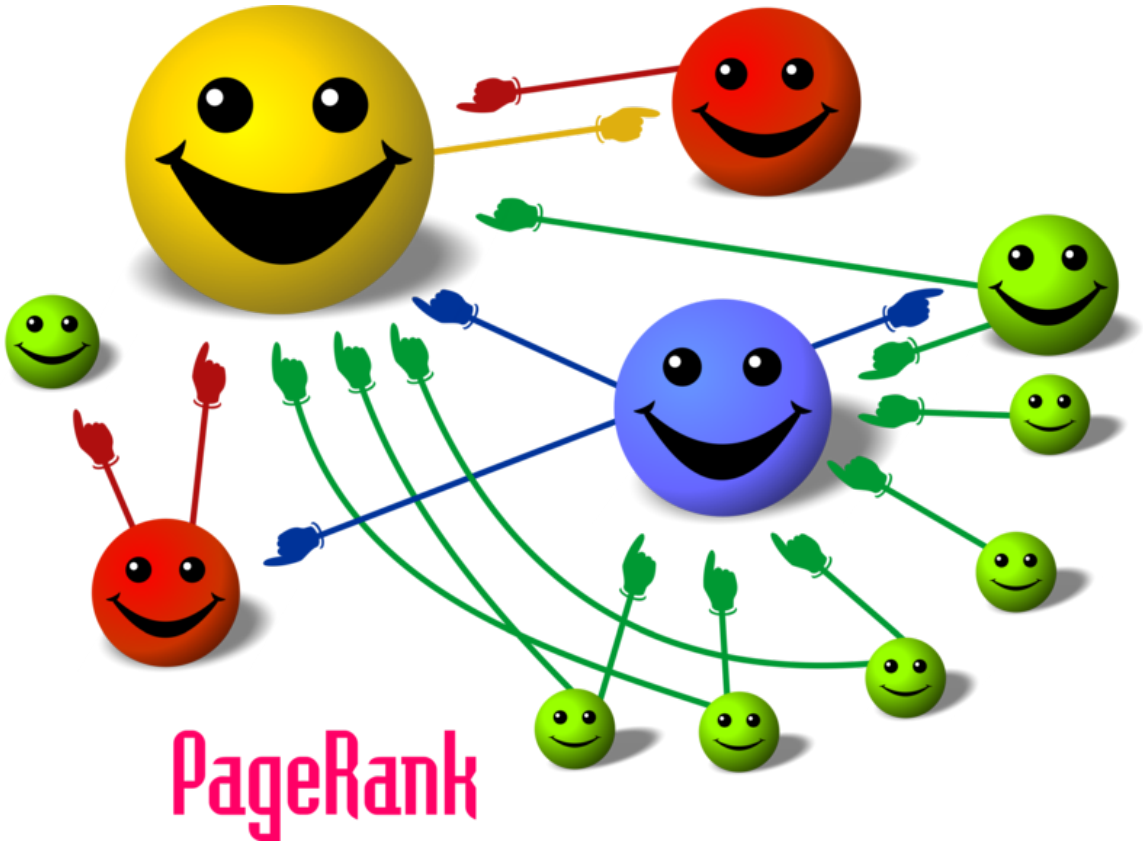
IR2: Link Analysis



Thx to Dan Weld, James Moody, Dragomir Radev

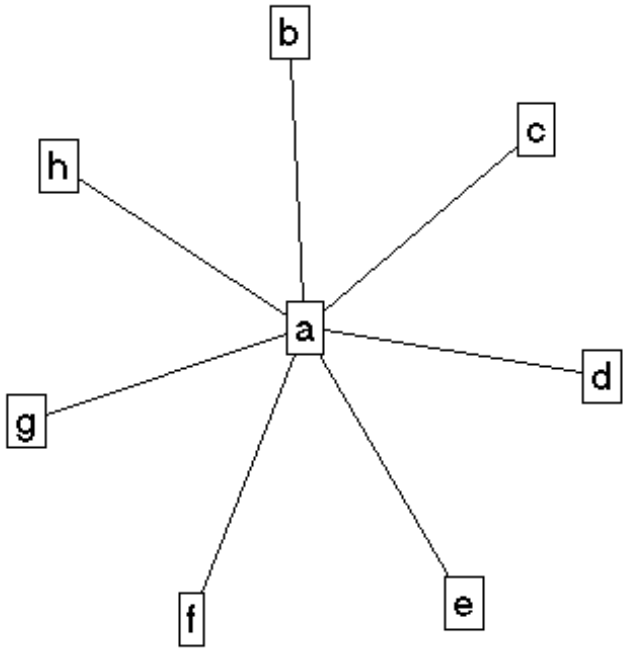
Challenges

- Three challenges in web search:
  - Result relevance
  - Processing speed
  - Scaling to many documents

- We'll continue to cover result relevance today

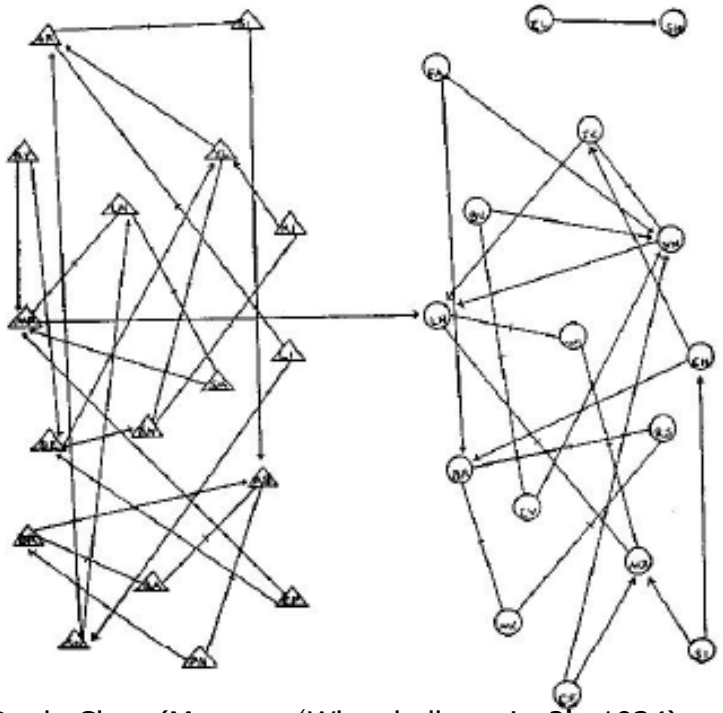
Graphs (or Networks)

- Describe relation among items
- Symmetric or directed
- Have been around for a long time
  - Friendship networks
  - Board membership
  - Paper citations
  - US power grid
  - Web pages



Early social network analysis

- School kids – favorite (and captive) subjects of study
- These days much more difficult because need parental consent to gather social network data

