CPSC156a: The Internet
Co-Evolution of Technology
and Society

Lecture 9: October 2, 2003
Web Searching and Google
Graphs: An Important Abstraction

Graphs with directed links are called “digraphs.”
Digraphs are Ubiquitous in Computer Science

• Used as *models* of systems
  - Nodes represent *components*.
  - Links represent *interactions or relationships*.

• Examples we’ve seen in CPSC156a:
  - Computer networks: Nodes represent computers (e.g., hosts or routers), and links represent direct (“hardwired”) connections.
  - The WWW: Nodes represent web pages, and links represent ... “links” (e.g., html code pointing from one page to another).
Two Aspects of WWW Searching

• Analyze contents of pages
  - Text (e.g., search terms)
  - Structure (e.g., HTML tags)

• Analyze structure of WWW digraph
  - Links to page P indicate interest in the contents of P.
  - Importance depends on who is interested.
  - Requires global analysis of digraph.
The WWW Digraph

- Massive, *Distributed, Explicit* Digraph
- More than 3.3 Billion Nodes (Pages)
- Sparse: Average Degree (links per page) is 5-15.
- Can be *crawled* (i.e., every node visited) in time linear in the total number of links (using classical methods).
“Hot” Research Area

- Graph Representation
- Duplicate Elimination
- Clustering
- Ranking Search Results
Finding Information on the Internet

The Internet is so successful partly because it is so easy to publish information on the World Wide Web.

• No central authority on what pages exist, where they exist, or when they exist.
• Too much to sort through, anyway.
• Question: How do we find what we need on the web?
WWW Search Engines

• **Answer:** Set up websites that people can use to search for information by performing a *search query*.

• Not such an easy solution! In addition to the technical problems, we have these business questions:
  - How do people know about the search engine websites?
  - How do you make money off of this? (Especially now that the service is free.)
Examples of Search Websites

- Website directories that have grown to become portals
  - Yahoo! (first searches its own hand-made directory, then Google index)
  - Lycos
  - Excite

- ISP + portals that now include search
  - AOL / Netscape (agreement with Google, as of 6/2002)
  - MSN (agreement with Inktomi - the search engine technology also used by Yale’s website)

- InfoSpace / MetaCrawler, a “search engine searcher”
- AskJeeves, a “natural language” search engine
- Google, a “traditional search” website that remains dedicated to searching
Solutions (?) to Technical Problems

• How do we keep track of what pages are on the WWW?
  - Have a *crawler* or *spider* scan the web and links between pages to find new, updated, and removed pages.

• How do we store the content we find?
  - Design a way to map keywords in queries to documents so we can return a *usefully ordered list* to the user.

• What happens when pages are temporarily unavailable?
  - Use *caching*: keep a local copy of documents as we crawl the web. (Need lots of space!)
Solutions (?) to Technical Problems (continued)

• How do we store all the information?
  - Use a large network of disks (and maybe a clever method of compression) that can be easily searched.

• How do we handle so many different requests?
  - Use a cluster of computers that work together to process queries.

There is still ongoing research to find better ways to solve these problems!
• Full name: Google, Inc.
• Privately held company. Funding partners include Kleiner Perkins Caufield & Byers and Sequoia Capital.
• Employees: over 1000 worldwide (more than 60 with Ph.D.)
• Mission: “Organize the world’s information and make it universally accessible and useful.”
• Award-winning search engine that has indexed over 3.3 billion web pages (note: index size 1.6B in 12/2001.)
Google History

• 1998: Founders Larry Page and Sergey Brin (Ph.D. students at Stanford) raise $1 million from family, friends, and angel investors. Google is incorporated Sept. 7. Site receives 10,000 queries per day and is listed in PC Magazine’s top 100 search websites list.

• 1st half 1999: Google has 8 employees and answers 500,000 queries/day. Red Hat (Linux distributor) becomes first customer. Google gets $25 million equity funding.
Google History (2)

- 2\textsuperscript{nd} half 1999: 39 employees, 3 million queries/day. Partners with Virgilio of Italy to provide search services.

