Search Engines: Technology, Society, and Business

Prof. Marti Hearst

Sept 24, 2007
How Search Engines Work

Three main parts:

i. Gather the contents of all web pages (using a program called a crawler or spider)

ii. Organize the contents of the pages in a way that allows efficient retrieval (indexing)

iii. Take in a query, determine which pages match, and show the results (ranking and display of results)
Standard Web Search Engine Architecture
Standard Web Search Engine Architecture

Crawler machines

Crawl the web

Check for duplicates, store the documents

DocIds

Create an inverted index

Search engine servers

Inverted index
Standard Web Search Engine Architecture

Crawler machines → Crawl the web

Check for duplicates, store the documents

DocIds

Create an inverted index

Inverted index

Search engine servers

Show results To user

user query
i. Spiders (crawlers)

- How to find web pages to visit and copy?
  - Can start with a list of domain names, visit the home pages there.
  - Look at the hyperlink on the home page, and follow those links to more pages.
    - Use HTTP commands to GET the pages
  - Keep a list of urls visited, and those still to be visited.
  - Each time the program loads in a new HTML page, add the links in that page to the list to be crawled.
Four Laws of Crawling

- A Crawler must show identification
- A Crawler must obey the robots exclusion standard
  [http://www.robotstxt.org/wc/norobots.html](http://www.robotstxt.org/wc/norobots.html)
- A Crawler must not hog resources
- A Crawler must report errors
Example robots.txt file

www.whitehouse.gov/robots.txt
(just the first few lines)

User-agent: *
Disallow: /cgi-bin
Disallow: /search
Disallow: /query.html
Disallow: /help
Disallow: /360pics/text
Disallow: /911/911day/text
Disallow: /911/heroes/text
Disallow: /911/messages/text
Disallow: /911/patriotism/text
Disallow: /911/patriotism2/text
Disallow: /911/progress/text
Disallow: /911/remembrance/text
Disallow: /911/response/text
Disallow: /911/sept112002/text
Disallow: /911/text
Disallow: /ConferenceAmericas/text
Disallow: /GOVERNMENT/text
Disallow: /QA-test/text
Disallow: /aci/text
Disallow: /afac/text
Disallow: /africanamerican/text
Disallow: /africanamericanhistory/text
Disallow: /agencycontact/text
Disallow: /americancompetitiveness/text
Disallow: /apec/2003/text
Disallow: /apec/2004-summit/text
Disallow: /apec/2004/text
Lots of tricky aspects

- Servers are often down or slow
- Hyperlinks can get the crawler into cycles
- Some websites have junk in the web pages
- Now many pages have dynamic content
  - The “hidden” web
  - E.g., schedule.berkeley.edu
    - You don’t see the course schedules until you run a query.
- The web is HUGE
“Freshness”

- Need to keep checking pages
  - Pages change (25%, 7% large changes)
    - At different frequencies
    - Who is the fastest changing?
    - Pages are removed
  - Many search engines cache the pages (store a copy on their own servers)
What really gets crawled?

- A small fraction of the Web that search engines know about; no search engine is exhaustive
- Not the “live” Web, but the search engine’s index
- Not the “Deep Web”
- Mostly HTML pages but other file types too: PDF, Word, PPT, etc.
ii. Index (the database)

Record information about each page

• List of words
  ■ In the title?
  ■ How far down in the page?
  ■ Was the word in boldface?

• URLs of pages pointing to this one

• Anchor text on pages pointing to this one
Inverted Index

• How to store the words for fast lookup

• Basic steps:
  ■ Make a “dictionary” of all the words in all of the web pages
  ■ For each word, list all the documents it occurs in.
  ■ Often omit very common words
    ■ “stop words”
  ■ Sometimes stem the words
    ■ (also called morphological analysis)
    ■ cats -> cat
    ■ running -> run
Inverted Index Example

Inverted Index

- In reality, this index is HUGE
- Need to store the contents across many machines
- Need to do optimization tricks to make lookup fast.
iii. Results ranking

- Search engine receives a query, then
- Looks up the words in the index, retrieves many documents, then
- Rank orders the pages and extracts “snippets” or summaries containing query words.
  - Most web search engines assume the user wants all of the words (Boolean AND, not OR).
- These are complex and highly guarded algorithms unique to each search engine.
Some ranking criteria

