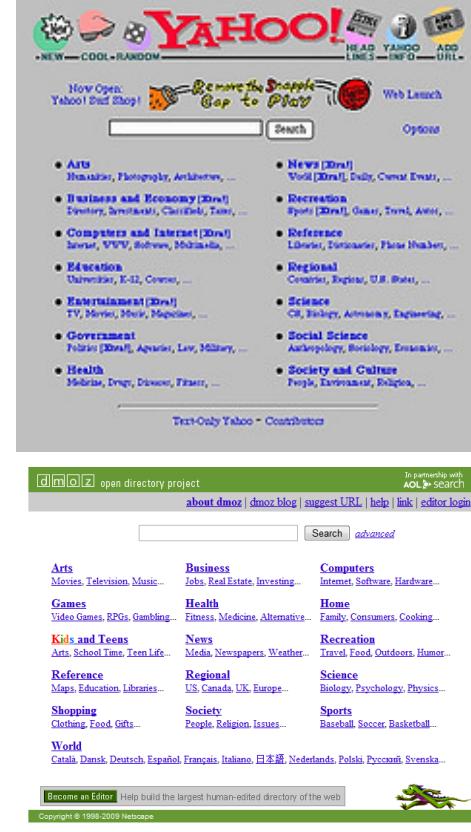


Link Analysis: PageRank and HITS

How to Organize the Web?

- **How to organize the Web?**
- **First try:** Human curated
Web directories
 - Yahoo, DMOZ, LookSmart
- **Second try:** **Web Search**
 - **Information Retrieval** attempts to find relevant docs in a small and trusted set
 - Newspaper articles, Patents, etc.
 - **But:** Web is **huge**, full of untrusted documents, random things, web spam, etc.
 - **So we need a good way to rank webpages!**



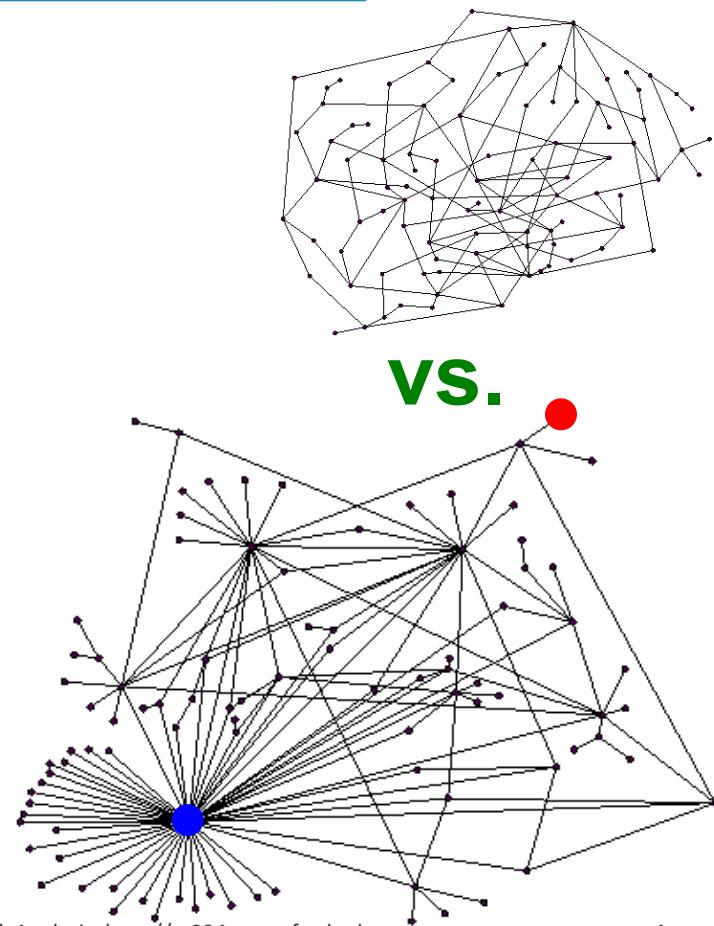
Web Search: 2 Challenges

2 challenges of web search:

- **(1) Web contains many sources of information**
Who to “trust”?
 - **Insight:** Trustworthy pages may point to each other!
- **(2) What is the “best” answer to query “newspaper”?**
 - No single right answer
 - **Insight:** Pages that actually know about newspapers might all be pointing to many newspapers

Ranking Nodes on the Graph

- All web pages are not equally “important”
www.joe-schmoe.com vs. www.stanford.edu
- We already know:
There is large diversity
in the web-graph
node connectivity.
- So, let's rank the pages
using the web graph
link structure!



Link Analysis Algorithms

- We will cover the following Link Analysis approaches to computing importance of nodes in a graph:
 - Hubs and Authorities (HITS)
 - Page Rank
 - Random Walk with Restarts

Sidenote: Various notions of node centrality: Node u

- Degree centrality = degree of u
- Betweenness centrality = #shortest paths passing through u
- Closeness centrality = avg. length of shortest paths from u to all other nodes of the network
- Eigenvector centrality = like PageRank

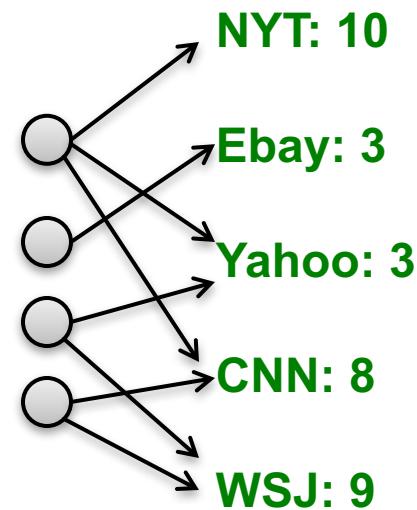
Hubs and Authorities

Link Analysis

- **Goal** (back to the newspaper example):
 - Don't just find newspapers. Find "experts" – pages that 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?
- **Hubs and Authorities**

Each page has **2** scores:

 - **Quality as an expert (hub):**
 - Total sum of votes of pages pointed to
 - **Quality as a content provider (authority):**
 - Total sum of votes of experts
 - **Principle of repeated improvement**

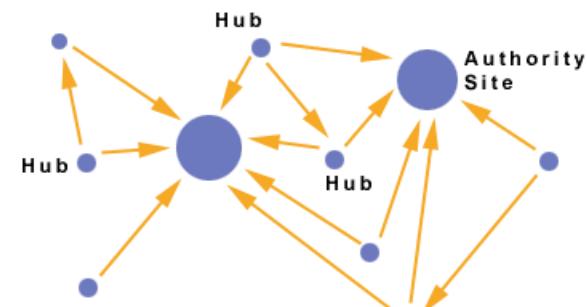


Hubs and Authorities

Interesting pages fall into two classes:

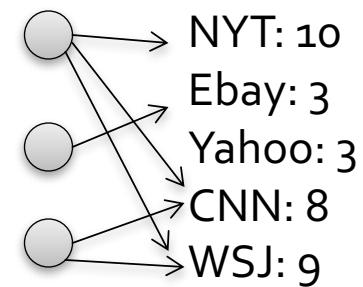
1. **Authorities** are pages containing useful information