Google History (3)

• 2001: Acquires Deja.com’s Usenet archive, adding newsgroups to Google’s index. Improves and adds services including browser plug-ins, image searching, PDF searching, cell-phone and handheld compatibility, and queries and document searches in many languages. Advertising services used by over 350 Premium Sponsorship customers.

• Spring 2003: 3.3 billion web pages, 800 million newsgroup messages, and 425 million images indexed. Serves 200 million queries/day.
Google Partners

- Yahoo!
- Palm
- Nextel
- Netscape
- Cisco Systems
- Virgin Net
- Netease.com
- RedHat
- Virgilio
- Washingtonpost.com
Google’s Business Model

Scalable Search Services:
• Google provides customized search services for websites.
• Has become the primary search engine used by popular portal and ISP websites.

Advertising:
• **Premium Sponsorship**: sponsored text links at the top of search results based on search category.
• **AdWords**: keyword-targeted, self-service advertising method. Choose keywords or phrases where text ads will appear to the right of the search result list.
• No banner ads or graphics!
Google Advertising Screenshot

- **Premium Sponsorships**
- **Self-Service AdWords**

Google search results page with highlighted sponsored links.
Technical Highlights

- **PageRank Technology**: Linear-algebraic, objective calculations of the “importance” of a webpage.
  - Link from Page A to Page B is a “vote” for B.
  - Importance of A is factored into the vote.
  - Page owners cannot pay to have their PageRanks modified. (Note the difference between buying a “sponsored link” and getting a higher PageRank.)
  - Google employees can modify a PageRank in exceptional circumstances (e.g., security threats).
Technical Highlights (2)

- Readings on how PageRank works:
  http://www.google.com/technology/index.html

  “Google’s PageRank explained, and how to make the most of it,”


• **Hypertext-Matching Analysis:** The HTML tags are taken into account when examining the contents of a page. Headings, fonts, positions, and content of neighboring pages influence the analysis.
Technical Highlights (3)

• **Scalable Core Technology:** Calculations are performed by the largest commercial Linux cluster of over 10,000 servers. *Can grow with the Internet!*

• **Complex-File Searching:** Google can now index files in “non-Internet” formats, *e.g.*:
  - PostScript, PDF (Adobe)
  - Word, Excel, PowerPoint, Works (Microsoft)
  - WordPro, 1-2-3 (IBM/Lotus SmartSuite)
  - MacWrite
  - Rich Text (RTF), plain text
Technical Highlights (4)

• **Bayesian Spelling-Suggestion Program:** Offers suggestions for misspelled words in queries, making searching easier. (“Did you mean...?”)

• **Internationalization:**
  - Google is developing technology to index pages with complex scripts, *e.g.*:
    - Some East Asian languages have no spaces between words.
    - Hebrew and Arabic are written right-to-left; Chinese is sometimes top-to-bottom.
  - Google has a translation engine and provides its interface in many languages.
  - Current research question: How to detect the language(s) of a page?
Life of a Query

1. The user enters a query on a web form sent to the Google web server.

2. The web server sends the query to the Index Server cluster, which matches the query to documents.

3. The match is sent to the Doc Server cluster, which retrieves the documents to generate abstracts and cached copies.

4. The list, with abstracts, is displayed by the web server to the user, sorted (using a secret formula involving PageRank).
Searching Habits

Google's Zeitgeist has interesting statistics about people's searches by logging the search queries!

http://www.google.com/press/zeitgeist.html

Languages used to access Google (March 2001 - August 2003)

## Searching Habits (continued)

### Top Ten Gaining Queries (Week Ending 9/29/03)

| 1.  | robert palmer               |
| 2.  | george plimpton             |
| 3.  | japan earthquake            |
| 4.  | edward said                 |
| 5.  | tsunami                     |
| 6.  | the bachelor                |
| 7.  | rosh hashanah               |
| 8.  | hurricane juan              |
| 9.  | do not call                 |
| 10. | dannii minogue              |

### Top Ten Declining Queries (Week Ending 9/29/03)

| 1.  | hurricane isabel            |
| 2.  | noaa                        |
| 3.  | kim bordenave               |
| 4.  | emmys                       |
| 5.  | champions league            |
| 6.  | anna lindh                  |
| 7.  | canadian idol               |
| 8.  | jennifer lopez              |
| 9.  | wesley clark                |
| 10. | kate beckinsale             |

