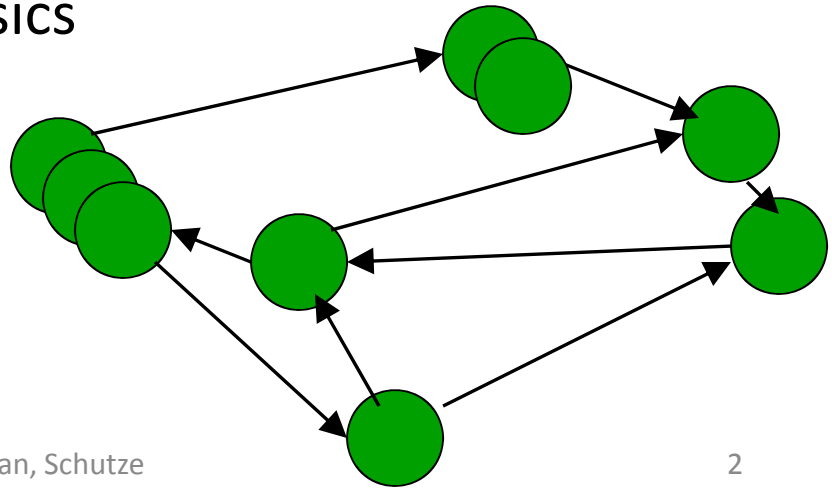
LINK ANALYSIS

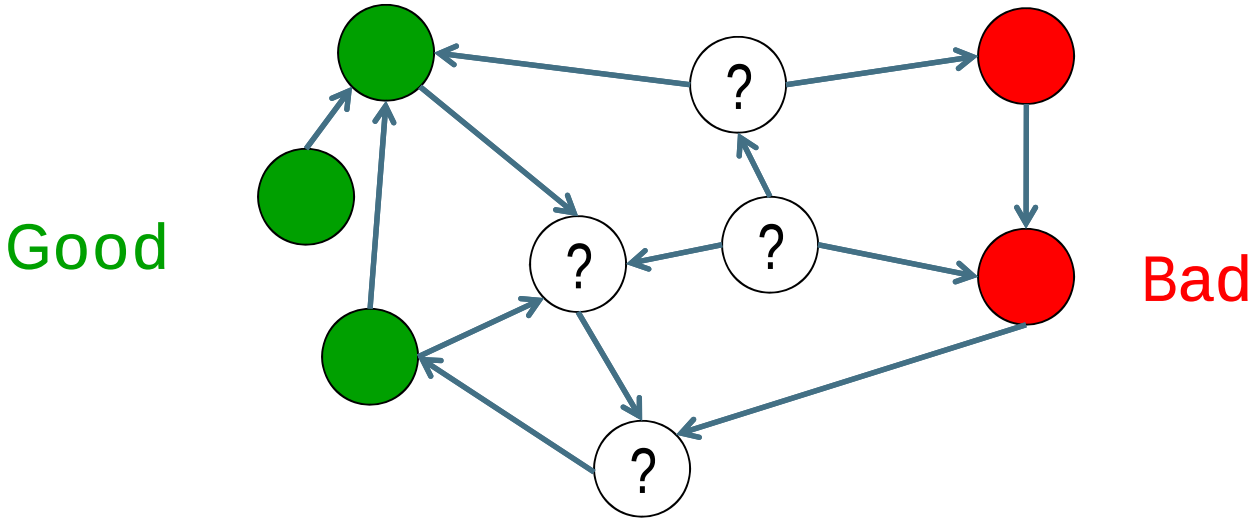
Today's lecture – hypertext and links

- We look beyond the *content* of documents
  - We begin to look at the hyperlinks between them
- Address questions like
  - Do the links represent a conferral of authority to some pages? Is this useful for ranking?
  - How likely is it that a page pointed to by the CERN home page is about high energy physics
- Big application areas
  - The Web
  - Email
  - Social networks



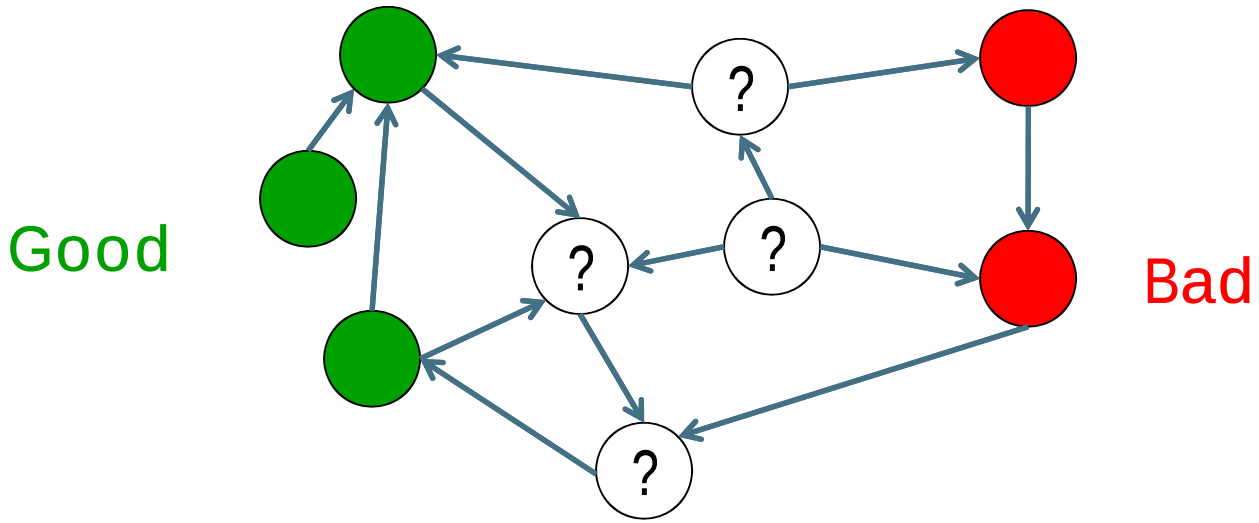
Links are everywhere

- Powerful sources of authenticity and authority
  - Mail spam – which email accounts are spammers?
  - Host quality – which hosts are "bad"?
  - Phone call logs
- The Good, The Bad and The Unknown



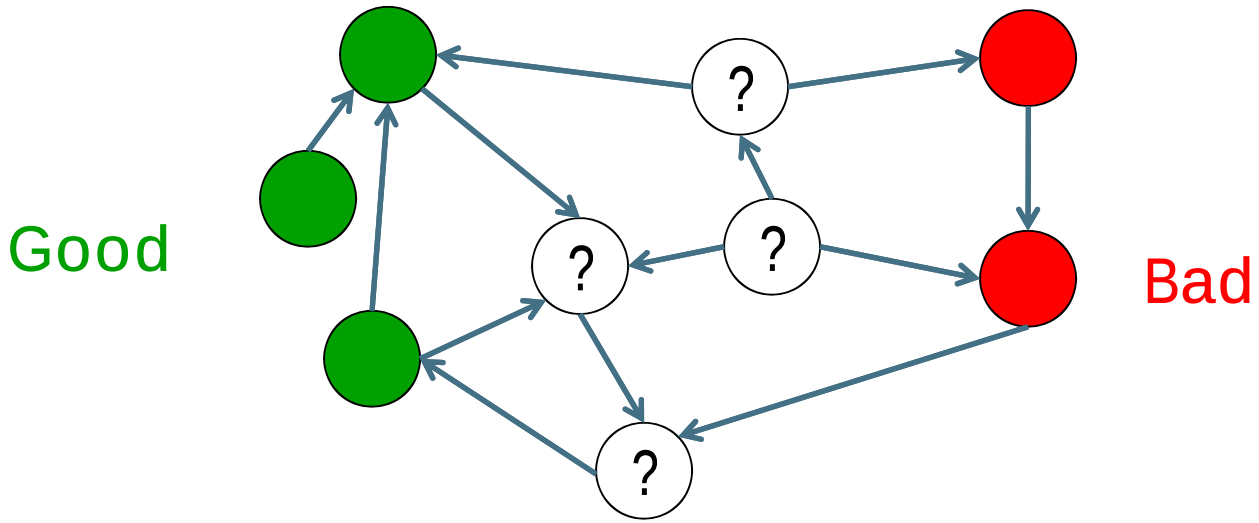
Simple iterative logic

- The Good, The Bad and The Unknown
  - Good nodes won't point to Bad nodes
  - All other combinations plausible



