CSCE 670 - Information Storage and Retrieval

Lecture 6: Link Analysis (HITS and Topic-Sensitive PageRank)

Yu Zhang

yuzhang@tamu.edu

September 11, 2025

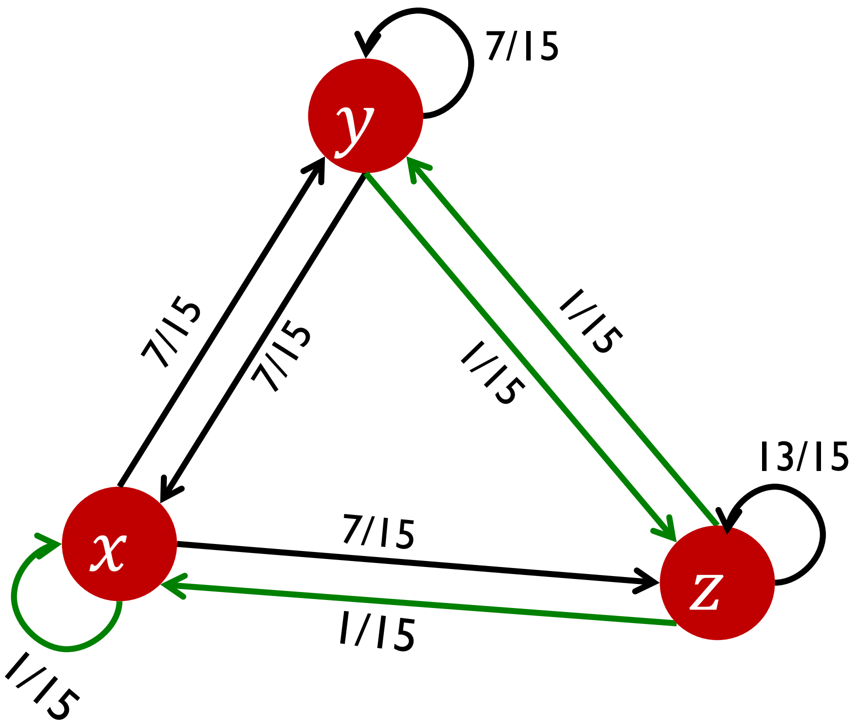
Course Website: https://yuzhang-teaching.github.io/CSCE670-F25.html

Adapted from the slides by Prof. Jure Leskovec (Stanford)

Recap: PageRank

Teleportation ($\beta = 0.8$):

$$\boldsymbol{A} = 0.8 \times \begin{bmatrix} 0 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1 \end{bmatrix} + 0.2 \times \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix}$$
$$= \begin{bmatrix} 1/15 & 7/15 & 1/15 \\ 7/15 & 7/15 & 1/15 \\ 7/15 & 1/15 & 13/15 \end{bmatrix}$$



Power Iteration:

	$\boldsymbol{r}^{(0)}$	$\boldsymbol{r}^{(1)}$	$\boldsymbol{r}^{(2)}$	$\boldsymbol{r}^{(3)}$	…	Finally
x	1/3	0.20	0.20	0.18	…	0.15
y	1/3	0.33	0.28	0.26	…	0.21
z	1/3	0.47	0.52	0.56	…	0.64

Our Plan: Ranking

- ✅ Why is ranking important?
- ✅ What factors impact ranking?
- Two foundational text-based approaches
  - ✅ TF-IDF
  - ✅ BM25
- Two foundational link-based approaches
  - ✅ PageRank (and some variants)
  - HITS
- Machine-learned ranking ("learning to rank")

HITS

- HITS (Hypertext-Induced Topic Selection) [Kleinberg, SODA'98]
  - Is a measure of webpage importance, similar to PageRank
  - Proposed at around same time as PageRank
- Goal: Say we want to find good newspapers
  - Don't just find newspapers.
  - Find "experts" – people who link in a coordinated way to good newspapers
- Idea: Links as votes
  - Page is more important if it has more links
  - In-coming links? Out-going links?

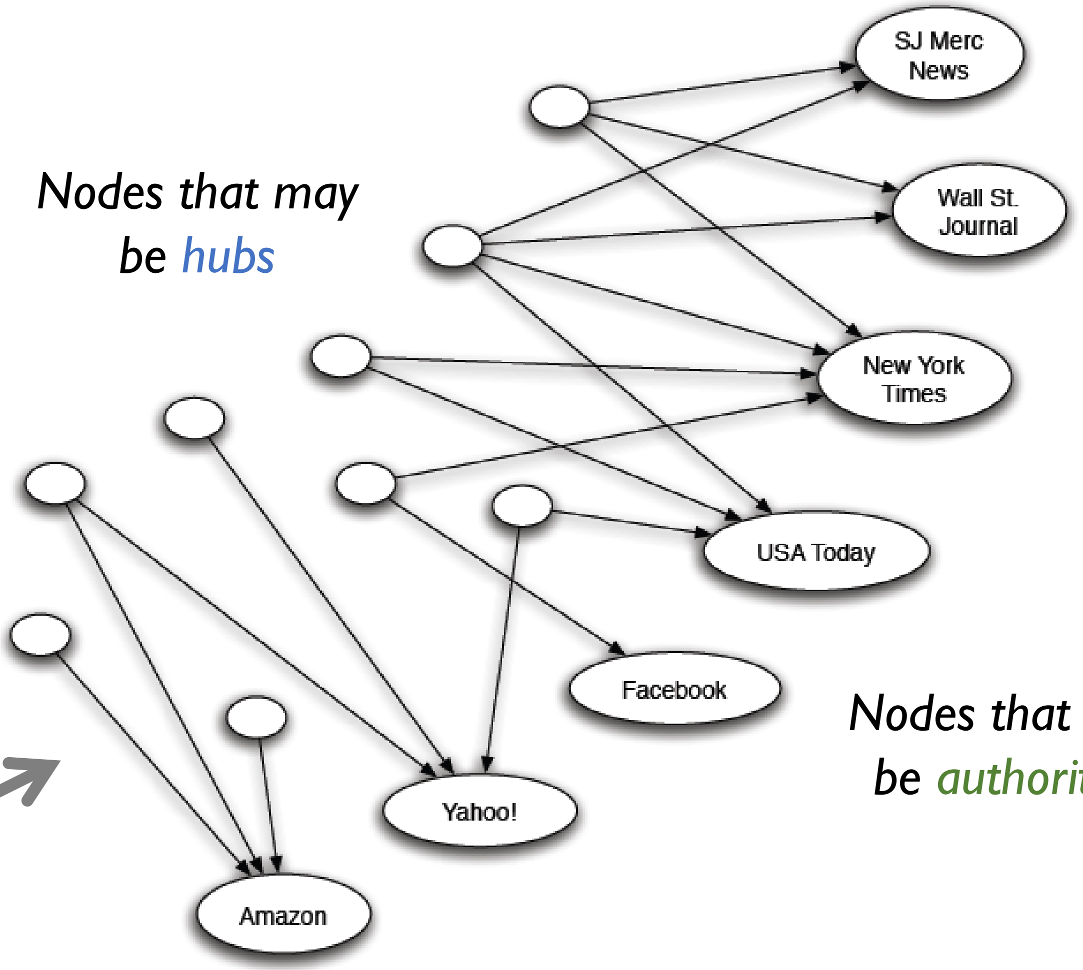
Finding Newspapers

- Each page has 2 scores
  - Quality as content (authority)
  - Quality as an expert (hub)

- Interesting pages fall into two classes:
  - Authorities are pages containing useful information
  - Hubs are pages that link to authorities

Note this is idealized example. In practice, the graph is not bipartite, and each page has both hub and authority scores.

