Database Management Systems

Winter 2003

CMPUT 391: Information Retrieval and the Web

Dr. Osmar R. Zaïane

University of Alberta

Chapter 27 of Textbook

Course Content

- Introduction
- Database Design Theory
- Query Processing and Optimisation
- Concurrency Control
- Data Base Recovery and Security
- Object-Oriented Databases
- **Inverted Index for IR**
- XML
- Data Warehousing
- Data Mining
- Parallel and Distributed Databases
- Other Advanced Database Topics

Objectives of Lecture 7

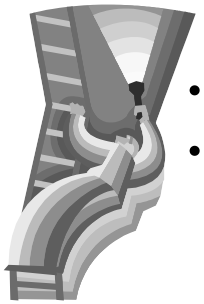
Inverted Indexes and Information Retrieval

- Get a general idea about the technologies behind search engines
- Get acquainted with inverted indexes
- Discuss ranking issues

Inverted Indexes and IR

- Inverted Indexes and Information Retrieval
- Signature Files
- Anatomy of a Search Engine
- Web Crawler
- Ranking Results
- Authorities, Hubs and PageRank

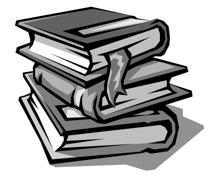
Everyday Activity



- We use search engines whenever we look for resources on the Internet
- How do these search engines work?
- How come they give different results while the results come from the same Web?
- The results are often very disappointing. Why aren't we satisfied?

Information Retrieval

- Find resources (documents) that contain a certain list of keywords



Find the pages where the phrase "alpha beta" occurs.

Searching sequentially is too expensive.

You would need an index to directly find the pages.

Creating an Index

documents ⇨ index

For each document

Document D_i ⇨ D_i: w_a, w_b, w_c…

documents ⇨
D_1: w_a, w_b, w_c…
D_2: w_a, w_d, w_e…
D_3: w_a, w_b, w_d…
…
D_n: w_x, w_y, w_z…

⇨

w_a: D_1, D_2, D_3 …
w_b: D_1 , D_3 …
w_c: D_1, …
w_d: D_2, D_3, …
…

Inverted Index

Querying

Which document contains W_a **and** W_b ? →

Inverted Index
w_a: D_1, D_2, D_3 …
w_b: D_1 , D_3 …
w_c: D_1, …
w_d: D_2, D_3, …
…

→ D_1, D_2, D_3 … $\cap$ D_1 , D_3 …

Which document contains W_a **or** W_b ? →

Inverted Index
w_a: D_1, D_2, D_3 …
w_b: D_1 , D_3 …
w_c: D_1, …
w_d: D_2, D_3, …
…

→ D_1, D_2, D_3 … $\cup$ D_1 , D_3 …

Inverted Indexes and IR

- Inverted Indexes and Information Retrieval
- Signature Files
- Anatomy of a Search Engine
- Web Crawler
- Ranking Results
- Authorities, Hubs and PageRank

Indexing for Text Search

- Text database: Collection of text documents
- Important class of queries: Keyword searches
  - Boolean queries: Query terms connected with AND, OR and NOT. Result is list of documents that satisfy the boolean expression.
  - Ranked queries: Result is list of documents ranked by their "relevance".
  - IR: Precision (percentage of retrieved documents that are relevant) and recall (percentage of relevant objects that are retrieved)
- Inverted indexes is not the only approach in IR. Signature files are also used for document retrieval.

