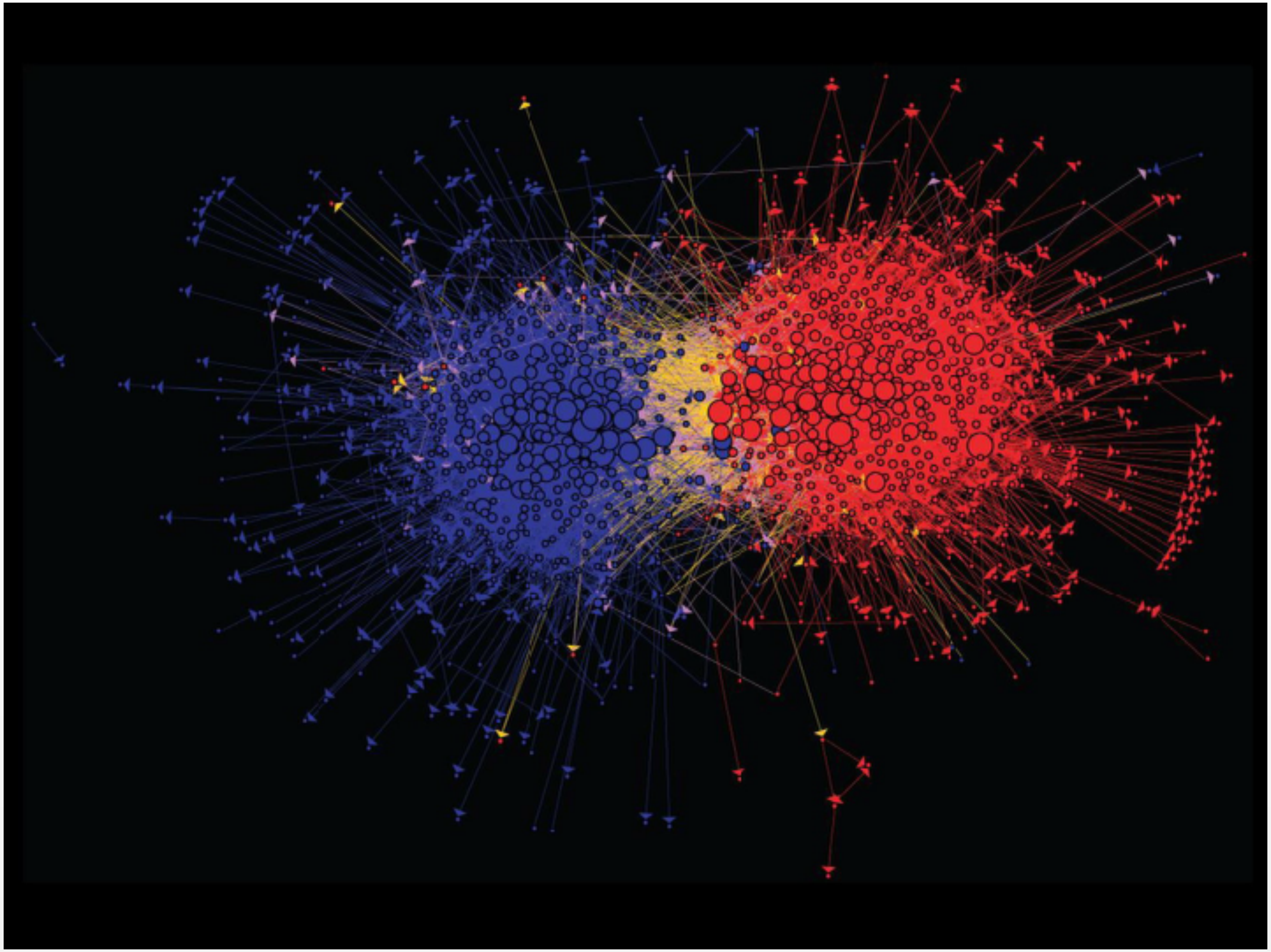
Data Mining Techniques

CS 6220 - Section 3 - Fall 2016

Lecture 17: Link Analysis

Jan-Willem van de Meent
(credit: Yijun Zhao, Yi Wang,
Tan et al., Leskovec et al.)

Graph Data: Media Networks



Connections between political blogs

Polarization of the network [Adamic-Glance, 2005]

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Schedule Updates

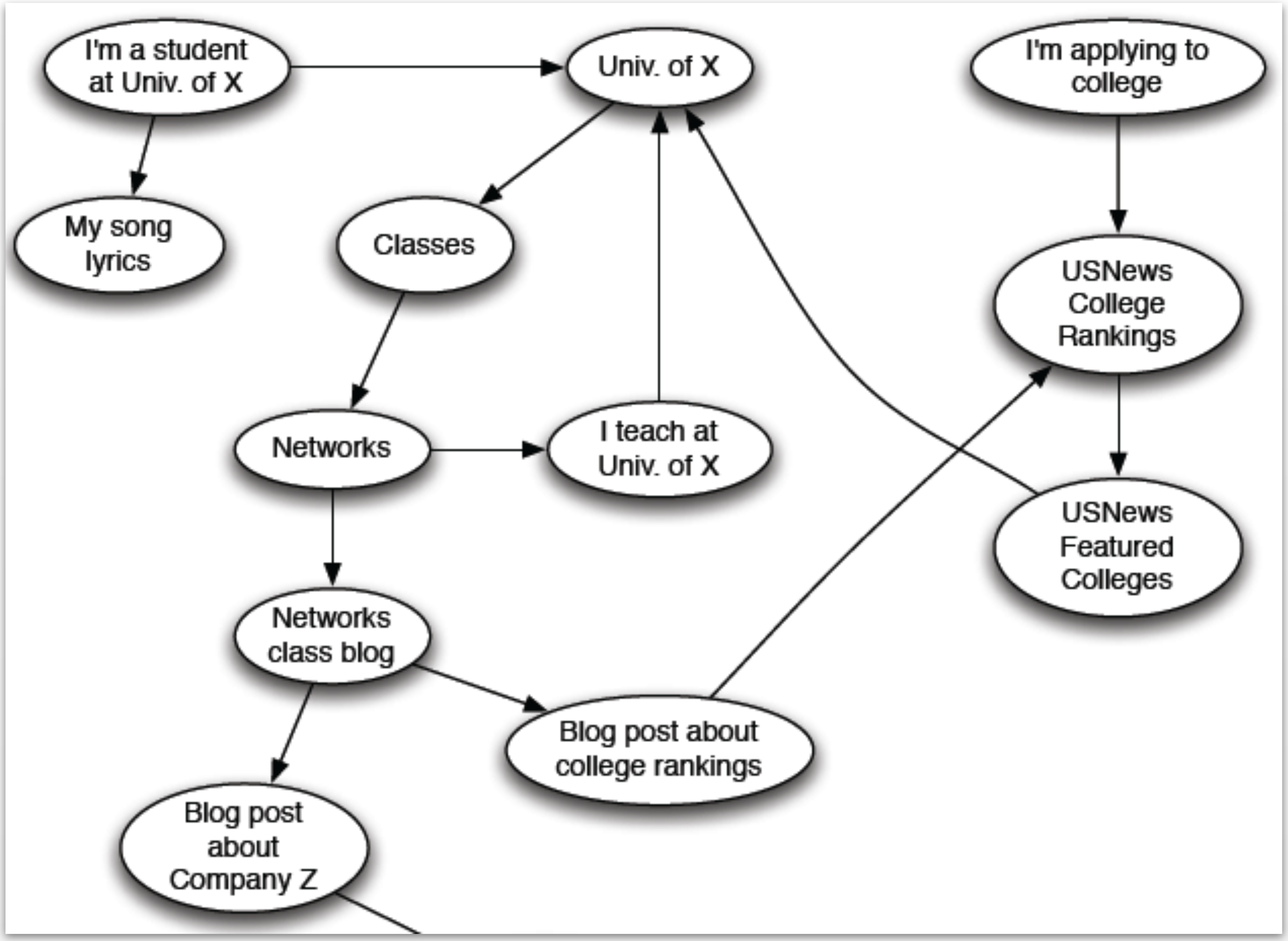
8	26 Oct	Midterm exam			
	28 Oct	Project Proposal presentations		Proposals due	
9	04 Nov	Frequent Pattern Mining 1: Apriori			HKP: 6; HTF: 14; Aggarwal: 4,5; TSK: 6
	07 Nov	Frequent Pattern Mining 2: PCY, FP-Growth			HKP: 6; HTF: 14; Aggarwal: 4,5; TSK: 6
10	09 Nov	Link Analysis: Page-rank, Trust-rank			LRU: 5; Aggarwal: 18.4
	11 Nov	(Veteran's Day)	#3 due		
11	16 Nov	Time Series: Hidden Markov Models			Bishop: 13.1-2; HKP: 13.1.1
	18 Nov	Community Detection: Betweenness, Spectral Clustering	#4 due		LRU: 10
12	23 Nov	(Thanksgiving Holiday)			
	25 Nov	(Thanksgiving Holiday)			
13	30 Nov	Bonus Topic: Deep Learning			
	02 Dec	Project Presentations			
14	07 Dec	(Review)			
	09 Dec	(Review)		Reports due	
15	14 Dec	Final Exam			
16	19 Dec	(Final grades posted)			

