Link analysis

- The web as a directed graph
  - Set of web pages with associated textual content
  - Hyperlinks between webpages (potentially with anchor text)
    → Directed graph
- Our focus: Which pages are “relevant” (to a query)?
  - Analysis of link structure instrumental for web search
  - Assumption: incoming link is a quality signal (endorsement)
  - Page has high quality ≈ links from/to high-quality pages
    - (We are ignoring anchor text in this lecture.)
- Gives rise to HITS and PageRank algorithms
- Similarly: citations of scientific papers, social networks, ...
Outline

1. Background: Power Method
2. HITS
3. Background: Markov Chains
4. PageRank
5. Summary
Eigenvectors and diagonalizable matrices

- Denote by $A$ an $n \times n$ real matrix
- Recap eigenvectors
  - $v$ is a **right eigenvector** with eigenvalue $\lambda$ of $A$ if $Av = \lambda v$
  - $v$ is a **left eigenvector** with eigenvalue $\lambda$ of $A$ if $vA = \lambda v$
  - If $v$ is a right eigenvector of $A$, then $v^T$ is a left eigenvector of $A^T$ (and vice versa)
- $A$ is **diagonalizable** if it has $n$ linearly independent eigenvectors
  - Some matrices are not diagonalizable (called **defective**)
  - If $A$ is symmetric (our focus), it is diagonalizable
  - If $A$ is symmetric, $v_1, \ldots, v_n$ can be chosen to be real and orthonormal
    - These eigenvectors then form an orthonormal basis of $\mathbb{R}^n$
  - Denote by $\lambda_1, \ldots, \lambda_n$ are the corresponding eigenvalues (potentially 0)
  - Then for every $x \in \mathbb{R}^n$, there exist $c_1, \ldots, c_n$ such that
    \[ x = c_1 v_1 + c_2 v_2 + \cdots + c_n v_n \]
  - And therefore
    \[ Ax = \lambda_1 c_1 v_1 + \lambda_2 c_2 v_2 + \cdots + \lambda_n c_n v_n \]
- Eigenvectors “explain” effect of linear transformation $A$
Example

\[ \lambda_1 v_1 \lambda_2 v_2 \]

\[ \lambda_1 = 2, \lambda_2 = 1 \]

\[ \tilde{x} = Ax \]
Power method

- Simple method to determine the largest eigenvalue \( \lambda_1 \) and the corresponding eigenvector \( \mathbf{v}_1 \)

Algorithm

1. Start at some \( \mathbf{x}_0 \)
2. While not converged
   1. Set \( \tilde{x}_{t+1} \leftarrow \mathbf{A} \mathbf{x}_t \)
   2. Normalize: \( \mathbf{x}_{t+1} \leftarrow \tilde{x}_{t+1}/\|\tilde{x}_{t+1}\| \)

- What happens here?
  - Observe that \( \mathbf{x}_t = \mathbf{A}^t \mathbf{x}_0 / C \), where \( C = \|\mathbf{A}^t \mathbf{x}_0\| \)
  - Assume that \( \mathbf{A} \) is real symmetric
  - Then \( \mathbf{x}_t = (\lambda_1^t c_1 \mathbf{v}_1 + \lambda_2^t c_2 \mathbf{v}_2 + \cdots + \lambda_n^t c_n \mathbf{v}_n) / C \)
  - If \( |\lambda_1| > |\lambda_2| \), then

\[
\lim_{t \to \infty} \frac{\lambda_2^t c_2}{\lambda_1^t c_1} = \lim_{t \to \infty} \left( \frac{\lambda_2}{\lambda_1} \right)^t \frac{c_2}{c_1} = 0
\]

- So as \( t \to \infty \), \( \mathbf{x}_t \) converges to \( \mathbf{v}_1 \)
Power method (example)

$n = 0$

$n = 1$

$n = 1$ (normalized)

$n = 2$

$n = 2$ (normalized)

$n = 100$
Discussion

- Easy to implement and parallelize

- We will see: useful for understanding link analysis

- Convergence
  - Works if $A$ is real symmetric, $|\lambda_1| > |\lambda_2|$, and $x_0 \not\perp v_1$ (i.e., $c_1 \neq 0$)
  - Speed depends on eigengap $|\lambda_1|/|\lambda_2|
  - Also works in many other settings (but not always)
Power method and singular vectors

