Information Retrieval and Web Search Engines
Lecture 12: Link Analysis
July 6, 2011

Wolf-Tilo Balke and Joachim Selke
Institut für Informationssysteme
Technische Universität Braunschweig

Exercise 9.1
What is a Web crawler? What is its basic mode of operation?

Exercise 9.2
What features does a good Web crawler have?

Exercise 9.3
Why do crawlers need to check for duplicate URIs? How do they do it?

Exercise 9.4
Why do crawlers need to check for duplicate content? How does shingling work and what problems does it solve?

Exercise 9.5
What do you need focused crawling for? How does it work?
A typical Web search engine:

**User interface**

**Today's topic**

**Web crawler**

**Indexer**

**Retrieval algorithms**

**User**

**The Web**

**An Overview of Web Retrieval**

Networks of social interactions are formed...
- Between academics by co-authoring
- Between movie personnel by directing and acting
- Between people making phone calls
- Between people transmitting infections
- Between scientific papers through citations
- Between countries via trading relations
- Between musicians, soccer stars, friends, and relatives
- And, of course, between Web pages through links...

Lecture 12: Link Analysis

1. Link Structures
2. PageRank
3. HITS
Models of Social Networks

- It has been quite common for decades to model social networks using directed graphs:

<table>
<thead>
<tr>
<th>A</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Directed graph

Adjacency matrix

$A_{ij} = 1$ if and only if node $i$ links to node $j$

The Recursive Nature of Prestige

- Using the graph model, it has been clear that in-degree is a good first-order indicator of prestige.

- In 1949, the sociologist John R. Seeley realized the recursive nature of prestige in a social network:
  - A person’s status is a function of the status of those who choose him.
  - And their status is a function of those who choose them.
  - And so ad infinitum...

A Model of Prestige

- Seeley modeled prestige as follows:
  - Every node $u$ has a notion of prestige $p(u)$ associated with it, which is simply a positive real number.
  - Recursiveness constraint: The prestige of each node $u$ should be proportional to the total sum of prestige of all nodes that link to $u$, i.e.,
    $$p(u) = \alpha \cdot \sum_{v \in \text{in-neighbors}(u)} p(v)$$
  - Over all nodes, we represent the prestige score as a real column vector $p$ having exactly one entry for each node.
  - Equivalent fixpoint condition:
    $$p = \alpha \cdot A \cdot p$$

- This approach fits well to ideas from linear algebra (later).

A Model of Prestige (2)

- Another interesting notion is centrality.
- Definitions:
  - The distance $d(u, v)$ between two nodes $u$ and $v$ in a directed graph is the smallest number of links via which one can go from $u$ to $v$.
  - The radius of a node $u$ is $r(u) = \max_{v \in \text{in-neighbors}(u)} d(u, v)$, i.e., the distance to $u$'s most distant node.
  - The center of the graph is arg min$_{u \in V}$ $r(u)$, i.e., the node that has the smallest radius.
Boyack et al. (2005) visualized similarity data based on co-citations created from over 1 million journal articles published in 2000.

• The scientific citation graph:
  - Link a paper $u$ to a paper $v$, i.e. set $u \to v$, if $u$ cites $v$
  - Papers having a small radius are likely to be very influential

• The scientific collaboration graph:
  - Link two authors $u$ and $v$, i.e. set $u \leftrightarrow v$, if they co-authored a paper
  - The Erdős number of an author $u$ is his/her distance to the famous mathematician Paul Erdős

Another important measure is co-citation
• If document $u$ cites documents $v$ and $w$, then $v$ and $w$ are said to be co-cited by $u$
• If documents $v$ and $w$ are co-cited by many documents, then $v$ and $w$ are somehow related to each other

In terms of the adjacency matrix $A$:
- Link a document $u$ to a paper $v$, i.e. set $u \to v$, if $u$ cites $v$
- The number of documents co-citing $v$ and $w$ is the entry corresponding to $v$ and $w$ in the matrix $A^2$:
  \[
  A^2[v, w] = \sum_x A[v, x] A[x, w] = \sum_x A[u, v] A[u, w] = |\{u | u \to v \text{ and } u \to w\}|
  \]