Web search before PageRank

- Human-curated
  (e.g. Yahoo, Looksmart)
  - Hand-written descriptions
  - Wait time for inclusion
- Text-search
  (e.g. WebCrawler, Lycos)
  - Prone to term-spam

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Web as a Directed Graph



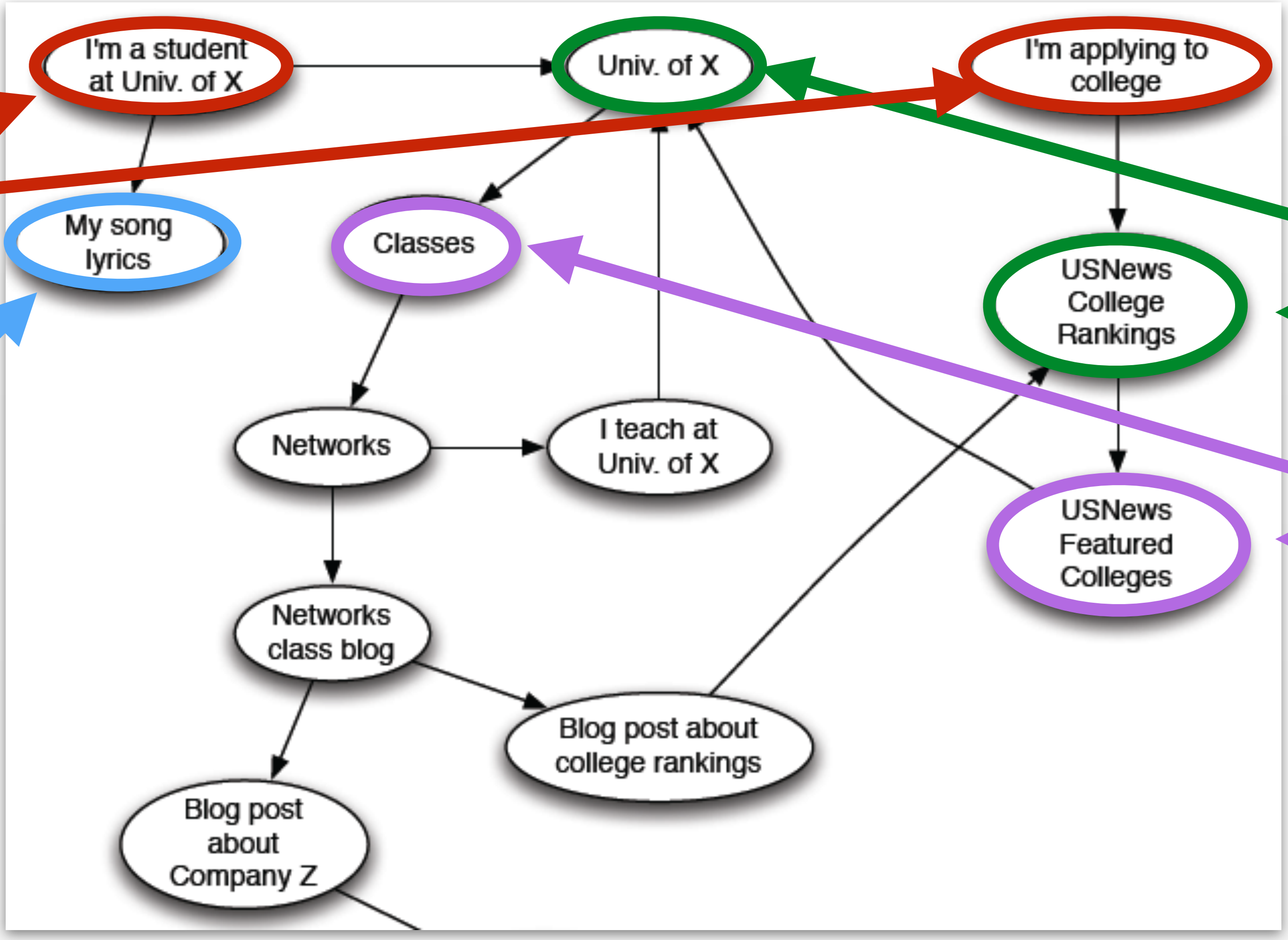
(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