- Unit vectors $u$ and $v$ are **left and right singular vectors** of $A$ if
  \[
  A^T u = \sigma v \quad \text{and} \quad Av = \sigma u
  \]
- $\sigma$ is the corresponding **singular value**
- The SVD decomposition is formed of the singular values ($\Sigma$) and corresponding left and right singular vectors (columns of $U$ and $V$)
- $u$ is an eigenvector of $AA^T$ with eigenvalue $\sigma^2$ since
  \[
  AA^T u = A\sigma v = \sigma Av = \sigma^2 u
  \]
- Similarly $v$ is an eigenvector of $A^T A$ with eigenvalue $\sigma^2$
- Power method for principal singular vectors
  1. $u_{t+1} \leftarrow Av_t / \|Av_t\|$
  2. $v_{t+1} \leftarrow A^T u_{t+1} / \|A^T u_{t+1}\|$
- Why does it work?
  - $AA^T$ and $A^T A$ are symmetric (and positive semi-definite)
  - $u_{t+2} = Av_{t+1} / \|Av_{t+1}\| = AA^T u_{t+1} / \|AA^T u_{t+1}\|$
Outline

1. Background: Power Method

2. HITS

3. Background: Markov Chains

4. PageRank

5. Summary
Asking Google for search engines

Web search engine - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Web_search_engine - Diese Seite übersetzen
A web search engine is a software system that is designed to search for information on the World Wide Web. The search results are generally presented in a line ...

Ixquick Search Engine
https://www.ixquick.com/
Ixquick search engine provides search results from over ten best search engines in full privacy. Search anonymously with Ixquick Search Engine!

Dogpile Web Search
www.dogpile.com/ - Diese Seite übersetzen
All the best search engines piled into one. Web; |; Images; |; Video; |; News; |; Local - White Pages. Search Results from: Google, Yahoo!, Yandex, And More.
White Pages - Make Dogpile Your Homepage - Local - About Dogpile

Search Engine Colossus: Find search engines from across the world
www.searchenginecolossus.com/ - Diese Seite übersetzen
International Directory of Search Engines. Giving you links to search engines from the USA, EU countries, Australia, Canada, China, India, Japan, Brazil, Russia, ...
Asking Bing for search engines

**Dogpile Web Search**
www.dogpile.com
Dogpile.com makes searching the Web easy, because it has all the best *search engines* piled into one. Go Fetch!

**AltaVista - Yahoo! Search - Web Search**
www.altavista.com
The *search engine* that helps you find exactly what you're looking for. Find the most relevant information, video, images, and answers from all across the Web.

**DuckDuckGo**
duckduckgo.com
Wir glauben an eine bessere Suche und echte Privatsphäre gleichzeitig. Wir erklären warum. Testen Sie DuckDuckGo für eine Woche.

**Ähnliche Suchvorgänge für search engine**

**Uncensored Search Engine**
**Rapidshare Search Engine**
**Torrent Search Engine**
**Ixquick Search Engine**
**Best Search Engine**
**Image Search Engine**
**Meta Search Engine**
**All Search Engines**

**Google**
www.google.com
*Search* the world's information, including webpages, images, videos and more. Google has many special features to help you find exactly what you're looking for.
Searching the WWW

- Some difficulties in web search
  - "search engine": many of the search engines do not contain phrase "search engine"
  - "Harvard": millions of pages contain "Harvard", but www.harvard.edu may not contain it most often
  - "lucky": there is an “I’m feeling lucky” button on google.com, but google.com is (probably) not relevant (popularity)
  - "automobile": some pages say “car” instead (synonymy)
  - "jaguar": the car or the animal? (polysemy)

- Query types
  1. Specific queries ("name of Michael Jackson’s dog") → Scarcity problem: few pages contain required information
  2. Broad-topic queries ("Java") → Abundance problem: large number of relevant pages
  3. Similar-page queries ("Pages similar to java.com")

- Our focus: broad-topic queries
  - Goal is to find “most relevant” pages
Hyperlink Induced Topic Search (HITS)

