GLOBAL WEB SEARCH

U Manber

Rapporteur: Martin Beet

Global Web Search

Udi Manber

Department of Computer Science

University of Arizona

udi@cs.arizona.edu

http://glimpse.cs.arizona.edu/udi.html

Global Web Search

There are two types of global web search:

1. spider-based (lycos, webcrawler, altavista, hotbot, microsoft?)

2. manual categorization (yahoo)

Spider-based — the Big Inhale

- All the documents in the world are dropped on the living room floor and you can search them all
- Spiders traverse all known sites and pick *samples*. (It used to be *everything*, but they cannot handle the volume anymore.)
- They index all of that in one big flat database.
- Updates are done continuously, with complete coverage once every two weeks (hotbot) to several months.

The Business Side

Competition is fierce.

The search engines differ by strategies for sampling, speed of crawling, search features, and alliances with content partners.

They now see themselves mainly as content companies.

Like broadcast TV, they don't work for you, they work for the advertizers.

Ranking

- Unlike traditional IR, web information is not passive and unbiased.

- It is an information that wants to be found, wants to be attractive, wants *you*, or rather wants your money.

- Getting your site to be ranked high on search engines is now an industry by itself.

How to Improve Ranking

- Add lots of keywords, relevant or not, in an unreadable form (black on black, 3-point font)
- Check your competitors daily and add the keywords that make them appear before you.
- Change your site (especially title) every day to appear up-to-date
- There are *automated* trial-and-error tools to improve your ranking

A new arm race

Is web search too easy to use?

- To attract naive users, all search engines provide very easy to use interfaces — one box, put everything you want into it
- Is it the right way for the long term?
- The general analogy to playing music.

Yahoo

- Two types of classifications, by subject and by region
- Very labor intensive.
- A search typically takes a lot of time with many page downloads. Perfect for business.

My own recent projects

- **GlimpseHTTP** — search
- **Harvest** — collect, extract, and search
- **WebGlimpse** — collect, organize, and search
- **Siff** — find similar documents
- **NetShell** — customize your browser
- **HAC** — organize your file system
- **The Search Broker** — focused web search
- **WordSmyth** and the *conceptuary* project
- **Information Matching**
- **Negotiators**
- **Security**

Glimpse

Main features

- versatility
- customization
- space-efficiency
- speed (both search and indexing)
- incremental indexing
- (**NEW**) Temporal features

  (e.g., ''search only files changed in the last week'')

GlimpseHTTP

GlimpseHTTP is a set of scripts to support search in http servers. It provides a browsing and searching combination.

When you search, glimpse automatically searches only the subtree below where you currently are (and it still uses only one global index).

This way the logical structure of the data is preserved.

GlimpseHTTP sites (over 850)

- **Governments** (New Zeland, Canada, Australia, Czech Republic, Singapore)
- **Organizations** (American Red Cross, EMBL, World Conservation Monitoring Centre, The British Library, The National Weather Service, The MacArthur Foundation, AIDS Authority)
- **Government Labs** (NASA, LBL, Los Alamos, Sandia, Army Corps of Engineers, Brookhaven)
- **Universities** (Washington, Berkeley, Columbia, NYU, Wisconsin, Georgia Tech, Penn, Duke, Georgetown, University of Melbourne, Oxford)
- **Medical Information** (The cancer lists, Iowa Virtual Hospital, NYU medical center, Mental Health Net, AIDS FAQ)
- **Newspapers** (MIT Tech, The San Diego Source, The Yale Daily News, ENews, The Scientist, Cleveland Free Times, Arizona Daily Wildcat)
- **Companies** (AT&T, Digital, Prodigy, Convex, Kodak, Group Bull, Computer Sciences Corporation, Encore, Land's End, Xerox)
- **Others** (The Rock and Roll Hall of Fame, Air Force HQ, U.S. Embassy in Bucharest, The Protein Data Bank, ACM Journal Abstracts, Cleveland Indians, FAQ search, CS Bibliography)

Harvest main features

(http://harvest.transarc.com)

- Integrated set of tools for gathering, extracting, indexing, searching, caching, and replicating.
- Focused collections gathered from widely distributed repositories.
- Multi-level scalable organization of data.
- Distributed gathering architecture very efficient on network, servers, disk.
- Customizable with umpteen features and options.
- Easy to build and easy to use servers (brokers).

Information discovery in the global net should be multi-level, topical, and highly customizable.

WebGlimpse

(http://glimpse.cs.arizona.edu/webglimpse/)

Incorporate the context of your

current location into your search.

- A search for "fast AND reliable" should give different results when done from the marketing, accounting, and/or engineering pages.

- A search for "Network" or "Computer" may be useless when the whole department is searched, but very useful when limited to the research group in, say, Statistics.

- If you have to switch to the "Search Page" you may lose your "train of traversal".

- **Goal**: Combine the paradigms of browsing and searching.

(Speech recognition would do wonders)

The Search Broker

(http://sb.cs.arizona.edu/sb/)

- Current web search lacks **Focus**
- We often don't want more, we want *less*.
- There are already thousands of *specialized* databases in various topics.
- Can we connect them all?

The Search Broker forwards your query to a search engine dealing *specifically* with the subject of your question.