Nodes that may be hubs



Nodes that may be authorities

Hubs and Authorities

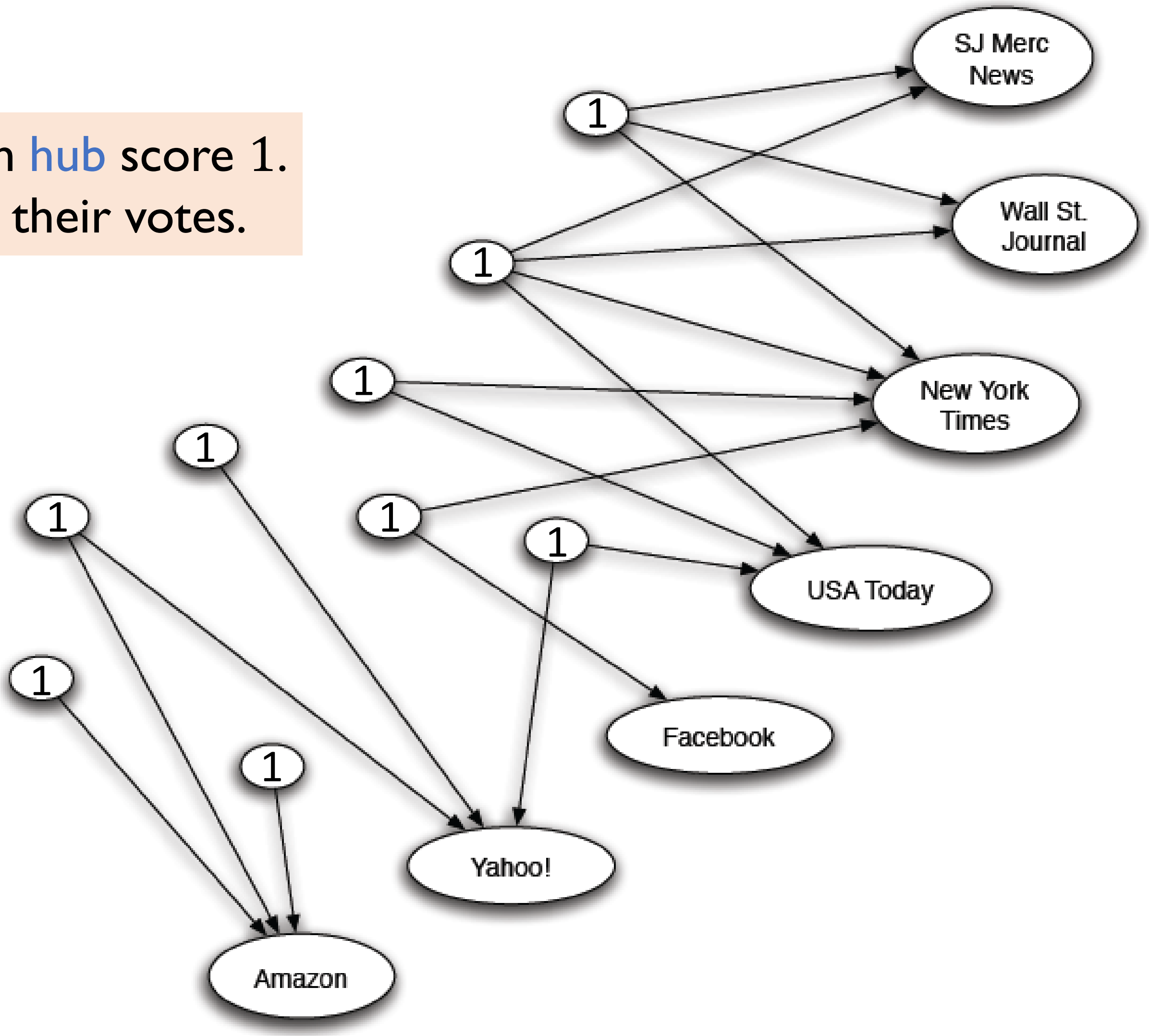
- Authorities are pages containing useful information
  - Newspaper homepages
  - Course homepages
  - Homepages of auto manufacturers

- Hubs are pages that link to authorities
  - List of newspapers
  - Course bulletin
  - List of US auto manufacturers

- Mutually recursive definition
  - A good hub links to many good authorities
  - A good authority is linked from many good hubs

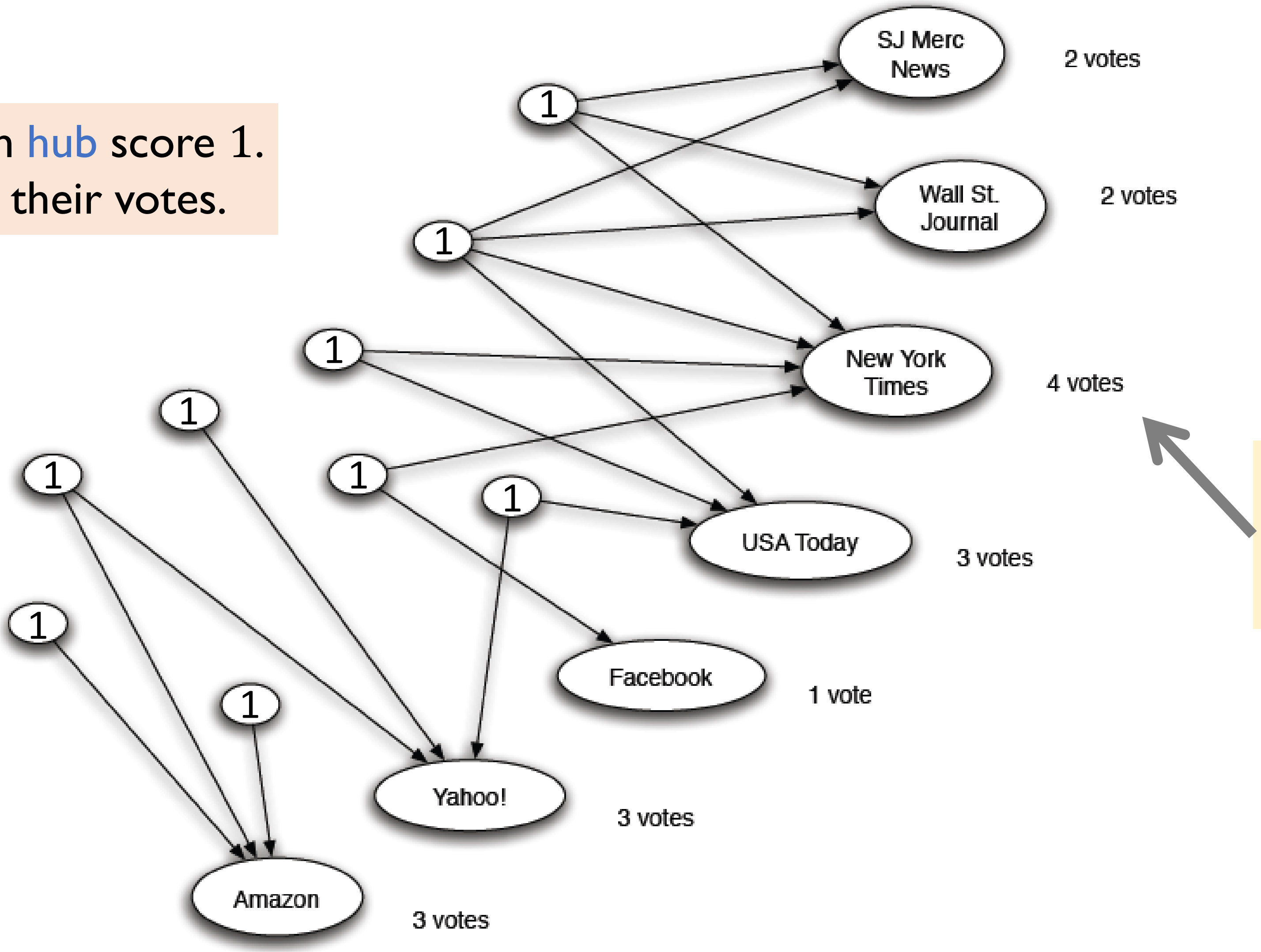
Principle of Repeated Improvement

Each page starts with hub score 1.
Authorities collect their votes.



Principle of Repeated Improvement

Each page starts with hub score 1.
Authorities collect their votes.