- Newspaper home pages
- Course home pages
- Home pages of auto manufacturers

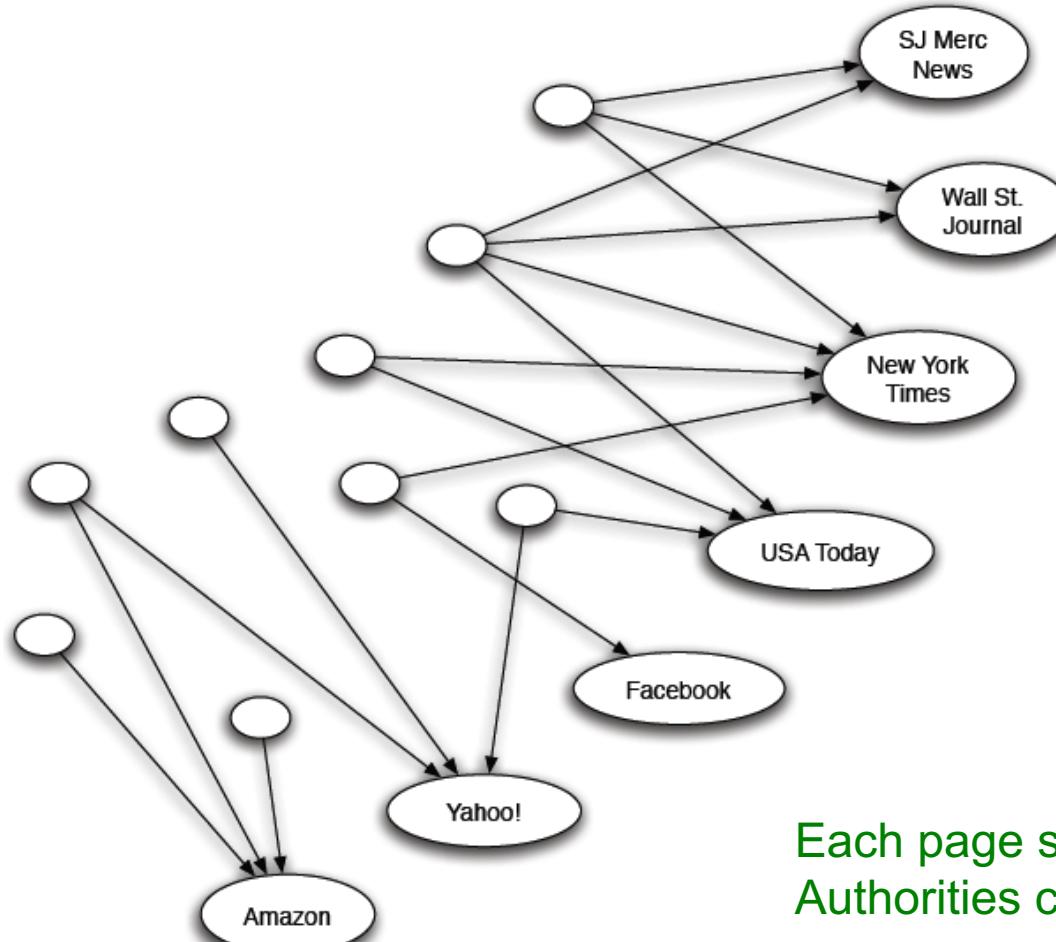


2. **Hubs** are pages that link to authorities

- List of newspapers
- Course bulletin
- List of U.S. auto manufacturers



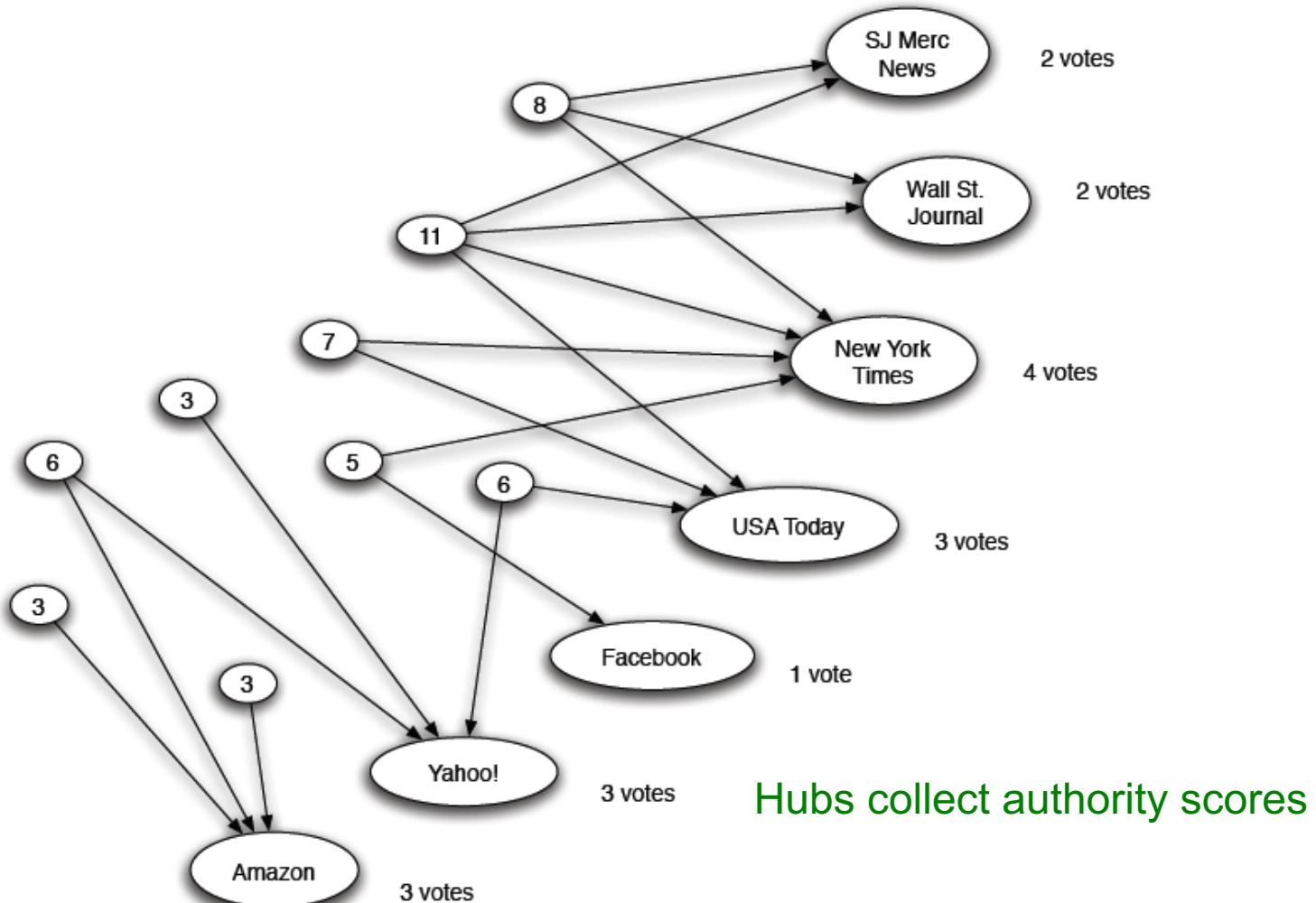
Counting in-links: Authority



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

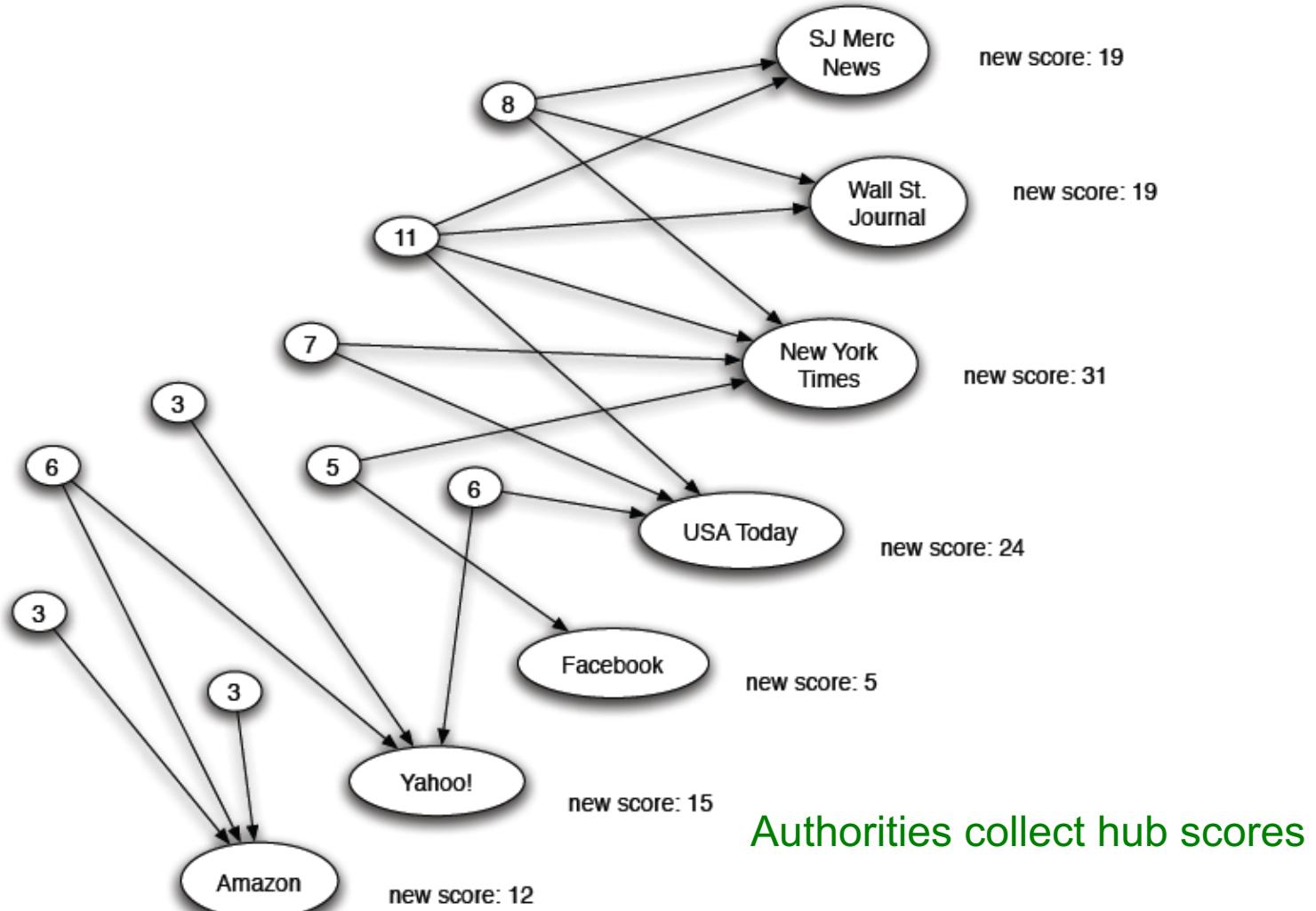
(Note this is an idealized example. In reality graph is not bipartite and each page has both the hub and the authority score)

Expert Quality: Hub



(Note this is an idealized example. In reality graph is not bipartite and each page has both the hub and authority score)

Reweighting



(Note this is an idealized example. In reality graph is not bipartite and each page has both the hub and authority score)

Mutually Recursive Definition

- A good hub links to many good authorities
- A good authority is linked from many good hubs
 - Note a self-reinforcing recursive definition
- Model using two scores for each node:
 - Hub score and Authority score
 - Represented as vectors h and a , where the i-th element is the hub/authority score of the i-th node

Hubs and Authorities

- **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**:

- $\forall i$: Authority: $a_i^{(t+1)} = \sum_{j \rightarrow i} h_j^{(t)}$

- $\forall i$: Hub: $h_i^{(t+1)} = \sum_{i \rightarrow j} a_j^{(t)}$

- $\forall i$: Normalize:

$$\sum_i (a_i^{(t+1)})^2 = 1, \sum_j (h_j^{(t+1)})^2 = 1$$

Convergence criteria:

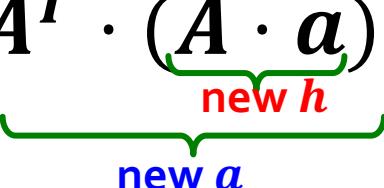
$$\sum_i (h_i^{(t)} - h_i^{(t+1)})^2 < \varepsilon$$

$$\sum_i (a_i^{(t)} - a_i^{(t+1)})^2 < \varepsilon$$

Hubs and Authorities

- **Hits in the vector notation:**
 - Vector $a = (a_1 \dots, a_n)$, $h = (h_1 \dots, h_n)$
 - Adjacency matrix A ($n \times n$): $A_{ij} = 1$ if $i \rightarrow j$
- **Can rewrite** $h_i = \sum_{i \rightarrow j} a_j$ **as** $h_i = \sum_j A_{ij} \cdot a_j$
- **So:** $h = A \cdot a$ And similarly: $a = A^T \cdot h$
- **Repeat until convergence:**
 - $h^{(t+1)} = A \cdot a^{(t)}$
 - $a^{(t+1)} = A^T \cdot h^{(t)}$
 - Normalize $a^{(t+1)}$ and $h^{(t+1)}$

Hubs and Authorities

- What is $a = A^T \cdot h$?
- Then: $a = A^T \cdot (A \cdot a)$

- a is updated (in 2 steps):
$$a = A^T(A \cdot a) = (A^T A) \cdot a$$
- h is updated (in 2 steps):
$$h = A(A^T h) = (A \cdot A^T) \cdot h$$
- Thus, in $2k$ steps:
$$a = (A^T \cdot A)^k \cdot a$$

$$h = (A \cdot A^T)^k \cdot h$$

Repeated matrix powering

Hubs and Authorities

- Definition: Eigenvectors & Eigenvalues
- Let $R \cdot x = \lambda \cdot x$
for some scalar λ , vector x , matrix R
 - Then x is an **eigenvector**, and λ is its **eigenvalue**
- The steady state (HITS has converged) is:
 - $A^T \cdot A \cdot a = c' \cdot a$
 - $A \cdot A^T \cdot h = c'' \cdot h$

Note constants c', c''
don't matter as we
normalize them out
every step of HITS
- So, **authority a** is eigenvector of $A^T A$
(associated with the largest eigenvalue)
Similarly: **hub h** is eigenvector of AA^T

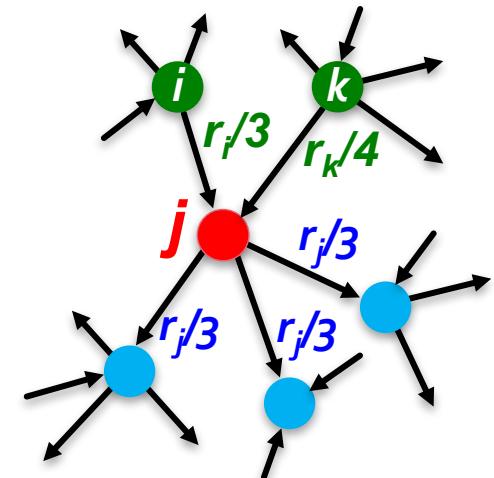
PageRank

Links as Votes

- Still the same idea: **Links as votes**
 - Page is more important if it has more links
 - In-coming links? Out-going links?
- Think of in-links as votes:
 - www.stanford.edu has 23,400 in-links
 - www.joe-schmoe.com has 1 in-link
- Are all in-links equal?
 - Links from important pages count more
 - Recursive question!