### Top Ten Brand Names Searched: (Year, 2002)

| 1.  | Ferrari                     |
| 2.  | Sony                        |
| 3.  | Nokia                       |
| 4.  | Disney                      |
| 5.  | Ikea                        |
| 6.  | Dell                        |
| 7.  | Ryanair                     |
| 8.  | Microsoft                   |
| 9.  | Porsche                     |
| 10. | HP                          |
The Hub-and-Authority Framework

- Linear-algebraic interlude for technically minded students.
- NOT required for the exam!
- Introduced simultaneously with Google’s PageRank.
  - Like PageRank, uses “wisdom” implied by WWW links.
  - Like PageRank, has provable mathematical properties.
  - Specific algorithm differs from that of PageRank.
- Invented by Jon Kleinberg, then at IBM, now at Cornell.
"Abundance" Problem


- Given a query find:
  - Good Content ("Authorities")
  - Good Sources of Links ("Hubs")
- Mutually Reinforcing
- Simple (Core) Algorithm
\[ T \overset{\wedge}{=} \{n \text{ Pages}\}, \ A \overset{\wedge}{=} \{\text{Links}\} \]

\( X_p \in \mathbb{R}^2_0, p \in T \)  \hspace{1cm} \text{non-negative “Authority Weights”}

\( Y_p \in \mathbb{R}^2_0, p \in T \)  \hspace{1cm} \text{non-negative “Hub Weights”}

**I** operation  \hspace{1cm} \text{Update Authority Weights}

\[ X_p \leftarrow \sum_{(q,p) \in A} Y_q \]

**O** operation  \hspace{1cm} \text{Update Hub Weights}

\[ Y_p \leftarrow \sum_{(p,q) \in A} X_q \]

**Normalize:**  \hspace{1cm} \sum_{p \in T} X_p^2 = \sum_{p \in T} Y_p^2 = 1
Core Algorithm

\[ Z \leftarrow (1,1,\ldots,1) \]

\[ X \leftarrow Y \leftarrow Z \]

Repeat until Convergence

Apply I /* Update Authority weights */
Apply O /* Update Hub Weights */
Normalize

Return Limit \((X^*, Y^*)\)
Convergence of 

\((X_i, Y_i) \xrightarrow{\triangle} (OI)_i(Z,Z)\) 

\(A \triangleq n \times n \text{ "Adjacency Matrix"}\)

Rewrite I and O:

\[ X \leftarrow A^T Y \quad ; \quad Y \leftarrow AX \]

\[ X^i = (A^TA)^{i-1} A^T Z \quad ; \quad Y^i = (AA^T)^i Z \]

\(AA^T\) Symm., Non-negative and \(Z = (1,1,\ldots, 1) \Rightarrow \)

\[ X^* \triangleq \lim_{i \to \infty} X^i = \omega_1(A^TA) \]

\[ Y^* \triangleq \lim_{i \to \infty} Y^i = \omega_1(-AA^T) \]
Whole Algorithm \( (k,d,c) \)

\[
q \Rightarrow \text{Search Engine} \Rightarrow \vert S \vert \leq k
\]

Base Set \( T \):

\( \text{(In } S, S \to , \to S \text{) and } \leq d \text{ links/page} \)

Remove “Internal Links”

Run Core Algorithm on \( T \)

From Result \((X,Y)\), Select

- \( C \) pages with max \( X^* \) values
- \( C \) pages with max \( Y^* \) values
Examples (k= 200, d=5)

q = censorship + net
www.EFF.org
www.EFF.org/BlueRib.html
www.CDT.org
www.VTW.org
www.ACLU.prg

q = Gates
www.roadahead.com
www.microsoft.com
www.ms.com/corpinfo/bill-g.html

[Compares well with Yahoo!, Galaxy, etc.]
Approach to “Massiveness”: Throw Out Most of G!!

• Non-principal Eigenvectors correspond to “Non-principal Communities”
• Open (?):
  Objective Performance Criteria
  Dependence on Search Engine
  Nondeterministic Choice of S and T