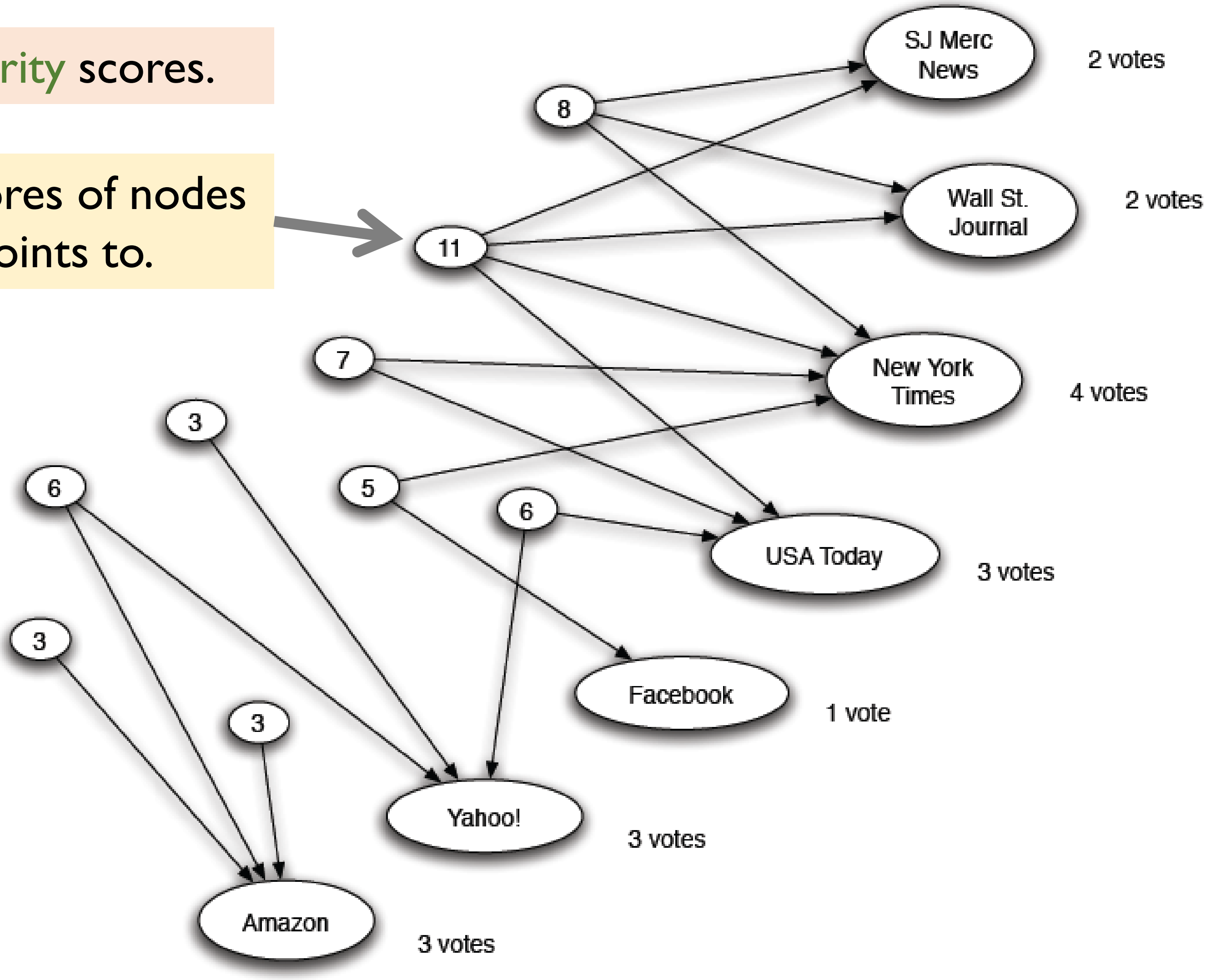


Sum of hub scores of nodes pointing to NYT

Principle of Repeated Improvement

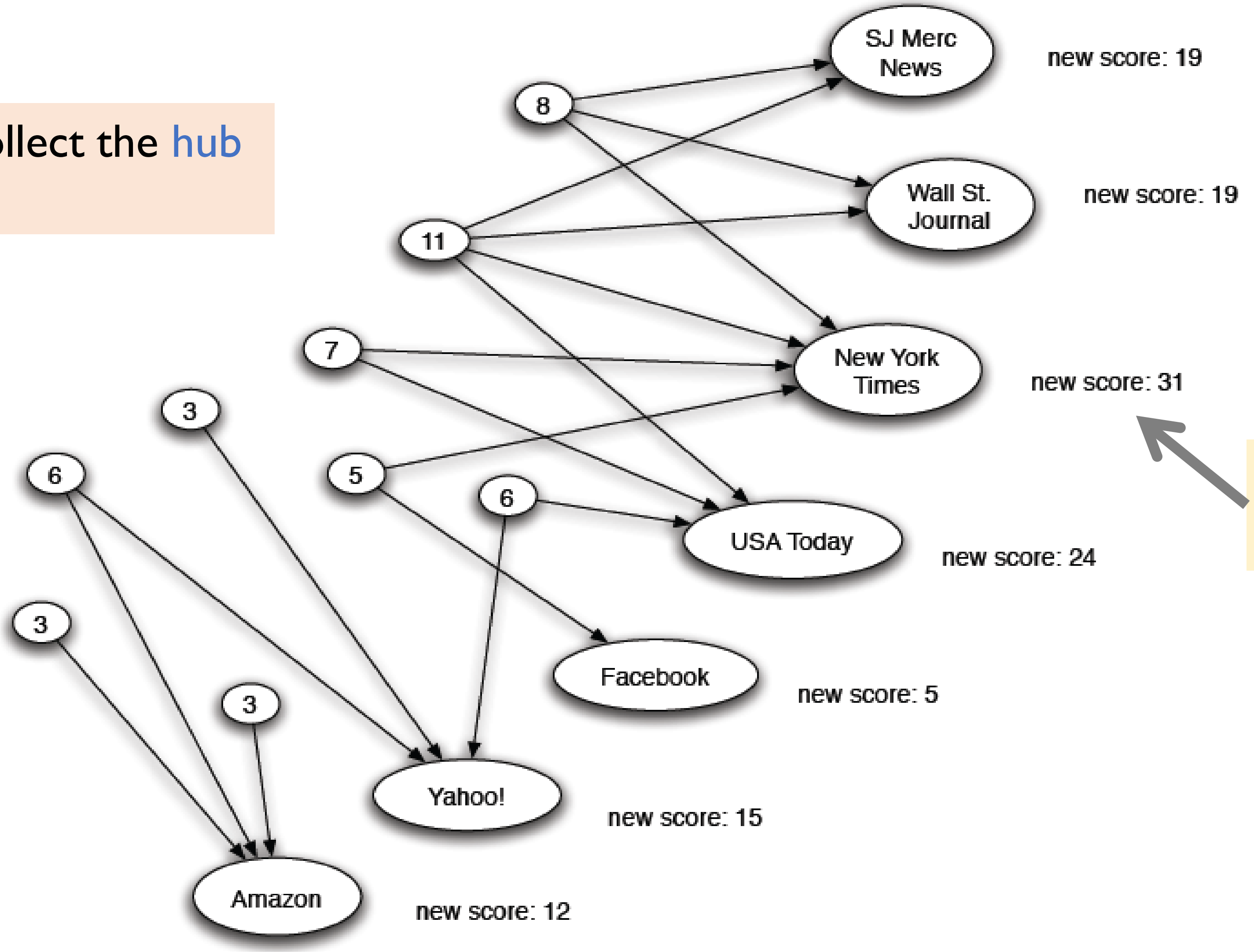
Hubs collect authority scores.

Sum of authority scores of nodes that the node points to.



Principle of Repeated Improvement

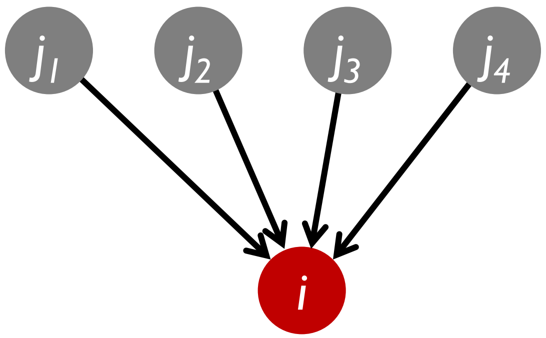
Authorities again collect the hub scores.