- For a given candidate result page, use:
  - Number of matching query words in the page
  - Proximity of matching words to one another
  - Location of terms within the page
  - Location of terms within tags e.g. `<title>`, `<h1>`, link text, body text
  - Anchor text on pages pointing to this one
  - Frequency of terms on the page and in general
  - Link analysis of which pages point to this one
  - (Sometimes) Click-through analysis: how often the page is clicked on
  - How “fresh” is the page

- Complex formulae combine these together.
Machine Learned Ranking

• Goal: Automatically construct a ranking function
  ▪ Input:
    □ Large number training examples
    □ Features that predict relevance
    □ Relevance metrics
  ▪ Output:
    □ Ranking function

• Enables rapid experimental cycle
  ▪ Scientific investigation of
    □ Modifications to existing features
    □ New feature
What is Machine Learning?

• We don’t know how to program computers to learn the way people do

• Instead, we devise algorithms that find patterns in data.
Machine Learning Example

- Devise algorithms that find patterns in data.
- Example:
  - Start with 2 classes (Italian food or Chinese food)
  - Show the algorithm examples of both
  - Look at the features of each
    - Ingredients
    - Cooking style
  - Figure out which features are distinct to each class, and (optionally) how frequently they occur.
  - See a new dish: try to guess which cuisine it is in.
A Toy Example

Data Examples

Chicken parmigiana: chicken, cheese, garlic, tomatoes; bake
Spaghetti w/pesto: pasta, basil, garlic, pine nuts; saute
Pizza: flour, tomatoes, garlic, ham; bake

Kung Pao chicken: chicken, chili peppers, garlic, rice; saute
Rice noodles with shrimp: shrimp, peppers, soy, rice; saute
Pork buns: pork, onions, soy, flour; steam

Derived Rules

If ingredient == tomatoes OR ingredient != rice:
then recipe == Italian
If cooking_method == steam:
then recipe == Chinese
Ranking Features  (from Jan Pedersen’s lecture)

- A0 - A4 anchor text score per term
- W0 - W4 term weights
- L0 - L4 first occurrence location (encodes hostname and title match)
- SP spam index: logistic regression of 85 spam filter variables (against relevance scores)
- F0 - F4 term occurrence frequency within document
- DCLN document length (tokens)
- ER Eigenrank
- HB Extra-host unique inlink count
- ERHB ER*HB
- A0W0 etc. A0*W0
- QA Site factor - logistic regression of 5 site link and url count ratios
- SPN Proximity
- FF family friendly rating
- UD url depth
A0w0 < 22.3
L0 < 18+1
L1 < 18+1
L1 < 509+1
W0 < 856
F0 < 2 + 1
F2 < 1 + 1
R=0.0015
R=-0.0545
R=-0.0199
R=-0.2368
R=-0.0790
R=-0.0039
R=-0.1185
R=-0.0790
R=-0.1604
R=-0.1185
R=-0.0199
R=-0.2368

The importance of anchor text

A terrific course on search engines

The anchor text summarizes what the website is about.
Measuring Importance of Linking

- PageRank Algorithm
  - Idea: important pages are pointed to by other important pages
  - Method:
    - Each link from one page to another is counted as a “vote” for the destination page
    - But the importance of the starting page also influences the importance of the destination page.
    - And those pages scores, in turn, depend on those linking to them.

Measuring Importance of Linking

- Example: each page starts with 100 points.
- Each page’s score is recalculated by adding up the score from each incoming link.
  - This is the score of the linking page divided by the number of outgoing links it has.
  - E.g., the page in green has 2 outgoing links and so its “points” are shared evenly by the 2 pages it links to.
- Keep repeating the score updates until no more changes.