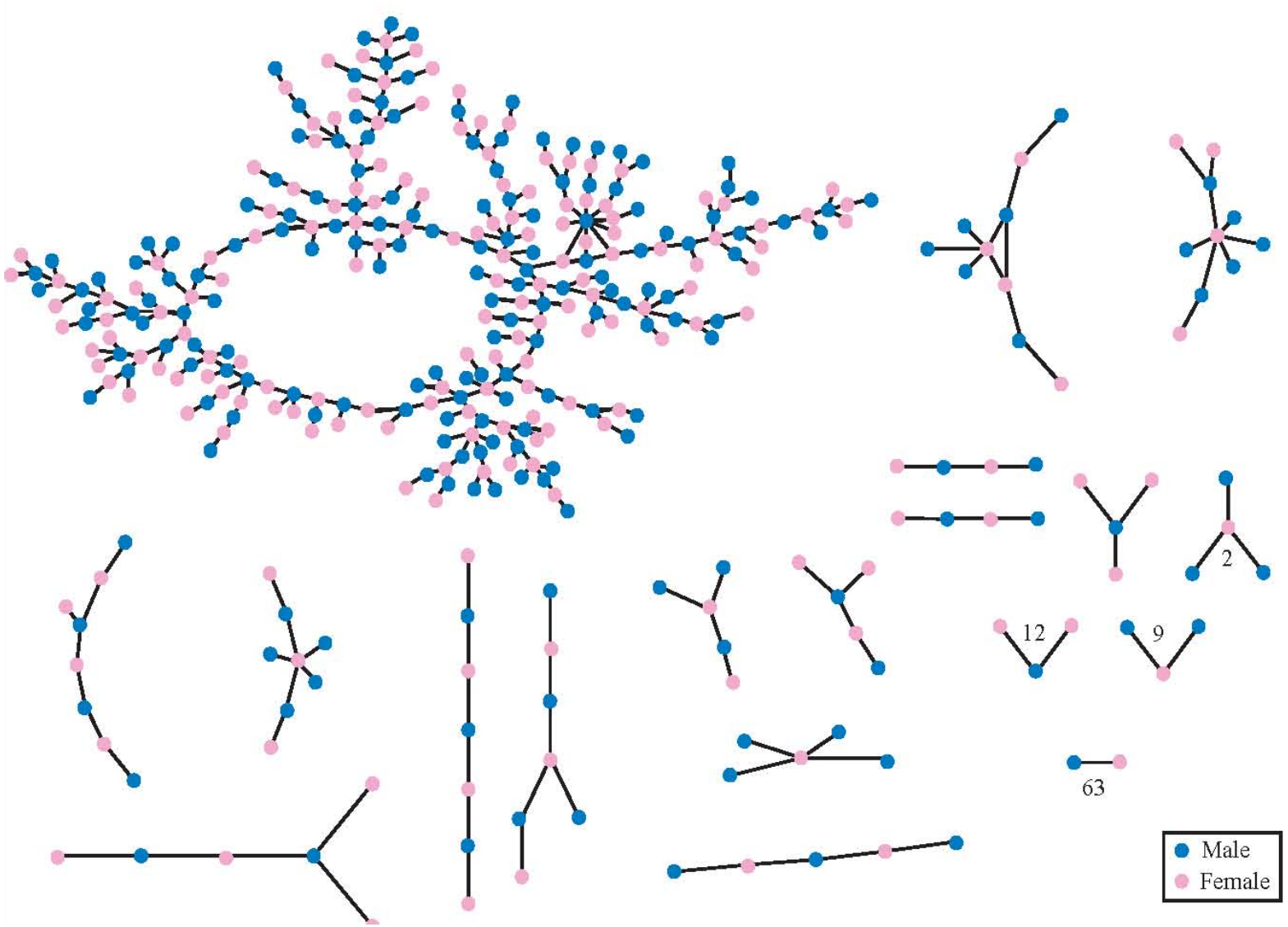


△ *boys*

○ *girls*

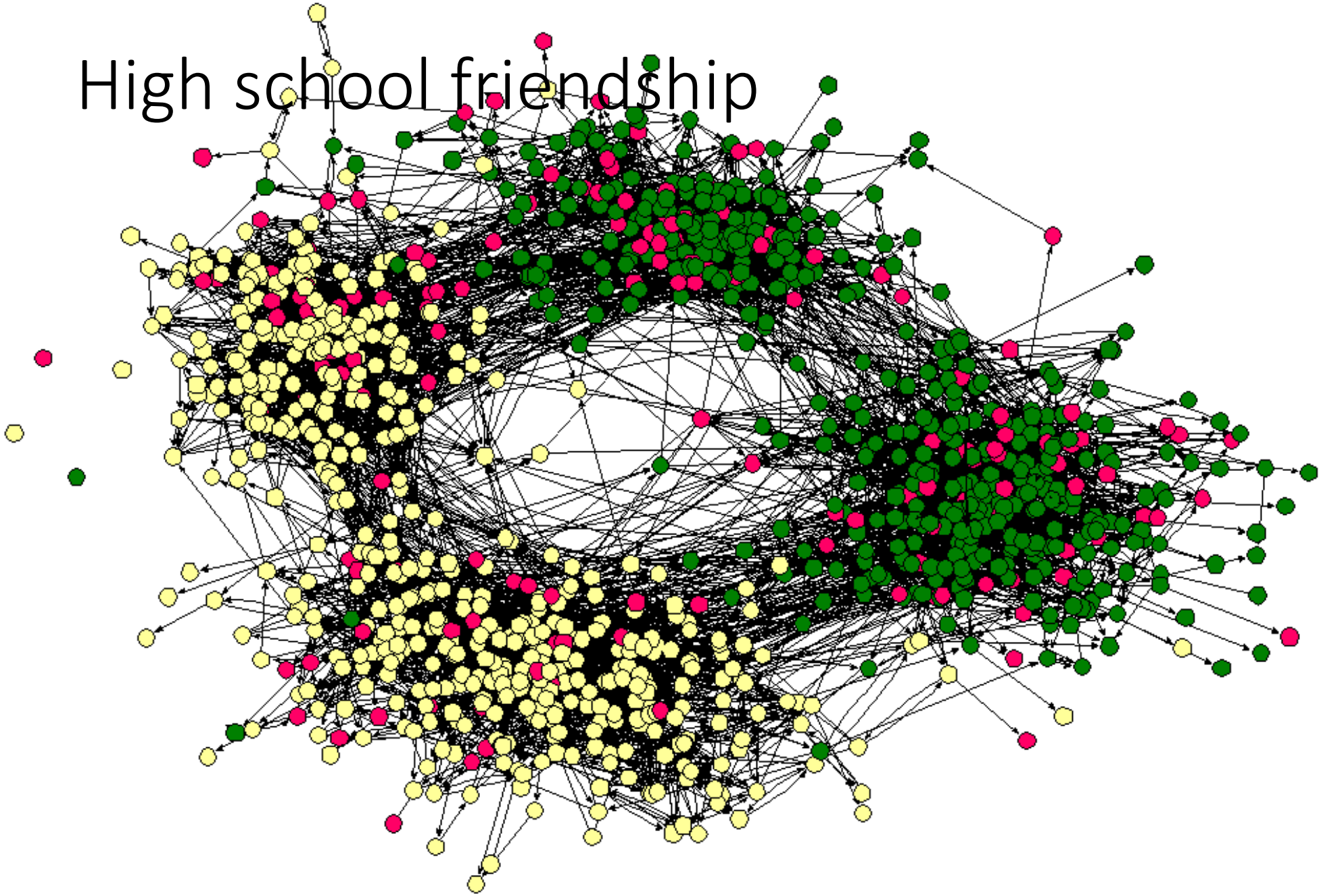
An Attraction Network in a Fourth Grade Class (Moreno, 'Who shall survive?', 1934).