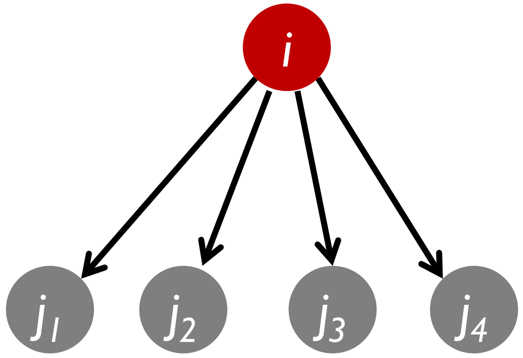
Updated score of NYT

HITS Algorithm: Formal Description

- Each page i has 2 scores:
  - Authority score: a_i
  - Hub score: h_i
- HITS algorithm
  - Initialize: $a_j^{(0)} = 1/\sqrt{N},\ h_j^{(0)} = 1/\sqrt{N}$
  - Then keep iterating until convergence:
    - $\forall i$, update the authority score: $a_i^{(t+1)} = \sum_{j\to i} h_j^{(t)}$
    - $\forall i$, update the hub score: $h_i^{(t+1)} = \sum_{i\to j} a_j^{(t)}$
    - $\forall i$, normalize: $\sum_i \left(a_i^{(t+1)}\right)^2 = 1, \sum_j \left(h_j^{(t+1)}\right)^2 = 1$



$$a_i = \sum_{j\to i} h_j$$



$$h_i = \sum_{i\to j} a_j$$

Matrix Version

- Notation:
  - Vectors $\boldsymbol{a} = \begin{pmatrix} a_1 \\ \cdots \\ a_n \end{pmatrix}$ and $\boldsymbol{h} = \begin{pmatrix} h_1 \\ \cdots \\ h_n \end{pmatrix}$ denote the authority/hub scores of all pages
  - Adjacency matrix $\boldsymbol{A}$, where $A_{ij} = \begin{cases} 1, & \text{if } i \to j \\ 0, & \text{otherwise} \end{cases}$
- Then, $h_i = \sum_{i \to j} a_j$ can be rewritten as $h_i = \sum_j A_{ij} a_j$
  - In other words, $\boldsymbol{h} = \boldsymbol{A}\boldsymbol{a}$
- Similarly, $a_i = \sum_{j \to i} h_j$ can be rewritten as $a_i = \sum_j A_{ji} h_j$
  - In other words, $\boldsymbol{a} = \boldsymbol{A}^T \boldsymbol{h}$

Matrix Version

- $\boldsymbol{h} = \boldsymbol{A}\boldsymbol{a}$
- $\boldsymbol{a} = \boldsymbol{A}^T\boldsymbol{h}$

- If we ignore the normalization step
  - $\boldsymbol{a} = \boldsymbol{A}^T\boldsymbol{h} = \boldsymbol{A}^T\boldsymbol{A}\boldsymbol{a}$
    - Power Iteration with the matrix $\boldsymbol{A}^T\boldsymbol{A}$
  - $\boldsymbol{h} = \boldsymbol{A}\boldsymbol{a} = \boldsymbol{A}\boldsymbol{A}^T\boldsymbol{h}$
    - Power Iteration with the matrix $\boldsymbol{A}\boldsymbol{A}^T$

Recall Power Iteration in PageRank

- Given the adjacency matrix $\boldsymbol{A}$,
  - The authority vector $\boldsymbol{a}$ we are looking for is an eigenvector of $\boldsymbol{A}^T\boldsymbol{A}$
  - The hub vector $\boldsymbol{h}$ we are looking for is an eigenvector of $\boldsymbol{A}\boldsymbol{A}^T$

Existence and Uniqueness

- Theorem: Under reasonable assumptions about $\boldsymbol{A}$, HITS converges to hub/authority vectors $\boldsymbol{h}^*$ and $\boldsymbol{a}^*$, where
  - $\boldsymbol{h}^*$ is the eigenvector of matrix $\boldsymbol{A}\boldsymbol{A}^T$ corresponding to its largest eigenvalue
  - $\boldsymbol{a}^*$ is the eigenvector of matrix $\boldsymbol{A}^T\boldsymbol{A}$ corresponding to its largest eigenvalue
- Proof (similar to PageRank but easier):
  - Both $\boldsymbol{A}\boldsymbol{A}^T$ and $\boldsymbol{A}^T\boldsymbol{A}$ are real symmetric matrices
    - The eigenvalues of a real symmetric matrix are all real numbers: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$
    - The eigenvectors of a real symmetric matrix are orthogonal to each other and form a basis of the entire vector space: $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N$
      - When considering eigenvectors of a real symmetric matrix, we often normalize $\boldsymbol{x}_i$ so that $\|\boldsymbol{x}_i\|^2 = \boldsymbol{x}_i{}^T\boldsymbol{x}_i = 1$
      - This explains why we use $1/\sqrt{N}$ for initialization and normalize the vectors to unit length after each iteration in HITS

Existence and Uniqueness

- Proof (Cont'd)
- $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N$ form a basis, so we can write $\boldsymbol{h}^{(\mathbf{0})} = c_1\boldsymbol{x}_1 + c_2\boldsymbol{x}_2 + \cdots + c_N\boldsymbol{x}_N$
- $\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{h}^{(\mathbf{0})} = \boldsymbol{A}\boldsymbol{A}^T(c_1\boldsymbol{x}_1 + c_2\boldsymbol{x}_2 + \cdots + c_N\boldsymbol{x}_N)$

$$= c_1\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{x}_1 + c_2\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{x}_2 + \cdots + c_N\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{x}_N$$

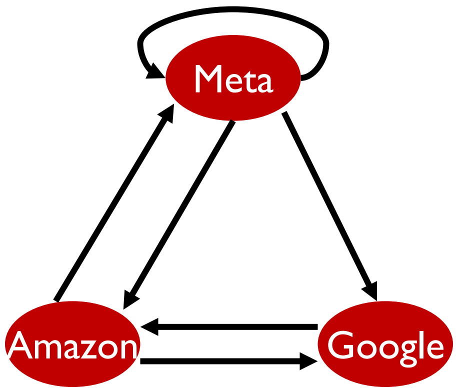
$$= c_1\lambda_1\boldsymbol{x}_1 + c_2\lambda_2\boldsymbol{x}_2 + \cdots + c_N\lambda_N\boldsymbol{x}_N$$

- Repeated multiplication on both sides
- $(\boldsymbol{A}\boldsymbol{A}^T)^k\boldsymbol{h}^{(\mathbf{0})} = c_1\lambda_1^k\boldsymbol{x}_1 + c_2\lambda_2^k\boldsymbol{x}_2 + \cdots + c_N\lambda_N^k\boldsymbol{x}_N$

$$= \lambda_1^k\left(c_1\boldsymbol{x}_1 + c_2\left(\frac{\lambda_2}{\lambda_1}\right)^k\boldsymbol{x}_2 + \cdots + c_N\left(\frac{\lambda_N}{\lambda_1}\right)^k\boldsymbol{x}_N\right)$$

$$\rightarrow \lambda_1^k c_1\boldsymbol{x}_1 \qquad \text{(when } k \rightarrow \infty\text{, if } \lambda_1 > \lambda_2\text{)}$$

Example



$$
\begin{array}{cccc}
 & \text{Meta} & \text{Amazon} & \text{Google} \\
\boldsymbol{A} = & \multicolumn{3}{c}{\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}}
\end{array}
$$

$$
\boldsymbol{A}^T = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}
$$

Hub	$\boldsymbol{h}^{(0)}$	$\boldsymbol{h}^{(1)}$	$\boldsymbol{h}^{(2)}$	$\boldsymbol{h}^{(3)}$	…	Finally
Meta	0.58	0.80	0.80	0.79	…	0.788
Amazon	0.58	0.53	0.53	0.57	…	0.577
Google	0.58	0.27	0.27	0.23	…	0.211

Authority	$\boldsymbol{a}^{(0)}$	$\boldsymbol{a}^{(1)}$	$\boldsymbol{a}^{(2)}$	$\boldsymbol{a}^{(3)}$	…	Finally
Meta	0.58	0.58	0.62	0.62	…	0.628
Amazon	0.58	0.58	0.49	0.49	…	0.459
Google	0.58	0.58	0.62	0.62	…	0.628

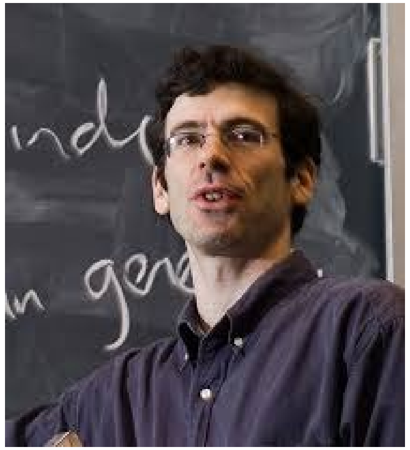
PageRank and HITS

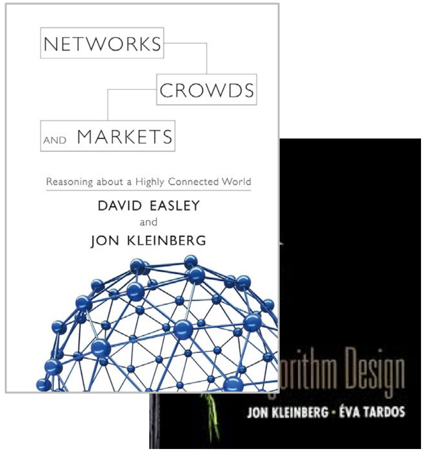
- PageRank and HITS are two solutions to the same problem:
  - How to identify important pages given the hyperlink graph of webpages?
- The destinies of PageRank and HITS after 1998 were very different