PageRank: The “Flow” Model

- A “vote” from an important page is worth more:
 - Each link’s vote is proportional to the **importance** of its source page
 - If page i with importance r_i has d_i out-links, each link gets r_i/d_i votes
 - Page j ’s own importance r_j is the sum of the votes on its in-links



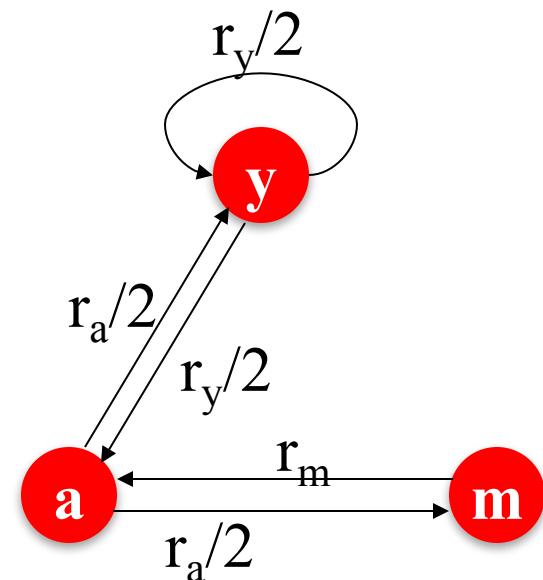
PageRank: The “Flow” Model

- A page is important if it is pointed to by other important pages
- Define a “rank” r_j for node j

$$r_j = \sum_{i \rightarrow j} \frac{r_i}{d_i}$$

d_i ... out-degree of node i

The web in 1839



“Flow” equations:

$$r_y = r_y/2 + r_a/2$$

$$r_a = r_y/2 + r_m$$

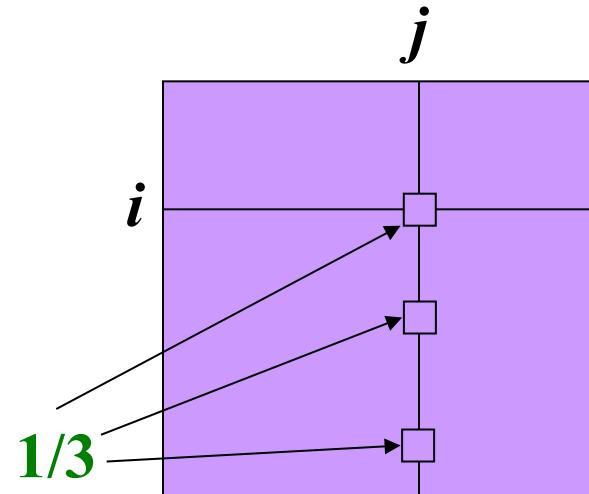
$$r_m = r_a/2$$

You might wonder: Let's just use Gaussian elimination to solve this system of linear equations. Bad idea!

PageRank: Matrix Formulation

■ Stochastic adjacency matrix M

- Let page j have d_j out-links
- If $j \rightarrow i$, then $M_{ij} = \frac{1}{d_j}$
 - M is a **column stochastic matrix**
 - Columns sum to 1



■ Rank vector r : An entry per page

M

- r_i is the importance score of page i
- $\sum_i r_i = 1$

■ The flow equations can be written

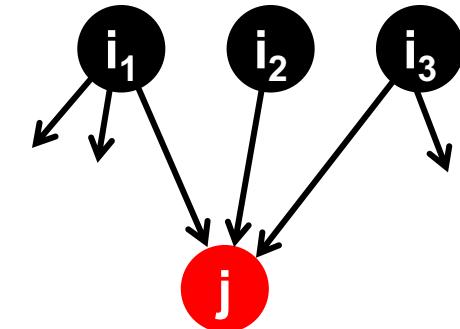
$$\mathbf{r} = M \cdot \mathbf{r}$$

$$r_j = \sum_{i \rightarrow j} \frac{r_i}{d_i}$$

Random Walk Interpretation

- **Imagine a random web surfer:**

- At any time t , surfer is on some page i
- At time $t + 1$, the surfer follows an out-link from i uniformly at random
- Ends up on some page j linked from i
- Process repeats indefinitely



$$r_j = \sum_{i \rightarrow j} \frac{r_i}{d_{\text{out}}(i)}$$

- **Let:**

- $p(t)$... vector whose i^{th} coordinate is the prob. that the surfer is at page i at time t
- So, $p(t)$ is a probability distribution over pages

The Stationary Distribution

- **Where is the surfer at time $t+1$?**

- Follows a link uniformly at random

$$p(t + 1) = M \cdot p(t)$$

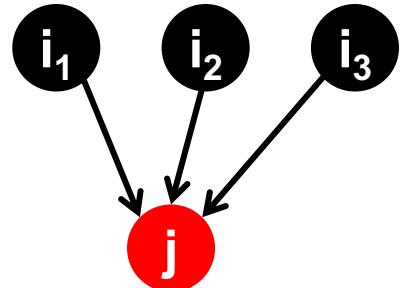
- Suppose the random walk reaches a state

$$p(t + 1) = M \cdot p(t) = p(t)$$

then $p(t)$ is **stationary distribution** of a random walk

- **Our original rank vector r satisfies $r = M \cdot r$**

- **So, r is a stationary distribution for the random walk**



$$p(t + 1) = M \cdot p(t)$$

PageRank: How to solve?

PageRank: How to solve?

Given a web graph with n nodes, where the nodes are pages and edges are hyperlinks

- Assign each node an initial page rank
- Repeat until convergence ($\sum_i |r_i^{(t+1)} - r_i^{(t)}| < \varepsilon$)
 - Calculate the page rank of each node

$$r_j^{(t+1)} = \sum_{i \rightarrow j} \frac{r_i^{(t)}}{d_i}$$

d_i out-degree of node i

PageRank: How to solve?

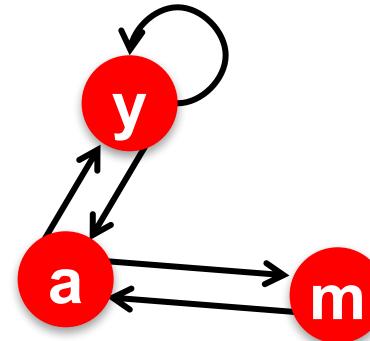
■ Power Iteration:

- Set $r_j \leftarrow 1/N$
- 1: $r'_j \leftarrow \sum_{i \rightarrow j} \frac{r_i}{d_i}$
- 2: $r \leftarrow r'$
- If $|r - r'| > \varepsilon$: goto 1

■ Example:

$$\begin{pmatrix} r_y \\ r_a \\ r_m \end{pmatrix} = \begin{matrix} 1/3 \\ 1/3 \\ 1/3 \end{matrix}$$

Iteration 0, 1, 2, ...



	y	a	m
y	1/2	1/2	0
a	1/2	0	1
m	0	1/2	0

$$r_y = r_y/2 + r_a/2$$

$$r_a = r_y/2 + r_m$$

$$r_m = r_a/2$$

PageRank: How to solve?

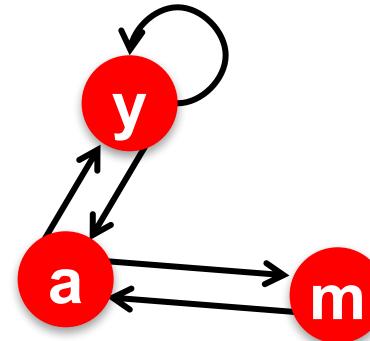
■ Power Iteration:

- Set $r_j \leftarrow 1/N$
- 1: $r'_j \leftarrow \sum_{i \rightarrow j} \frac{r_i}{d_i}$
- 2: $r \leftarrow r'$
- If $|r - r'| > \varepsilon$: goto 1

■ Example:

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{matrix} 1/3 & 1/3 & 5/12 & 9/24 & 6/15 \\ 1/3 & 3/6 & 1/3 & 11/24 & \dots & 6/15 \\ 1/3 & 1/6 & 3/12 & 1/6 & 3/15 \end{matrix}$$

Iteration 0, 1, 2, ...



	y	a	m
y	1/2	1/2	0
a	1/2	0	1
m	0	1/2	0

$$r_y = r_y/2 + r_a/2$$

$$r_a = r_y/2 + r_m$$

$$r_m = r_a/2$$

PageRank: Three Questions

$$r_j^{(t+1)} = \sum_{i \rightarrow j} \frac{r_i^{(t)}}{d_i}$$

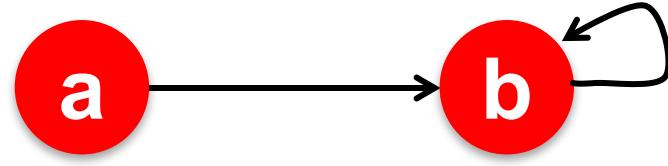
or
equivalently

$$r = Mr$$

- Does this converge?
- Does it converge to what we want?
- Are results reasonable?

Does this converge?

- The “Spider trap” problem:



$$r_j^{(t+1)} = \sum_{i \rightarrow j} \frac{r_i^{(t)}}{d_i}$$

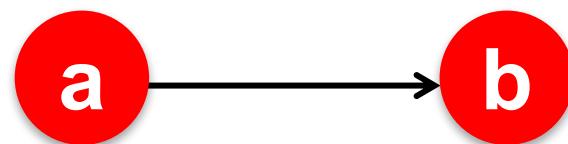
- Example:

Iteration: 0, 1, 2, 3...

r_a	=	1		0		0		0
r_b		0		1		1		1

Does it converge to what we want?