There are many other notions of centrality, e.g., cuts:
• A cut is a (usually small) number of edges that, when removed, disconnect a given pair of vertices
  - One may look for a small set of vertices that, when removed, will decompose the graph into two or more connected components
  - This is useful for the study of epidemics, espionage, or suspected terrorist communication on telephone networks

• The entry in the $A^2$ matrix corresponding to $[v, w]$ is the co-citation index of $v$ and $w$ and a measure of relatedness between $v$ and $w$
• One may use this pairwise relatedness measure in a clustering algorithm, such as multidimensional scaling
  - MDS is similar to the singular value decomposition
  - It uses a similarity matrix to embed the documents into a low-dimensional Euclidean space (e.g., a plane)
  - Visualizing clusters based on co-citation reveals important social structures between and within link communities

Information Retrieval and Web Search Engines — Wolf-Tilo Balke and Joachim Selke — Technische Universität Braunschweig

Back to the Web

• Classical IR:
  - The worth of a document with regard to a query is intrinsic to the document
  - Documents are self-contained units, and are generally descriptive and truthful about their contents

• Modern Web search:
  - Apply ideas from network analysis to the Web graph...
  - Links are recommendations
  - Anchor texts can be used as document descriptions
Assumption 1:
A hyperlink is a signal of quality or popular interest
– In some sense, a link is a democratic vote

Assumption 2:
The anchor text of a link (or its surrounding text) describes the target page

Both assumptions clearly do not hold always
– But exploiting them has proved to be much better than not exploiting them

We will concentrate on the first assumption:
“Links are quality signals”

Two highly popular algorithms:
– PageRank (Page et al., 1998)
– HITS (Kleinberg, 1999)

Before 1993:
– There are no search engines...
– Tim Berners-Lee maintains a list of Web servers:
  W3 servers
  – In Germany: LEO, “Link Everything Online”

1993–1998:
– Many new search engines, most popular:
  – Lycos, AltaVista, Excite, Inktomi, HotBot, Ask Jeeves
  – All of them mainly rely on classical IR techniques and focus on the problem of scaling
– Google is founded
– The first engine that heavily exploits the Web’s link structure
– Google’s success has a name: PageRank

1998–Today:
– Large companies try to keep up with Google
– Most noteworthy: Yahoo and Microsoft
The next big thing in Web search?

- Clustering?
- Natural language query processing?
- The “Semantic Web”?
- Advanced personalization?
- Open source search engines?
- Metasearch that works?
- Federated search?
- Innovative user interfaces?
- Something else?

Lecture 12: Link Analysis

1. Link Structures
2. PageRank
3. HITS

Problem:
- How to assign a query-independent measure of prestige to each Web resource?

A good but infeasible solution:
- Rank Web resources by their popularity (measured by traffic?)

The PageRank solution:
- Apply John R. Seeley’s model of prestige to the Web graph!
- The number of in-links is correlated to a resource’s prestige.
- Links from good resources should count more than links from bad ones

The Random Surfer Model

Imagine a Web surfer doing a random walk on the Web:

- 90% of the time, the surfer clicks a random hyperlink
- 10% of the time, the surfer types in a random URI

PageRank = The long-term visit rate of each node

This is a crude, but useful, Web surfing model
- No one chooses links with equal probability, surfing usually is topic-driven
- How to surf to a random page?
- What about the back button or bookmarks?

A more detailed version of the model:
1. Start at a random page, chosen uniformly
2. Flip a coin that shows “tails” with probability λ.
3. If the coin shows “heads” AND the current page has a positive out-degree:
   - Randomly follow one of the pages out-links
   - Continue at (2)
4. If the coin shows “tails” OR the current page has no out-links:
   - Surf to a random Web page, chosen uniformly

Example:

Set $\lambda = 0.25$
Convergence (3)

- If the surfer is at page 3 in step $t$
  - He/she will be at page 1 in step $t+1$ with a probability of 5%.
  - He/she will be at page 2 in step $t+1$ with a probability of 80%.
  - He/she will be at page 3 in step $t+1$ with a probability of 5%.
  - He/she will be at page 4 in step $t+1$ with a probability of 5%.
  - He/she will be at page 5 in step $t+1$ with a probability of 5%.