Sergey Brin

Larry Page

Co-founders of Google





Jon Kleinberg

Professor at Cornell University
Member of NAS and NAE

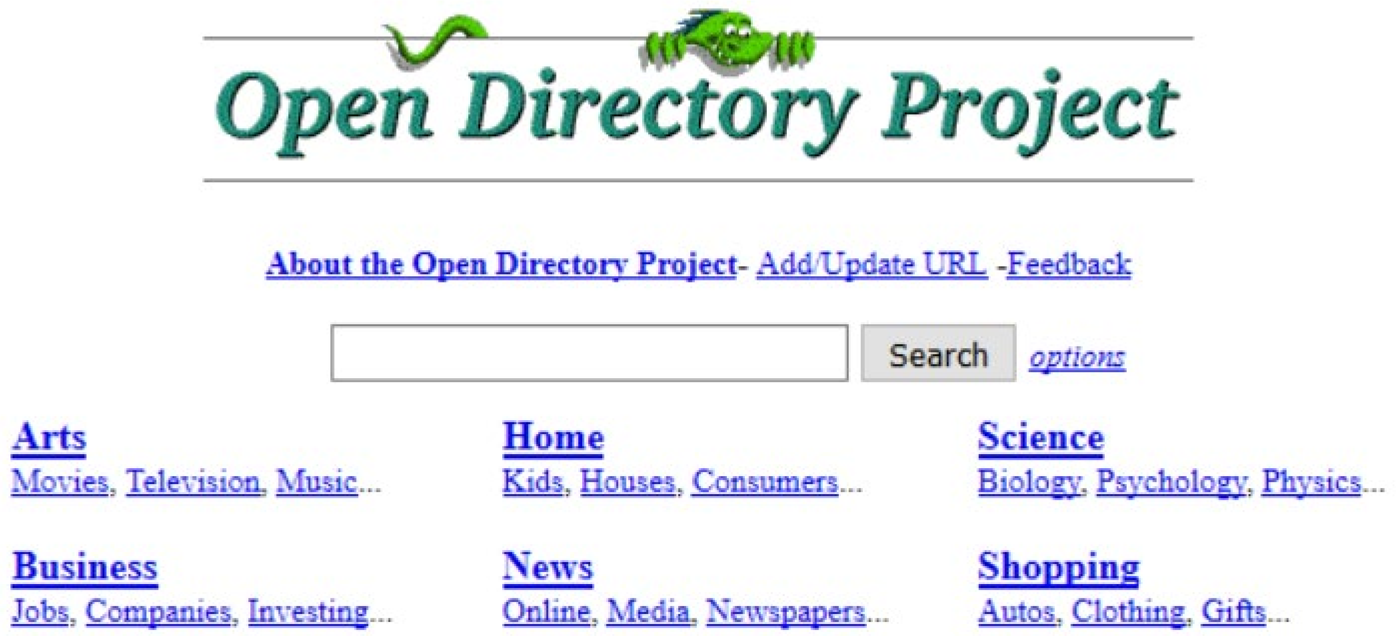
Questions?

Topic-Sensitive PageRank (a.k.a., Personalized PageRank)

- PageRank measures generic importance of a page
  - Can we measure page importance within a topic?
- Goal: Evaluate Web pages not just according to their popularity, but by how close they are to a particular topic, e.g., "*sports*" or "*history*"
  - Allow search queries to be answered based on interests of the user
- Idea: Modify the teleportation mechanism
  - Standard PageRank: The random surfer can teleport to any page with equal probability
    - To avoid dead-end and spider-trap problems
  - Topic-Sensitive PageRank: The random surfer can only teleport to a topic-specific set of "relevant" pages

Topic-Sensitive PageRank (a.k.a., Personalized PageRank)

- Topic-Sensitive PageRank: The random surfer can only teleport to a topic-specific set of "relevant" pages (denoted as S)
  - S contains only pages that are relevant to the topic
    - E.g., Open Directory (DMOZ) pages for a given topic/query



Matrix Formulation

- Standard PageRank

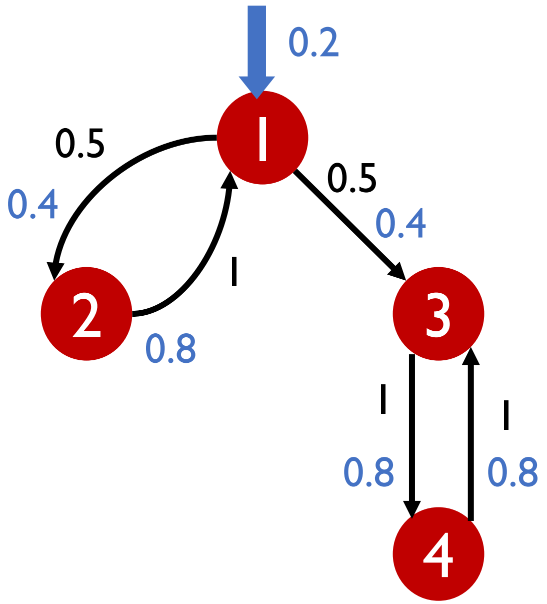
$$A_{ij} = \beta M_{ij} + (1-\beta)\frac{1}{N}, \qquad \forall \text{ pages } i, j$$

- Topic-Sensitive PageRank

$$A_{ij} = \begin{cases} \beta M_{ij} + (1-\beta)\frac{1}{|S|}, & \text{if } i \in S \\ \beta M_{ij}, & \text{otherwise} \end{cases}$$

- We weighted all pages in S equally
    - Could also assign different weights to pages!
- The computation is similar to that of standard PageRank
    - Power Iteration

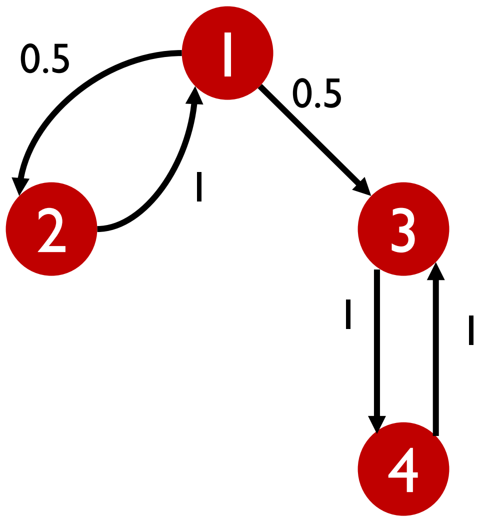
Example



Suppose $S = \{1\}$ and $\beta = 0.8$

	$r^{(0)}$	$r^{(1)}$	$r^{(2)}$	…	Finally
1	0.25	0.40	0.28	…	0.294
2	0.25	0.10	0.16	…	0.118
3	0.25	0.30	0.32	…	0.327
4	0.25	0.20	0.24	…	0.261

Example



$S = \{1\}$
$\beta = 0.9$

Node	Score
1	0.17
2	0.07
3	0.40
4	0.36

$S = \{1\}$
$\beta = 0.8$

Node	Score
1	0.29
2	0.12
3	0.33
4	0.26

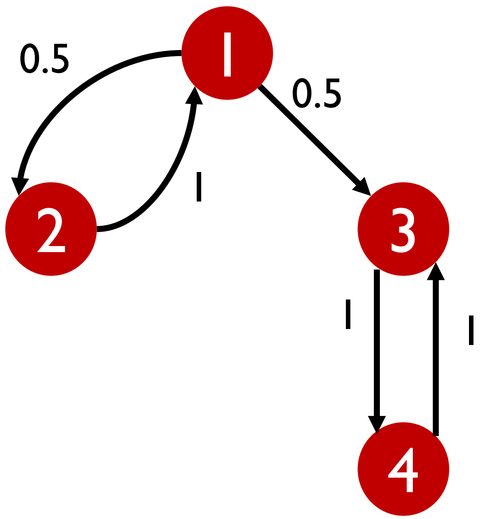
$S = \{1\}$
$\beta = 0.7$

Node	Score
1	0.39
2	0.14
3	0.27
4	0.19

Trend?

