example:

Netscape acknowledges Microsoft's existence



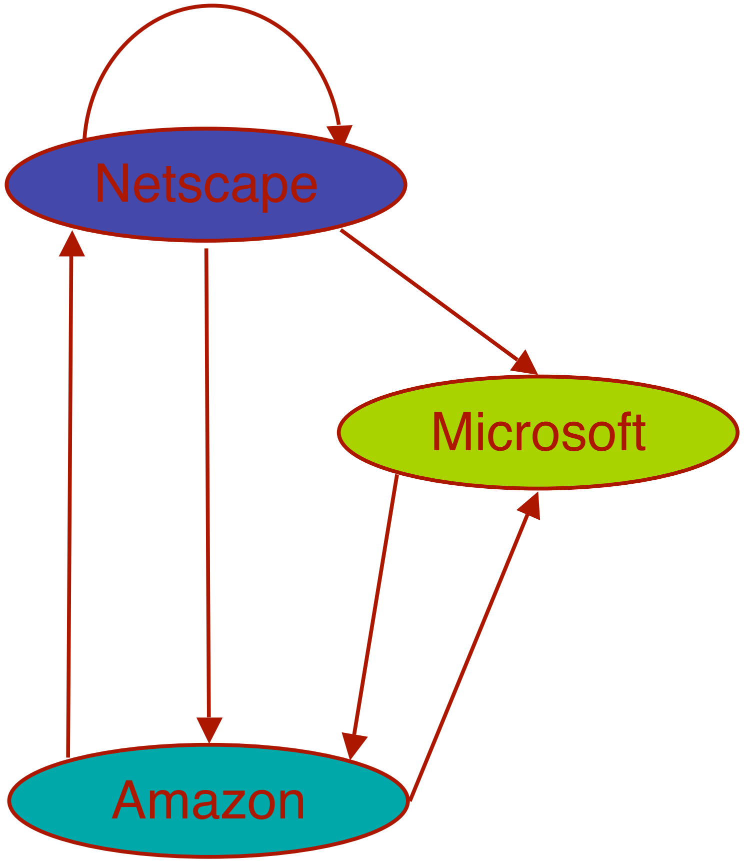
Matrices:

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \qquad A^T = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

$$AA^T = \begin{pmatrix} 3 & 1 & 2 \\ 1 & 1 & 0 \\ 2 & 0 & 2 \end{pmatrix} \qquad A^TA = \begin{pmatrix} 2 & 2 & 1 \\ 2 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$$

Hubs and Authorities example:

Netscape acknowledges Microsoft's existence



Iteratively solving the equations for a and h assuming $\lambda = \mu = 1$:

a(N)	=	1	5	24	114	...	1.3 a(A)
a(M)	=	1	5	24	114	...	1.3 a(A)
a(A)	=	1	4	18	84	...	a(A)
h(N)	=	1	6	28	132	...	
h(M)	=	1	2	8	36	...	
h(A)	=	1	4	20	96	...	

Authorities and Hubs pragmatics

- The system is jump-started by obtaining a set of root pages from a standard text index such as AltaVista or Google.
- The iterative process settles very rapidly.
  - A root set of 3,000 pages requires just 5 rounds of calculations!
  - The results are independent of the initial estimates
- Algorithm naturally separates web sites into clusters
  - e.g. a search for “abortion” partitions the Web into a pro-life and a pro-choice community