High school dating



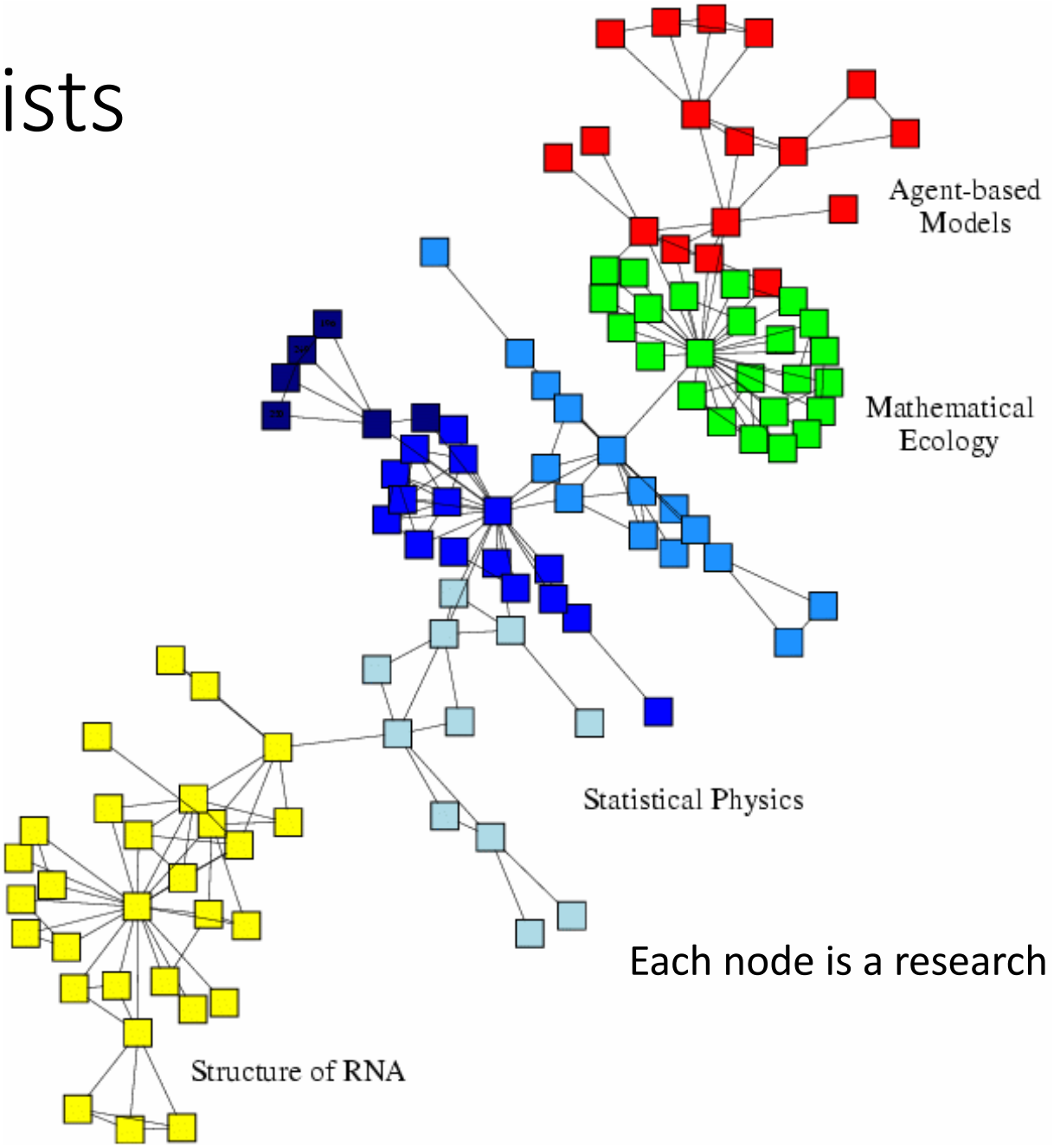
Chains of affection: The structure of adolescent romantic and sexual networks, Bearman, et al., American Journal of Sociology 110, 44-91 (2004)

High school friendship



High school friendship: James Moody, Race, school integration, and friendship segregation in America, American Journal of Sociology 107, 679-716 (2001).

Scientists

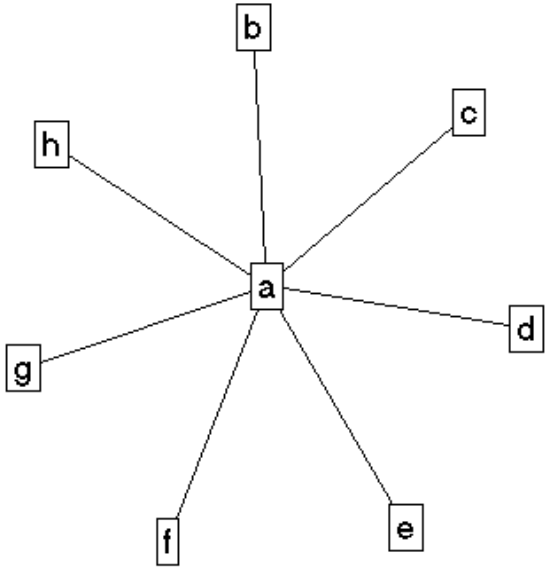


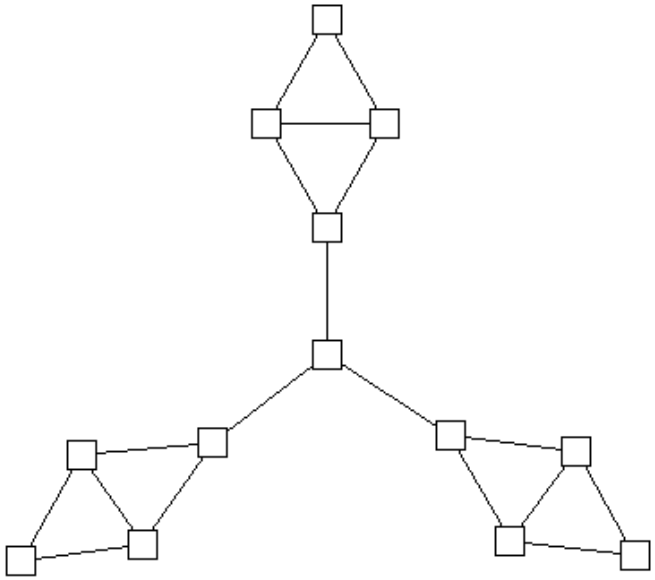
Each node is a research scientist

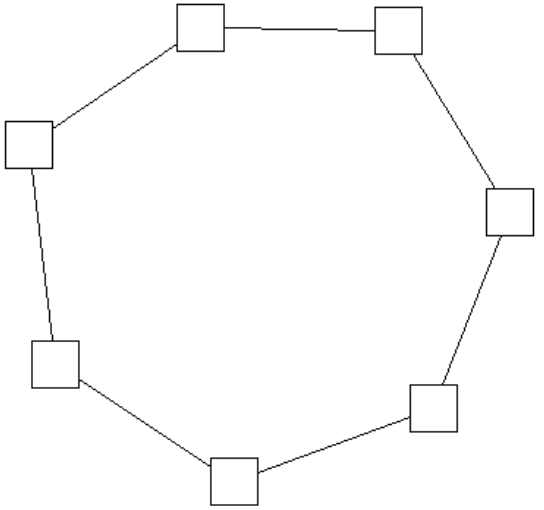
M. Girvan and M. E. J. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. USA 99, 8271-8276 (2002).

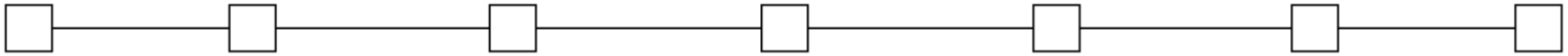
Prestige and importance

- Which node(s) are the most important?
- How would you measure it?