- The more you want to emphasize relevance to the topic node set S, the smaller you should set β.
  - A smaller β directs more votes $(1 - \beta)$ toward S in each iteration.
  - Drawback: The general importance of each page is also considered less

Example



$S = \{1\}$
$\beta = 0.8$

Node	Score
1	0.29
2	0.12
3	0.33
4	0.26

$S = \{1,2\}$
$\beta = 0.8$

Node	Score
1	0.26
2	0.20
3	0.29
4	0.23

$S = \{1,2,3\}$
$\beta = 0.8$

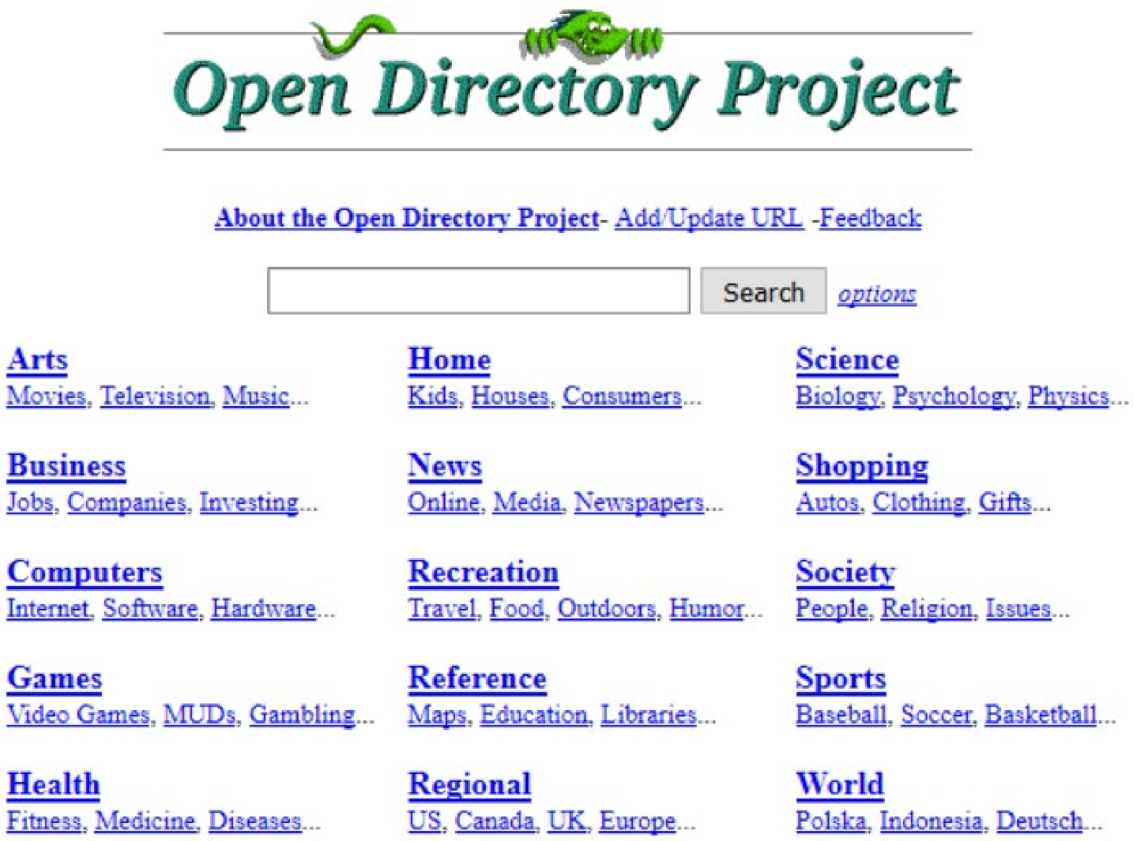
Node	Score
1	0.17
2	0.13
3	0.38
4	0.30

Trend?

- As S covers more nodes, relevance to the topic becomes increasingly less important.
- When S includes all nodes, topic-sensitive PageRank reduces to standard PageRank.

How to get *S*?

- The 15 DMOZ top-level categories:
  - arts, business, sports, …
  - Compute different PageRank scores for different topics
- Which topic ranking to use?
  - Users can pick from a menu
  - Classify the query into a topic
  - Query context, e.g., search history
  - User context, e.g., user's bookmarks



Questions?

Link Spamming

- Once Google became the dominant search engine, spammers began to work out ways to fool Google.
  - Imagine an "evil" user who, after creating his personal homepage, tries to manipulate its PageRank score to make it appear higher in people's search results.

- Spam farms were developed to concentrate PageRank on a single page.

- Link spam: Creating link structures that boost PageRank of a particular page

Link Spamming

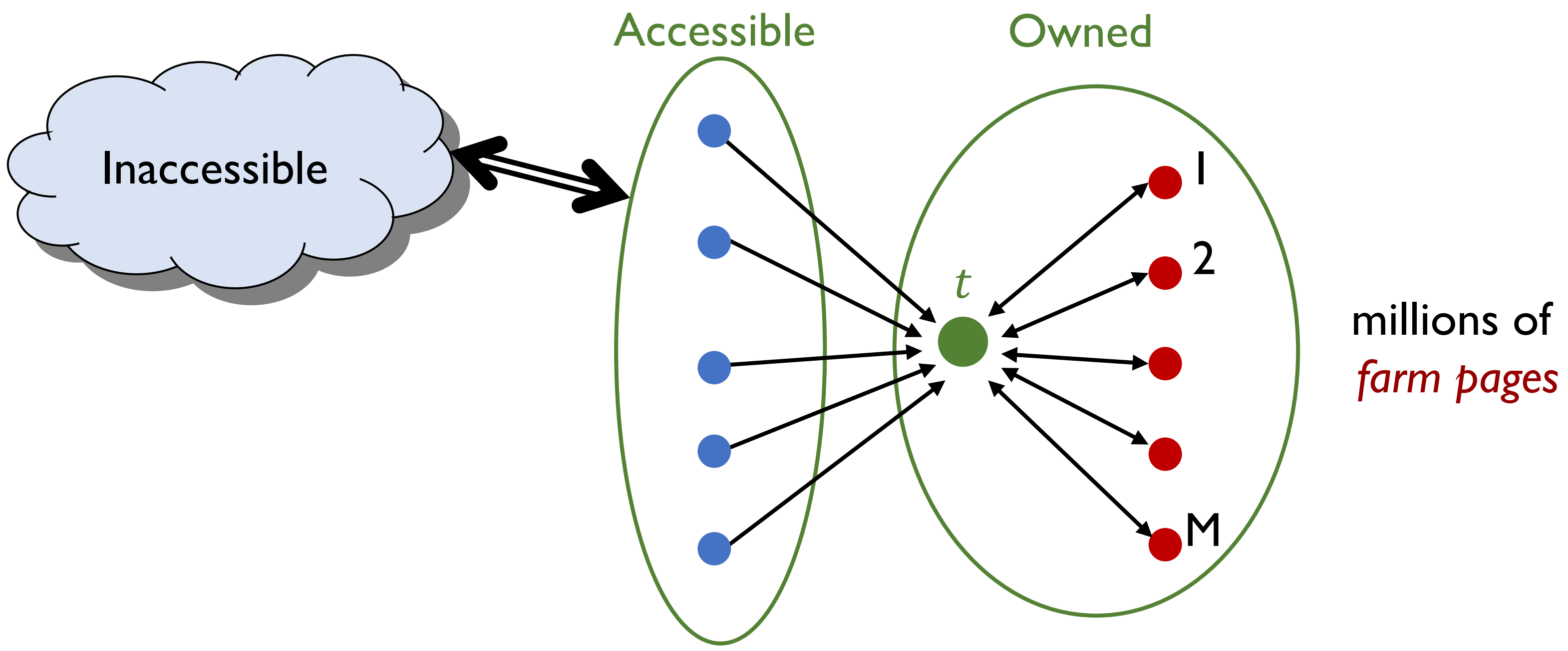
- Three kinds of web pages from a spammer's point of view
  - Inaccessible pages
    - E.g., official homepage of CNN
  - Accessible pages
    - E.g., social media comment pages
    - The spammer can post links to his pages
  - Owned pages
    - Completely controlled by spammer
    - E.g., register several new GitHub accounts, and use each account to create a personal homepage.