- HITS analyzes the link structure to mitigate these challenges
  - Uses links as source of exogenous information
  - Key idea: If $p$ links to $q$, $p$ confers “authority” on $q$
    - Try to find authorities through links that point to them
  - HITS aims to balance between relevance to a query (content) and popularity (in-links)

- HITS uses two notions of relevance
  - **Authority** page directly answers information need
    - Page pointed to by many hubs for the query
  - **Hub** page contains link to pages that answer information need
    - Points to many authorities for the query
  - Note: circular definition

- Algorithm
  1. Create a focused subgraph of the WWW based on the query
  2. Score each page w.r.t. to authority and hub
  3. Return the pages with the largest authority scores
Hubs and authorities (example)

- **hubs**
  - www.bestfares.com
  - www.airlinesquality.com
  - blogs.usatoday.com/sky
  - aviationblog.dallasnews.com

- **authorities**
  - www.aa.com
  - www.delta.com
  - www.united.com

Manning et al., 2008
Creating a focused subgraph

- **Desiderata**
  1. Should be small (for efficiency)
  2. Should contain most (or many) of the strongest authorities (for recall)
  3. Should be rich in relevant pages (for precision)

- Using all pages that contain query may violate (1) and (2)

- **Construction**
  - **Root set**: the highest-ranked pages for the query (regular web search)
    → Satisfies (1) and (3), but often not (2)
  - **Base set**: pages that point to or are pointed to from the root set
    → Increases number of authorities, addressing (2)
  - **Focused subgraph** = induced subgraph of base set
    → Consider all links between pages in the base set
Root set and base set

[Diagram showing the relationship between root set and base set with nodes and arrows]
Heuristics

- Retain efficiency
  - Focus on $t$ highest ranked pages for the query (e.g., $t = 200$)
    - Small root set
  - Allow each page to bring in at most $d$ pages pointing to it (e.g., $d = 50$)
    - Small base set ($\approx 5000$ pages)

- Try to avoid links that serve a purely navigational function
  - E.g., link to homepage
  - Keep transverse links (to different domain)
  - Ignore intrinsic links (to same domain)

- Try to avoid links that indicate collusion/advertisement
  - E.g., “This site is designed by...”
  - Allow each page to be pointed to at most $m$ times from each domain ($m \approx 4–8$)
Hubs and authorities

- Simple approach: rank pages by in-degrees in focused subgraph
  - Works better than on whole web
  - Still problematic: some pages are “universally popular” regardless of underlying query topic

- Key idea: weight links from different pages differently
  - Authoritative pages have high in-degree and a common topic
    → Considerable overlap in sets of pages that point to authorities
  - Hub pages “pull together” authorities on a common topic
    → Considerable overlap in sets of pages that are pointed to by hubs
  - Mutual reinforcement
    ★ Good hub points to many good authorities
    ★ Good authority is pointed to by many good hubs
Hub and authority scores

- Denote by $G = (V, E)$ the focused subgraph
- Assign to page $p$
  - A non-negative **hub weight** $u_p$
  - A non-negative **authority weight** $v_p$
- Larger means “better”
- Authority weight = sum of weights of hubs pointing to the page

$$v_p \leftarrow \sum_{(q,p) \in E} u_q$$

- Hub weight = sum of weights of authorities pointed to by the page

$$u_p \leftarrow \sum_{(p,q) \in E} v_p$$

- HITS iterates until it reaches a fixed point
  - Normalize vectors to length 1 after every iteration (does not affect ranking)
Example

\[ \mathbf{u} = (0.63, 0.46, 0.55, 0.29, 0.00, 0.00, 0.00)^T \quad \text{(hubs)} \]

\[ \mathbf{v} = (0.00, 0.00, 0.00, 0.21, 0.42, 0.46, 0.75)^T \quad \text{(authorities)} \]
Authorities for Chicago Bulls

0.85  www.nba.com/bulls
0.25  www.essex1.com/people/jmiller/bulls.htm
      “da Bulls”
0.20  www.nando.net/SportServer/basketball/nba/chi.html
      “The Chicago Bulls”
0.15  users.aol.com/rynocub/bulls.htm
      “The Chicago Bulls Home Page”
0.13  www.geocities.com/Colosseum/6095
      “Chicago Bulls”
Top-authority for Chicago Bulls