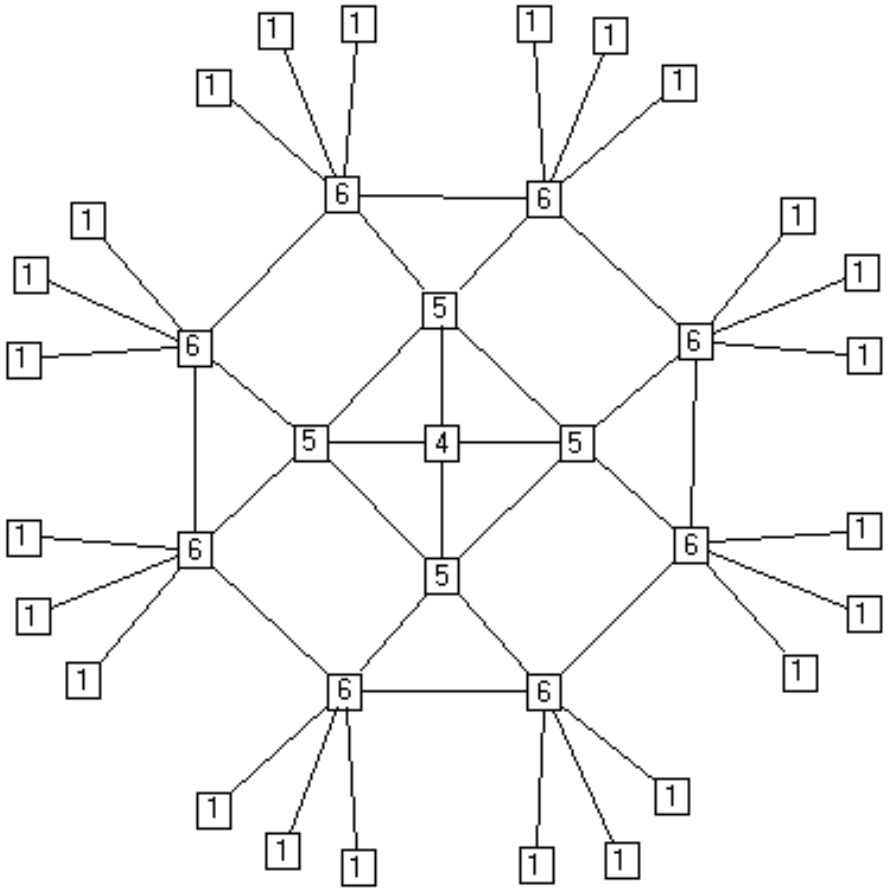


Prestige and importance

- Which node(s) are the most important?
- How would you measure it?
  - # links?
  - # "2-deep links"?
  - position in the graph?
- This is also sometimes called determining "centrality", especially in social network research

Prestige and importance

- Degree centrality is one way
  - Just count the links!



Prestige and importance

- Another way: measure closeness
- Node is important if it is close to all others
- Based on inverse of distance from each node to every other node

$$C_c(n_i) = \left[\sum_{j=1}^{g} d(n_i, n_j) \right]^{-1}$$

Prestige and impor

$$C_c(n_i) = \left[\sum_{j=1}^{g} d(n_i, n_j) \right]^{-1}$$

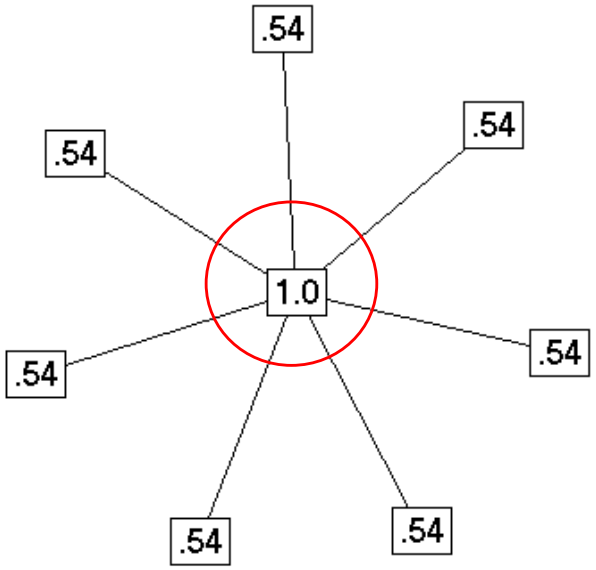
- Draw a graph that would cause one node to achieve the best possible score

- What would that node's score be?

Prestige and impor

$$C_c(n_i) = \left[\sum_{j=1}^{g} d(n_i, n_j) \right]^{-1}$$

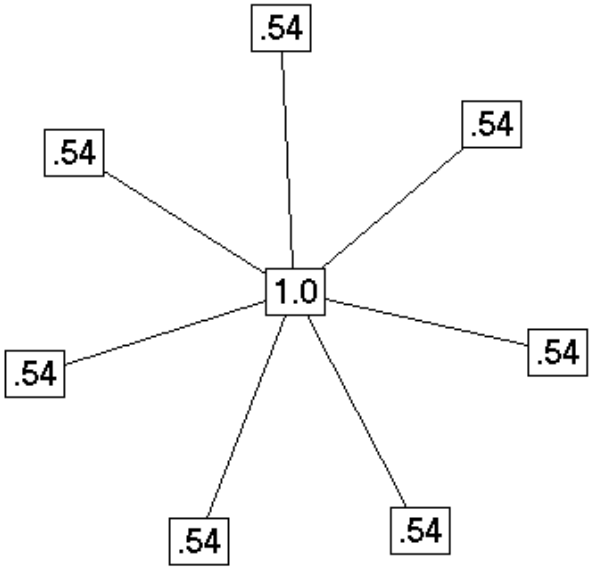
- Draw a graph that would cause one node to achieve the best possible score



- What would that node's score be?
  - 1 / (n-1)
  - 1 / 7 = 0.143 in this example
- We'll normalize every score in the graph to this

Closeness

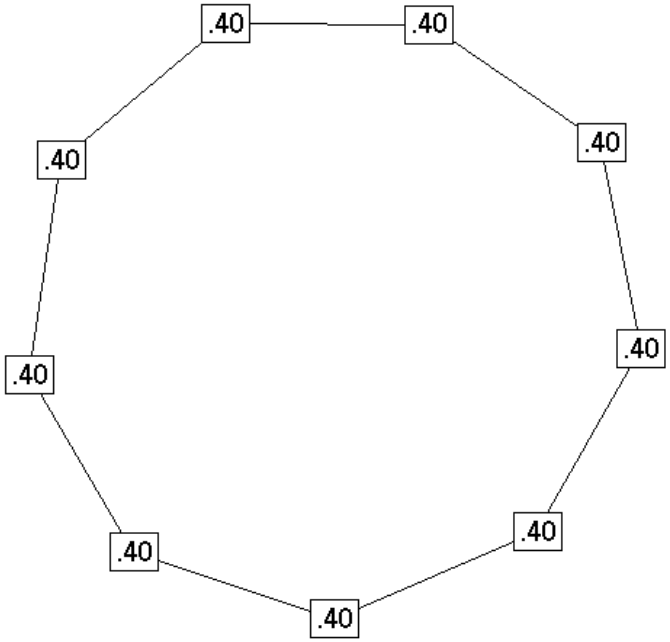
$$C_c(n_i) = \left[\sum_{j=1}^{g} d(n_i, n_j) \right]^{-1}$$



	Distance	Closeness	normalized
0 1 1 1 1 1 1		.143	1.00
1 0 2 2 2 2 2		.077	.538
1 2 0 2 2 2 2		.077	.538
1 2 2 0 2 2 2		.077	.538
1 2 2 2 0 2 2		.077	.538
1 2 2 2 2 0 2		.077	.538
1 2 2 2 2 2 0		.077	.538
1 2 2 2 2 2 2	0	.077	.538

Closeness

$$C_c(n_i) = \left[\sum_{j=1}^{g} d(n_i, n_j) \right]^{-1}$$



Distance	Closeness	normalized
0 1 2 3 4 4 3 2 1	.050	.400
1 0 1 2 3 4 4 3 2	.050	.400
2 1 0 1 2 3 4 4 3	.050	.400
3 2 1 0 1 2 3 4 4	.050	.400
4 3 2 1 0 1 2 3 4	.050	.400
4 4 3 2 1 0 1 2 3	.050	.400
3 4 4 3 2 1 0 1 2	.050	.400
2 3 4 4 3 2 1 0 1	.050	.400
1 2 3 4 4 3 2 1 0	.050	.400