Reply: https://XXX.github.io

…

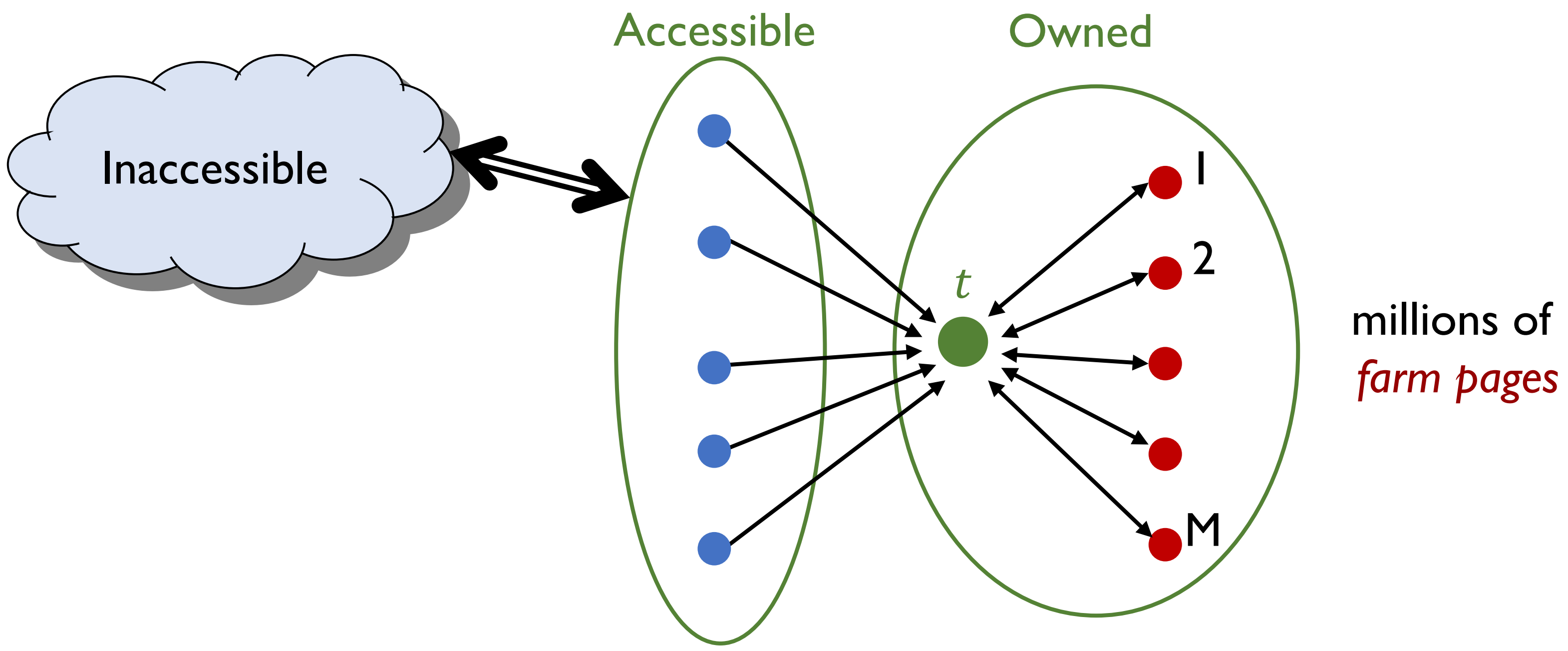
Link Farms

- Spammer's goal: Maximize the PageRank score of a target page t
- Technique:
  - Get as many links from accessible pages as possible to the target page t
  - Construct a "link farm" to get a PageRank multiplier effect



Analysis

- Let x be the PageRank score of the target page t
  - What is the PageRank score of each "farm" page? $\beta \frac{x}{M} + (1-\beta)\frac{1}{N}$
- Let y be the PageRank scores contributed by accessible pages to t
- So $x = y + \beta M \left[\beta \frac{x}{M} + (1-\beta)\frac{1}{N}\right] + (1-\beta)\frac{1}{N}$



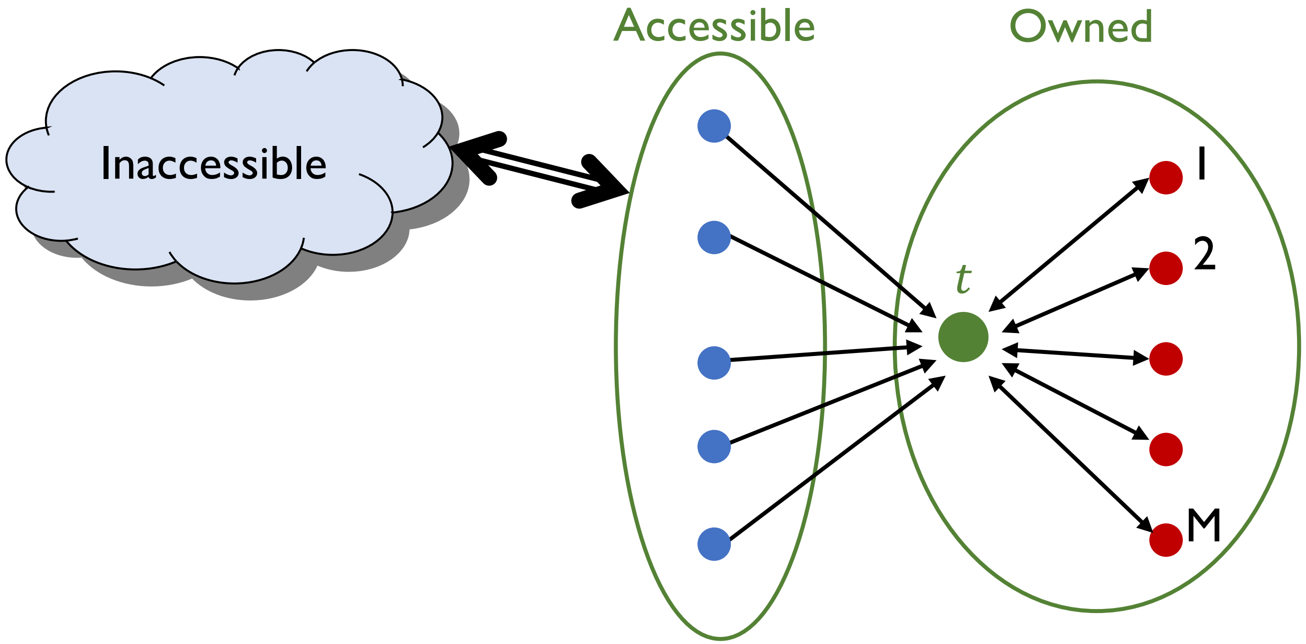
Analysis

- Let x be the PageRank score of the target page t
- $x = y + \beta M \left[\beta \frac{x}{M} + (1-\beta)\frac{1}{N}\right] + (1-\beta)\frac{1}{N}$