Manning et al., 2008
Hubs for Chicago Bulls

1.62  www.geocities.com/Colosseum/1778
      “Unbelieveabulls!!!!!”
1.24  www.webring.org/cgi-bin/webring?ring=chbulls
      “Erin’s Chicago Bulls Page”
0.74  www.geocities.com/Hollywood/Lot/3330/Bulls.html
      “Chicago Bulls”
0.52  www.nobull.net/web_position/kw-search-15-M2.htm
      “Excite Search Results: bulls”
0.52  www.halcyon.com/wordsld/bball/bulls.htm
      “Chicago Bulls Links”
What happens here?

- Adjacency matrix $A$ ($A_{pq} = 1$ if $p$ links to $q$)
  - $v_p \leftarrow \sum_{(q,p) \in E} u_q = (A*_{p})^T u$
  - Thus: $v \leftarrow A^T u$
  - Similarly $u \leftarrow A v$

- This is the power method for principal singular vectors
  - $u$ and $v$ correspond to principal left and right singular vectors of $A$
  - $u$ is principal eigenvector of $AA^T$ (co-citation matrix)
  - $v$ is principal eigenvector of $A^T A$ (bibliographic coupling matrix)

$$A = \begin{pmatrix}
0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$
Discussion

- Hub and authority weights depend on query
  → Scores need to be computed online

- HITS can find relevant pages regardless of content
  - Pages in base set often do not contain query keywords
  - Once base set is constructed, we only do link analysis

- Potential topic drift
  - Pages in base set may not be relevant to the topic
  - May also return Japanese pages for English query (if appropriately connected)

- Sensitive to manipulation
  - E.g., adversaries can create densely coupled hub and authority pages
Outline

1. Background: Power Method
2. HITS
3. Background: Markov Chains
4. PageRank
5. Summary
Markov chains

- A **stochastic process** is a family of random variables \( \{ X_t : t \in T \} \)
  - Here: \( T = \{ 1, 2, \ldots \} \) and \( t \) is called *time*
  - Thus we get sequence \( X_1, X_2, \ldots \)
  - Instance of a discrete-time stochastic process

\( \{ X_t \} \) is **Markov chain** if it is memory-less

\[
P(X_{t+1} = j \mid X_1 = i_1, \ldots, X_{t-1} = i_{t-1}, X_t = i) = P(X_{t+1} = j \mid X_t = i)
\]

- If \( X_t = i \), we say that Markov chain is in **state** \( i \) at time \( t \)

<table>
<thead>
<tr>
<th></th>
<th>( X_1 )</th>
<th>( X_2 )</th>
<th>( X_3 )</th>
<th>( X_4 )</th>
<th>( X_5 )</th>
<th>( X_6 )</th>
<th>( X_7 )</th>
<th>( X_8 )</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coin flips</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>MC</td>
</tr>
<tr>
<td>Invert</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>MC</td>
</tr>
<tr>
<td>First-one</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>MC</td>
</tr>
<tr>
<td>1 on odd time</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>MC</td>
</tr>
<tr>
<td>Sum</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>MC</td>
</tr>
<tr>
<td>Sum (2-window)</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>( \neg MC )</td>
</tr>
</tbody>
</table>
Finiteness and time-homogeneity

- Markov chain is **finite** if it has a finite number of states
- Markov chain is **time-homogeneous** if

\[ P(X_{t+1} = j \mid X_t = i) = P(X_1 = j \mid X_0 = i) \]

- We assume finite, time-homogeneous Markov chains from now on

<table>
<thead>
<tr>
<th>Properties</th>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$X_3$</th>
<th>$X_4$</th>
<th>$X_5$</th>
<th>$X_6$</th>
<th>$X_7$</th>
<th>$X_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coin flips</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Invert</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>First-one</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1 on odd time</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Sum</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Sum (2-window)</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>
Markov chains and graphs

Markov chains can be represented as graph

- \( V = \text{set of states} \)
- \((i, j) \in E \text{ if } P(X_1 = j \mid X_0 = i) > 0\)
- \( w_{ij} = P(X_1 = j \mid X_0 = i) \)