Class Exercise

• Students as web pages and a search engine
  • Web pages:
    ■ Web site = where you live
    ■ Hyperlinks = who you know in class
    ■ Web page = Beatle’s song title

Unit 2

Jane Tran

I
Wanna
Hold
Your
hand
Class Exercise

- Crawlers: follow the links between web pages
- Indexers: record information about each document
- Ranking algorithm: compute which documents to retrieve, and their order
- Human: search the web!
Crawler

- Get the first page (student) (from a pre-defined list).
- Write down the other students that this student links to (the people hyperlinks)
- Assign each document (student) a unique ID (number)
- Visit each of these in turn
- Be sure to eliminate duplicates!
Indexers

- Record the following information
- Write down each word that appears in the document
- Write down also the ID of that document (student)
- If you’ve seen that word before, add this document to that word’s list of document IDs
Ranking Algorithm

• For a given query:
  ▪ Ask the indexers to tell it the document IDs that contain those words
  ▪ Compute a score based on:
    □ How often the words of the query occur in the document (if the word falls in the doc multiple times, that is better)
    □ How popular the web site (student housing location) is.
    □ How long the document is (shorter is better)
  ▪ Formula:
    □ Score for a document =
      ■ # hits in query + #pages in site - length(document)
  ▪ List the results in sorted order.
Test Your Understanding

- What is the difference between the WWW and the Internet?
Internet vs. WWW

• Internet and Web are not synonymous

• Internet is a global communication network connecting millions of computers.

• World Wide Web (WWW) is one component of the Internet, along with e-mail, chat, etc.

• Now we’ll talk about both.
Test Your Understanding

• How many queries are there per day to a major search engine?

• How much data is in the index of a major search engine?

• How many computers act as servers for a major search engine?
Test Your Understanding

• How many queries are there per day to a major search engine?
  ■ Hundreds of millions (NYTimes article)

• How much data is in the index of a major search engine?
  ■ Billions of documents
  ■ Petabytes of data

• How many computers act as servers for a major search engine?
  ■ Hundreds of thousands, maybe millions
What is a Petabyte?
Start with Orders of Magnitude


An order of magnitude is the class of scale or magnitude of any amount, where each class contains values of a fixed ratio to the class preceding it. The ratio most commonly used is 10.

<table>
<thead>
<tr>
<th>In words</th>
<th>Decimal</th>
<th>Power of ten</th>
<th>Order of magnitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>ten thousandths</td>
<td>0.0001</td>
<td>$10^{-4}$</td>
<td>-4</td>
</tr>
<tr>
<td>(these terms may be confusing)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>thousandth</td>
<td>0.001</td>
<td>$10^{-3}$</td>
<td>-3</td>
</tr>
<tr>
<td>hundredth</td>
<td>0.01</td>
<td>$10^{-2}$</td>
<td>-2</td>
</tr>
<tr>
<td>tenth</td>
<td>0.1</td>
<td>$10^{-1}$</td>
<td>-1</td>
</tr>
<tr>
<td>one</td>
<td>1</td>
<td>$10^0$</td>
<td>0</td>
</tr>
<tr>
<td>ten</td>
<td>10</td>
<td>$10^1$</td>
<td>1</td>
</tr>
<tr>
<td>hundred</td>
<td>100</td>
<td>$10^2$</td>
<td>2</td>
</tr>
<tr>
<td>thousand</td>
<td>1,000</td>
<td>$10^3$</td>
<td>3</td>
</tr>
<tr>
<td>ten thousand</td>
<td>10,000</td>
<td>$10^4$</td>
<td>4</td>
</tr>
<tr>
<td>million</td>
<td>1,000,000</td>
<td>$10^6$</td>
<td>6</td>
</tr>
<tr>
<td>billion</td>
<td>1,000,000,000</td>
<td>$10^9$</td>
<td>9</td>
</tr>
</tbody>
</table>
What is a Petabyte?