$$= y + \beta^2 x + \frac{\beta(1-\beta)M}{N} \boxed{+ (1-\beta)\frac{1}{N}}$$

very small, can be ignored

$$\Rightarrow \quad x = \frac{y}{1-\beta^2} + \frac{\beta}{1+\beta}\frac{M}{N}$$

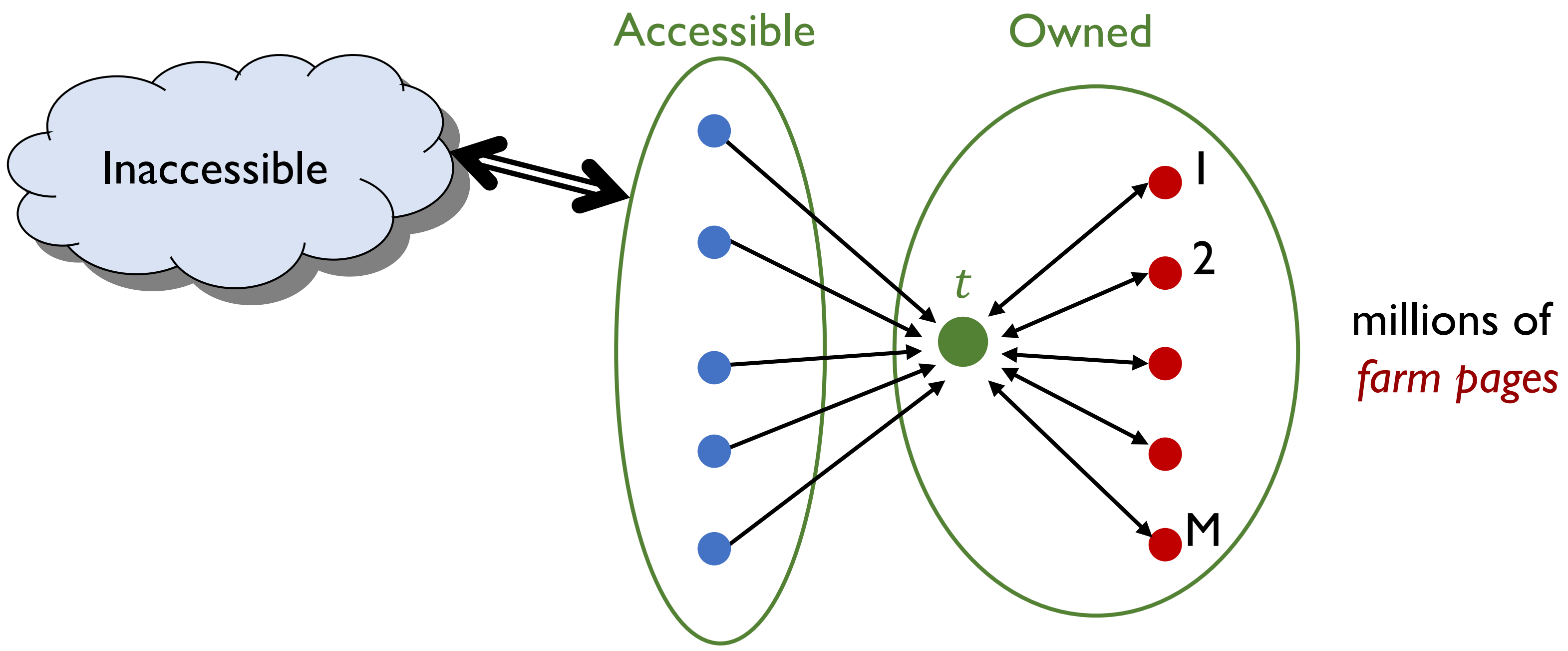


millions of *farm pages*

Analysis

$$x = \frac{y}{1-\beta^2} + \frac{\beta}{1+\beta}\frac{M}{N}$$

- If $\beta = 0.8$, then $x = 2.78y + 0.44\frac{M}{N}$
- By making M large, we can make x as large as we want



Extended Content
(will not appear in quizzes or the exam)

How to combat link spamming?

- Naïve Idea: detecting and blacklisting structures that look like spam farms
  - Leads to another war: hiding and detecting spam farms
- More Advanced Idea: Topic-Sensitive PageRank with teleportation to trusted pages
  - Example of trusted pages: *.edu* domains
- Step 1: Sample a set of seed pages from the web
  - Each page can be good (i.e., trusted) or bad (i.e., spam)
- Step 2: Ask humans to identify the good/bad pages in the seed set
  - An expensive task, so we must make seed set as small as possible

How to combat link spamming?

- Step 1: Sample a set of seed pages from the web
- Step 2: Ask humans to identify the good/bad pages in the seed set
- Step 3: Perform Topic-Sensitive PageRank with $S = \{\text{seed pages identified as good}\}$
  - Essentially propagate trust through links
  - Each page gets a trust value between 0 and 1

- Given a webpage, how to judge whether it is spam or not?
- Solution 1: Use a threshold value and mark all pages below the trust threshold as spam
  - Why should this work?
  - Are there cases where this may not work?

Why should Topic-Sensitive PageRank work here?

- Basic principle: Approximate isolation
  - It is rare for a trusted page to point to a spam page
- Trust attenuation: The degree of trust conferred by a trusted page decreases with the distance in the graph
- Trust splitting: The larger the number of out-links from a page, the less scrutiny the page author gives each out-link
  - Trust is split across out-links

How to pick the seed set?

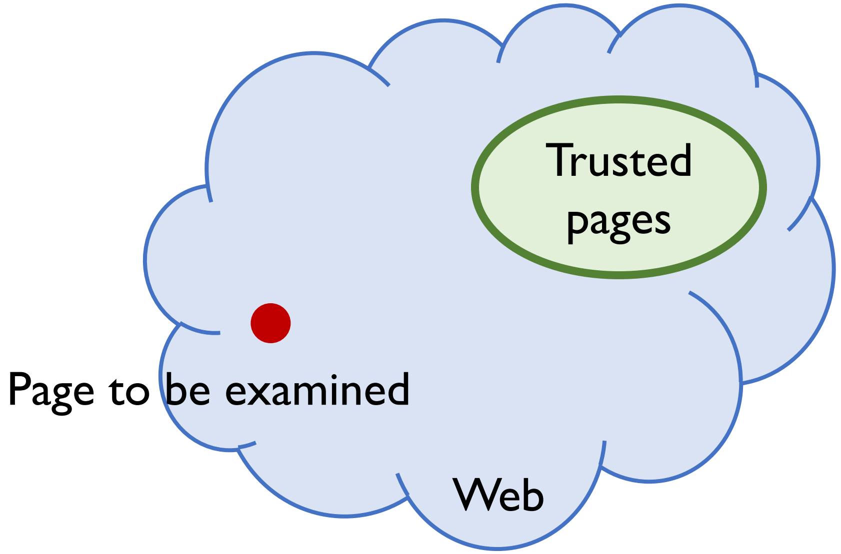
- Two conflicting considerations:
  - Humans have to inspect each seed page, so the seed set must be as small as possible
  - Must ensure every good page gets adequate trust rank, so need make all good pages reachable from seed set by short paths

- How to pick the seed set then?
  - PageRank: Pick the top k pages according to the standard PageRank score. The intuition is that you cannot get a bad page's rank really high
  - Use trusted domains whose membership is controlled, like *.edu*, *.mil*, and *.gov*

Spam Mass

- Solution 1: Use a threshold value and mark all pages below the trust threshold as spam
  - Are there cases where this may not work?
  - When will a node get a low Topic-Sensitive PageRank score?
    - Case 1: It is far away from S (i.e., trusted page)
    - Case 2: It has a low Standard PageRank score
      - This does not imply the node is a spam. Maybe it is just newly created.

- Solution 2: We can calculate what fraction of a page's PageRank comes from spam pages
  - In practice, we do not know all the spam pages, so we need to estimate.

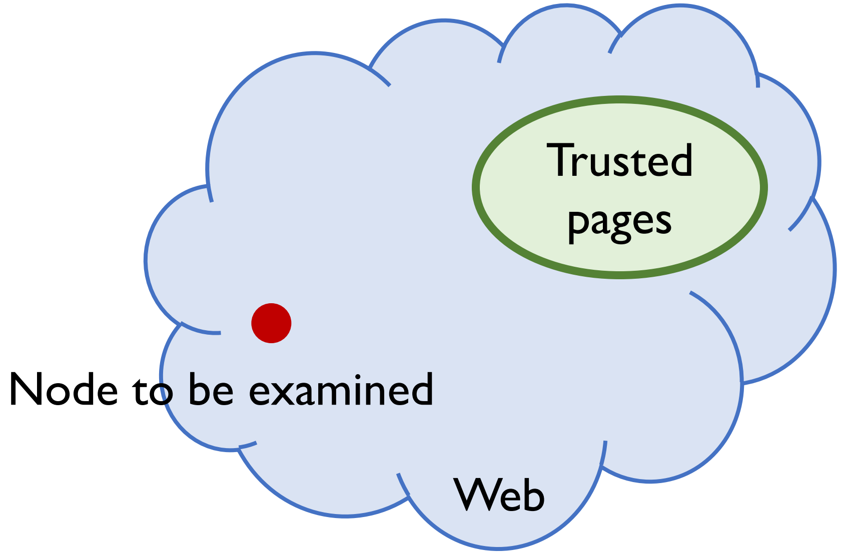


Spam Mass Estimation

- r_p = Standard PageRank score of page p
- r_p^+ = Topic-Sensitive PageRank of page p with teleportation into trusted pages only
  - r_p^+ may be small simply because r_p is small. We need to exclude this case.
- What fraction of a page's PageRank comes from spam pages?

$$r_p^- = r_p - r_p^+$$

- Spam mass of p is defined as $\frac{r_p^-}{r_p}$.
- Pages with high spam mass are judged as spam.



Thank You!

Course Website: https://yuzhang-teaching.github.io/CSCE670-F25.html