Subjects include stocks, flight schedule, languages, nutrition, medical, government, movies, patent, software, science, TV, hotels, history, art, driving directions, people, law, maps, books, and 400 more.

The Search Broker — Implementation

The subject is determined by the first word of the query.

The Search Broker finds the subject in its database, and then performs the following:

1. selects the appropriate specialized search engine (e.g,, the one from USDA for calories)

2. reformats the query

3. forwards the query to the right place

4. forwards the results back to you

If the subject is not found, the query is forwarded to Lycos.

The Search Broker — Examples

stocks ibm
hotel phoenix
fly sfo jfk
poison ammonium
calories pizza
howto buy a car
medline fatigue
travel fiji
convert 6 inch to cm
directions Summit,NJ to Washington, DC
french-english amour
car-prices 1992 Chevrolet, Camaro
phone-reverse 111-222-3333
flag barbados
movie round midnight

The Search Broker — Issues

- How do we select the subjects and the engines?
- How do users know which subjects to use?
- How do we compress several fields into one box?
- Can the subjects be inferred?
- Can we send queries to many places at once?
- Can the Search Broker be integrated into a general web search engine?

Non-Keyword Search

Can we utilize the *structure* of the web for search?

People in *bibliometrics* have long studied the citation structure of scientific articles (e.g., to assess *impact*).

WWW Hubs and Authorities:
Jon Kleinberg, Cornell Univ.

- Compute a measure of quality for a page based on how many (and which) other pages point to it.
- Use the same information for clustering

Authorities are ''good'' documents, with high indegree

Hubs point to many authorities

Kleinberg's algorithm

Assume that each page P has a hub weight and an authority weight.

An I operation: set the authority weight of P to be the sum of hub weights of the pages pointing to it.

An O operation: set the hub weight of P to be the sum of authority weights of the pages it points to.

Basic algorithm:

Start with a set of reasonably relevant documents

Apply the I operations

Apply the O operations

Normalize

Continue until you converge.

Examples

Starting with an Altavista search for ''Web Browsers'' and applying the algorithm they got

.225 Mosaic Windows Home Page
.202 Welcome to Netscape
.196 TradeWave Corporation
.188 SlipKnot HomePage
.188 winWeb and MacWeb
.185 Microsoft Internet Explorer

None of these showed up in the original search.

Examples

Starting with an Altavista search for ''Java'' and applying the algorithm they got

.328 Gamelan
.251 javaSoft Home Page
.190 The Java Developer: How Do I..
.190 The Java Book Pages
.183 comp.lang.java FAQ

Similar pages

If you start with a page and consider it an authority, you would expect the other authorities in the same ''community'' to be similar.

Take a set of pages pointing to the initial page, and apply the algorithm to them:

When they started with *www.honda.com* they got

.202 Toyota
.199 Honda
.192 Ford
.173 BMW
.162 Volvo
.158 Saturn
.155 Nissan

DISCUSSION

Rapporteur: Martin Beet

Lecture One

The discussion after Professor Manber's talks on Global Search touched on several current research issues. One of the main points of discussion was the lack of uniformity among search facilities, especially with respect to classification of resources. Professor Manber was asked whether there were efforts to adopt a standard classification scheme such as the Dewey numbering scheme for services which categorize resources, such as Yahoo or AltaVista with the new LiveTopics facility. Professor Manber replied that widespread use of such a standard classification faced mainly two difficulties. Firstly, automatic classification was little used in existing services, which, like Yahoo, mostly rely on manual classification. Secondly, a universally applicable classification scheme would be too unwieldy to be readily usable. He shared Professor Farber's view that there was a lot of scope for applying automatic classification to global searching. Mr Butler complained about the widely differing query interfaces to existing search facilities. Professor Manber agreed that standards for interfaces and indexing were desirable, but was rather pessimistic as this was the main distinguishing feature of search services. Dr Coleman enquired about existing services' handling of emerging metadata schemes such as the Dublin Core. Professor Manber said these were mostly ignored by the services. The future of such schemes would depend largely on the availability of tools to generate metadata. In his opinion, these were at the moment difficult to use, employed many different formats and relied on sensible HTML coding, which could not be readily assumed, even for pages produced using HTML generators. In answering a question referring to reports charging spidering programs with excessive network resource consumption, Professor Manber expressed the opinion that the search facilities provided a valuable service for relatively little consumption.

Finally, he was asked by Professor Randell to comment on the success of the Yahoo service which seemed to be incompatible with its manual approach. In answering, Professor Manber stated that a search facility's success nowadays depended much more on strategic alliances and the versatility of such a service rather than the technology. He remarked that one of the reasons behind Yahoo's success was its lack of speed and the necessity to view a lot of pages during the search process, as this made it very attractive for advertising.

Lecture Two

Professor Manber's second talk concentrated on the demonstration of the Search Broker (http://sb.cs.arizona.edu/sb/), his most recent project. Questions from the audience were mostly concerned with using the Search Broker.

One issue that was addressed repeatedly was the need to classify or search data based on geographic location. It was pointed out that Yahoo now offered a number of regional directories, and some search engines allow to specify the domain of the host in a query. Professor Manber nevertheless acknowledged that the need for locating information based on geographic proximity would become increasingly important with the growing number of local, community-oriented information providers.