PageRank: Links as Votes

Not all pages are equally important

Few/no inbound links

Many inbound links



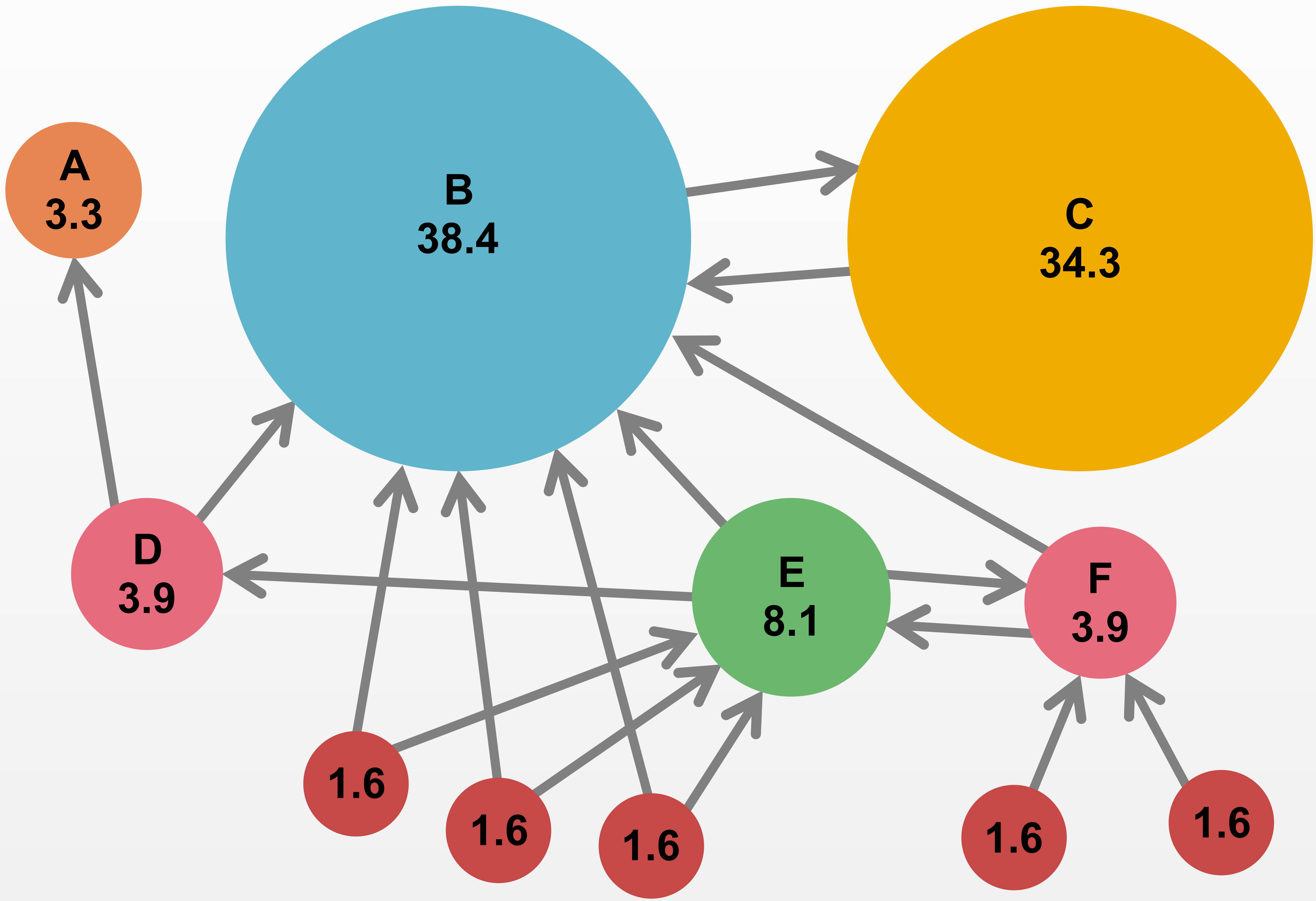
Links from unimportant pages

Links from important pages

- Pages with **more inbound links** are more **important**
- Inbound **links from important pages** carry **more weight**

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

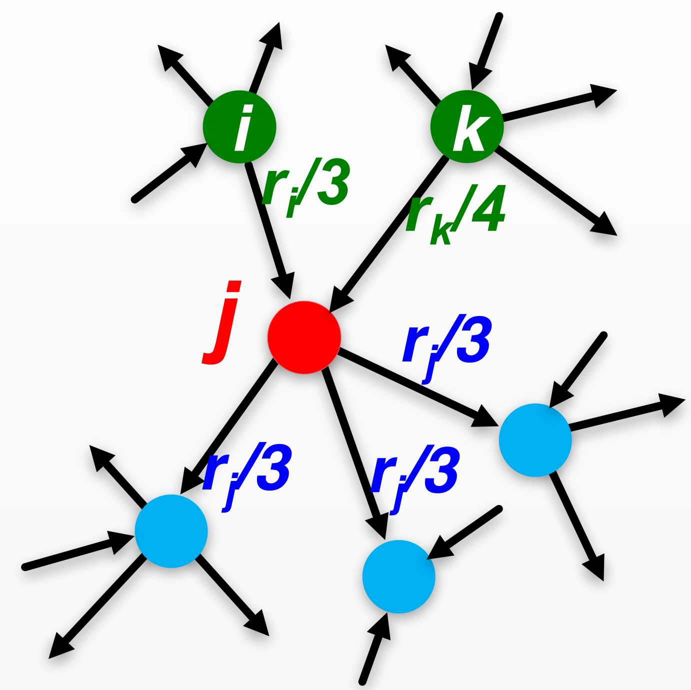
Example: PageRank Scores



(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Simple Recursive Formulation

$r_j = r_i/3 + r_k/4$

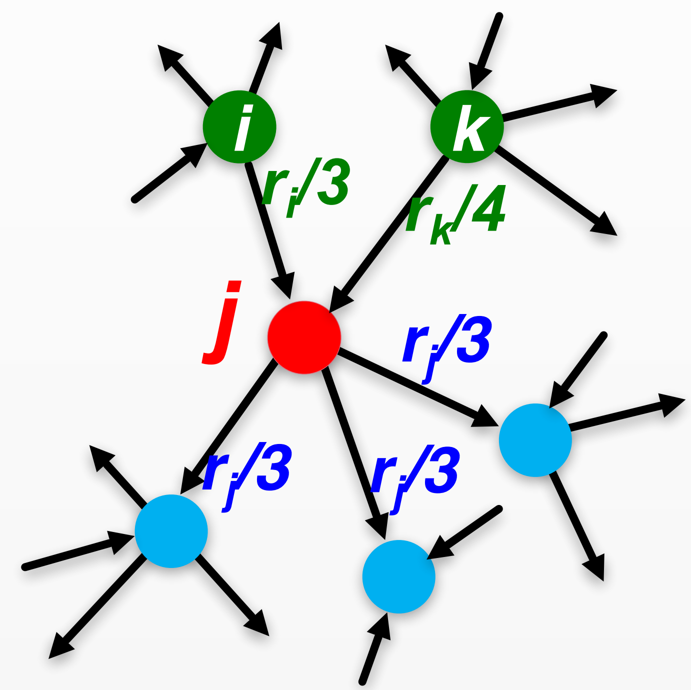


- A link's vote is proportional to the importance of its source page
- If page j with importance r_j has n out-links, each link gets r_j / n votes
- Page j's own importance is the sum of the votes on its in-links

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Equivalent Formulation: Random Surfer

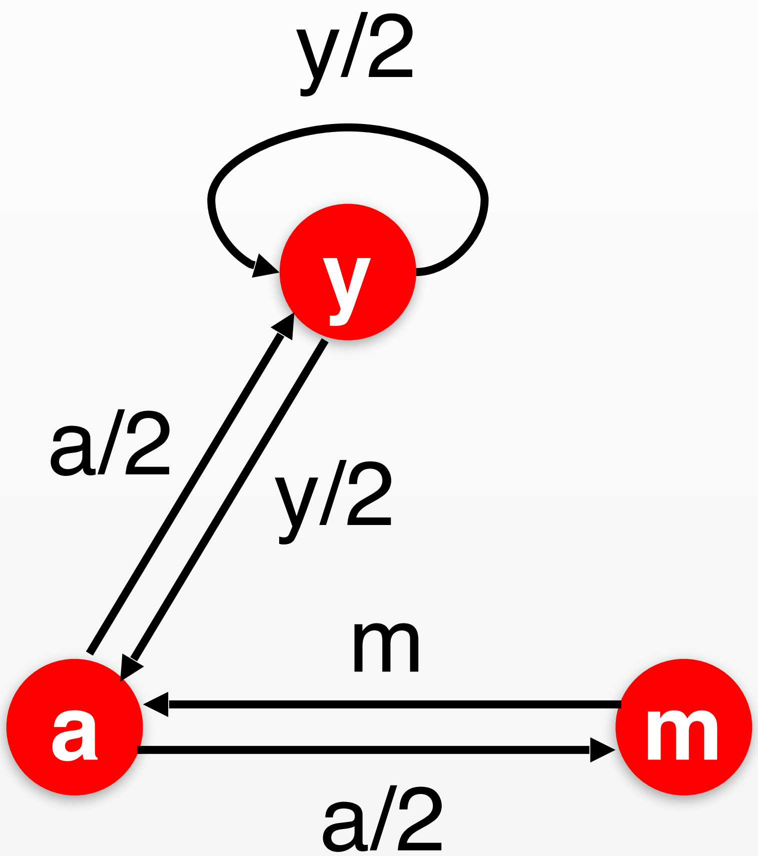
$r_j = r_i/3 + r_k/4$



- At time *t* a surfer is on some page *i*
- At time *t+1* the surfer follows a link to a new page at random
- Define rank r_i as fraction of time spent on page *i*