Signature Files

- Index structure (the signature file) with one data entry for each document
- Hash function hashes words to bit-vector.
- Data entry for a document (the signature of the document) is the OR of all hashed words.
- Signature S1 matches signature S2 if S2&S1=S2

Signature Files: Query Evaluation

- Boolean query consisting of conjunction of words:
  - Generate query signature Sq
  - Scan signatures of all documents.
  - If signature S matches Sq, then retrieve document and check for false positives.
- Boolean query consisting of disjunction of k words:
  - Generate k query signatures S1, …, Sk
  - Scan signature file to find documents whose signature matches any of S1, …, Sk
  - Check for false positives

Signature Files: Example

Word	Hash
Agent	010
James	100
Mobile	001

RID	Document	Signature
1	Agent James	110
2	Mobile agent	011

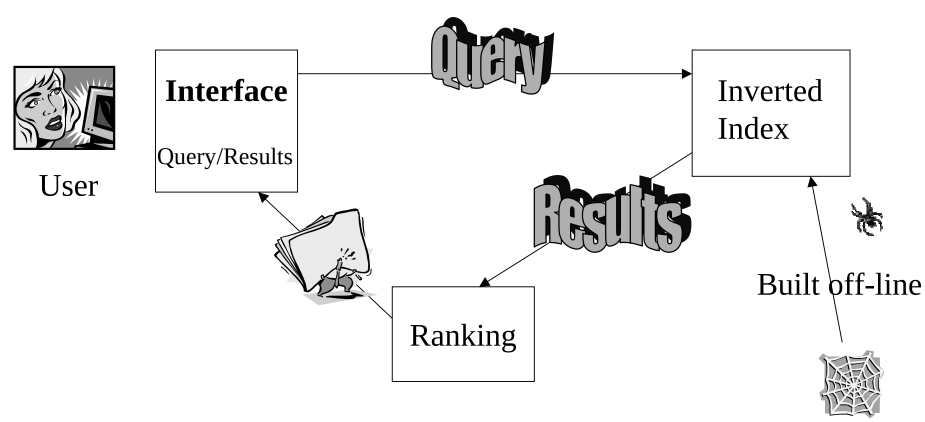
Inverted Indexes and IR

- Inverted Indexes and Information Retrieval
- Signature Files
- Anatomy of a Search Engine
- Web Crawler
- Ranking Results
- Authorities, Hubs and PageRank

Search Engine Components

- A Search Engine has an interface to enter queries
- A search engine has access to an inverted index already built
- A search engine ranks the results found in the index

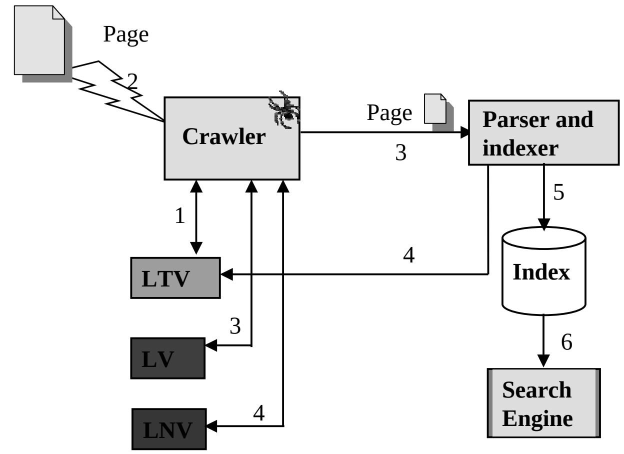
A Search Engine Blocs



Inverted Indexes and IR

- Inverted Indexes and Information Retrieval
- Signature Files
- Anatomy of a Search Engine
- Web Crawler
- Ranking Results
- Authorities, Hubs and PageRank

Search Engine General Architecture



Search Engines are not Enough

- Most of the knowledge in the World-Wide Web is buried inside documents.
- Search engines (and crawlers) barely scratch the surface of this knowledge by extracting keywords from web pages.
- There is text mining, text summarization, natural language statistical analysis, etc., but not the scope of this course.