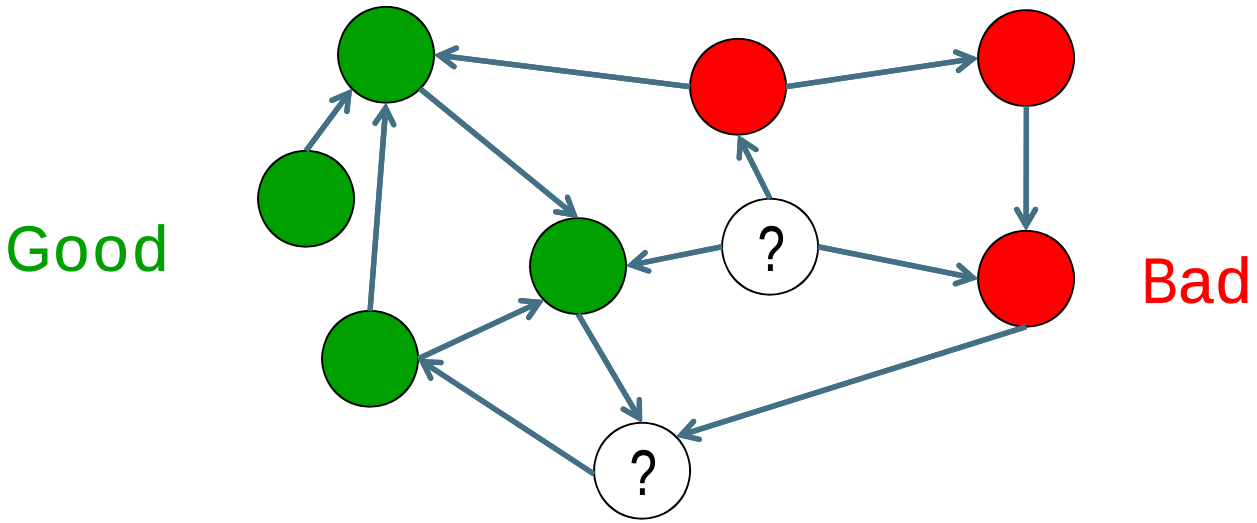
Simple iterative logic

- Good nodes won't point to Bad nodes
  - If you point to a Bad node, you're Bad
  - If a Good node points to you, you're Good



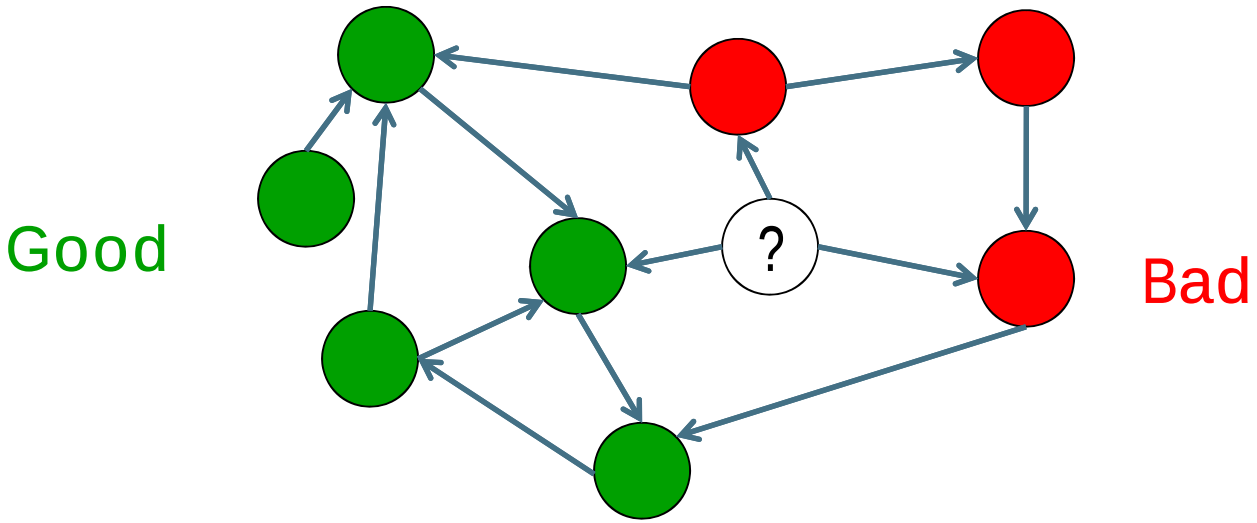
Simple iterative logic

- Good nodes won't point to Bad nodes
  - If you point to a Bad node, you're Bad
  - If a Good node points to you, you're Good



Simple iterative logic

- Good nodes won't point to Bad nodes
  - If you point to a Bad node, you're Bad
  - If a Good node points to you, you're Good



Sometimes need probabilistic analogs – e.g., mail spam

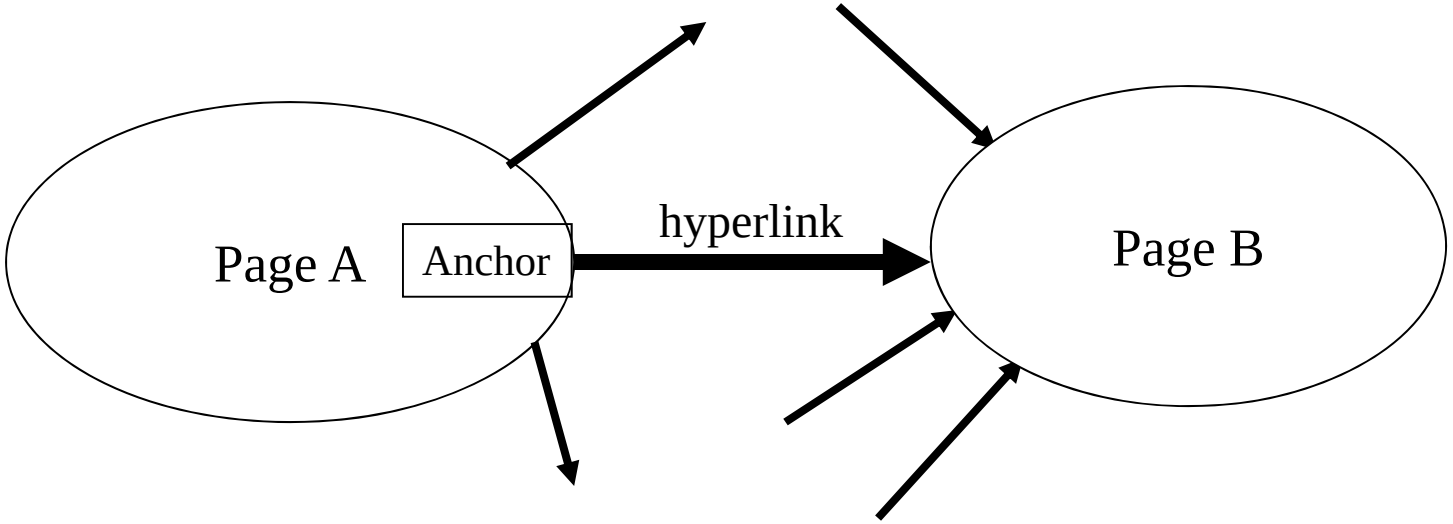
Many other examples of link analysis

- Social networks are a rich source of grouping behavior
- E.g., Shoppers' affinity – Goel+Goldstein 2010
  - Consumers whose friends spend a lot, spend a lot themselves
- http://www.cs.cornell.edu/home/kleinber/networks-book/

Our primary interest in this course

- Analogs of most IR functionality based purely on text
  - Scoring and ranking
  - Link-based clustering – topical structure from links
  - Links as features in classification – documents that link to one another are likely to be on the same subject
- Crawling
  - Based on the links seen, where do we crawl next?

The Web as a Directed Graph

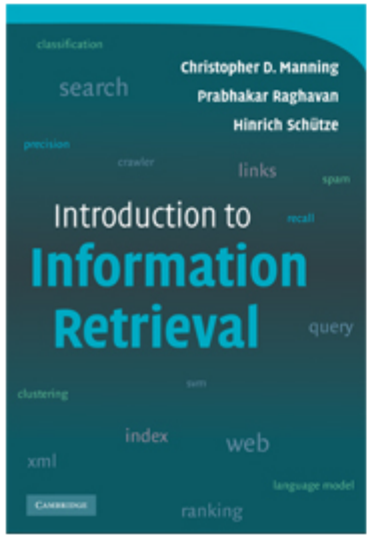


Assumption 1: A hyperlink between pages denotes a conferral of authority (quality signal)

Assumption 2: The text in the anchor of the hyperlink describes the target page (textual context)

Assumption 1: reputed sites

Introduction to Information Retrieval



This is the companion website for the following book.

Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, *Introduction to Informat*

You can order this book at CUP, at your local bookstore or on the internet. The best search

The book aims to provide a modern approach to information retrieval from a computer scie
University and at the University of Stuttgart.