Convergence (4)

- In essence, a Markov chain is a probabilistic finite state machine.
- Knowledge about the current state of a Markov chain can be expressed by probability vectors of length $n$.
- Remember our example:
  - Knowing for sure that the current state of the chain is state $i$.
  - For example, $(0.2, 0.5, 0.3)$ means that the chain’s probability to be in the first, second, and third state is 20%, 50%, and 30%, respectively.

Convergence (5)

- Stochastic matrices are closely related to Markov chains:
  - A Markov chain consists of $n$ states and an $n \times n$ stochastic matrix $T$.
  - Each row and column of $T$ corresponds to a state, respectively.
  - At any point in time, the Markov chain is in exactly one of these states.
  - Time is discrete, i.e., it runs in discrete steps: $t = 0, 1, 2, ...$
  - From time step to time step, the chain’s current state changes according to the stochastic matrix $T$.

Convergence (2)

- State transitions can be formalized using matrix-vector multiplication.
- Let $T$ be a transition matrix and $p$ a probability vector that models the chain’s state probabilities at time $t$.
- What are the state probabilities $p'$ at time $t+1$?

**Example ($n = 2$):**

$$T = \begin{pmatrix}
0.5 & 0.5 \\
0.5 & 0.5
\end{pmatrix}$$

$$p = (p_1, p_2)$$

$$p' = \sum_{k=1}^{2} T_{1,k} \cdot p_k$$
PageRank

- In the random surfer model, there is a unique stationary probability vector $p$.
- Node $u$’s PageRank is its stationary probability $p[u]$.

\[
\begin{array}{c|cccccc}
  \text{Time} & 1 & 2 & 3 & 4 & 5 & \text{Avg} \\
  \hline 
  T=1 & 0.05 & 0.05 & 0.05 & 0.05 & 0.8 & \\
  T=2 & 0.11 & 0.29 & 0.27 & 0.25 & 0.09 & \\
  T=3 & 0.36 & 0.27 & 0.17 & 0.07 & 0.13 & \\
  T=4 & 0.36 & 0.23 & 0.13 & 0.11 & 0.13 & \\
\end{array}
\]

- This fits Seeley’s notion of prestige:
  \[ p(u) = \alpha \cdot \sum v p(v) \]

According to the Perron–Frobenius theorem from linear algebra, the following is true:
- Every stochastic matrix containing only positive entries has 1 as one of its eigenvalues.
- Furthermore, 1 is the largest eigenvalue of the matrix.
- There is only one eigenvector having the eigenvalue 1.
- Since we do a random teleport with probability $\lambda > 0$ in the random surfer model, this theorem applies.
- Therefore, we can be sure that there is a probability vector $p$ satisfying $p = T^T \cdot p$.
- Such a vector $p$ is called the Markov chain’s stationary probability vector.

PageRank Quiz

A Web graph:

Which of the following node lists is ordered by PageRank?

- a) E > B = D > A = C
- b) B = E = D > A = C
- c) E > D > B = A > C
- d) D > E > A = C > B
How to compute the PageRank?

A very simple method for eigenvalue and eigenvector computation is the so-called power iteration, which can be applied to any quadratic matrix $A$:

1. Start with an arbitrary initial vector $b_0$.
2. Set $i = 0$.
3. Set $b_{i+1} = A \cdot b_i$, i.e. normalize $b_{i+1}$ to unit length.
4. Set $b_{i+2} = b_{i+1} / |b_{i+1}|$, i.e. normalize $b_{i+1}$ to unit length.
5. Set $i = i + 1$.
6. GOTO 2.

One can prove that the power iteration converges to the eigenvector of $A$ having the largest eigenvalue.

In our case, the largest eigenvalue is 1.

The power iteration finds the stationary probability vector $p$.

How many iterations are needed?

Actually, the number is quite low since we don’t need a perfect result anyway.

How to compute the PageRank for a Web graph containing 60 billion nodes?

Use a highly scalable distributed algorithm.

Actually, this is one of Google’s secrets.