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

PageRank: The “Flow” Model



$$r_j = \sum_{i \to j} \frac{r_i}{\mathrm{d}_i}$$

“Flow” equations:

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

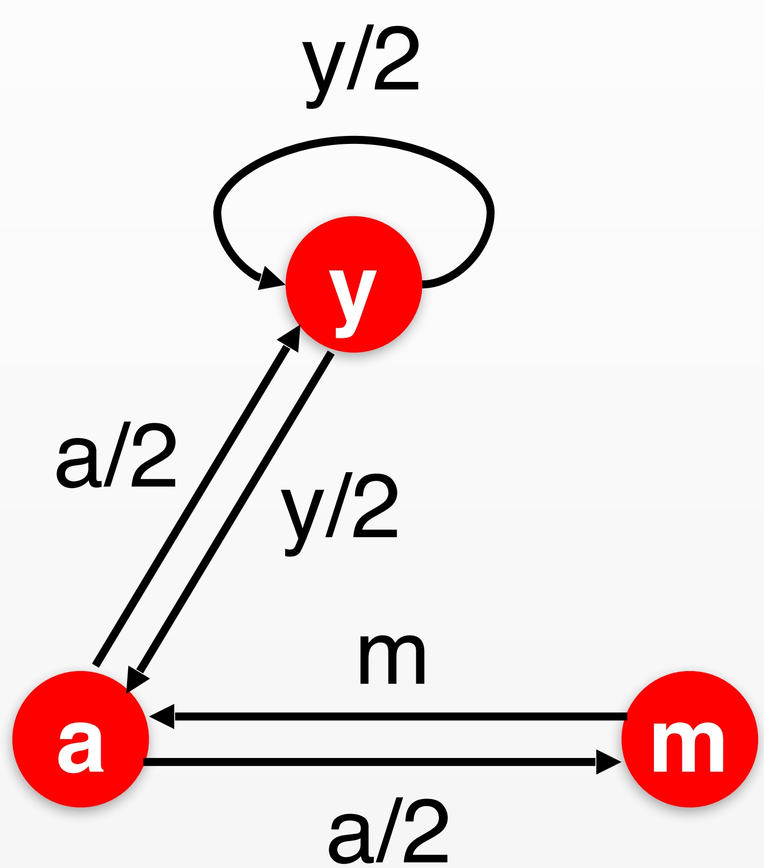
$$r_a = r_y/2 + r_m$$

$$r_m = r_a/2$$

- 3 equations, 3 unknowns
- Impose constraint: $r_y + r_a + r_m = 1$
- Solution: $r_y = 2/5,\ r_a = 2/5,\ r_m = 1/5$

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

PageRank: The “Flow” Model



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

“Flow” equations:

$r_y = r_y/2 + r_a/2$

$r_a = r_y/2 + r_m$

$r_m = r_a/2$

r = M · r

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 1 \\ 0 & \frac{1}{2} & 0 \end{bmatrix} \begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix}$$

Matrix *M* is stochastic (i.e. columns sum to one)

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

PageRank: Eigenvector Problem

- PageRank: Solve for eigenvector $r = M\,r$ with eigenvalue $\lambda = 1$
- Eigenvector with $\lambda = 1$ is guaranteed to exist since M is a stochastic matrix (i.e. if $a = M\,b$ then $\Sigma\, a_i = \Sigma\, b_i$)
- *Problem*: There are billions of pages on the internet. How do we solve for eigenvector with order 10^{10} elements?

PageRank: Power Iteration

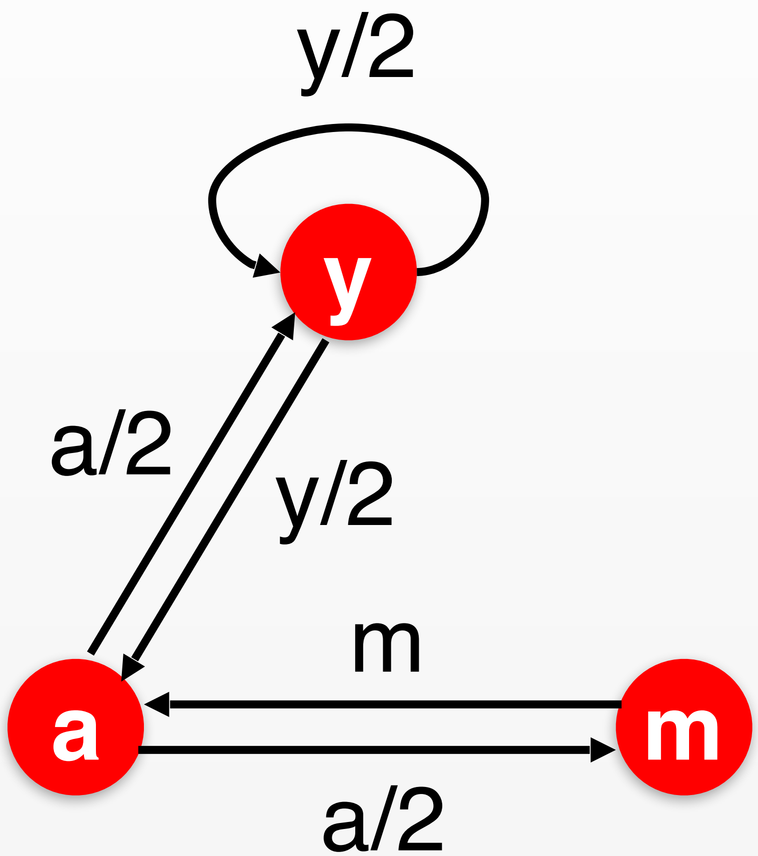
Model for random Surfer:

- At time $t = 0$ pick a page at random
- At each subsequent time t follow an outgoing link at random

Probabilistic interpretation:

$$
\begin{aligned}
p(z_0 = i) &= 1/N \\
p(z_t = i \,|\, z_{t-1} = j) &= M_{ij} \\
p(z_t = i) &= \textstyle\sum_j p(z_t = i, z_{t-1} = j) \\
&= \textstyle\sum_j M_{ij} p(z_{t-1} = j)
\end{aligned}
$$

PageRank: Power Iteration



$$\boldsymbol{p}^t = M\boldsymbol{p}^{t-1} = M^t\boldsymbol{p}^0$$

$$\boldsymbol{p}^0 = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \quad M = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 1 \\ 0 & \frac{1}{2} & 0 \end{bmatrix}$$

$$\boldsymbol{p}^t = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \begin{bmatrix} 2/6 \\ 3/6 \\ 1/6 \end{bmatrix} \begin{bmatrix} 5/12 \\ 4/12 \\ 3/12 \end{bmatrix} \begin{bmatrix} 9/24 \\ 11/24 \\ 4/24 \end{bmatrix} \begin{bmatrix} 20/48 \\ 17/48 \\ 11/48 \end{bmatrix} \simeq \begin{bmatrix} 2/5 \\ 2/5 \\ 1/5 \end{bmatrix}$$

p^t converges to r. Iterate until $|p^t - p^{t-1}| < \varepsilon$

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Aside: Ergodicity

- PageRank is assumes a *random walk* model for individual surfers
- *Equivalent assumption*: flow model in which equal fractions of surfers follow each link at every time
- *Ergodicity:* The equilibrium of the flow model is the same as the asymptotic distribution for an individual random walk

$$\boldsymbol{r} = \boldsymbol{M}\boldsymbol{r} \qquad \boldsymbol{p}^t = M\boldsymbol{p}^{t-1} \qquad \lim_{t\to\infty} \boldsymbol{p}^t = \boldsymbol{r}$$