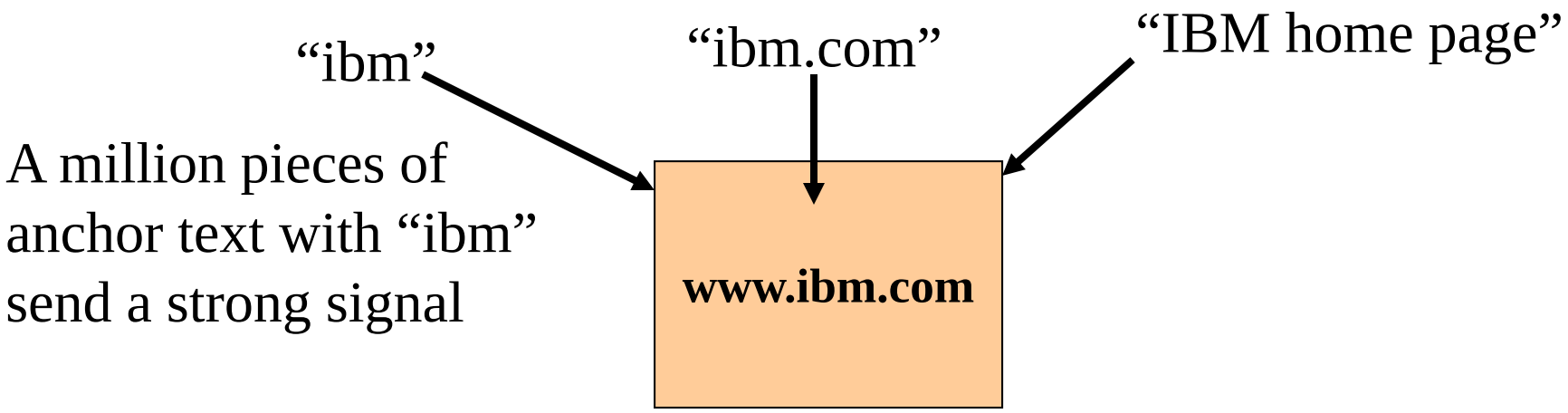
We'd be pleased to get feedback about how this book works out as a textbook, what is m
comments to: informationretrieval (at) yahoogroups (dot) com

Assumption 2: annotation of target

Anchor Text

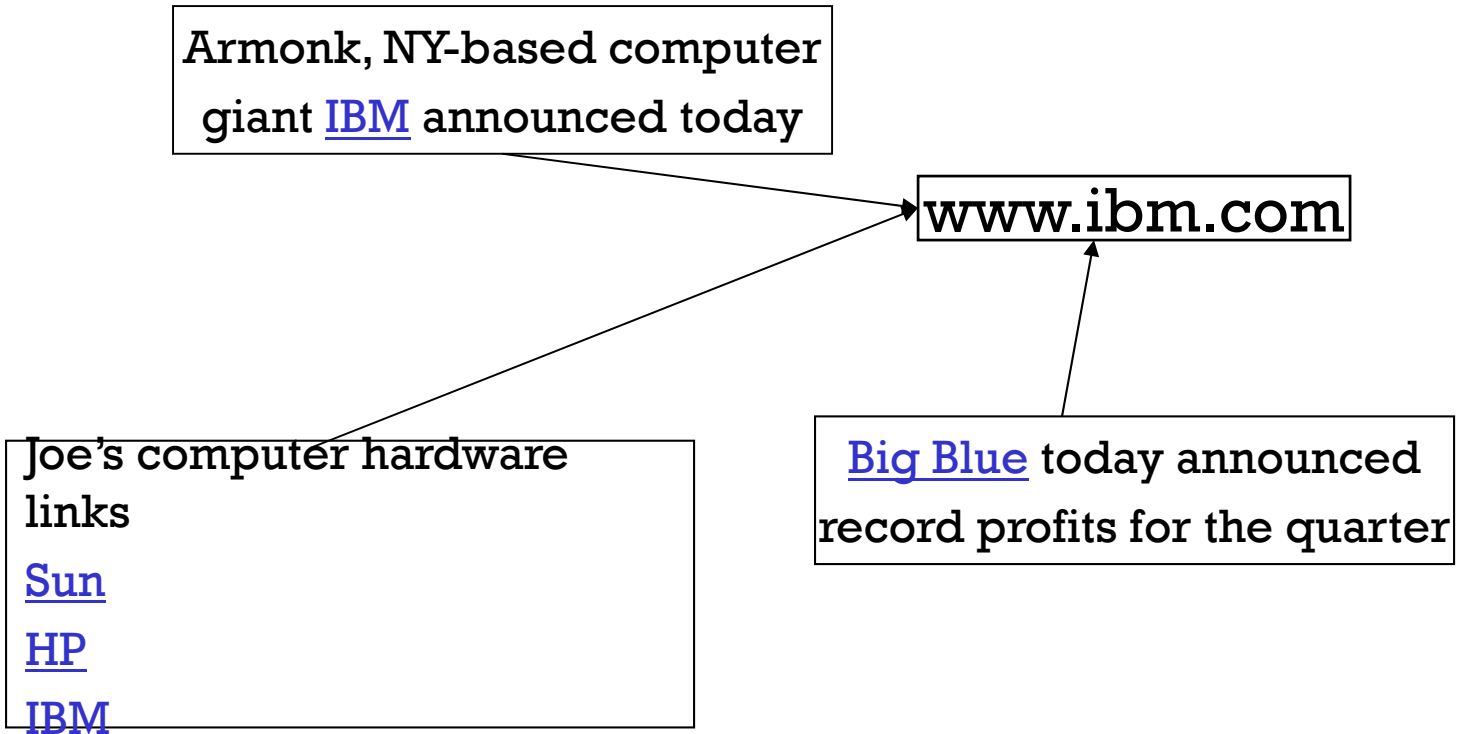
WWW Worm - McBryan [Mcbr94]

- For ***ibm*** how to distinguish between:
  - IBM's home page (mostly graphical)
  - IBM's copyright page (high term freq. for 'ibm')
  - Rival's spam page (arbitrarily high term freq.)



Indexing anchor text

- When indexing a document *D*, include (with some weight) anchor text from links pointing to *D*.



Indexing anchor text

- Can sometimes have unexpected side effects - *e.g.*, ***evil empire***.
- Can score anchor text with weight depending on the authority of the anchor page's website
  - E.g., if we were to assume that content from cnn.com or yahoo.com is authoritative, then trust the anchor text from them

Anchor Text

- Other applications
  - Weighting/filtering links in the graph
  - Generating page descriptions from anchor text