A search engine myth: “PageRank is the most important component of ranking”

The reality:

- There are several components that are at least as important: Anchor text, phrases, proximity, …
- Google uses hundreds of different features for ranking.
- There are rumors that PageRank in its original form (as presented here) has a negligible effect on ranking.
- However, variants of PageRank are still an essential part of ranking.
- Addressing link spam is difficult and crucial!

A disadvantage of PageRank is that it computes only a single overall score for each web resource.

- A web resource might be unimportant from a global view but highly important for a specific topic.

Topic-sensitive PageRank tries to address this issue:

- Define a set of popular topics (e.g. football, Windows, Obama).
- Use classification algorithms to assign each web resource to one (or more) of these topics.
- For each topic, compute a topic-sensitive PageRank by limiting the random teleports to pages of the current topic.
- At query time, detect the query’s topics and use the corresponding PageRank scores…
Comparison to PageRank (precision at 10):

- Eliminate navigational links:
  - Most web pages contain navigational structures
  - The quality assumption does only hold if a hyperlink was created as a result of editorial judgment
  - Therefore, navigational links should be removed before computing the PageRank

- Eliminate nepotistic links:
  - Nepotism = favoritism based on kinship
  - Links between pages authored by the same person also are problematic
  - Again, they should be removed before doing any computations
  - Unfortunately, it's much harder to detect them than detecting navigational links…

Possible Enhancements

- Google Toolbar: http://toolbar.google.com
- Web pages having the highest PageRank: http://www.seocompany.ca/pagerank/pr-10-pages.php

More Applications

- The PageRank can be used for crawling:
  - Decide how deep to crawl a web site
  - Decide how often to update a resource

- Other applications:
  - Many more…

HITS

- HITS stands for hyperlink induced topic search
- Invented by Jon Kleinberg

- Problem setting:
  - For any information need, there are hubs and authorities
  - Authority: Definitive high-quality information (query-dependent)
  - Hub: Comprehensive list of links to authorities (query-dependent)
  - To a certain degree, each page is a hub as well as an authority

- Task:
  - Given a query, estimate the degree of authority and hubness of each Web page
HITS (2)

• Obvious:
  The authority and hubness scores are query-dependent, therefore the computation has to be done at query time.

HITS (3)

• Idea (continued):
  Finally, compute hub and authority scores on the base set.

HITS (4)

- By combining both equations we arrive at:
  \[ a = \alpha \beta A^T \cdot h \]
  \[ h = \beta \cdot A \cdot a \]

HITS (5)

Example (query: Japan elementary schools):

- Authority scores:
  - The authority score of a page is proportional to the sum of hub scores of the pages linking to it.

HITS (6)

- As PageRank, HITS has been patented:
  - US patent 6,112,202
  - "Method and system for identifying authoritative information resources in an environment with content-based links between information resources"
  - Inventor: Jon Kleinberg
  - Assignee: IBM

Connection to LSI/SVD

- There is a direct mapping between finding the singular value decomposition of \( A \) and finding an eigen-decomposition of \( AA^T \) and \( A^T A \).
- A short recap from Lecture 4:
  - Let \( A = USV \) be the SVD of \( A \).
  - The singular vectors of \( A \) are the eigenvectors of \( AA^T \) and \( A^T A \);
  - The matrix \( S \) contains the corresponding eigenvalues.
  - Similarly, \( Y \)'s rows are the eigenvectors of \( A^T A \).

- Therefore, HITS is equivalent to running the SVD on the adjacency matrix of the base set.

If the query is ambiguous (e.g. "Java" or "jaguar") or polarized (e.g. "abortion" or "cold fusion"), the base set will contain a few, almost disconnected, link communities.

Then, the principal eigenvectors found by HITS will reveal hubs and authorities in the largest link community.

One can tease of this structure by computing not only the principal eigenvectors but some more.

PageRank can be precomputed, HITS has to be computed at query time.

- HITS is very expensive

Different choices regarding the formal model

- HITS models hubs and authorities
- HITS uses a subset of the Web graph
- But: We could also apply PageRank to a subset and HITS on the whole Web graph...

On the Web, a good hub usually is also a good authority.

The difference between HITS and PageRank is not that large...

Extensions

Next Lecture

- Spam detection
- Metasearch
- Privacy issues