Aside: Ergodicity

- PageRank is assumes a *random walk* model for individual surfers
- *Equivalent assumption*: flow model in which equal fractions of surfers follow each link at every time
- *Ergodicity:* The equilibrium of the flow model is the same as the asymptotic distribution for an individual random walk

$$p(z_t = i) = \sum_j M_{ij} p(z_{t-1} = j)$$

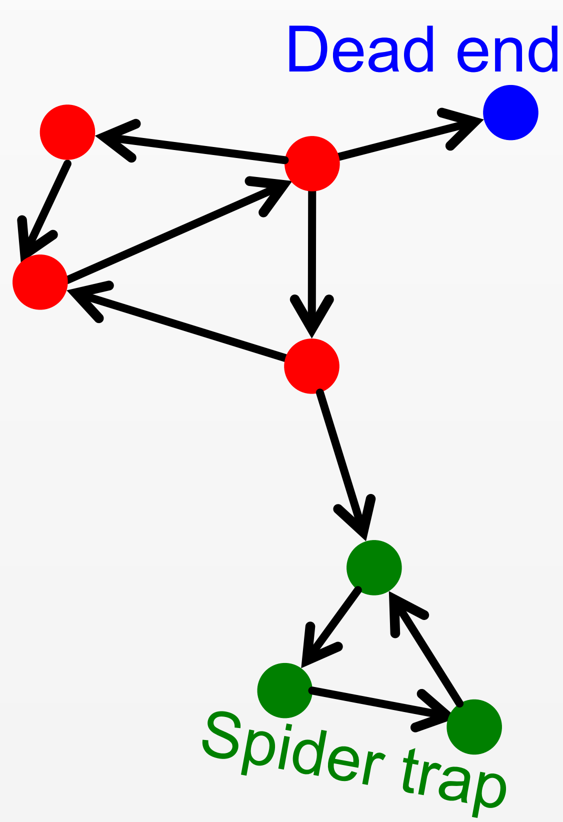
$$\lim_{T \to \infty} \mathbb{E}\left[\frac{1}{T} \sum_{t=1}^{T} I[z_t = i] \right] = r_i$$

Aside: Ergodicity

- PageRank is assumes a *random walk* model for individual surfers
- *Equivalent assumption*: flow model in which equal fractions of surfers follow each link at every time
- *Ergodicity:* The equilibrium of the flow model is the same as the asymptotic distribution for an individual random walk

Averaging over individuals is equivalent to averaging single individual over time

PageRank: Problems

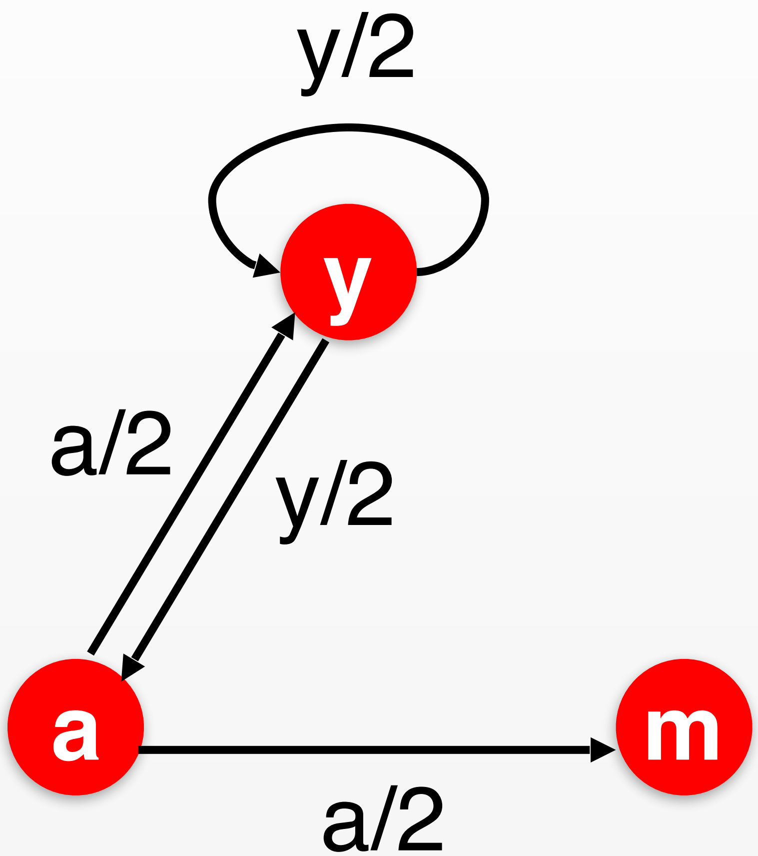


1. *Dead Ends*
   - Nodes with no outgoing links.
   - Where do surfers go next?

2. *Spider Traps*
   - Subgraph with no outgoing links to wider graph
   - Surfers are “trapped” with no way out.

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Power Iteration: Dead Ends



$$\boldsymbol{p}^t = M\boldsymbol{p}^{t-1} = M^t\boldsymbol{p}^0$$

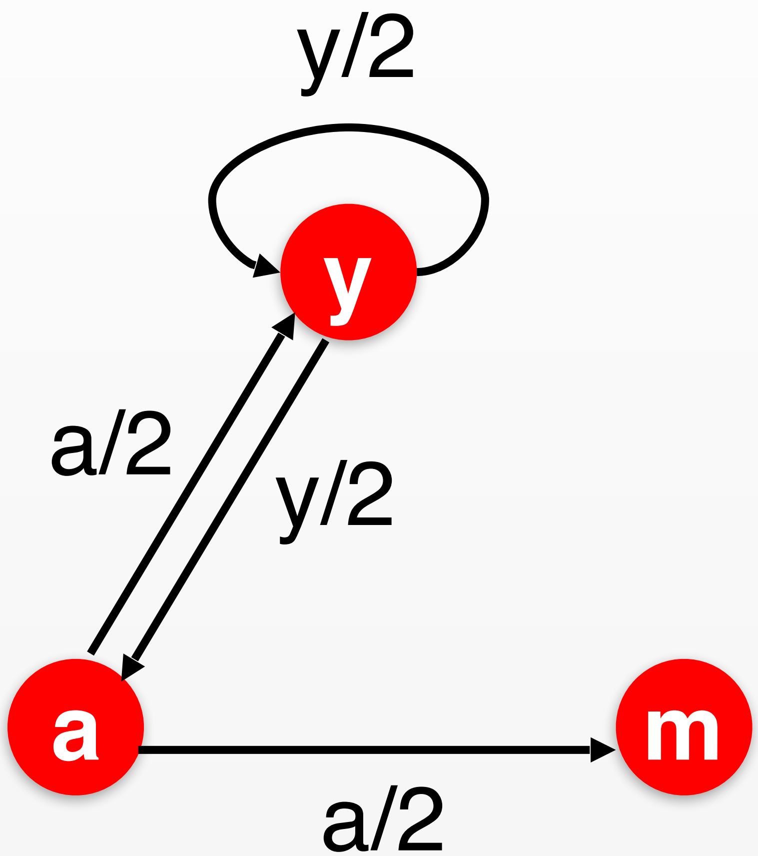
$$\boldsymbol{p}^0 = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \quad M = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \end{bmatrix}$$

$$\boldsymbol{p}^t = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \begin{bmatrix} 2/6 \\ 1/6 \\ 1/6 \end{bmatrix} \begin{bmatrix} 3/12 \\ 1/12 \\ 1/12 \end{bmatrix} \dots \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Probability not conserved