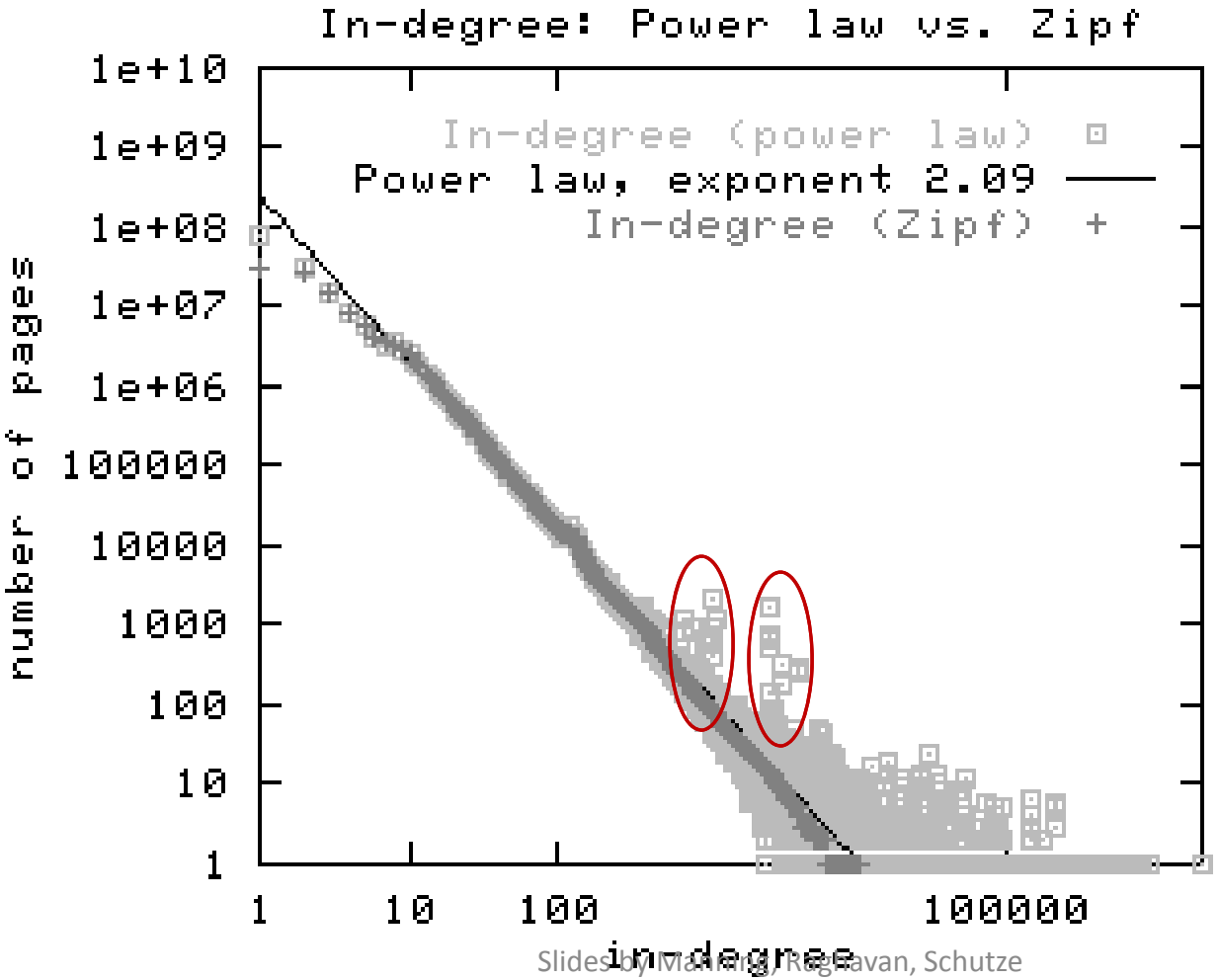
Citation Analysis

- Citation frequency
- Bibliographic coupling frequency
  - Articles that co-cite the same articles are related
- Citation indexing
  - Who is this author cited by? (Garfield 1972)
- Pagerank preview: Pinsker and Narin ’60s

The web isn't scholarly citation

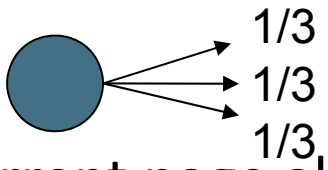
- Millions of participants, each with self interests
- Spamming is widespread
- Once search engines began to use links for ranking (roughly 1998), link spam grew
  - You can join a group of websites that heavily link to one another

In-links to pages – unusual patterns ☺



Pagerank scoring

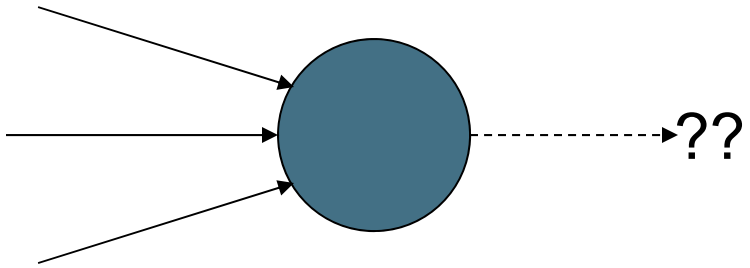
- Imagine a browser doing a random walk on web pages:
  - Start at a random page
  - At each step, go out of the current page along one of the links on that page, equiprobably



- "In the steady state" each page has a long-term visit rate - use this as the page's score.

Not quite enough

- The web is full of dead-ends.
  - Random walk can get stuck in dead-ends.
  - Makes no sense to talk about long-term visit rates.



Teleporting

- At a dead end, jump to a random web page.
- At any non-dead end, with probability 10%, jump to a random web page.
  - With remaining probability (90%), go out on a random link.
  - 10% - a parameter.

Result of teleporting

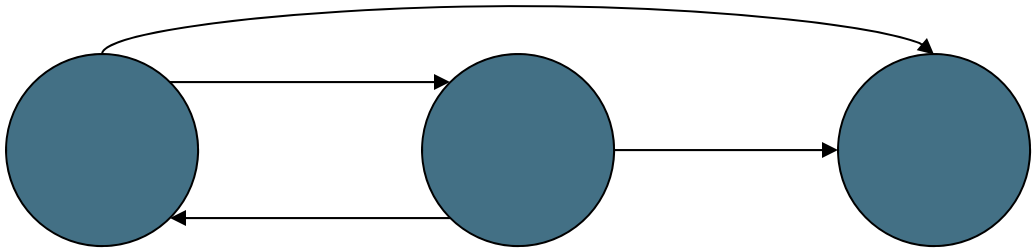
- Now cannot get stuck locally.
- There is a long-term rate at which any page is visited (not obvious, will show this).
- How do we compute this visit rate?

Markov chains

- A Markov chain consists of n states, plus an $n \times n$ transition probability matrix **P**.
- At each step, we are in exactly one of the states.
- For $1 \leq i,j \leq n$, the matrix entry P_{ij} tells us the probability of j being the next state, given we are currently in state i.

Markov chains

- Clearly, for all *i*, $\sum_{j=1}^{n} P_{ij} = 1.$
- Markov chains are abstractions of random walks.
- *Exercise*: represent the teleporting random walk from 3 slides ago as a Markov chain, for this case:



Ergodic Markov chains

- For any (ergodic) Markov chain, there is a unique long-term visit rate for each state.
  - *Steady-state probability distribution*.
- Over a long time-period, we visit each state in proportion to this rate.
- It doesn't matter where we start.

Theory of Markov chains

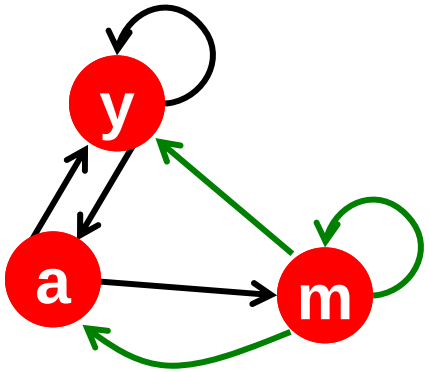
- Fact: For any start vector, the power method applied to a Markov transition matrix P will converge to a unique positive stationary vector as long as P is **stochastic**, **irreducible** and **aperiodic**.

Make M Stochastic

- **Stochastic:** Every column sums to 1
- **A possible solution:** Add green links

$$S = M + a^T \left(\frac{1}{n}\mathbf{1}\right)$$

- a_i…=1 if node *i* has out deg 0, =0 else
- **1**…vector of all 1s

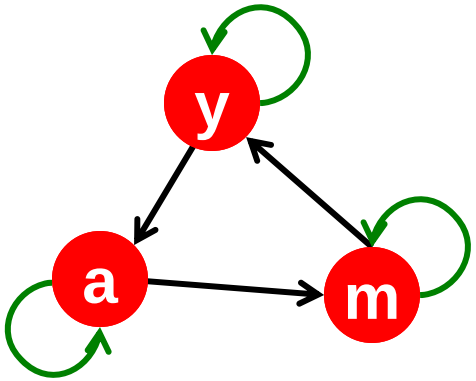


	y	a	m
y	½	½	1/3
a	½	0	1/3
m	0	½	1/3

$$r_y = r_y/2 + r_a/2 + r_m/3$$
$$r_a = r_y/2 + r_m/3$$
$$r_m = r_a/2 + r_m/3$$

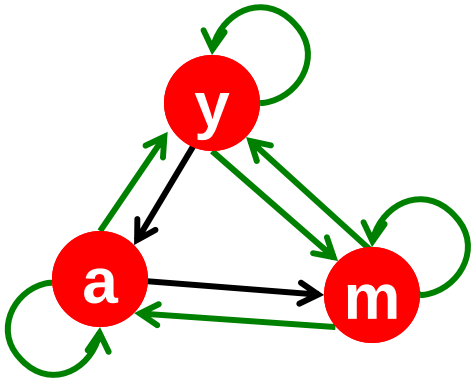
Make M Aperiodic

- A chain is **periodic** if there exists $k > 1$ such that the interval between two visits to some state s is always a multiple of k.
- **A possible solution:** Add green links



Make M Irreducible

- From any state, there is a non-zero probability of going from any one state to any another
- **A possible solution:** Add green links



Probability vectors

- A probability (row) vector $\mathbf{x} = (x_1, \ldots x_n)$ tells us where the walk is at any point.
- E.g., (000…1…000) means we're in state *i*.

1 i n

More generally, the vector $\mathbf{x} = (x_1, \ldots x_n)$ means the walk is in state *i* with probability x_i.

$$\sum_{i=1}^{n} x_i = 1.$$

Change in probability vector

- If the probability vector is $\mathbf{x} = (x_1, \ldots x_n)$ at this step, what is it at the next step?
- Recall that row *i* of the transition prob. Matrix **P** tells us where we go next from state *i*.
- So from **x**, our next state is distributed as **xP**
  - The one after that is $\mathbf{xP}^2$, then $\mathbf{xP}^3$, etc.
  - (Where) Does the converge?