![Diagram of Markov chains and graphs]

<table>
<thead>
<tr>
<th></th>
<th>( X_1 )</th>
<th>( X_2 )</th>
<th>( X_3 )</th>
<th>( X_4 )</th>
<th>( X_5 )</th>
<th>( X_6 )</th>
<th>( X_7 )</th>
<th>( X_8 )</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coin flips</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Invert</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>MC, F, TH</td>
</tr>
<tr>
<td>First-one</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>MC, F, TH</td>
</tr>
</tbody>
</table>
Irreducibility and aperiodicity

A Markov chain is

- **irreducible**: for all $i, j \in V$, there is a path from $i$ to $j$
- **aperiodic**: for all $i$, $\gcd \{ t : P(X_t = i \mid X_0 = i) > 0 \} = 1$

### Properties

<table>
<thead>
<tr>
<th></th>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$X_3$</th>
<th>$X_4$</th>
<th>$X_5$</th>
<th>$X_6$</th>
<th>$X_7$</th>
<th>$X_8$</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coin flips</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0, 1, 3, 5, 6, 7</td>
</tr>
<tr>
<td>Invert</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0, 1, 3, 4, 5, 6, 7</td>
</tr>
<tr>
<td>First-one</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0, 1, 2, 3, 4, 5, 6, 7</td>
</tr>
</tbody>
</table>

- $MC$: Markov Chain
- $F$: First Return
- $TH$: Total Return
- $I$: Invariant
- $A$: Aperiodic
Transition matrix

- Consider the graph of a Markov chain
- Associated adjacency matrix $P$ is called **transition matrix**
  - $P$ is row-stochastic (rows sum to 1)

$$P = \begin{pmatrix}
0 & 0.9 & 0.1 \\
0.3 & 0.1 & 0.6 \\
0.5 & 0.5 & 0
\end{pmatrix}$$

![Graph with probabilities](image)
Surfing the chain

- $p_t = (p_{t,1} \cdots p_{t,n})$ = distribution of states after $t$ steps
  - i.e., $p_{t,i} = P(X_t = i)$
  - $p_0$ is initial distribution
- After one step, we have

$$p_{t+1,j} = \sum_i P(X_t = i)P(X_{t+1} = j \mid X_t = i) = \sum_i p_{t,i}P_{ij} = p_tP_{*j}$$

$$p_{t+1} = p_tP$$

- After $k$ steps, we have $p_{t+k} = p_tP^k$

\[ P = \begin{pmatrix} 0 & 0.9 & 0.1 \\ 0.3 & 0.1 & 0.6 \\ 0.5 & 0.5 & 0 \end{pmatrix} \]

\[ p_0 = (1 \ 0 \ 0) \]
\[ p_1 = (0 \ 0.9 \ 0.1) \]
\[ p_2 = (0.32 \ 0.13 \ 0.54) \]
\[ p_3 = (0.31 \ 0.57 \ 0.12) \]
Stationary distribution

- Distribution $\pi$ satisfying $\pi = \pi P$ is called **stationary distribution**
  - Distribution does not change if we make more steps
- Unique stationary distribution exists if chain is irreducible
- If additionally aperiodic, $\lim_{k \to \infty} p_0 P^k = \pi$ for any distribution $p_0$
  - This is just the power method
  - $\pi$ is the principal eigenvector of $P$
  - Corresponding eigenvalue is 1 and has multiplicity 1

$$P = \begin{pmatrix} 0 & 0.9 & 0.1 \\ 0.3 & 0.1 & 0.6 \\ 0.5 & 0.5 & 0 \end{pmatrix}$$

$$p_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad p_1 = \begin{pmatrix} 0 \\ 0.9 \\ 0.1 \end{pmatrix}, \quad p_2 = \begin{pmatrix} 0.32 \\ 0.13 \\ 0.54 \end{pmatrix}, \quad p_3 = \begin{pmatrix} 0.31 \\ 0.57 \\ 0.12 \end{pmatrix}, \quad \pi = \begin{pmatrix} 0.27 \\ 0.44 \\ 0.29 \end{pmatrix}$$
Outline