Closeness

$$C_c(n_i) = \left[\sum_{j=1}^{g} d(n_i, n_j) \right]^{-1}$$

Distance	Closeness	normalized
0 1 2 3 4 5 6	.048	.286
1 0 1 2 3 4 5	.063	.375
2 1 0 1 2 3 4	.077	.462
3 2 1 0 1 2 3	.083	.500
4 3 2 1 0 1 2	.077	.462
5 4 3 2 1 0 1	.063	.375
6 5 4 3 2 1 0	.048	.286

Prestige & Importance

- Other ideas:
  - Identify nodes with smallest max-distance to all other nodes
  - *Betweenness* - for what fraction of paths is the node along the path?
  - Bonacich Power Centrality, aka *proximity-to-prestige*. A node's importance depends on the importance of its neighbors
  - Academic impact analysis
- These ideas came about before the Web, but very relevant

Web Link Analysis

- Search in late 1990s was pretty bad
  - Content growth outstripped human editors
- Lots of Web interest in 1997-1999 in using the hyperlink graph
  - **PageRank**, Page
  - **HITS**, Kleinberg
  - **"Silk from a sow's ear"**, Pirolli, Pitkow, Rao
- Can measure "importance", but that's not all

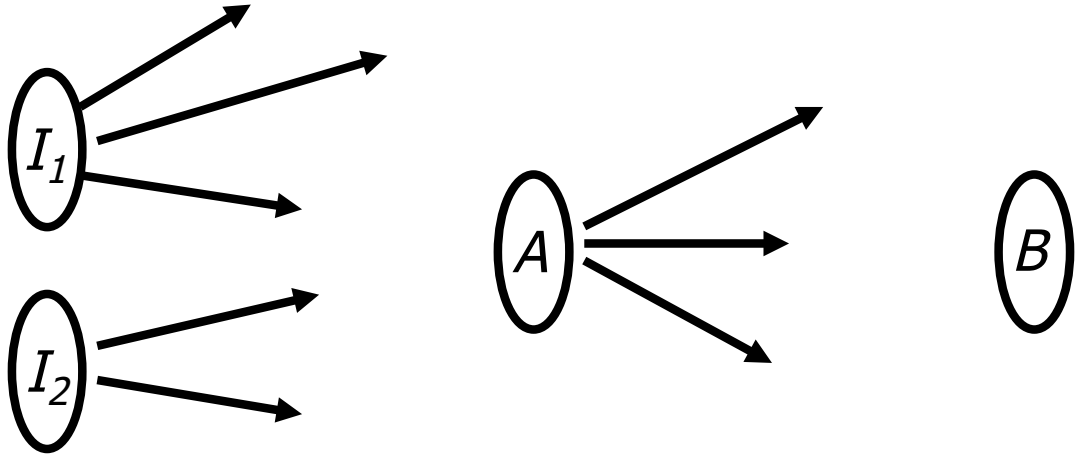
PageRank

- For first time, search engines got the right page
  - AltaVista used to rank pages by URL length
  - When PageRank hit, it was astonishing
- Intuition:
  - Web is a big directed graph
  - A "random surfer" clicks at random
  - Importance of a page = probability the surfer is on the page
  - Suppose *P* has *N* outgoing links; surfer clicks on link with probability *1/N*
  - Query-independent!!!

PageRank Intuition

- You have an adjacency matrix E where e[i,j]=1 if i cites j
  - It describes the Web
- Each node in the graph gets a PageRank score, p_u for node *u*
- Each site in the Web votes for important sites by linking to them
  - Weigh votes according to importance of sender
  - How is importance of sender determined?
  - With its PageRank score!
- PageRank is defined recursively (and computed iteratively)

PageRank



- A node with C links contributes 1/C of its PageRank to each target node

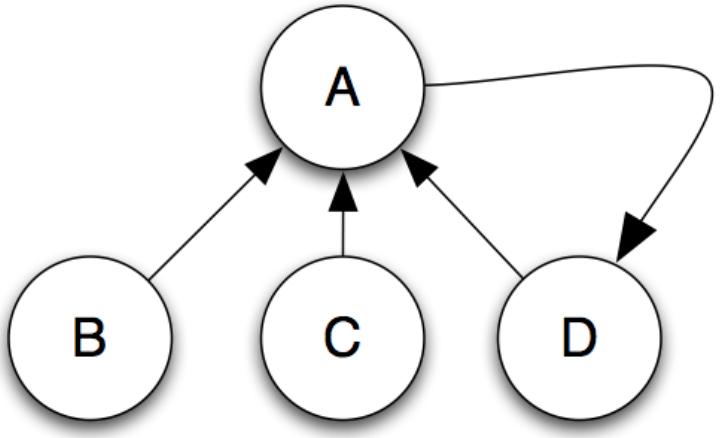
$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

- Damping factor d is usually 0.85

PageRank Example

$$PR(A) = \frac{(1-d)}{N} + d\sum_{i}\frac{PR(I_i)}{C(I_i)}$$

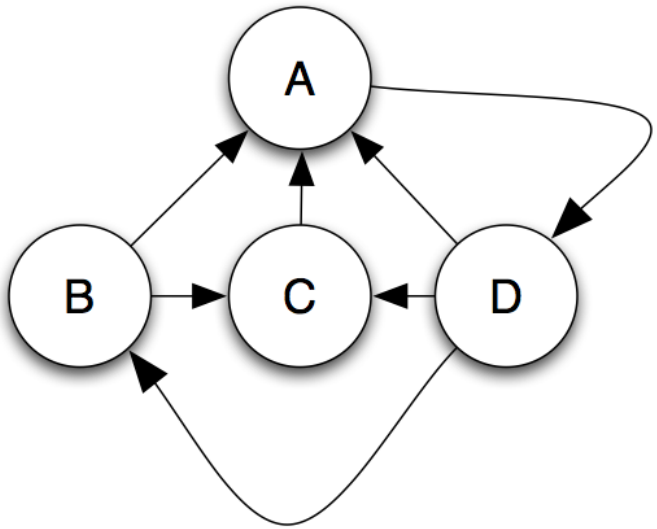
- Total PR = 1, so initialize each node to 0.25
- Set d = 0.85
- PR(A) = (0.15/4) +
  0.85 * (0.25/1 + 0.25/1 + 0.25/1)
- PR(A) = 0.675



PageRank Example

$$PR(A) = \frac{(1-d)}{N} + d\sum_{i}\frac{PR(I_i)}{C(I_i)}$$

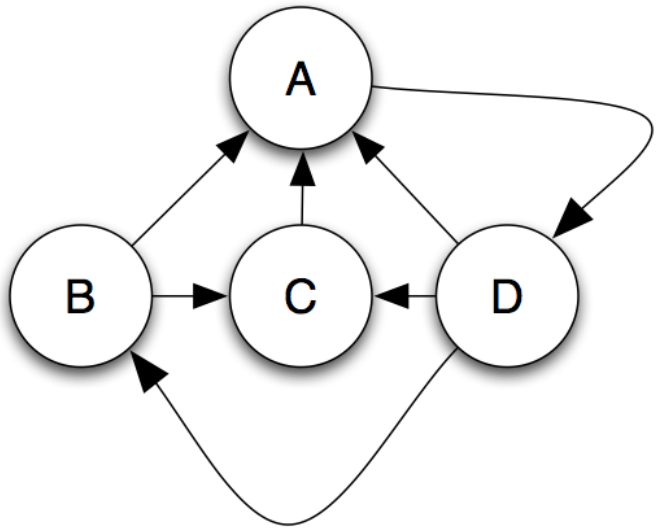
- Again, initialize all nodes to 0.25 and d=0.85
- PR(A) = *compute this*