How do we compute this vector?

- Let **a** = $(a_1, \ldots a_n)$ denote the row vector of steady-state probabilities.
- If our current position is described by **a**, then the next step is distributed as **aP**.
- But **a** is the steady state, so **a**=**aP**.
- Solving this matrix equation gives us **a**.
  - So **a** is the (left) eigenvector for **P**.
  - (Corresponds to the "principal" eigenvector of **P** with the largest eigenvalue.)
  - Transition probability matrices always have largest eigenvalue 1.

Pagerank summary

- Preprocessing:
  - Given graph of links, build matrix **P**.
  - From it compute **a** – left eigenvector of **P**.
  - The entry a_i is a number between 0 and 1: the pagerank of page *i*.
- Query processing:
  - Retrieve pages meeting query.
  - Rank them by their pagerank.
  - But this rank order is query-*independent* …

Solution: Random Jumps

- **Google's solution that does it all:**
  - Makes ***M*** stochastic, aperiodic, irreducible
- At each step, random surfer has two options:
  - With probability $1-\beta$, follow a link at random
  - With probability β, jump to some random page
- **PageRank equation** [Brin-Page, 98]

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1-\beta)\frac{1}{n}$$

From now on: We assume M has no dead ends
That is, we follow random teleport links
with probability 1.0 from dead-ends

d_i … out-degree

The Google Matrix

- **PageRank equation** [Brin-Page, 98]

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1 - \beta) \frac{1}{n}$$

- **The Google Matrix *A*:**

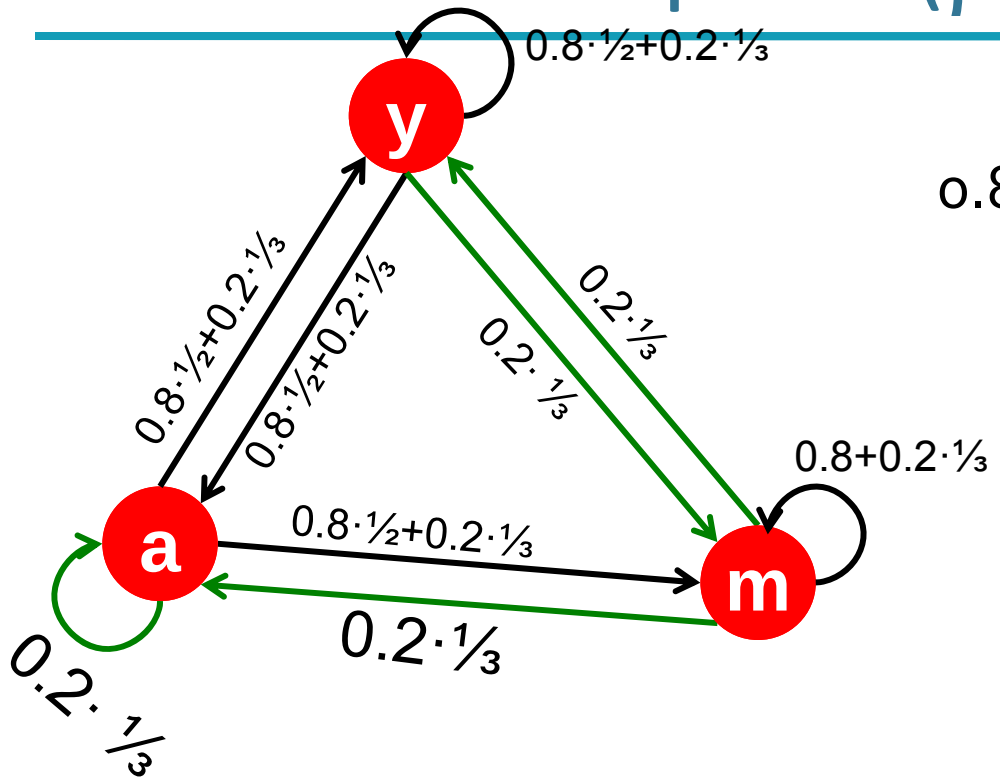
$$A = \beta\, S + (1 - \beta) \frac{1}{n} \mathbf{1} \cdot \mathbf{1}^T$$

- ***G* is stochastic, aperiodic and irreducible, so**

$$r^{(t+1)} = A \cdot r^{(t)}$$

- **What is β?**
  - In practice $\beta = 0.85$ (make 5 steps and jump)

Random Teleports ($\beta = 0.8$)



S $\quad 1/n \cdot 1 \cdot 1^T$

$$0.8 \begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 0 \\ 0 & 1/2 & 1 \end{bmatrix} + 0.2 \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix}$$

$$\begin{matrix} y \\ a \\ m \end{matrix} \begin{bmatrix} 7/15 & 7/15 & 1/15 \\ 7/15 & 1/15 & 1/15 \\ 1/15 & 7/15 & 13/15 \end{bmatrix}$$

A

$$\begin{matrix} y \\ a \\ m \end{matrix} = \begin{matrix} 1/3 & 0.33 & 0.24 & 0.26 \\ 1/3 & 0.20 & 0.20 & 0.18 \\ 1/3 & 0.46 & 0.52 & 0.56 \end{matrix} \quad \ldots \quad \begin{matrix} 7/33 \\ 5/33 \\ 21/33 \end{matrix}$$

Some Problems with Page Rank

- **Measures generic popularity of a page**
  - Biased against topic-specific authorities
  - **Solution:** Topic-Specific PageRank (next)
- **Susceptible to Link spam**
  - Artificial link topographies created in order to boost page rank
  - **Solution:** TrustRank (next)
- **Uses a single measure of importance**
  - Other models e.g., hubs-and-authorities
  - **Solution:** Hubs-and-Authorities (next)

Topic-Specific PageRank

- **Instead of generic popularity, can we measure popularity 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."
- **Allows search queries to be answered based on interests of the user**
  - **Example:** Query "Trojan" wants different pages depending on whether you are interested in sports or history.

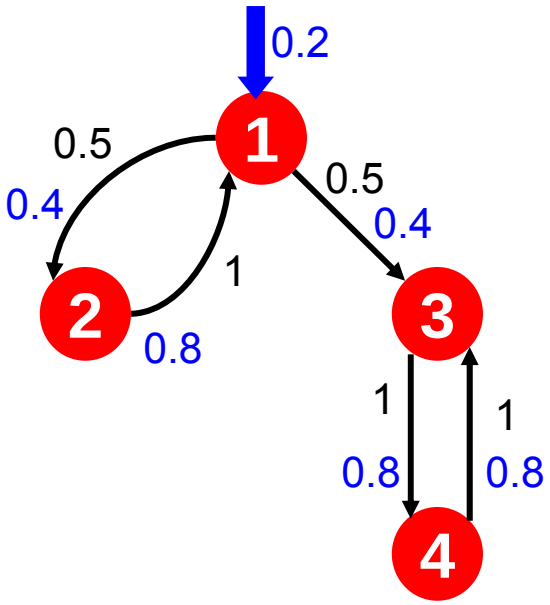
Topic-Specific PageRank

- Assume each walker has a small probability of "teleporting" at any step
- **Teleport can go to:**
  - Any page with equal probability
    - To avoid dead-end and spider-trap problems
  - A topic-specific set of "relevant" pages (teleport set)
    - For topic-sensitive PageRank.
- **Idea: Bias the random walk**
  - When walked teleports, she pick a page from a set *S*
  - *S* contains only pages that are relevant to the topic
    - E.g., Open Directory (DMOZ) pages for a given topic
  - For each teleport set *S*, we get a different vector r_S

Matrix Formulation

- **Let:**
  - $\boldsymbol{A}_{ij} = \beta \boldsymbol{M}_{ij} + (1-\beta) / |S|$ if $i \in S$
    $\beta \boldsymbol{M}_{ij}$ otherwise
  - **A** is stochastic!
- We have weighted all pages in the teleport set *S* equally
  - **Could also assign different weights to pages!**
- **Compute as for regular PageRank:**
  - Multiply by ***M***, then add a vector
  - Maintains sparseness

Example



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

Node	Iteration				
	0	**1**	**2**	**…**	**stable**
1	1.0	0.2	0.52		0.294
2	0	0.4	0.08		0.118
3	0	0.4	0.08		0.327
4	0	0	0.32		0.261

Note how we initialize the PageRank vector differently from the unbiased PageRank case.

Discovering the Topic

- **Create different PageRanks for different topics**
  - The 16 DMOZ top-level categories:
    - arts, business, sports,…
- **Which topic ranking to use?**
  - User can pick from a menu
  - Classify query into a topic
  - Can use the context of the query
    - E.g., query is launched from a web page talking about a known topic
    - History of queries e.g., “basketball” followed by “Jordan”
  - User context, e.g., user’s bookmarks, …

Web Spam

What is Web Spam?