- The “Dead end” problem:



$$r_j^{(t+1)} = \sum_{i \rightarrow j} \frac{r_i^{(t)}}{d_i}$$

- Example:

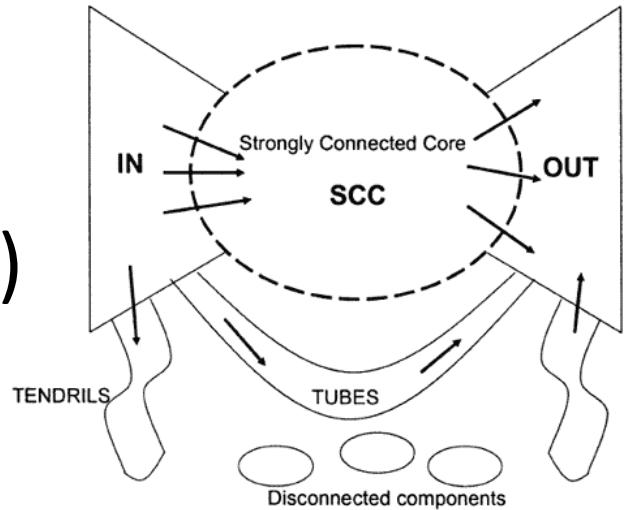
Iteration: 0, 1, 2, 3...

$$\begin{array}{c} r_a \\ r_b \end{array} = \begin{array}{c|c|c|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{array}$$

RageRank: Problems

2 problems:

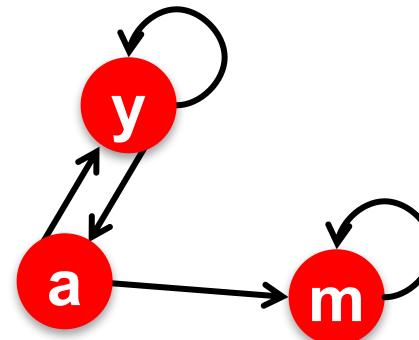
- (1) Some pages are **dead ends** (have no out-links)
 - Such pages cause importance to “leak out”
- (2) **Spider traps**
(all out-links are within the group)
 - Eventually spider traps absorb all importance



Problem: Spider Traps

Power Iteration:

- Set $r_j = \frac{1}{N}$
- $r_j = \sum_{i \rightarrow j} \frac{r_i}{d_i}$
- And iterate



	y	a	m
y	1/2	1/2	0
a	1/2	0	0
m	0	1/2	1

$$r_y = r_y/2 + r_a/2$$

$$r_a = r_y/2$$

$$r_m = r_a/2 + r_m$$

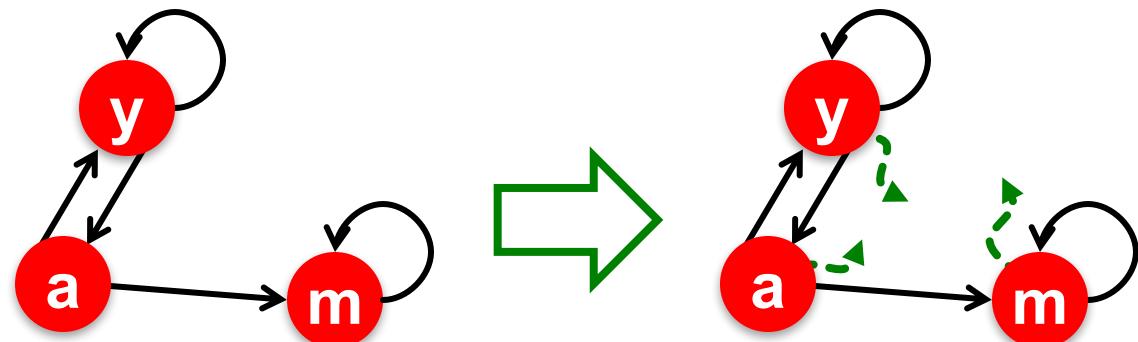
Example:

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{array}{c|c|c|c|c|c} & 1/3 & 2/6 & 3/12 & 5/24 & 0 \\ \hline r_y & 1/3 & 1/6 & 2/12 & 3/24 & \dots & 0 \\ \hline r_a & 1/3 & 3/6 & 7/12 & 16/24 & & 1 \end{array}$$

Iteration 0, 1, 2, ...

Solution: Random Teleports

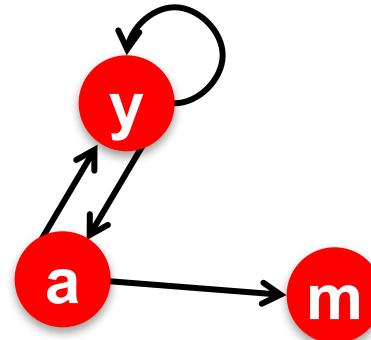
- The Google solution for spider traps: **At each time step, the random surfer has two options**
 - With prob. β , follow a link at random
 - With prob. $1-\beta$, jump to a random page
 - Common values for β are in the range 0.8 to 0.9
- **Surfer will teleport out of spider trap within a few time steps**



Problem: Dead Ends

Power Iteration:

- Set $r_j = \frac{1}{N}$
- $r_j = \sum_{i \rightarrow j} \frac{r_i}{d_i}$
- And iterate



	y	a	m
y	1/2	1/2	0
a	1/2	0	0
m	0	1/2	0

$$r_y = r_y/2 + r_a/2$$

$$r_a = r_y/2$$

$$r_m = r_a/2$$

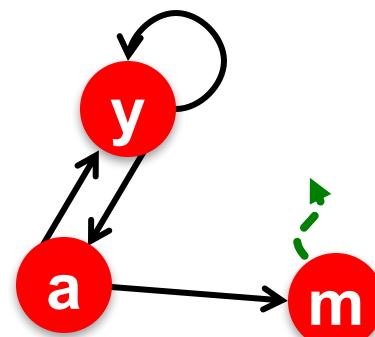
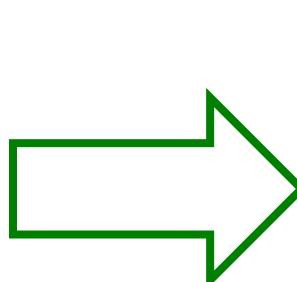
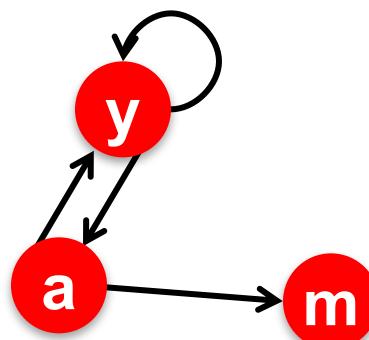
Example:

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{array}{c|c|c|c|c|c} & 1/3 & 2/6 & 3/12 & 5/24 & 0 \\ \hline r_y & 1/3 & 1/6 & 2/12 & 3/24 & \dots & 0 \\ \hline r_a & 1/3 & 1/6 & 1/12 & 2/24 & & 0 \end{array}$$

Iteration 0, 1, 2, ...

Solution: Always Teleport

- **Teleports:** Follow random teleport links with probability 1.0 from dead-ends
 - Adjust matrix accordingly



	y	a	m
y	$\frac{1}{2}$	$\frac{1}{2}$	0
a	$\frac{1}{2}$	0	0
m	0	$\frac{1}{2}$	0

	y	a	m
y	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{3}$
a	$\frac{1}{2}$	0	$\frac{1}{3}$
m	0	$\frac{1}{2}$	$\frac{1}{3}$

Final PageRank Equation

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

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

d_i ... out-degree
of node i

The above formulation assumes that M has no dead ends. We can either preprocess matrix M (**bad!**) or explicitly follow random teleport links with probability 1.0 from dead-ends. See P. Berkhin, *A Survey on PageRank Computing*, Internet Mathematics, 2005.

PageRank & Eigenvectors

- **PageRank as a principal eigenvector**

$$\mathbf{r} = \mathbf{M} \cdot \mathbf{r} \text{ or equivalently } r_j = \sum_i \frac{r_i}{d_i}$$

- **But we really want (**):**

$$r_j = \beta \sum_i \frac{r_i}{d_i} + (1 - \beta) \frac{1}{n}$$

- **Let's define:**

$$M'_{ij} = \beta M_{ij} + (1 - \beta) \frac{1}{n}$$

- **Now we get what we want:**

$$\mathbf{r} = \mathbf{M}' \cdot \mathbf{r}$$

- **What is $1 - \beta$?**

- In practice 0.15 (Jump approx. every 5-6 links)

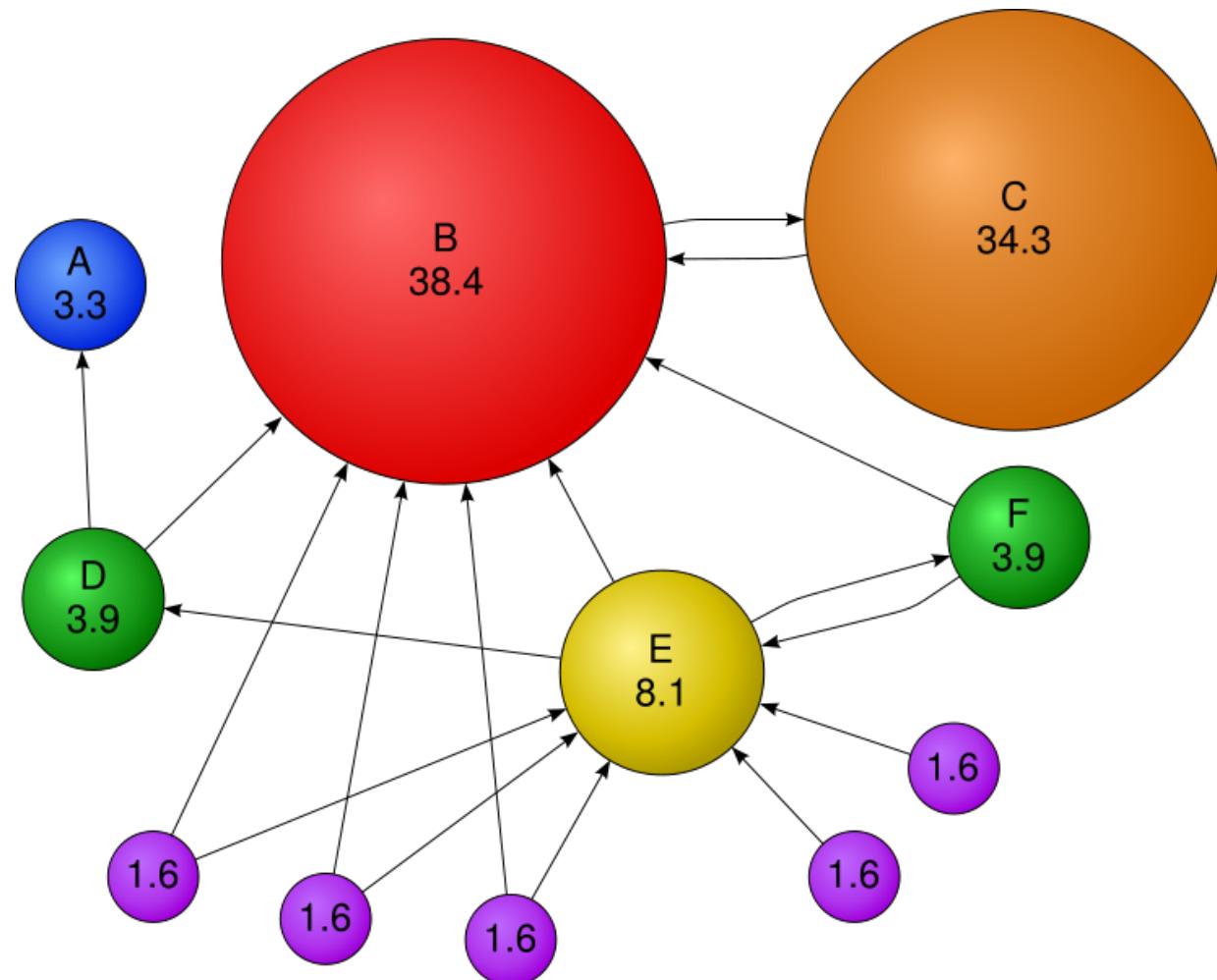
d_i ... out-degree
of node i

Note: M is a sparse matrix but \mathbf{M}' is dense (all entries $\neq 0$). In practice we never “materialize” M but rather we use the “sum” formulation (**)