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Power Iteration: Dead Ends



$$\boldsymbol{p}^t = M\boldsymbol{p}^{t-1} = M^t\boldsymbol{p}^0$$

$$\boldsymbol{p}^0 = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \quad M = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & 0 & \frac{1}{3} \\ 0 & \frac{1}{2} & \frac{1}{3} \end{bmatrix}$$

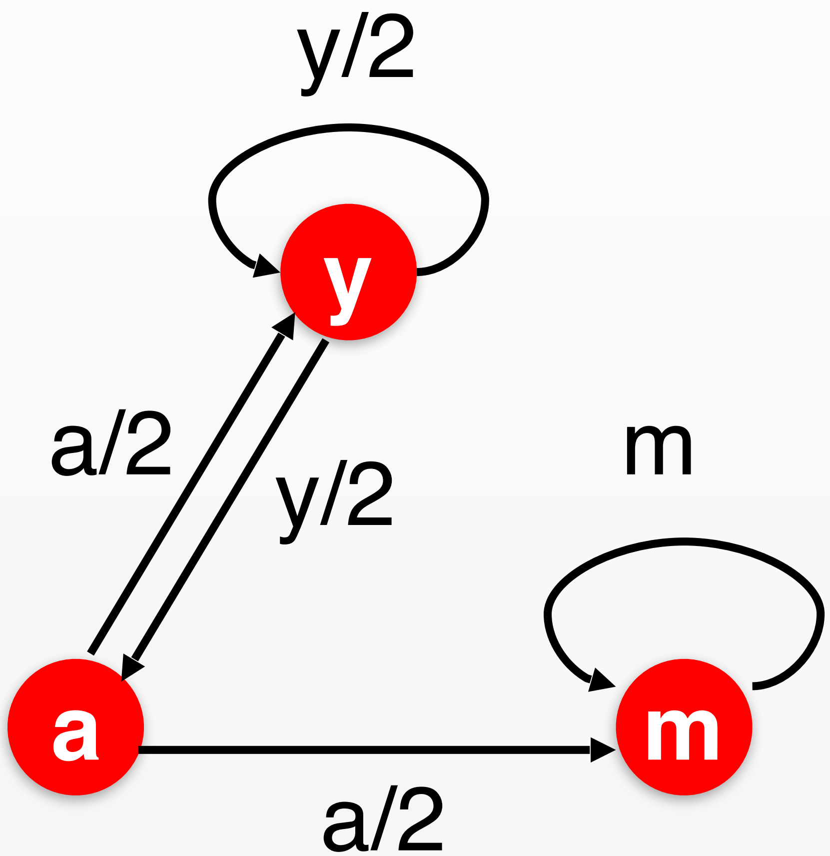
(teleport at dead ends)

$$\boldsymbol{p}^t = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \begin{bmatrix} 8/18 \\ 5/18 \\ 5/18 \end{bmatrix} \begin{bmatrix} 49/108 \\ 34/108 \\ 35/108 \end{bmatrix} \ldots$$

Fixes "probability sink" issue

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Power Iteration: Spider Traps



$$\boldsymbol{p}^t = M\boldsymbol{p}^{t-1} = M^t\boldsymbol{p}^0$$

$$\boldsymbol{p}^0 = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \quad M = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 1 \end{bmatrix}$$

$$\boldsymbol{p}^t = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \begin{bmatrix} 2/6 \\ 1/6 \\ 3/6 \end{bmatrix} \begin{bmatrix} 3/12 \\ 2/12 \\ 7/12 \end{bmatrix} \begin{bmatrix} 5/24 \\ 3/24 \\ 16/24 \end{bmatrix} \dots \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Probability accumulates in traps (surfers get stuck)

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Solution: Random Teleports

Model for teleporting random surfer:

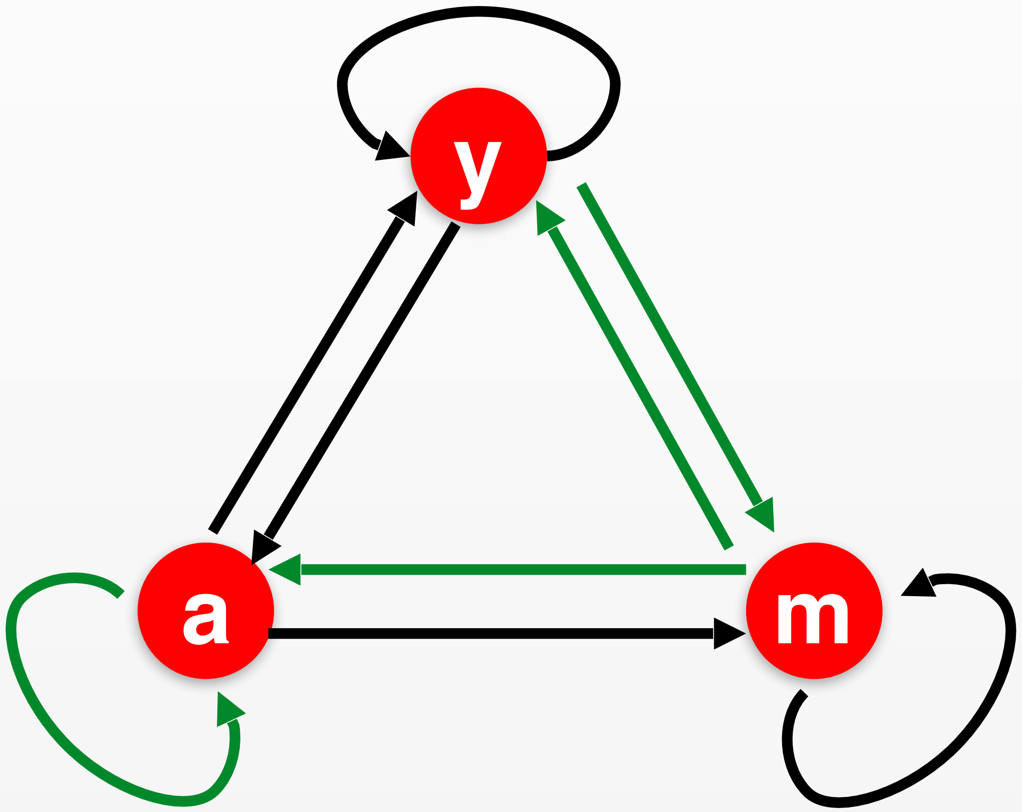
- At time $t = 0$ pick a page at random
- At each subsequent time t
  - With probability β follow an outgoing link at random
  - With probability 1-β teleport to a new initial location at random

PageRank Equation [Page & Brin 1998]