- **Spamming:**
  - any deliberate action to boost a web page's position in search engine results, incommensurate with page's real value
- **Spam:**
  - web pages that are the result of spamming
- This is a very broad definition
  - SEO industry might disagree!
  - SEO = search engine optimization
- Approximately 10-15% of web pages are spam

Web Search

- **Early search engines:**
  - Crawl the Web
  - Index pages by the words they contained
  - Respond to search queries (lists of words) with the pages containing those words
- **Early page ranking:**
  - Attempt to order pages matching a search query by "importance"
  - First search engines considered:
    - 1) Number of times query words appeared.
    - 2) Prominence of word position, e.g. title, header.

First Spammers

- As people began to use search engines to find things on the Web, those with commercial interests tried to exploit search engines to bring people to their own site – whether they wanted to be there or not.
- **Example:**
  - Shirt-seller might pretend to be about "movies."
- **Techniques for achieving high relevance/importance for a web page**

First Spammers: Term Spam

- **How do you make your page appear to be about movies?**
  - **1)** Add the word movie 1000 times to your page
  - Set text color to the background color, so only search engines would see it
  - **2)** Or, run the query "movie" on your target search engine
  - See what page came first in the listings
  - Copy it into your page, make it "invisible"
- **These and similar techniques are term spam**

Google's Solution to Term Spam

- **Believe what people say about you, rather than what you say about yourself**
  - Use words in the anchor text (words that appear underlined to represent the link) and its surrounding text

- PageRank as a tool to measure the "importance" of Web pages

Why It Works?

- Our hypothetical shirt-seller **loses**
  - Saying he is about movies doesn't help, because others don't say he is about movies
  - His page isn't very important, so it won't be ranked high for shirts or movies
- **Example:**
  - Shirt-seller creates 1000 pages, each links to his with "movie" in the anchor text
  - These pages have no links in, so they get little PageRank
  - So the shirt-seller can't beat truly important movie pages like IMDB

Google vs. Spammers: Round 2

- Once Google became the dominant search engine, spammers began to work out ways to fool Google
- **Spam farms** were developed to concentrate PageRank on a single page
- **Link spam:**
  - Creating link structures that boost PageRank of a particular page

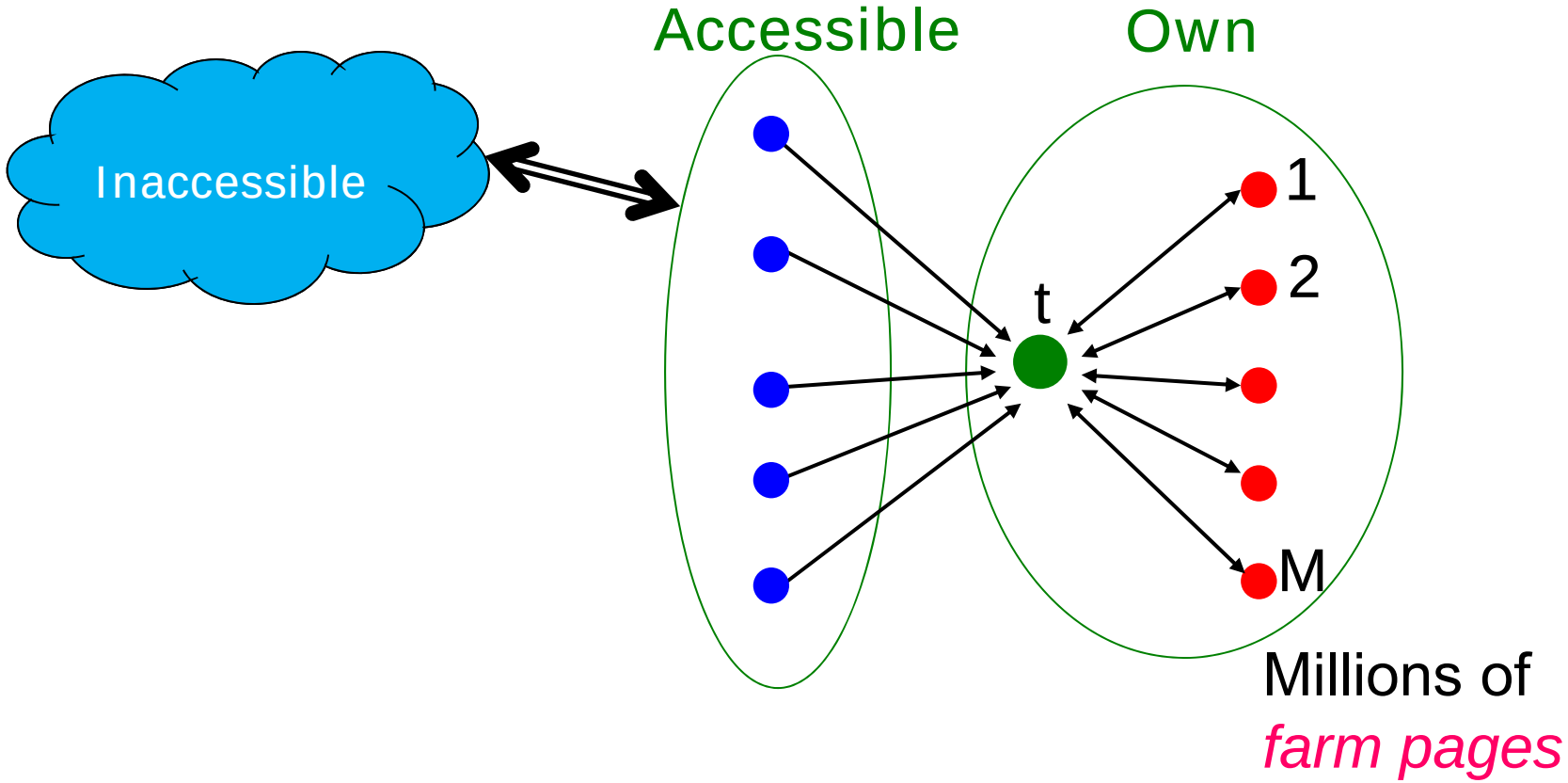
Link Spamming

- **Three kinds of web pages from a spammer's point of view:**
  - **Inaccessible pages**
  - **Accessible pages:**
    - e.g., blog comments pages
    - spammer can post links to his pages
  - **Own pages:**
    - Completely controlled by spammer
    - May span multiple domain names

Link Farms

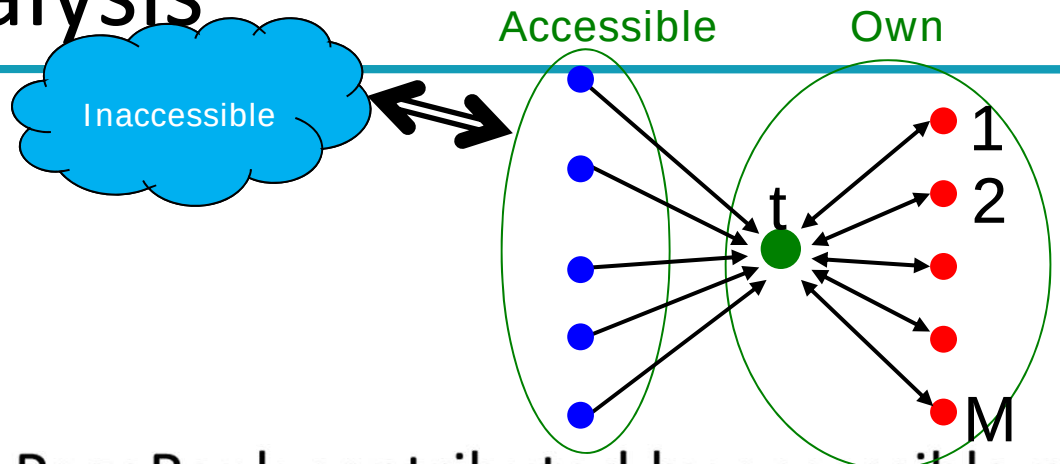
- **Spammer's goal:**
  - Maximize the PageRank of target page *t*
- **Technique:**
  - Get as many links from accessible pages as possible to target page *t*
  - Construct "link farm" to get PageRank multiplier effect