The PageRank Algorithm

- **Input:** Graph G and parameter β
 - Directed graph G with **spider traps** and **dead ends**
 - Parameter β
- **Output:** PageRank vector r
 - **Set:** $r_j^{(0)} = \frac{1}{N}, t = 1$
 - **do:**
 - $\forall j: r_j'^{(t)} = \sum_{i \rightarrow j} \beta \frac{r_i^{(t-1)}}{d_i}$
 - $r_j'^{(t)} = 0$ if in-deg. of j is 0
 - Now re-insert the leaked PageRank:
 $\forall j: r_j^{(t)} = r_j'^{(t)} + \frac{1-S}{N}$ where: $S = \sum_j r_j'^{(t)}$
 - $t = t + 1$
 - **while** $\sum_j |r_j^{(t)} - r_j^{(t-1)}| > \varepsilon$

Example

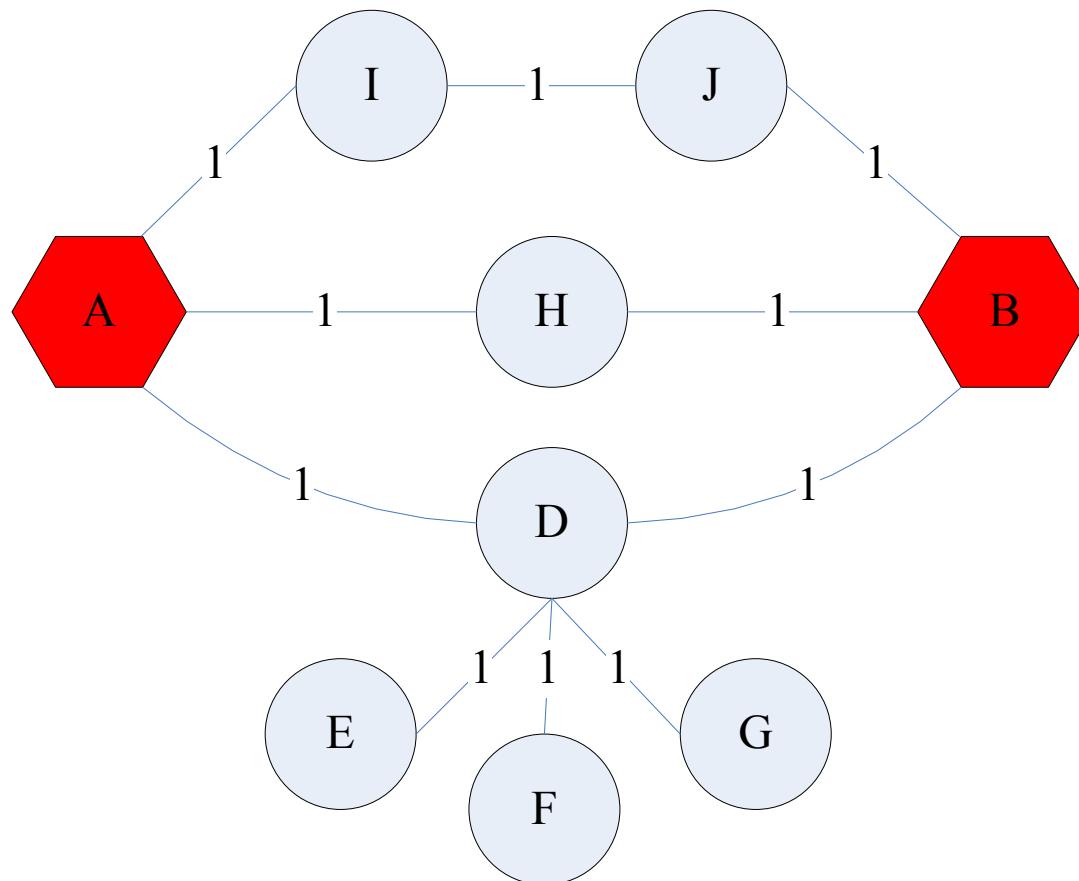


PageRank and HITS

- **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**

Random Walk with Restarts

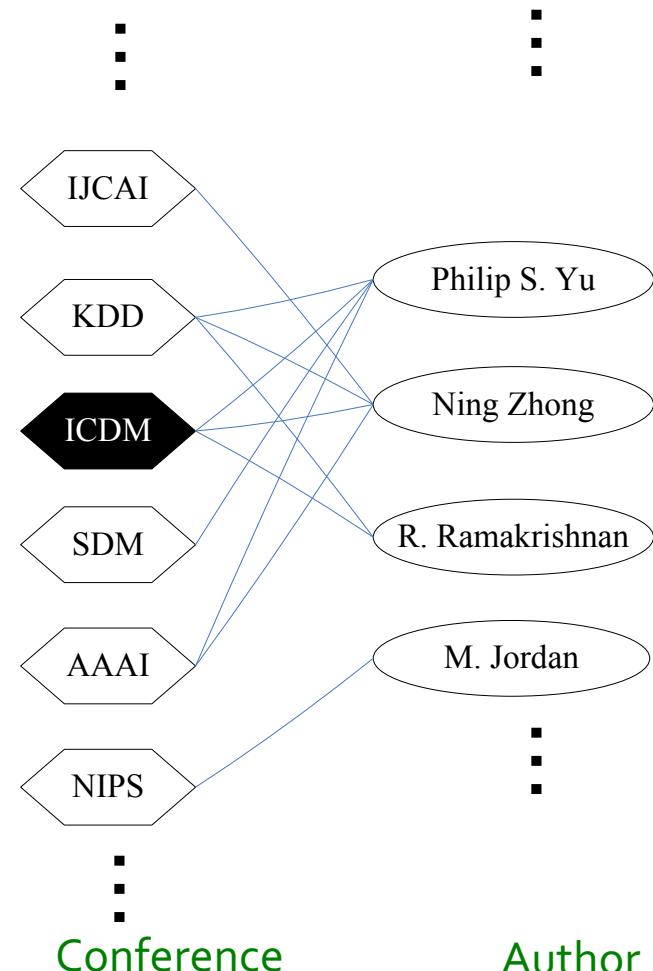
Proximity on Graphs



a.k.a.: Relevance, Closeness, 'Similarity'...

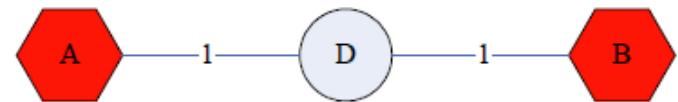
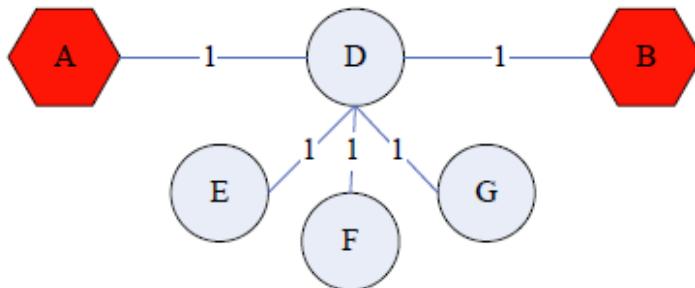
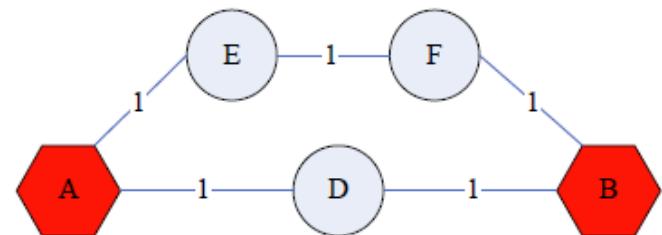
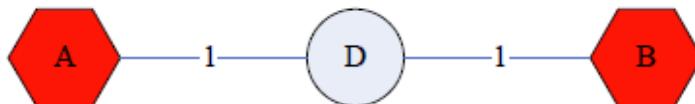
Example Application: Graph Search

- **Given:**
Conferences-to-authors
graph
- **Goal:**
Proximity on graphs
 - Q: What is most related
conference to ICDM?



Good proximity measure?

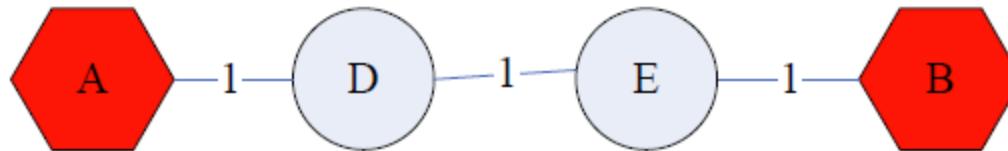
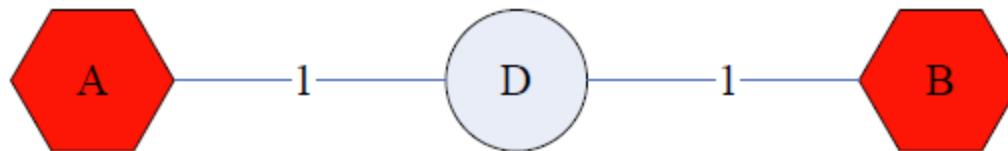
- Shortest path is not good:



- No influence for degree-1 nodes (E, F, G)!
- Multi-faceted relationships

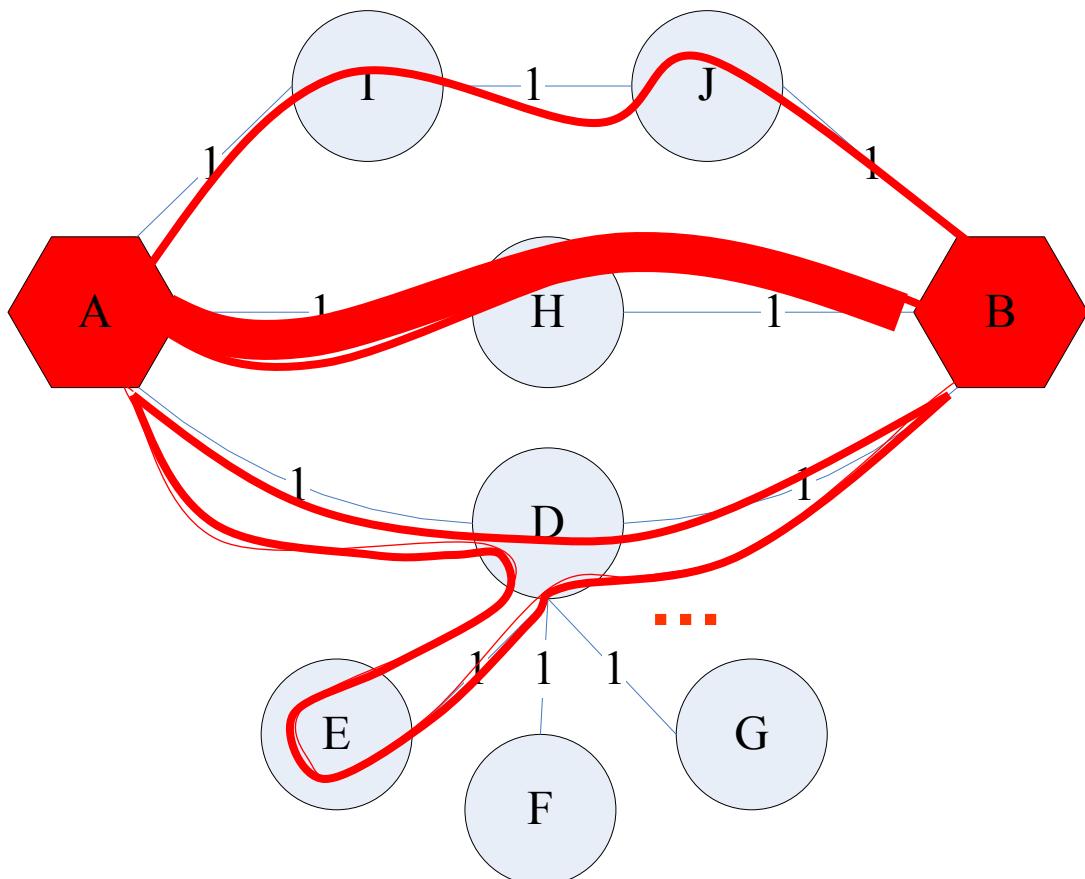
Good proximity measure?

- Network Flow is not good:



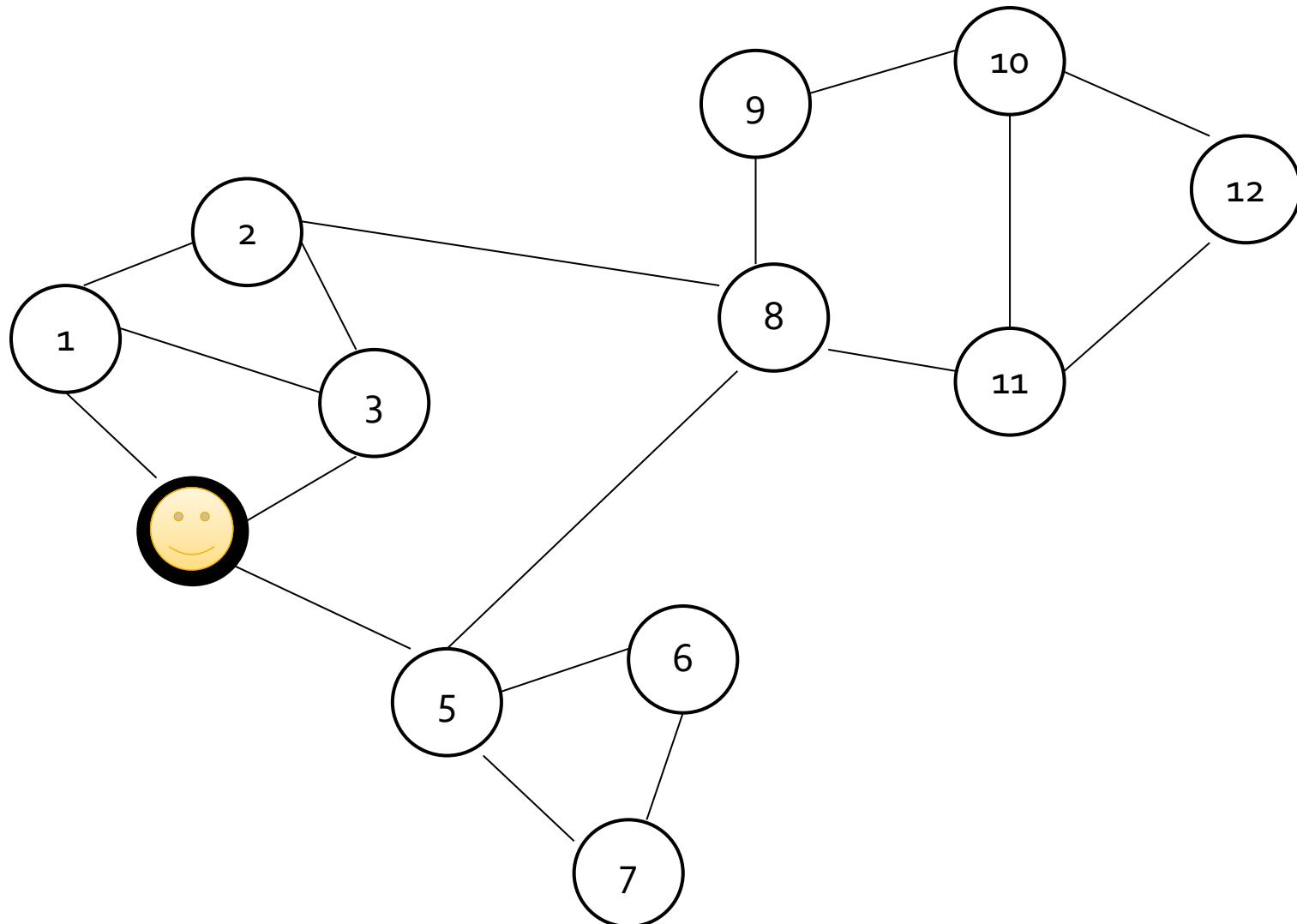
- Does not punish long paths

What is good notion of proximity?



- Multiple Connections
- Quality of connection
 - Direct & In-direct connections
 - Length, Degree, Weight...

Random Walk with Restarts



Personalized PageRank

- **Goal:** Evaluate pages not just by popularity but by how close they are to the topic
- **Teleporting can go to:**
 - Any page with equal probability
 - PageRank (we used this so far)
 - A topic-specific set of “relevant” pages
 - Topic-specific (personalized) PageRank (S ... teleport set)
- $$\begin{aligned} M'_{ij} &= \beta M_{ij} + (1 - \beta)/|S| && \text{if } i \in S \\ &= \beta M_{ij} && \text{otherwise} \end{aligned}$$
- **Random Walk with Restart:** S is a single element

PageRank: Applications

- **Graphs and web search:**

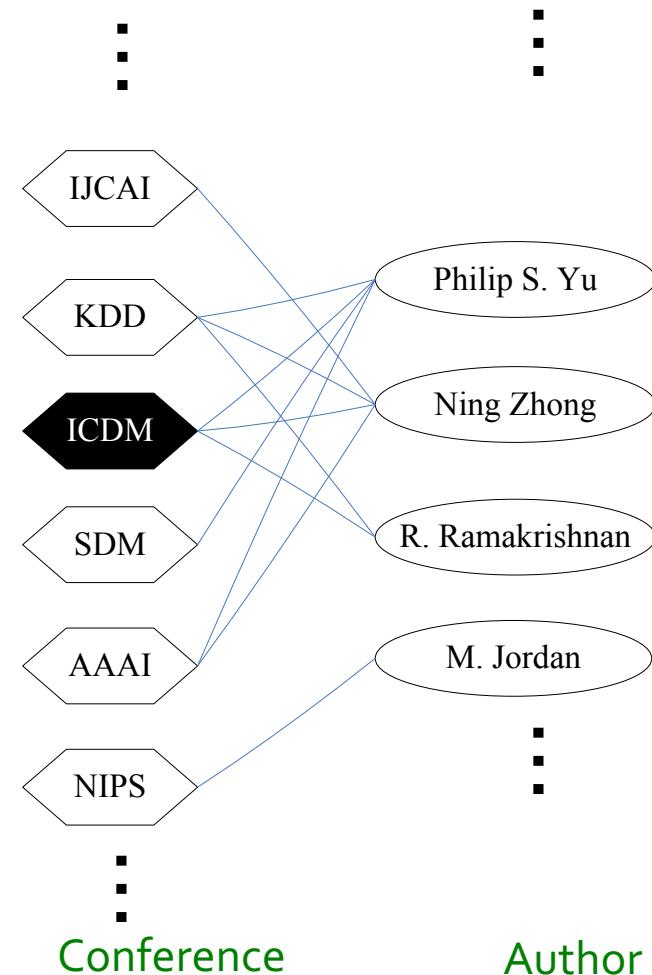
- Ranks nodes by “importance”

- **Personalized PageRank:**

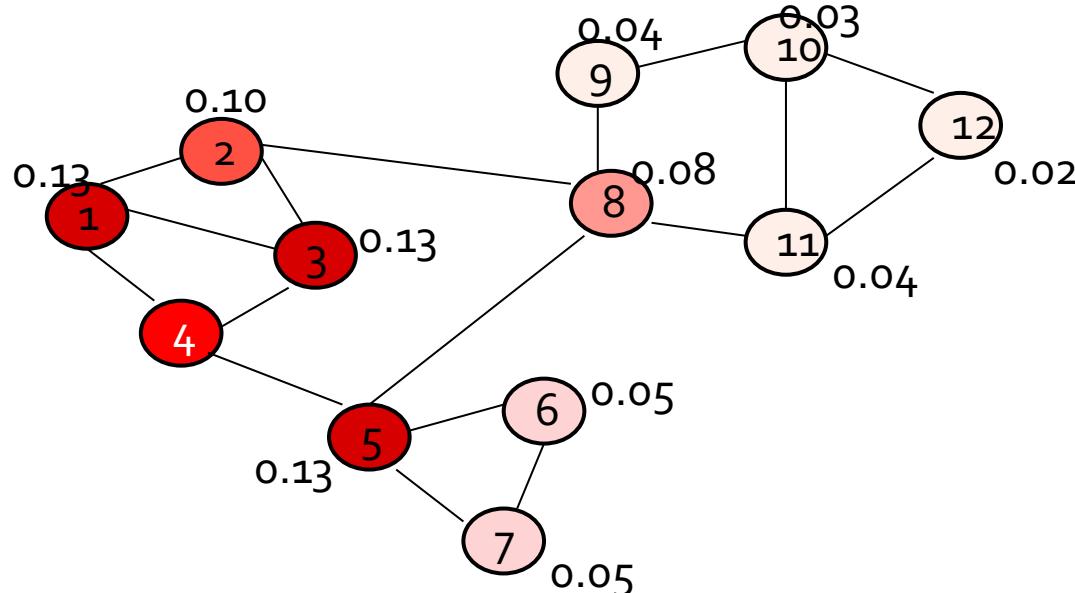
- Ranks proximity of nodes to the teleport nodes S