Inverted Indexes and IR

- Inverted Indexes and Information Retrieval
- Signature Files
- Anatomy of a Search Engine
- Web Crawler
- Ranking Results
- Authorities, Hubs and PageRank

Relevancy Ranking

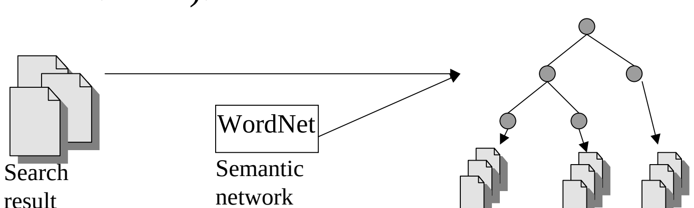
- Some search engine claim to have indexed about one billion documents
- Each search can yield a very large list of “supposedly relevant” documents
- Sifting through thousands of results is tedious and not necessary
- It is extremely important to rank the results since most users will look mainly at the 10 to 20 first documents.

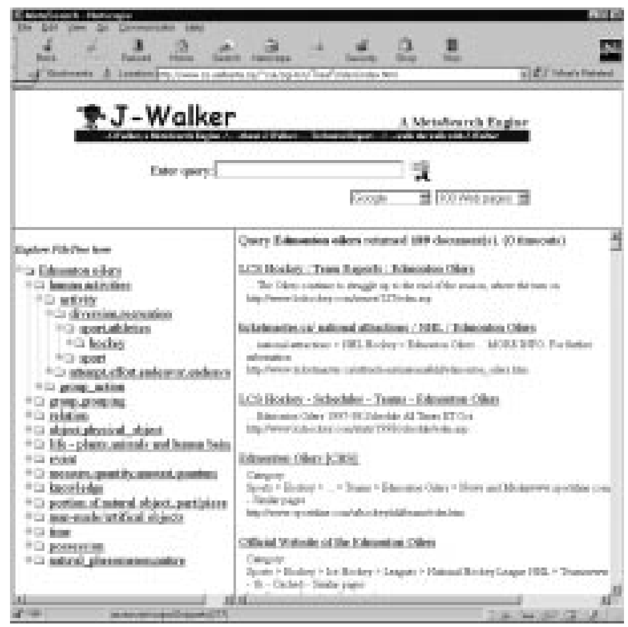
How do we Rank?

- Each Search Engine uses a different ranking function. Usually these ranking functions are not disclosed. (similarity measure)
- Parameters used in ranking:
  - Frequency of words
  - Location of words
  - Entirety of query
  - Size of document
  - Age of document
  - Existence in directory
  - Inward and outward Links
  - Metadata
  - Domain
  - And $$$$

Ontology for Search Results

- There are still too many results in typical search engine responses.
- Reorganize results using a semantic hierarchy (Zaïane et al. 2001).





Inverted Indexes and IR

- Inverted Indexes and Information Retrieval
- Signature Files
- Anatomy of a Search Engine
- Web Crawler
- Ranking Results
- Authorities, Hubs and PageRank

Hyperlink Induced Topic Search (HITS)

- Kleinberg's HITS algorithm (1998) uses a simple approach to finding quality documents and assumes that if document A has a hyperlink to document B, then the author of document A thinks that document B contains valuable information.

- If A is seen to point to a lot of good documents, then A's opinion becomes more valuable and the fact that A points to B would suggest that B is a good document as well.

General HITS Strategy

HITS algorithm applies two main steps.

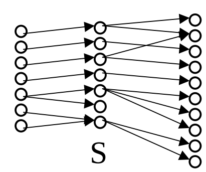
- A sampling component which constructs a focused collection of thousand web pages likely to be rich in authorities.

- A weight-propagation component, which determines the numerical estimates of hub and authority weights by an iterative procedure.