Link Farms



One of the most common and effective organizations for a link farm

Analysis



N…# pages on the web
M…# of pages spammer owns

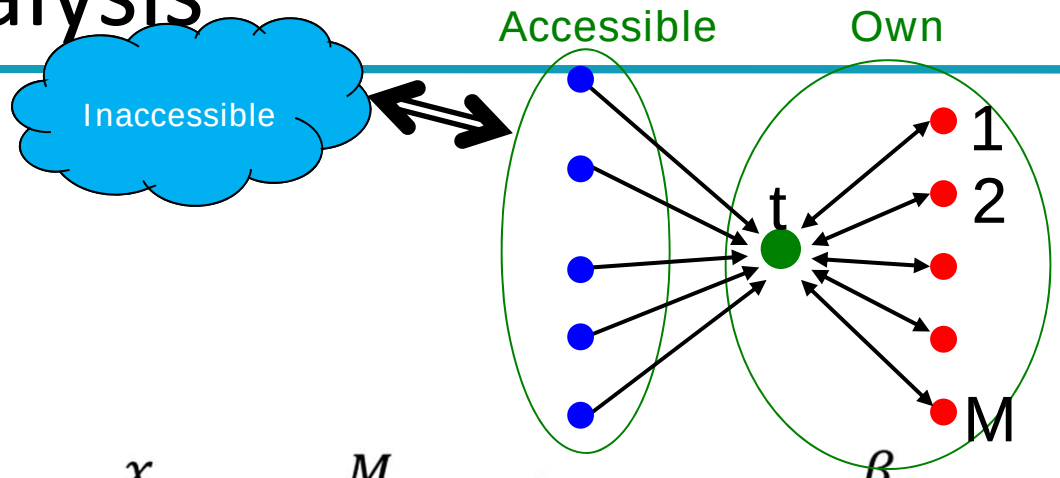
- x: PageRank contributed by accessible pages
- y: PageRank of target page t
- Rank of each "farm" page $= \frac{\beta y}{M} + \frac{1-\beta}{N}$
- $y = x + \beta M\left[\frac{\beta y}{M} + \frac{1-\beta}{N}\right] + \frac{1-\beta}{N}$

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

  Very small; ignore

  Now we solve for y
- $y = \frac{x}{1-\beta^2} + c\frac{M}{N}$ where $c = \frac{\beta}{1+\beta}$

Analysis



N…# pages on the web
M…# of pages spammer owns

- $y = \frac{x}{1-\beta^2} + c\frac{M}{N}$ where $c = \frac{\beta}{1+\beta}$
- For $\beta = 0.85$, $1/(1-\beta^2) = 3.6$
- Multiplier effect for "acquired" PageRank
- By making M large, we can make y as **large as we want**

Combating Web Spam

Combating Spam

- **Combating term spam**
  - Analyze text using statistical methods
  - Similar to email spam filtering
  - Also useful: Detecting approximate duplicate pages
- **Combating link spam**
  - Detection and blacklisting of structures that look like spam farms
    - Leads to another war – hiding and detecting spam farms
  - **TrustRank** = topic-specific PageRank with a teleport set of "trusted" pages
    - Example: .edu domains, similar domains for non-US schools

TrustRank: Idea

- **Basic principle: Approximate isolation**
  - It is rare for a "good" page to point to a "bad" (spam) page
- Sample a set of "seed pages" from the web
- Have an oracle (human) identify the good pages and the spam pages in the seed set
  - Expensive task, so we must make seed set as small as possible

Trust Propagation

- Call the subset of seed pages that are identified as "good" the "trusted pages"
- Perform a topic-sensitive PageRank with teleport set = trusted pages.
  - **Propagate trust through links:**
    - Each page gets a trust value between 0 and 1
- Use a threshold value and mark all pages below the trust threshold as spam

Why is it a good idea?

- **Trust attenuation:**
  - The degree of trust conferred by a trusted page decreases with distance
- **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

Picking the Seed Set

- **Two conflicting considerations:**
  - Human has to inspect each seed page, so 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

Approaches to Picking Seed Set

- Suppose we want to pick a seed set of *k* pages
- **PageRank:**
  - Pick the top *k* pages by PageRank
    - Theory is that you can't get a bad page's rank really high
- Use domains whose membership is controlled, like .edu, .mil, .gov

Spam Mass

- In the TrustRank model, we start with good pages and propagate trust
- **Complementary view:**

  What fraction of a page's PageRank comes from "spam" pages?
- In practice, we don't know all the spam pages, so we need to estimate

Spam Mass Estimation

- $r(p)$ = PageRank of page p
- $r^+(p)$ = page rank of p with teleport into "good" pages only
- **Then:**

  $r^-(p) = r(p) - r^+(p)$
- **Spam mass of $p = r^-(p)/ r(p)$**

The reality

- Pagerank is used in google and other engines, but is hardly the full story of ranking
  - Many sophisticated features are used
  - Some address specific query classes
  - Machine learned ranking (Lecture 19) heavily used
- Pagerank still very useful for things like crawl policy

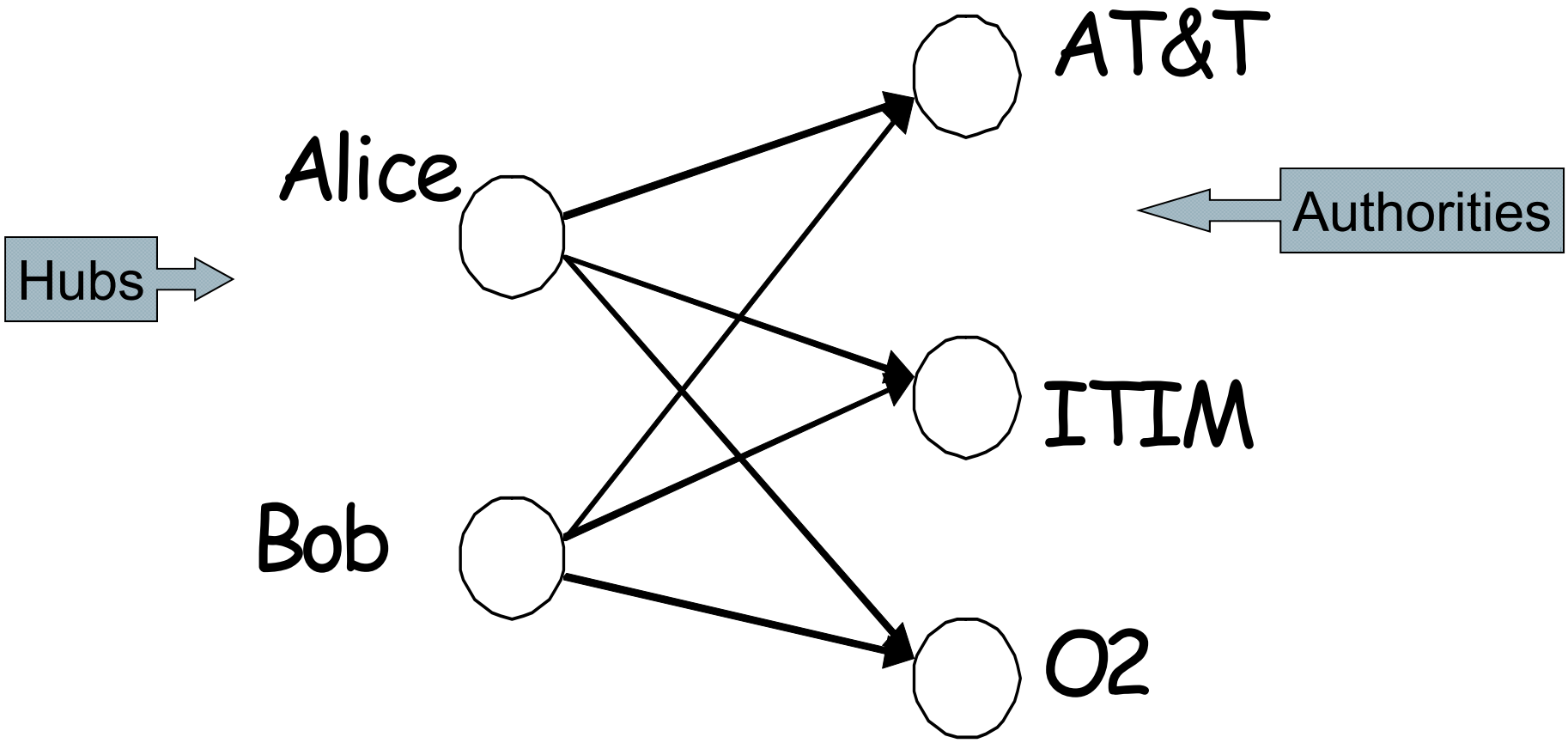
Hyperlink-Induced Topic Search (HITS)

- In response to a query, instead of an ordered list of pages each meeting the query, find two sets of inter-related pages:
  - *Hub pages* are good lists of links on a subject.
    - e.g., "Bob's list of cancer-related links."
  - *Authority pages* occur recurrently on good hubs for the subject.
- Best suited for "broad topic" queries rather than for page-finding queries.
- Gets at a broader slice of common *opinion*.

Hubs and Authorities

- Thus, a good hub page for a topic *points* to many authoritative pages for that topic.
- A good authority page for a topic is *pointed* to by many good hubs for that topic.
- Circular definition - will turn this into an iterative computation.