- **Proximity on graphs:**

- **Q:** What is most related conference to ICDM?
 - **Random Walks with Restarts**
 - Teleport back to the starting node:
 $S = \{ \text{single node} \}$



Random Walk with Restarts

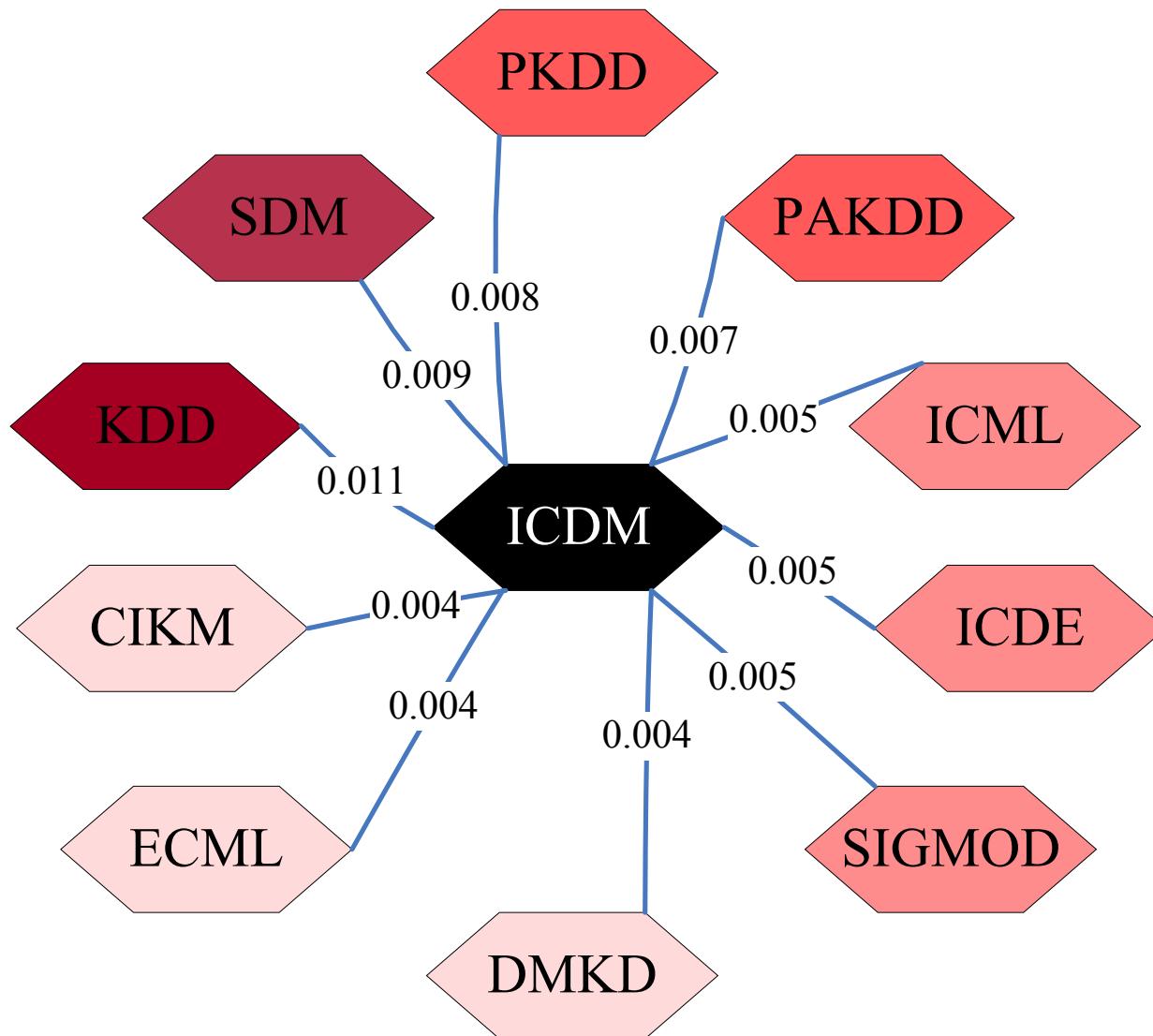


	Node 4
Node 1	0.13
Node 2	0.10
Node 3	0.13
Node 4	/
Node 5	0.13
Node 6	0.05
Node 7	0.05
Node 8	0.08
Node 9	0.04
Node 10	0.03
Node 11	0.04
Node 12	0.02

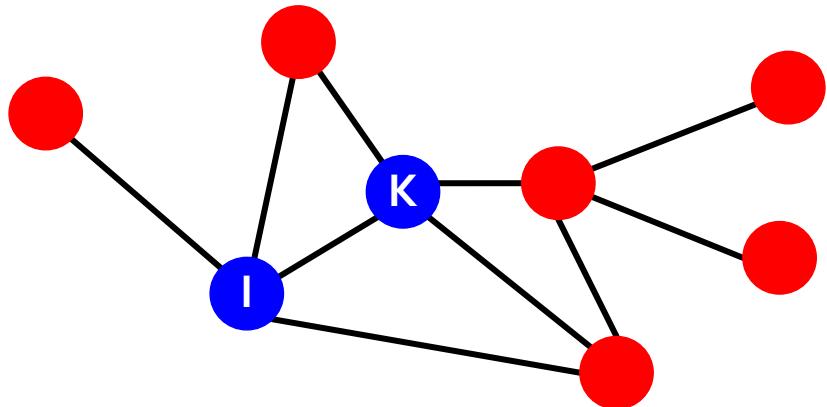
Nearby nodes, higher scores
More red, more relevant

Ranking vector

Most related conferences to ICDM



Personalized PageRank



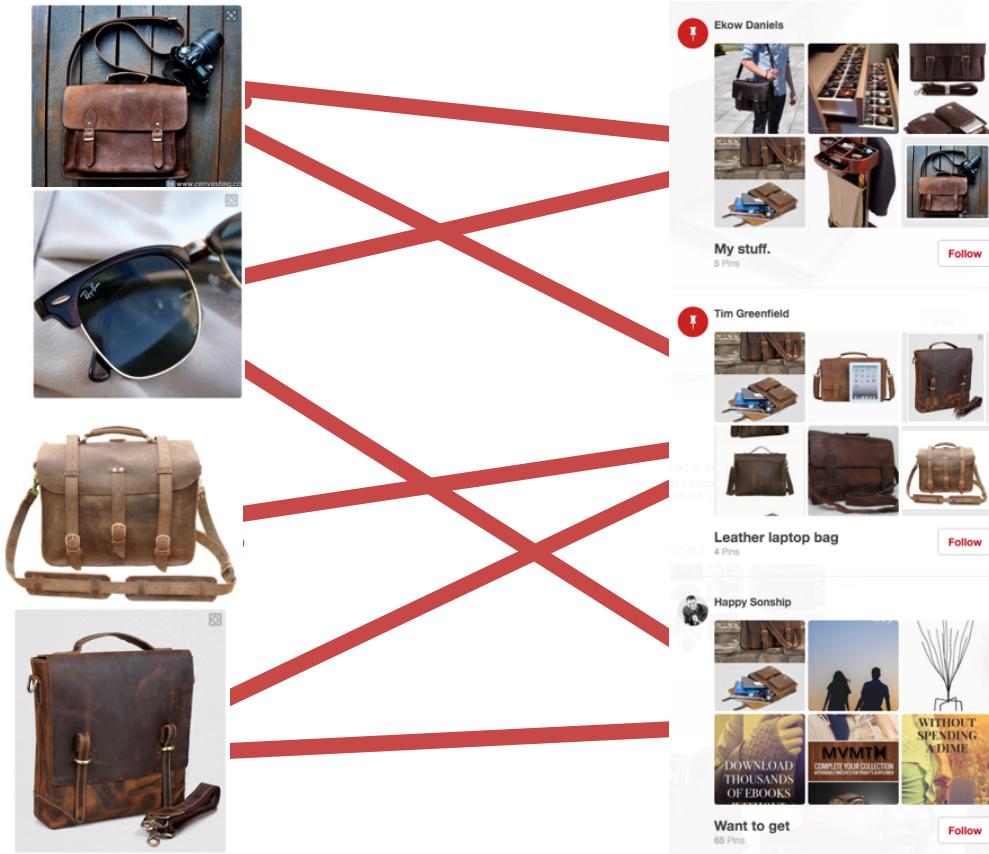
Graph of CS conferences

Q: Which conferences
are closest to KDD &
ICDM?

A: Personalized PageRank with teleport set $S=\{\text{KDD}, \text{ICDM}\}$

Pinterest is a Giant Bipartite Graph

- Pins belong to Boards



Pins to Pins Recommendations

Input:



HEALTHY CHOCOLATE STRAWBERRY SHAKE



Chocolate Strawberry Shake

249

This healthier chocolate strawberry shake is like sipping a...

One Lovely Life



Danielle Benzaia
Strawberries

Pins to Pins Recommendations

Input: Recommendations:



HEALTHY CHOCOLATE STRAWBERRY SHAKE



Chocolate Dipped Strawberry Smoothie

Chocolate Dipped Strawberry Smoothie. Just in time for...

Be Whole. Be You.
Ed Todd
Drinks- Smoothies

Chocolate Strawberry Shake

249

This healthier chocolate strawberry shake is like sipping a...

One Lovely Life

Danielle Benzaia
Strawberries



Tropical Orange Smoothie



Easy Breezy Tropical Orange Smoothie

80.1k

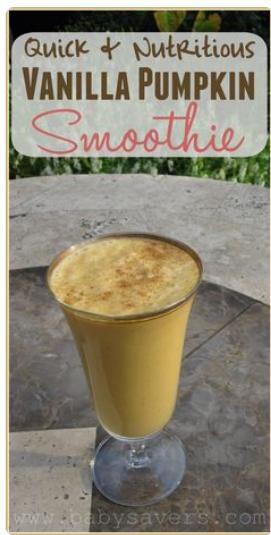


8 STAPLE SMOOTHIES
(THAT YOU SHOULD KNOW HOW TO MAKE)



8 Staple Smoothies You Should Know How to Make
8 Staple Smoothies That You Should Know How to Make

5.2k



Quick & Nutritious VANILLA PUMPKIN Smoothie

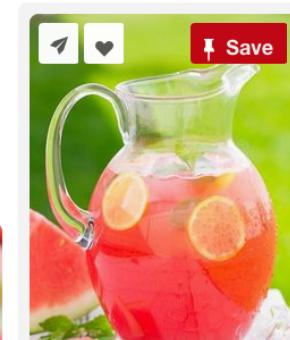
The Perfect Vanilla Pumpkin Smoothie: A Quick &...
The perfect vanilla pumpkin smoothie recipe. Quick, easy and...

BabySavers
Marybeth @ Bab...
Best Comfort Fo...