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

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Power Iteration: Teleports



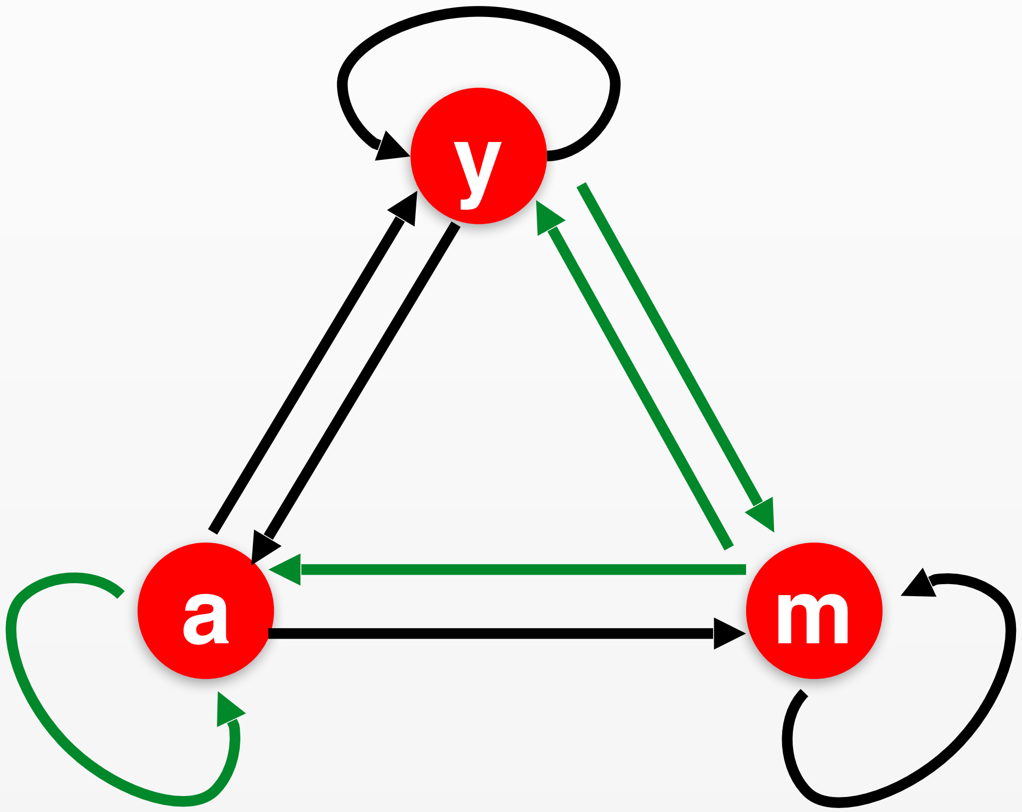
$$\boldsymbol{p}^t = \beta M \boldsymbol{p}^{t-1} + (1-\beta)\boldsymbol{p}^0 = \tilde{M}\boldsymbol{p}^{t-1}$$

$$\tilde{M} = \beta M + (1-\beta)\begin{bmatrix} - & p_1^0 & - \\ & \dots & \\ - & p_N^0 & - \end{bmatrix}$$

(can use power iteration as normal)

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Power Iteration: Teleports



$$\boldsymbol{p}^t = \beta M \boldsymbol{p}^{t-1} + (1-\beta)\boldsymbol{p}^0 = \tilde{M}\boldsymbol{p}^{t-1}$$

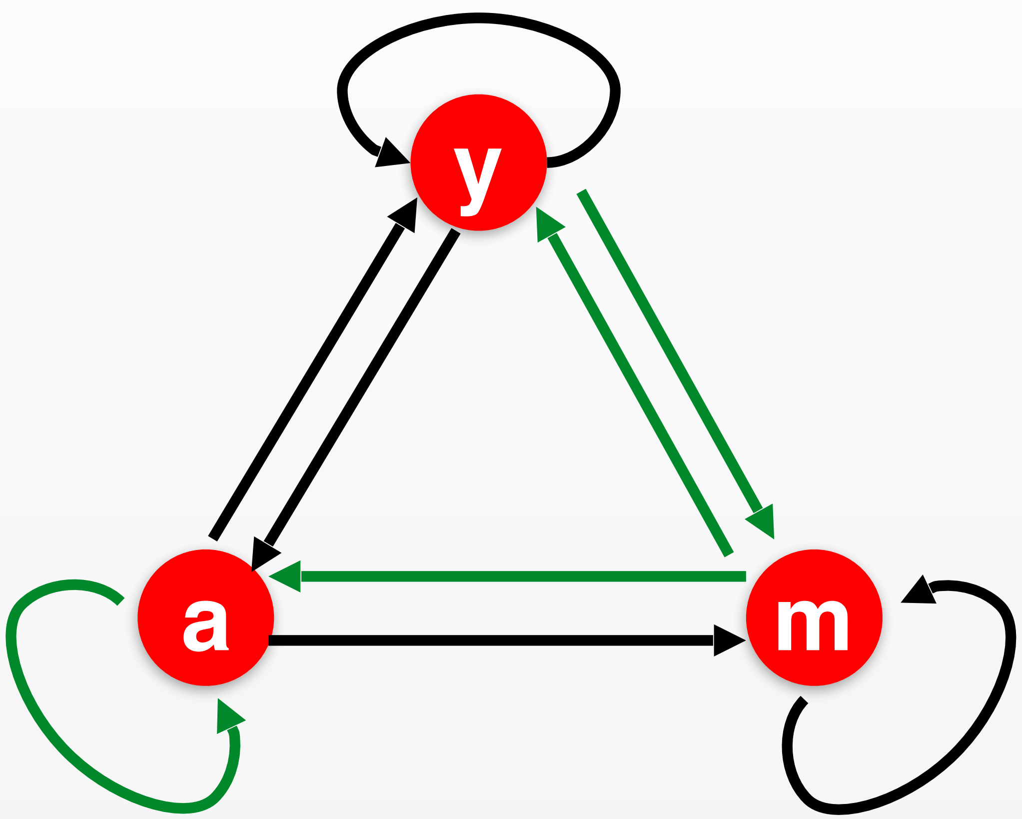
$$\tilde{M} = \beta M + (1-\beta)\begin{bmatrix} - & p_1^0 & - \\ & \dots & \\ - & p_N^0 & - \end{bmatrix}$$

(can use power iteration as normal)

$$\tilde{M} = 4/5\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 1 \end{bmatrix} + 1/5\begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} = \begin{bmatrix} \frac{7}{15} & \frac{7}{15} & \frac{1}{15} \\ \frac{7}{15} & \frac{1}{15} & \frac{1}{15} \\ \frac{1}{15} & \frac{7}{15} & \frac{1}{15} \end{bmatrix}$$

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Power Iteration: Teleports



$$\boldsymbol{p}^t = \tilde{M}\boldsymbol{p}^{t-1} = \tilde{M}^t\boldsymbol{p}^0$$

$$\boldsymbol{p}^0 = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \quad \tilde{M} = \begin{bmatrix} \frac{7}{15} & \frac{7}{15} & \frac{1}{15} \\ \frac{7}{15} & \frac{1}{15} & \frac{1}{15} \\ \frac{1}{15} & \frac{7}{15} & \frac{1}{15} \end{bmatrix}$$

(can use power iteration as normal)

$$\boldsymbol{p}^t = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \begin{bmatrix} 0.33 \\ 0.20 \\ 0.46 \end{bmatrix} \begin{bmatrix} 0.24 \\ 0.20 \\ 0.56 \end{bmatrix} \ldots \begin{bmatrix} 7/33 \\ 5/33 \\ 21/33 \end{bmatrix}$$

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Computing PageRank

$$\boldsymbol{p}^t = \beta M \boldsymbol{p}^t + \frac{1-\beta}{N}$$

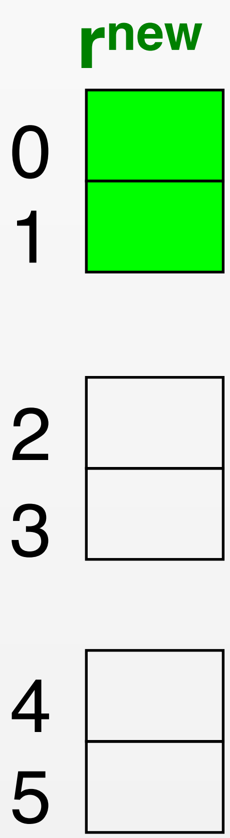
- *M* is sparse - only store nonzero entries
  - Space proportional roughly to number of links
  - Say 10N, or 4*10*1 billion = 40GB
  - Still won't fit in memory, but will fit on disk

source node	**degree**	**destination nodes**
0	3	1, 5, 7
1	5	17, 64, 113, 117, 245
2	2	13, 23