1. Background: Power Method
2. HITS
3. Background: Markov Chains
4. PageRank
5. Summary
A surfer
Random surfer (1)

- Consider a random surfer who
  1. Starts at a random web page
  2. Repeatedly clicks a random link to move to next web page
- PageRank is steady-state distribution of the random surfer
  - High PageRank = page frequently visited
  - Low PageRank = page infrequently visited
  - PageRank thus captures the “importance” of each webpage
- When is a page frequently visited?
  - When it has many in-links from frequently visited pages
- Still a circular definition, but now well-defined
Random surfer (2)

- Random surfer as a Markov chain
  - States = web pages
  - Transitions = normalized adjacency matrix (s.t. rows sum to 1)
  - Called walk matrix $D^{-1}W$
  - Note $L_{rw} = I - D^{-1}W$

**Pitfalls**
- How to handle dead ends? (there are many of them on the web)
- How to avoid getting stuck in subgraphs?

\[
W = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
\end{pmatrix}
\]

\[
D^{-1}W = \begin{pmatrix}
1/3 & 0 & 1/3 & 1/3 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 1/2 & 1/2 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
\end{pmatrix}
\]
A surfer with a problem
A surfer without a problem

Beam me up, Scotty
Teleportation

- A teleporting surfer
  - If no outgoing links, go to random site (handles dead ends)
  - With probability $\alpha$, teleport to a random site (handles subgraphs)
    - Can be thought of as typing URL into address bar
  - With probability $1 - \alpha$, follow random link
- Teleportation ensures irreducibility and aperiodicity
- PageRank of page $i = \pi_i$

$$W = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
\end{pmatrix}$$

$$P_{0.1} = \begin{pmatrix}
0.20 & 0.20 & 0.20 & 0.20 & 0.20 \\
0.32 & 0.02 & 0.32 & 0.32 & 0.02 \\
0.02 & 0.92 & 0.02 & 0.02 & 0.02 \\
0.02 & 0.47 & 0.47 & 0.02 & 0.02 \\
0.02 & 0.02 & 0.02 & 0.92 & 0.02 \\
\end{pmatrix}$$

$$\pi = \begin{pmatrix}
0.15 \\
0.36 \\
0.24 \\
0.20 \\
0.05 \\
\end{pmatrix}^T$$
Discussion

- PageRank is query-independent
  - Static, global ordering
- For web search, PageRank is one component of many
  - E.g., only pages satisfying the query are of interest
- Walks and teleportation can be done *non-uniformly*
  - Topic-specific PageRank
  - Personalized PageRank
  - Do not teleport to “dubious” websites (e.g., link farms)
Outline

1. Background: Power Method
2. HITS
3. Background: Markov Chains
4. PageRank
5. Summary
Lessons learned

- Link analysis exploits links structure for relevance assessment
  → We discussed HITS and PageRank

- Relevance score related to principal eigenvectors
  ▶ HITS: of co-citation and bibliographic coupling matrix
  ▶ PageRank: of walk matrix of a random, teleporting surfer
  ▶ Power method is simple way to compute these eigenvectors

<table>
<thead>
<tr>
<th>HITS</th>
<th>PageRank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distinguishes hubs and authorities</td>
<td>Single relevance score</td>
</tr>
<tr>
<td>Query dependent</td>
<td>Query independent</td>
</tr>
<tr>
<td>Computed online</td>
<td>Computed offline</td>
</tr>
<tr>
<td>Mutual reinforcement</td>
<td>Random surfer</td>
</tr>
<tr>
<td>No normalization</td>
<td>Out-degree normalization</td>
</tr>
<tr>
<td>(Was?) used by ask.com</td>
<td>google.com etc.</td>
</tr>
</tbody>
</table>
Suggested reading

- Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze
  *Introduction to Information Retrieval* (Chapter 21)
  Cambridge University Press, 2008

- Jon Kleinberg
  *Authoritative sources in a hyperlinked environment*

- Lawrence Page, Sergey Brin, Rajeev Motwani, Terry Winograd
  *The PageRank Citation Ranking: Bringing Order to the Web.*
  Technical Report, Stanford InfoLab, 1999
  http://ilpubs.stanford.edu:8090/422/