Spinach-Pear-Celery Smoothie
drink this daily and watch the pounds come off without fuss...

greenreset.com
Spring Stutzman
R - Drink Up



Pins to Pins Recommendations

Input:



Chocolate Strawberry Shake

249

This healthier chocolate strawberry shake is like sipping a...

One Lovely Life

Danielle Benzaia
Strawberries



Healthy Chocolate Peanut Butter Chips Muffins

119

Healthy Chocolate Peanut Butter Chip Muffins made with greek...

The First Year

Katie - You Brew ...
Healthy Recipes



221

The ULTIMATE Healthy Chocolate Chip Cookies -- so buttery...

Amv's Healthy Baking
Robin Guertin
healthy cooking

Pins to Pins Recommendations

Input:



This healthier chocolate strawberry shake is like sipping a...

One Lovely Life

Danielle Benzaia
Strawberries



Healthy Chocolate Peanut Butter Chip Muffins

Healthy Chocolate Peanut Butter Chip Muffins made with greek...

The First Year

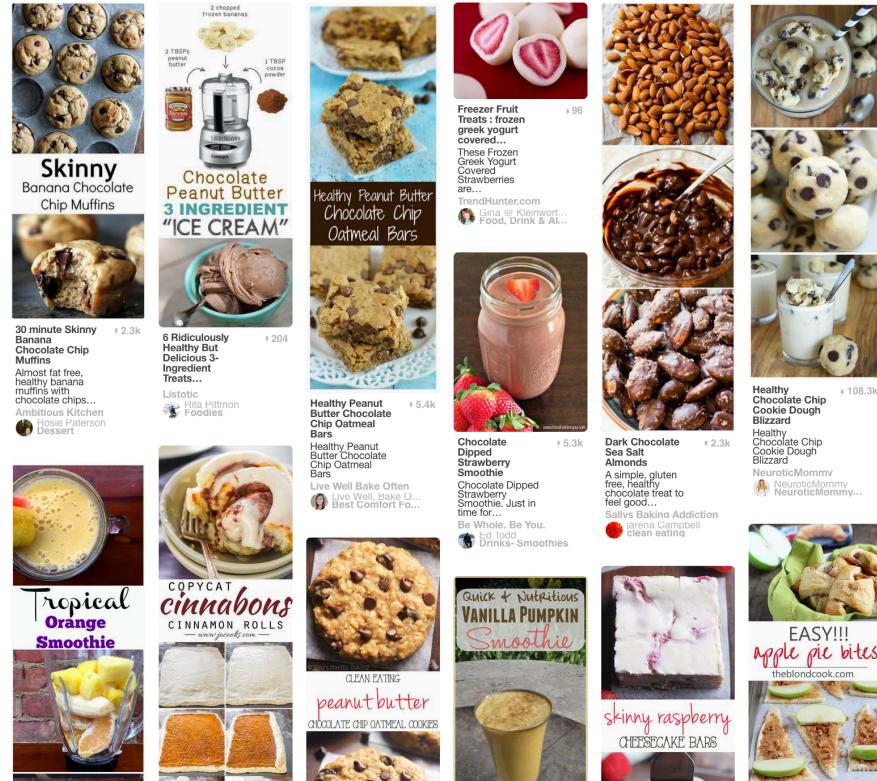
Katie - You Brew...
Healthy Recipes



The ULTIMATE Healthy Chocolate Chip Cookies -- so buttery...

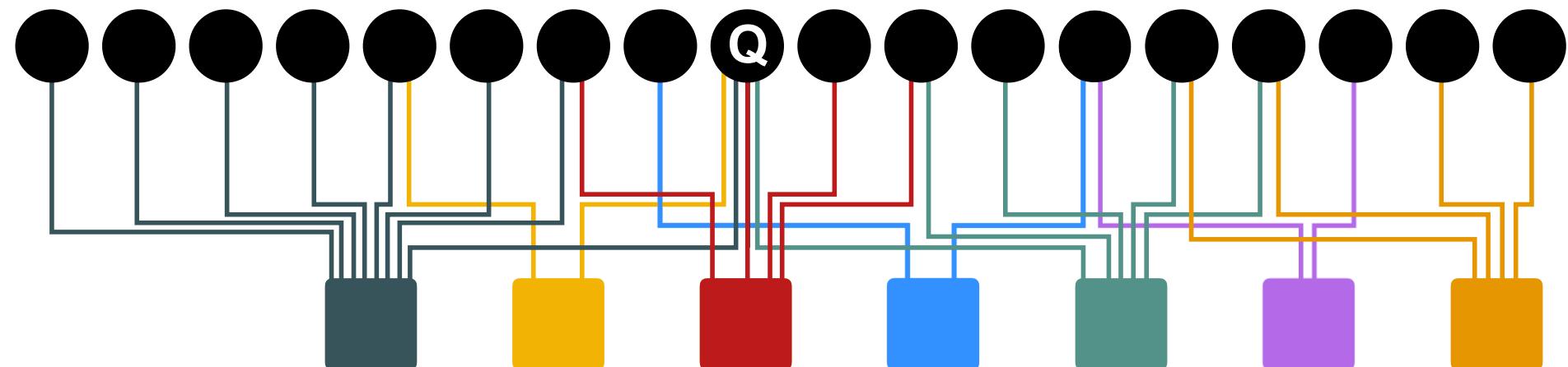
Amv's Healthy Baking
Robin Guertin
healthy cooking

Recommendations:



Random Walk with Restarts

- Idea
 - Every node has some importance
 - Importance gets evenly split among all edges and pushed to the neighbors
- Given a set of QUERY NODES Q, simulate a random walk:

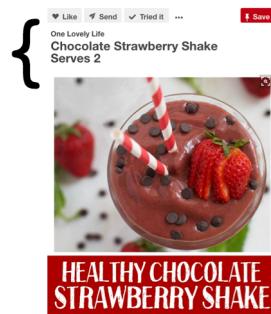


Random Walk Algorithm

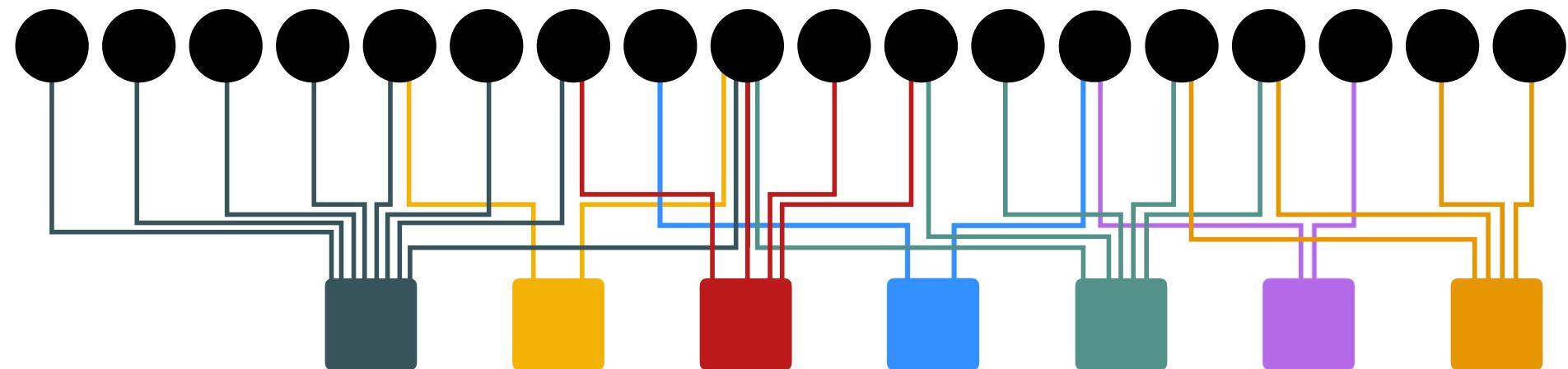
- Proximity to query node(s) Q :

```
ALPHA = 0.5
```

```
QUERY_NODES =
```



```
{ pin_node = QUERY_NODES.sample_by_weight()  
for i in range(N_STEPS):  
    board_node = pin_node.get_random_neighbor()  
    pin_node = board_node.get_random_neighbor()  
    pin_node.visit_count += 1  
    if random() < ALPHA:  
        pin_node = QUERY_NODES.sample_by_weight()
```



Random Walk Algorithm

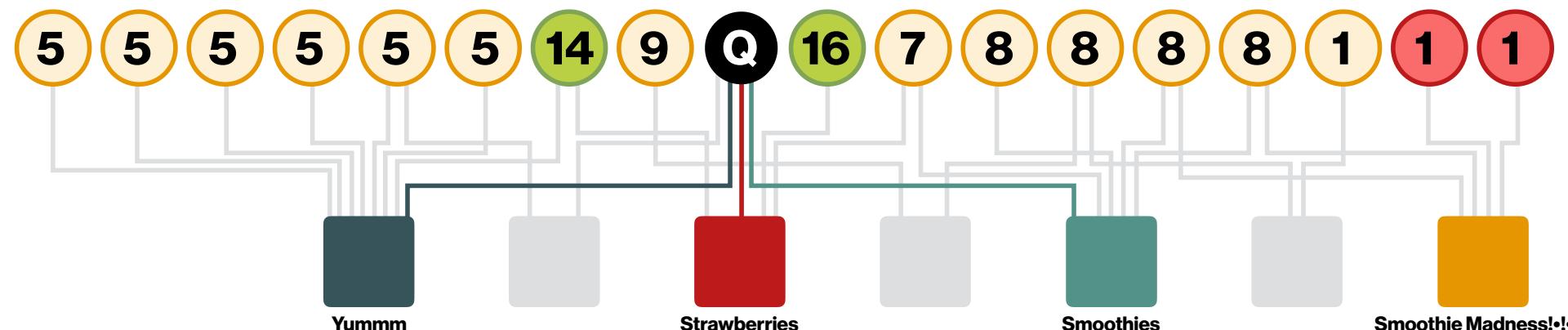
- Proximity to query node(s) Q :

```
ALPHA = 0.5
```

```
QUERY_NODES =
```



```
{ } pin_node = QUERY_NODES.sample_by_weight()  
for i in range(N_STEPS):  
    board_node = pin_node.get_random_neighbor()  
    pin_node = board_node.get_random_neighbor()  
    pin_node.visit_count += 1  
    if random() < ALPHA:  
        pin_node = QUERY_NODES.sample_by_weight()
```



How to make recommendations

- **Output top 1k pins with highest visit count**

Extensions:

- **Weighted edges:**

- The walk prefers to traverse certain edges:
 - Edges to pins in your local language

- **Early stopping:**

- Don't need to walk a fixed long number of steps
 - Walk until 1k-th pin has at least 20 visits

Benefits of Random Walks

- **Benefits:**
 - **Blazingly fast:** Given Q , we can output top 1k in 50ms (after doing 100k steps of the random walk)
 - Single machine can run 1500 walks in parallel! (1500 recommendation requests per second)
 - Can fit entire graph in RAM (17B edges, 3B nodes)
 - Can scale it by just adding more machines
 - Built in SNAP!
- Today about 80% of all the pins you see at Pinterest is recommended by random walks