PageRank Example

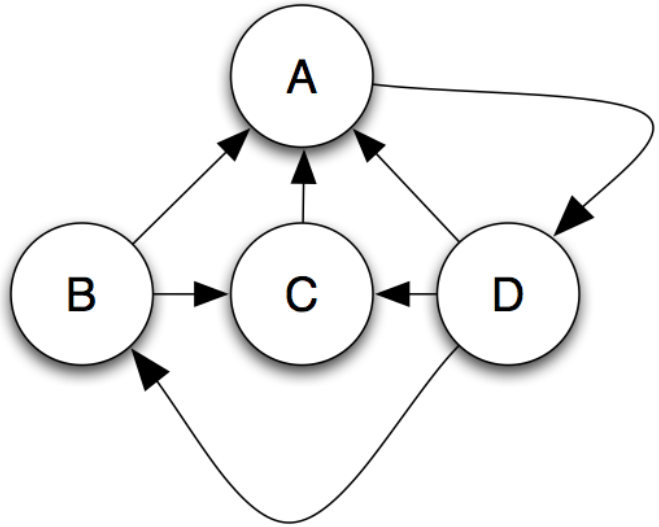
$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

- Again, initialize all nodes to 0.25 and d=0.85
- PR(A) = (0.15/4) +
  0.85 * (0.25/2 + 0.25/1 + 0.25/3)
- PR(A) = .05 + .85*(0.125 + 0.25 + 0.083)
- PR(A) = 0.4268



Example

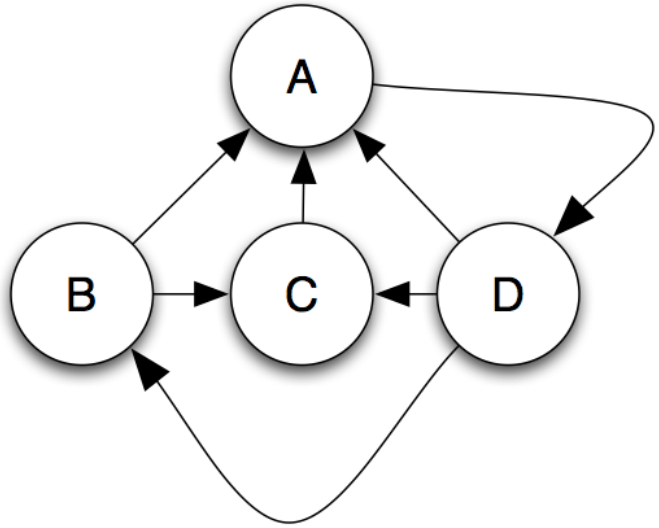
$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$



A	**B**	**C**	**D**
0.25	0.25	0.25	0.25

Example

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

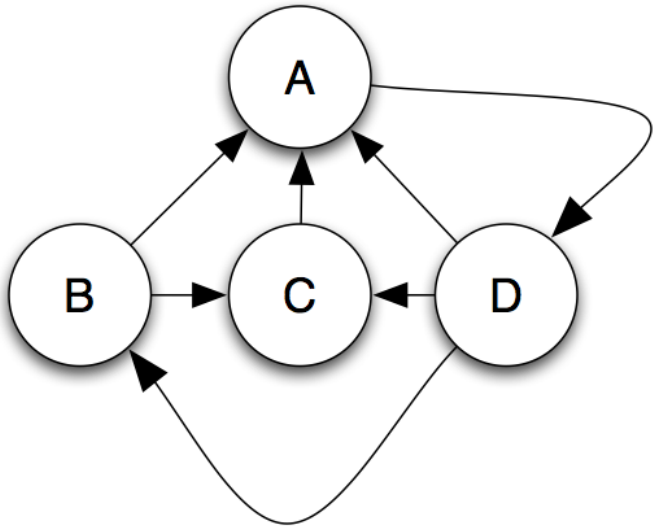


PR(A) = 0.0375 + 0.85(0.25/2 + 0.25/1 + 0.25/3)

A	**B**	**C**	**D**
0.25	0.25	0.25	0.25
0.428			

Example

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

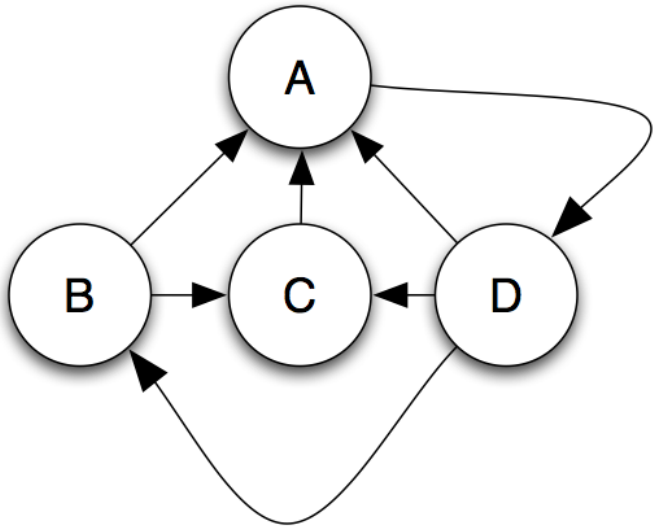


PR(B) = 0.0375 + 0.85(0.25/3)

A	**B**	**C**	**D**
0.25	0.25	0.25	0.25
0.428	0.109		

Example

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

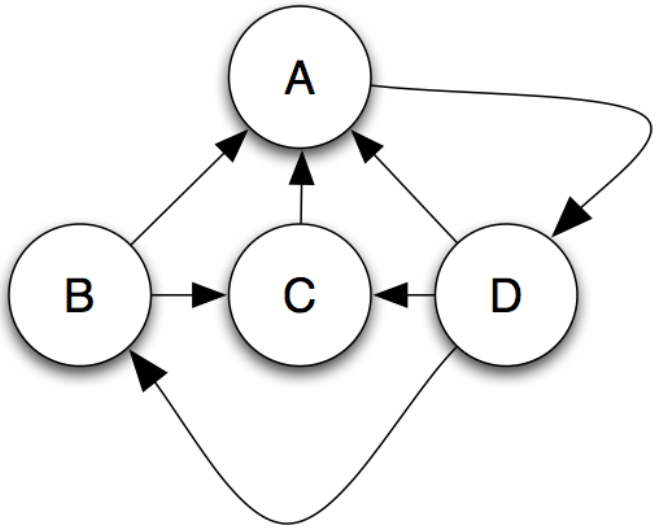


PR(C) = 0.0375 + 0.85(0.25/2 + 0.25/3)

A	**B**	**C**	**D**
0.25	0.25	0.25	0.25
0.428	0.109	0.215	

Example

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

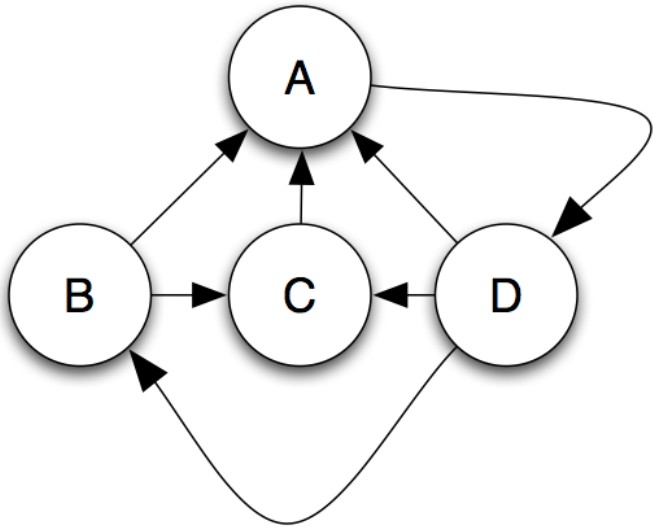


PR(D) = 0.0375 + 0.85(0.25/1)