It is 10 million gigabytes

<table>
<thead>
<tr>
<th>SI prefixes</th>
<th>Value</th>
<th>Historical use</th>
<th>Binary prefixes</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbol</td>
<td>Value</td>
<td>Symbol</td>
<td>Value</td>
<td></td>
</tr>
<tr>
<td>(name)</td>
<td></td>
<td>(name)</td>
<td>(name)</td>
<td></td>
</tr>
<tr>
<td>kB (kilobyte)</td>
<td>1000^1 = 10^3</td>
<td>KB</td>
<td>1024^1 = 2^10 kIB (kibibyte)</td>
<td>2^10</td>
</tr>
<tr>
<td>MB (megabyte)</td>
<td>1000^2 = 10^6</td>
<td>MB</td>
<td>1024^2 = 2^20 MiB (mebibyte)</td>
<td>2^20</td>
</tr>
<tr>
<td>GB (gigabyte)</td>
<td>1000^3 = 10^9</td>
<td>GB</td>
<td>1024^3 = 2^30 GiB (gibibyte)</td>
<td>2^30</td>
</tr>
<tr>
<td>TB (terabyte)</td>
<td>1000^4 = 10^12</td>
<td>TB</td>
<td>1024^4 = 2^40 TiB (tebibyte)</td>
<td>2^40</td>
</tr>
<tr>
<td>PB (petabyte)</td>
<td>1000^5 = 10^15</td>
<td>PB</td>
<td>1024^5 = 2^50 PiB (pebibyte)</td>
<td>2^50</td>
</tr>
<tr>
<td>EB (exabyte)</td>
<td>1000^6 = 10^18</td>
<td>EB</td>
<td>1024^6 = 2^60 ExB (exbibyte)</td>
<td>2^60</td>
</tr>
<tr>
<td>ZB (zettabyte)</td>
<td>1000^7 = 10^21</td>
<td>ZB</td>
<td>1024^7 = 2^70 ZaB (zebibyte)</td>
<td>2^70</td>
</tr>
<tr>
<td>YB (yottabyte)</td>
<td>1000^8 = 10^24</td>
<td>YB</td>
<td>1024^8 = 2^80 YiB (yobibyte)</td>
<td>2^80</td>
</tr>
</tbody>
</table>
Test Your Understanding

• Why is the empty text box special, from a software application point of view?
Comparison to State-of-the-art (from Jan Pedersen’s lecture)
Test Your Understanding

• Why is the search results page unchanged from 10 years ago? Why is it so plain?
Test Your Understanding

- What is needed for high-quality search results?
Test Your Understanding

- What is needed for high-quality search results?
- Good results for:
  - Ranking
  - Comprehensiveness
  - Freshness
  - Presentation
Test your Understanding

• What are three levels of user evaluation?
Test your Understanding

• What are three levels of user evaluation?
  • Micro
    - Small details about the UI; eye tracking
    - Milliseconds
  • Meso
    - Field studies
    - Days to weeks
  • Macro
    - Millions of users
    - Days to months
What do these mean?
Test your understanding

- What is meant by ambiguous and disambiguate?
What is meant by ambiguous and disambiguate?

- Words with more than one meaning or more than one sense
  - Jets: sports team or airplane?
  - Bass: fish or musical instrument?
Test Your Understanding

- What is morphological analysis, also known as stemming?
Test Your Understanding

• What is morphological analysis, also known as stemming?

  Convert a word to its base form:
  
  □ Running, ran, runs -> run
  □ Building, builder, builds -> build? Not always
Test Your Understanding

- Why is it not always a good idea to stem query terms?
Test Your Understanding

- Why is it not always a good idea to stem query terms?
  - Sometimes the form a word is used indicates something about the sense of the word.
  - apple vs apples
What does this mean? (from Dan Russell’s lecture)

Looking vs. Clicking

- Users view results one and two more often / thoroughly
- Users click most frequently on result one
What does this mean? (from Dan Russell’s lecture)

User: 16925

[ knitting patterns ]

[ knitting patterns socks ]

mining behavior
Test Your Understanding

- Are peoples’ search skills evolving? If so, how?

- What is “teleporting?” What is “orienteering?”
Test Your Understanding

• What are navigational queries?

• What other kinds of queries are there?

• What do these queries mean?
  • banana
  • Sgt Peppers Lonely Hearts Club Band
  • Why is my dog sick?
Search Operators

- How do “double quotes” work?
- What does * mean?
- What is AND vs. OR?
Know your search engine

- What is the default Boolean operator? Are other operators supported?
- Does it index other file types like PDF?
- Is it case sensitive?
- Phrase searching?
- Proximity searching?
- Truncation?
- Advanced search features?
Keyword search tips

- There are many books and websites that give searching tips; here are a few common ones:
  - Use unusual terms and proper names
  - Put most important terms first
  - Use phrases when possible
  - Make use of slang, industry jargon, local vernacular, acronyms
  - Be aware of country spellings and common misspellings
  - Frame your search like an answer or question