The hope



Mobile telecom companies

High-level scheme

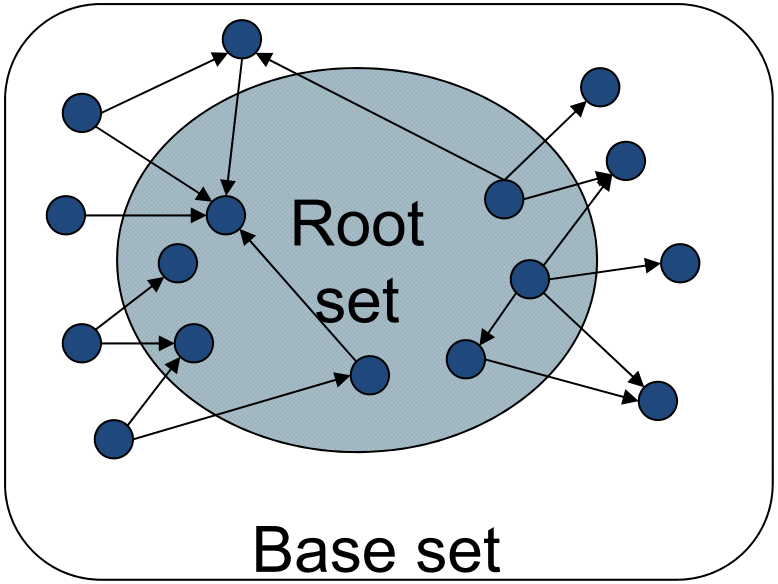
- Extract from the web a base set of pages that *could* be good hubs or authorities.
- From these, identify a small set of top hub and authority pages;

  →iterative algorithm.

Base set

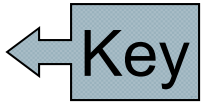
- Given text query (say ***browser***), use a text index to get all pages containing ***browser.***
  - Call this the root set of pages.
- Add in any page that either
  - points to a page in the root set, or
  - is pointed to by a page in the root set.
- Call this the base set.

Visualization



Get in-links (and out-links) from a *connectivity server*

Distilling hubs and authorities

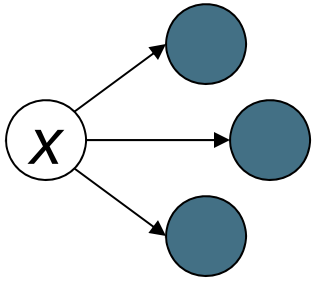
- Compute, for each page *x* in the base set, a hub score *h(x)* and an authority score *a(x).*
- Initialize: for all *x, h(x)←1; a(x) ←1;*
- Iteratively update all *h(x), a(x)*; 

- After iterations
  - output pages with highest *h()* scores as top hubs
  - highest *a()* scores as top authorities.

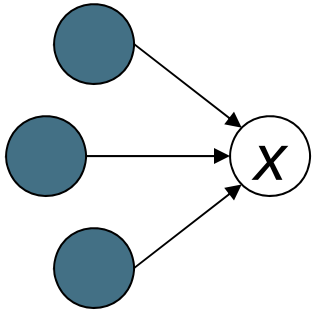
Iterative update

- Repeat the following updates, for all x:

$$h(x) \leftarrow \sum_{x \mapsto y} a(y)$$



$$a(x) \leftarrow \sum_{y \mapsto x} h(y)$$



Scaling

- To prevent the *h()* and *a()* values from getting too big, can scale down after each iteration.
- Scaling factor doesn't really matter:
  - we only care about the *relative* values of the scores.

How many iterations?

- Claim: relative values of scores will converge after a few iterations:
  - in fact, suitably scaled, *h()* and *a()* scores settle into a steady state!
  - proof of this comes later.
- In practice, ~5 iterations get you close to stability.

Japan Elementary Schools

Hubs

- schools
- LINK Page-13
- “ú–{‚ÌŠwZ
- a‰„¬ŠwZƒz[ƒƒy[ƒW
- 100 Schools Home Pages (English)
- K-12 from Japan 10/...rnet and Education)
- http://www...iglobe.ne.jp/~IKESAN
- ‚l‚f‚j¬ŠwZ‚U”N‚P‘g•¨Œê
- ÒŠ—’¬— § ÒŠ—“Œ¬ŠwZ
- Koulutus ja oppilaitokset
- TOYODA HOMEPAGE
- Education
- Cay's Homepage(Japanese)
- –y“ì¬ŠwZ‚Ìƒz[ƒƒy[ƒW
- UNIVERSITY
- ‰J—³¬ŠwZ DRAGON97-TOP
- Â‰ª¬ŠwZ‚T”N‚P‘gƒz[ƒƒy[ƒW
- ¶µ° é¼ÂÁ© ¥á¥Ë¥å¡¼ ¥á¥Ë¥å¡¼

Authorities

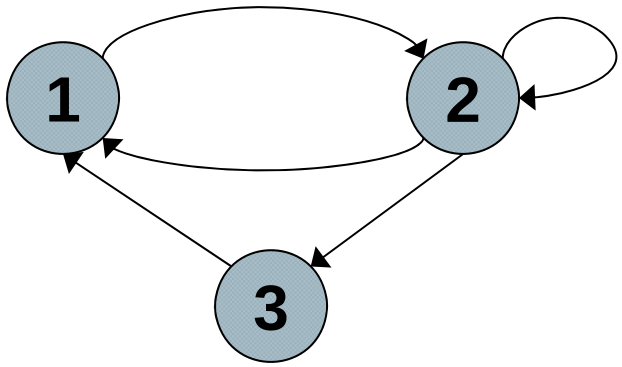
- The American School in Japan
- The Link Page
- ‰ªèsš— § ˆä“c¬ŠwZƒz[ƒƒy[ƒW
- Kids' Space
- ˆÀés— § ˆÀé¼•”¬ŠwZ
- ‹{é‹³ˆç‘åŠw•‘®¬ŠwZ
- KEIMEI GAKUEN Home Page (Japanese)
- Shiranuma Home Page
- fuzoku-es.fukui-u.ac.jp
- welcome to Miasa E&J school
- _“ÞìŒ § E‰¡•ls— § ’†ì¼¬ŠwZ‚Ìƒy
- http://www...p/~m_maru/index.html
- fukui haruyama-es HomePage
- Torisu primary school
- goo
- Yakumo Elementary,Hokkaido,Japan
- FUZOKU Home Page
- Kamishibun Elementary School...

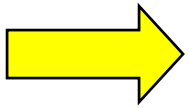
Things to note

- Pulled together good pages regardless of language of page content.
- Use *only* link analysis after base set assembled
  - iterative scoring is query-independent.
- Iterative computation after text index retrieval - significant overhead.

Proof of convergence

- $n \times n$ adjacency matrix **A**:
  - each of the n pages in the base set has a row and column in the matrix.
  - Entry $A_{ij} = 1$ if page i links to page j, else = 0.





	1	2	3
1	0	1	0
2	1	1	1
3	1	0	0

Hub/authority vectors

- View the hub scores *h()* and the authority scores *a()* as vectors with *n* components.
- Recall the iterative updates

$$h(x) \leftarrow \sum_{x \mapsto y} a(y)$$

$$a(x) \leftarrow \sum_{y \mapsto x} h(y)$$

Rewrite in matrix form

- $\mathbf{h}=\mathbf{Aa}$.
- $\mathbf{a}=\mathbf{A}^t\mathbf{h}$.

Recall A^t is the transpose of A.

Substituting, $\mathbf{h}=\mathbf{AA}^t\mathbf{h}$ and $\mathbf{a}=\mathbf{A}^t\mathbf{Aa}$.

Thus, **h** is an eigenvector of $\mathbf{AA}^t$ and **a** is an eigenvector of $\mathbf{A}^t\mathbf{A}$.

Further, our algorithm is a particular, known algorithm for computing eigenvectors: the *power iteration* method.

Guaranteed to converge.

Issues

- Topic Drift
  - Off-topic pages can cause off-topic "authorities" to be returned
    - E.g., the neighborhood graph can be about a "super topic"
- Mutually Reinforcing Affiliates
  - Affiliated pages/sites can boost each others' scores
    - Linkage between affiliated pages is not a useful signal

PageRank and HITS

- **PageRank and HITS are two solutions to the same problem:**
  - **What is the value of an in-link from *u* to *v*?**
  - In the PageRank model, the value of the link depends on the links **into** *u*
  - In the HITS model, it depends on the value of the other links **out of** *u*
- The destinies of PageRank and HITS post-1998 were very different

Resources

- IIR Chap 21
- Chapter 5, Mining Massive Datasets, by Anand Rajaraman and Jeff Ullman, Cambridge University Press, 2011