A	**B**	**C**	**D**
0.25	0.25	0.25	0.25
0.427	0.108	0.215	0.25

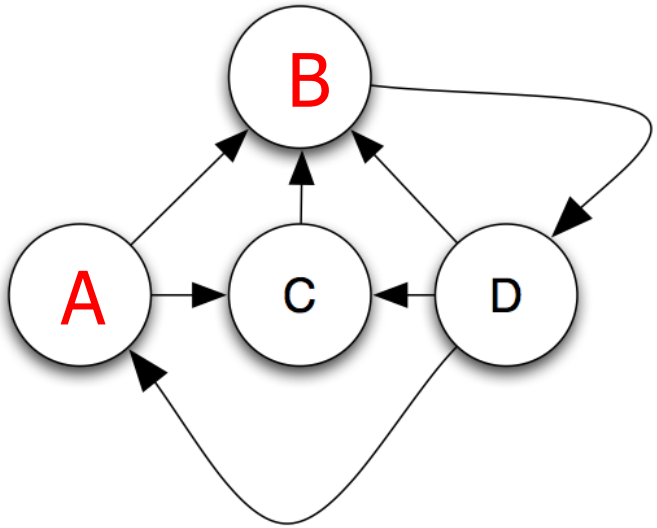
Example

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$



A	B	C	D
0.25	0.25	0.25	0.25
0.427	0.108	0.215	0.25
0.337	0.108	0.154	0.401
0.328	0.151	0.197	0.324
0.361	0.129	0.193	0.317

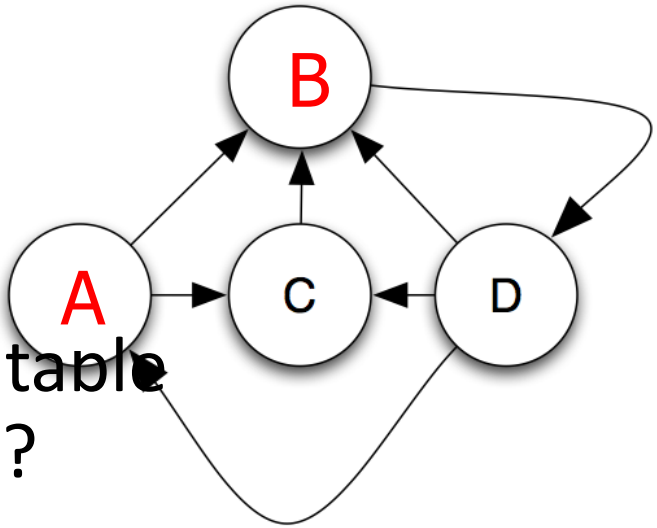
Exercise



- If I change the order of nodes in my table (or graph) will the page rank change?

~~A~~ **B**	~~B~~ **A**	**C**	**D**
0.25	0.25	0.25	0.25

Exercise



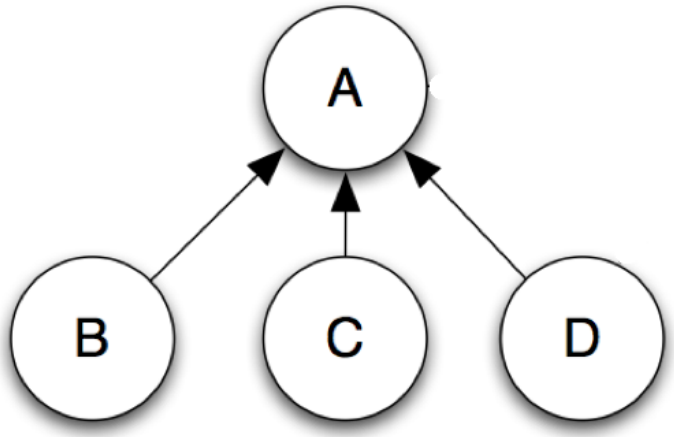
- If I change the order of nodes in my table (or graph) will the page rank change?
- No: labels don't matter, structure of the graph does

~~A~~ **B**	~~B~~ **A**	**C**	**D**
0.25	0.25	0.25	0.25
0.427	0.108	0.215	0.25
0.337	0.108	0.154	0.401
0.328	0.151	0.197	0.324
0.361	0.129	0.193	0.317

Sink nodes

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

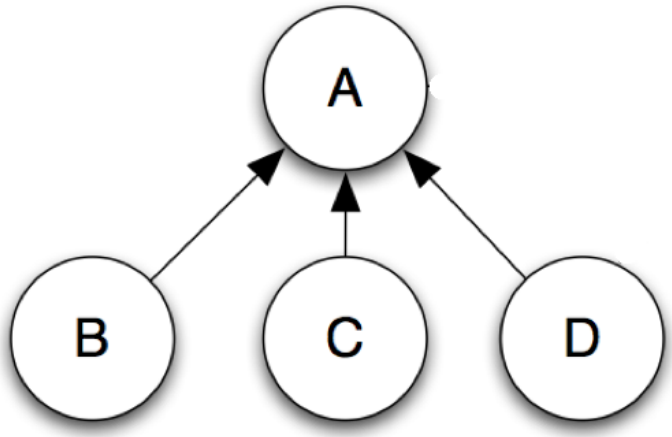
- What happens after many iterations?



Sink nodes

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

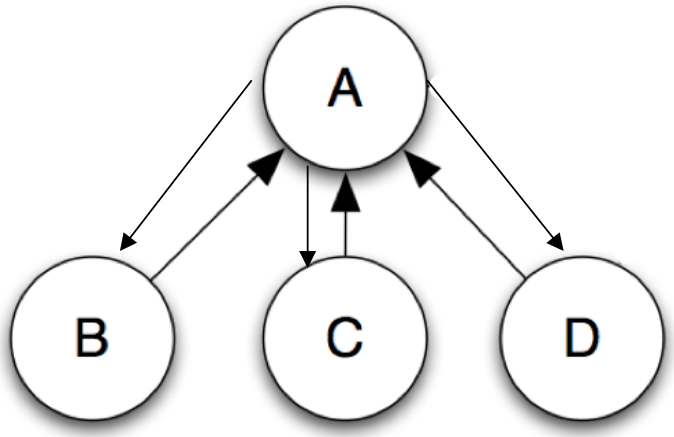
- What happens after many iterations?
  - PR(A) keeps increasing
  - PR(B) = PR(C) = PR(D) = (1-d)/N



Sink nodes

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

- Nodes with no outlinks are disallowed
- Can "drain rank" from rest of system
- Solution: Add edge from sink=>every node



Sink regions

- Must have non-zero probability of reaching every node from every other node
- Solution: with prob (1-d), random surfer types in a random URL instead of clicking a link

$$PR(A) = \frac{(1-d)}{N} + d\sum_i \frac{PR(I_i)}{C(I_i)}$$

Adding PageRank to a Search Engine

- Weighted sum of page importance and query-similarity
- Score(query, doc)=
  - w*sim(q, p) + (1-w) * PR(p)
    - If sim(q, p) > 0
    - Otherwise, 0
- Where:
  - 0 < w < 1
  - Values sim(q,p) and PR(p) are normalized

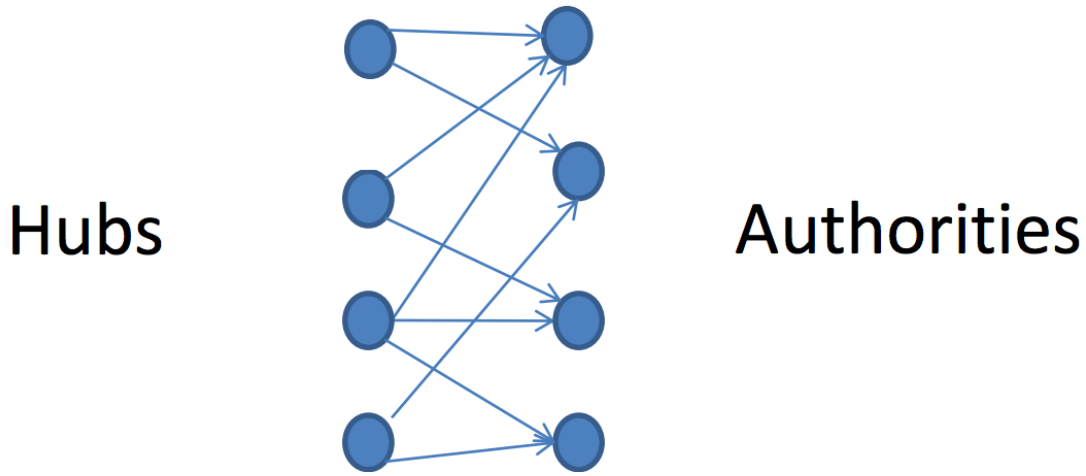
Hubs and Authorities

- Due to Kleinberg, 1997
- Unlike PageRank, is query-dependent

- A page is a good ***authority*** if it is pointed-to by many good ***hubs***
- A page is a good ***hub*** if it points to many good ***authorities***
- Good hubs and authorities reinforce each other