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

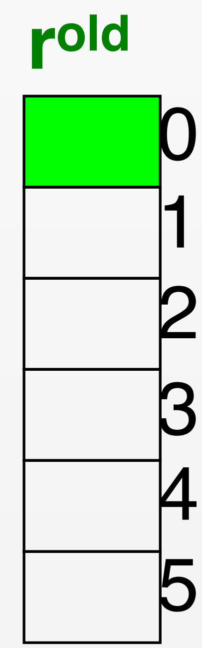
Block-based Update Algorithm

- Break r^{new} into k blocks that fit in memory
- Scan M and r^{old} once for each block



src	degree	destination
0	4	0, 1, 3, 5
1	2	0, 5
2	2	3, 4

M



(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Block-Stripe Update Algorithm

Break M into stripes: Each stripe contains only destination nodes in the corresponding block of r^{new}

r^{new}

0
1

2
3

4
5

src	degree	destination
0	4	0, 1
1	3	0
2	2	1

0	4	3
2	2	3

0	4	5
1	3	5
2	2	4

r^{old}

0
1
2
3
4
5

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

First Spammers: Term Spam

- How do you make your page appear to be about movies?
  - (1) Add the word movie 1,000 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

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

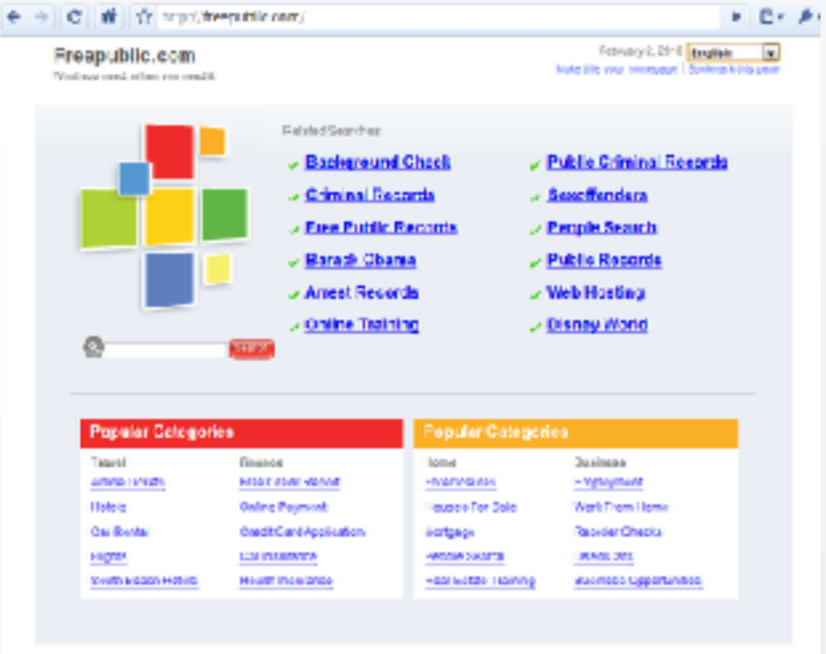
Google's Solution to Term Spam

- 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

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

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



(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

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
  - Owned pages
    - Completely controlled by spammer
    - May span multiple domain names

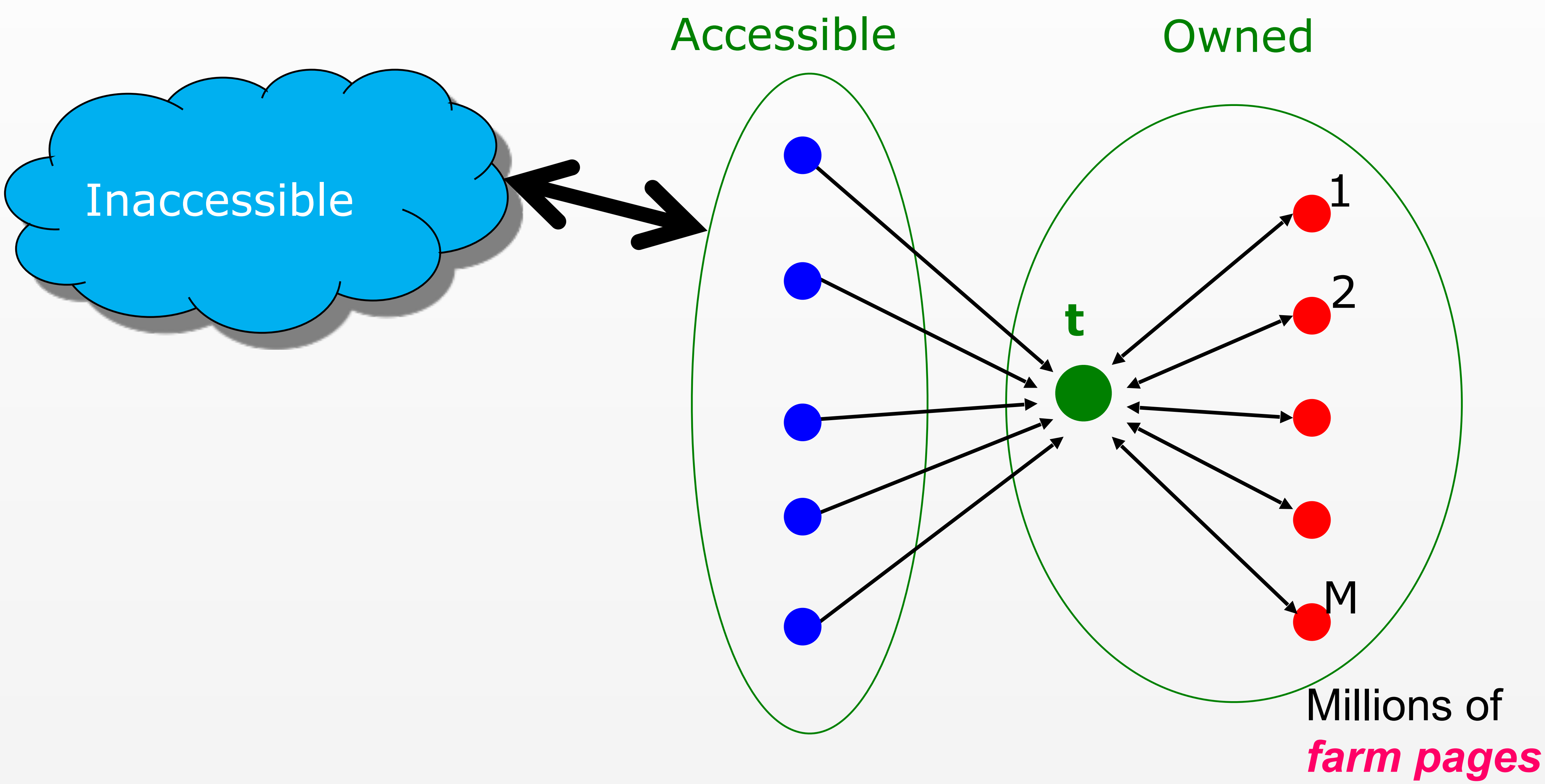
(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

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

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

Link Farms



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

(adapted from:: Mining of Massive Datasets, http://www.mmds.org)

PageRank: Extensions

$$\boldsymbol{p}^t = \beta M \boldsymbol{p}^{t-1} + (1-\beta)\boldsymbol{p}^0 = \tilde{M}\boldsymbol{p}^{t-1}$$

- *Topic-specific PageRank*:
  - Restrict teleportation to some set ***S*** of pages related to a specific topic
  - Set $p^0_i = 1/|S|$ if $i \in S$, $p^0_i = 0$ otherwise
- *Trust Propagation*
  - Use set ***S*** of trusted pages for teleport set