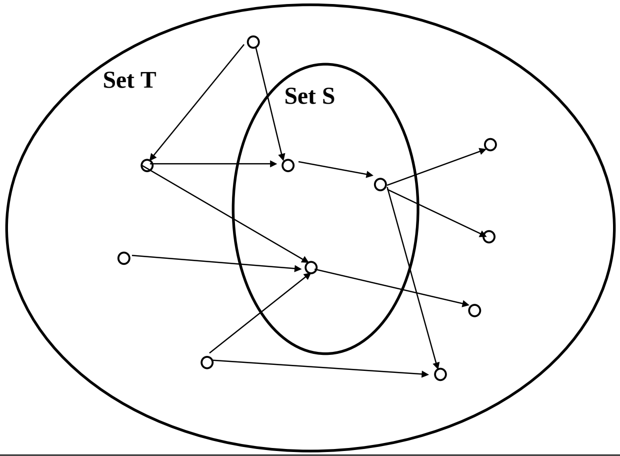
Steps of HITS Algorithm

- Starting from a user supplied query, HITS assembles an initial set S of pages:

  The initial set of pages is called root set. These pages are then expanded to a larger root set T by adding any pages that are linked to or from any page in the initial set S.



- HITS then associates with each page p a hub weight h(p) and an authority weight a(p), all initialized to one.



- HITS then iteratively updates the hub and authority weights of each page.

Let $p \rightarrow q$ denote "page p has an hyperlink to page q". HITS updates the hubs and authorities as follows:

$$a(p) = \sum_{q \rightarrow p} h(q)$$

Good authorities are linked by good hubs

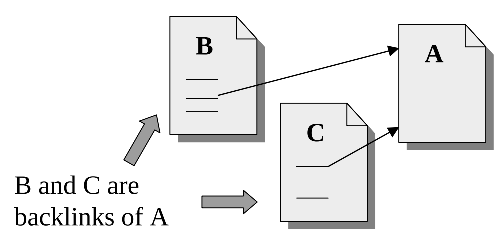
$$h(p) = \sum_{p \rightarrow q} a(q)$$

Good hubs link to good authorities

Ranking Pages Based on Popularity

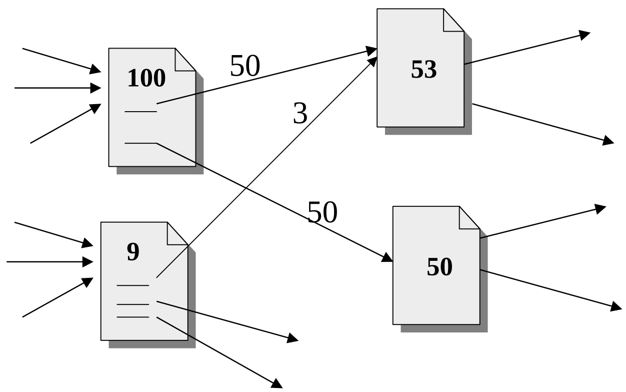
- Page-rank method (Brin and Page, 1998): Rank the "importance" of Web pages, based on a model of a "random browser.“
  - Initially used to select pages to revisit by crawler.
  - Ranks pages in Google's search results.
- In a simulated web crawl, following a random link of each visited page may lead to the revisit of popular pages (pages often cited).
- Brin and Page view Web searches as random walks to assign a topic independent "rank" to each page on the world wide web, which can be used to reorder the output of a search engine.
- The number of visits to each page is its PageRank. PageRank estimates the visitation rate => popularity score.

Page Rank: A Citation Importance Ranking



- Number of backpacks (~citations)

Idealized PageRank Calculation



Each Page p has a number of links coming out of it C(p) (C for citation), and number of pages pointing at page p_1, p_2 ….., p_n.

PageRank of P is obtained by

$$PR(p) = (1-d) + \sum_{k=1}^{n} \frac{PR(p_k)}{C(p_k)}$$

Summary

- Searching for relevant documents sequentially in a large collection of text documents is not a good solution.
- An inverted index is an index containing the list of documents per term. (documents containing the term)
- A web search engine does not crawl the web at query time. The Web is pre-indexed in an inverted index.
- Automatic crawling of the Web starts from seeds. If starting seeds are different, the resulting index is different.
- Ranking results is an important operation for Search engines. (only 20 to 30 first are usually seen).
- There is still a great deal of research related to search the Web