Hubs and Authorities

- A page is a good ***authority*** if it is pointed-to by many good ***hubs***
- A page is a good ***hub*** if it points to many good ***authorities***



HITS algorithm

$$auth(p) = \sum_{i=1}^{n} hub(i)$$

$$hub(p) = \sum_{i=1}^{n} auth(i)$$

1. Obtain *root set* using input query
2. Expand the root set by radius 1. This is called the *base set*
3. Authority update
4. Hub update
5. Iteratively compute hub, authority scores for each node in graph

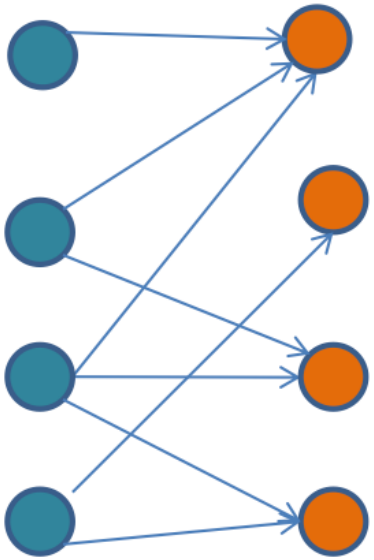
More HITS

1. Initialize all hub() and auth() scores to 1
2. For all nodes, update Auth() scores
3. For all nodes, update Hub() scores
4. Normalize scores
   - Divide each Auth by sqrt(sum($Auth^2$))
   - Divide each Hub by sqrt(sum(Hub^2))
5. If converges, terminate; else goto 2

- Note that unlike page rank, we need an explicit normalization step

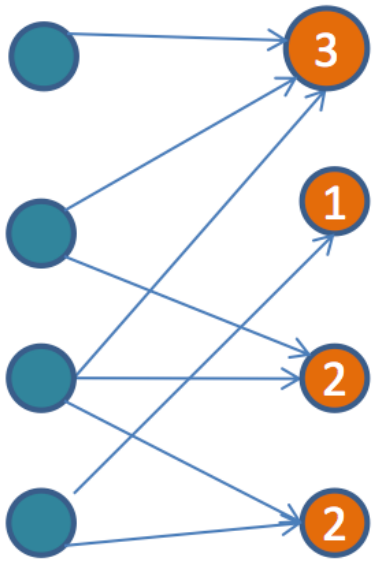
HITS Example

- All nodes start with value 1
- NOTE: this example omits normalization



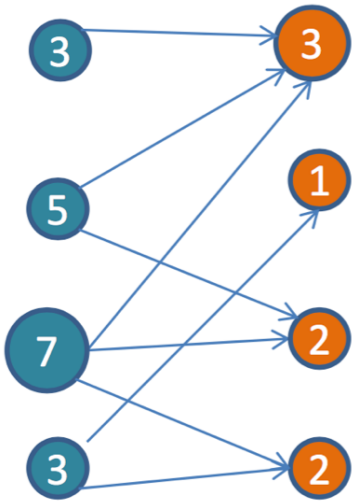
HITS Example

- Compute authorities



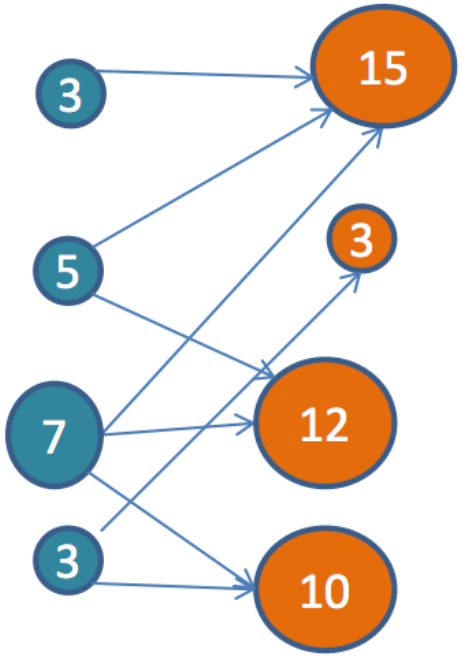
HITS Example

- Compute hubs



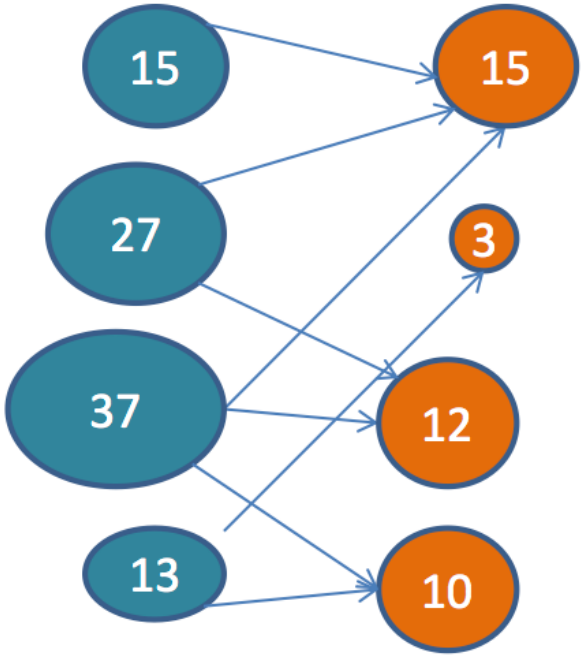
HITS Example

- Compute authorities



HITS Example

- Compute hubs



Latent Semantic Indexing

- Documents are about topics, not about words. How can we identify topics?
- Use Eigen-decomposition (singular value decomposition) of document-term matrix.
- The larger Eigen values correspond to topics that "characterize" documents.
- Corresponding Eigen vectors define a multi-dimension topic space.
- Represent documents and queries as vectors/points in this topic space.

Relevance Factors

- Many other things can be considered.
  - Which part of the page words appear in
  - How close together the words appear
  - Synonyms of specified words
  - Guesses of user intent
- Google says, “Relevancy is determined by over 200 factors, one of which is the PageRank for a given page.”

Search Engine Optimization

- Big bucks for good ranking
- Many nasty ways to improve ranking
  - E.g. Link farms
  - Search engines hate this
- But knowing what the search engine does can be used to improve ranking for your page/site.

Make it easy for the search engine

- In your site design, use text rather than images and Flash for important content
- Make your site work with JavaScript, Java and CSS disabled
- Avoid links that look like form queries http://www.mysite.com/info?about
- Have pages that focus on a particular topic
- Market your site by having other relevant sites link to yours

Link Spam

- Write comments on influential blogs etc.
  - Link these back to your site

    “Fantastic blog. <A href=http://mypornsite.com> Black belt ideas.</A>”
- Blog owner should use rel=“nofollow” attribute for such links in comments.
  - Such links are ignored for indexing purposes.

Challenges

- Three challenges in web search:
  - Result relevance
  - Processing speed
  - Scaling to many documents
- So far we've discussed result relevance
- Next time, we'll cover speed and scaling