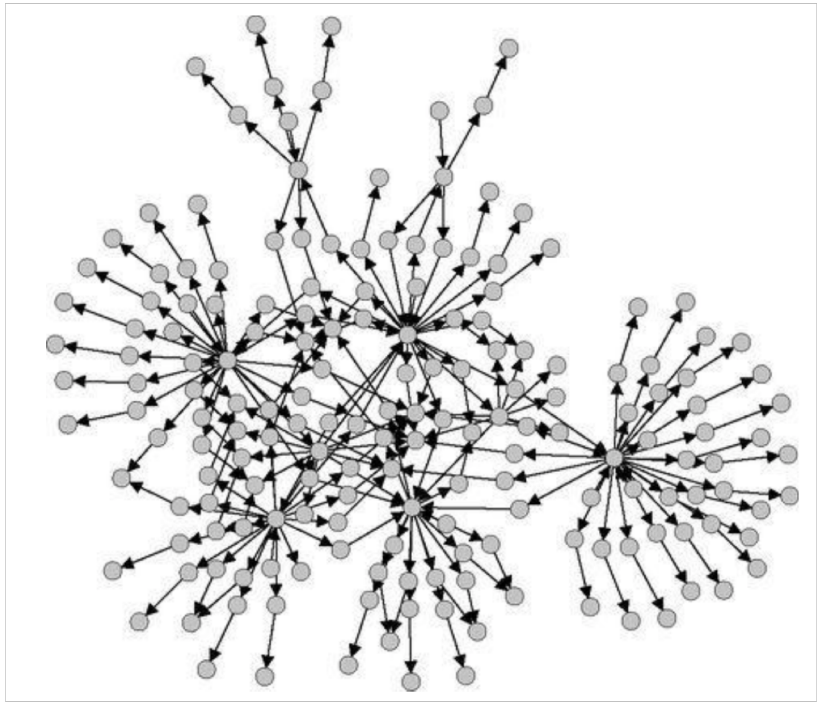
Project Proposal due Wed (9/11)

- Two deliverables
- Report
  - What is the problem? Why do we care? How is your execution plan?
  - Due before class
- Presentation
  - 5-10 mins
  - High-level explanation
    - Report
  - Seek for feedback from class

Web and PageRank

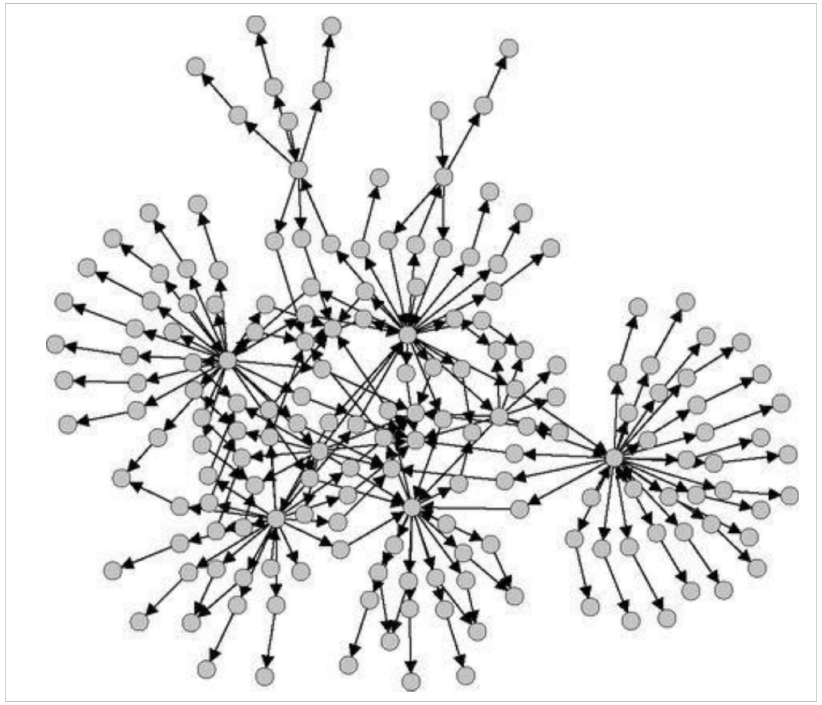
World Wide Web (WWW)

- Network of web pages
  - Each website is a node
  - Each hyperlink from a website to another is an edge
  - Dynamic pages are created when accessed



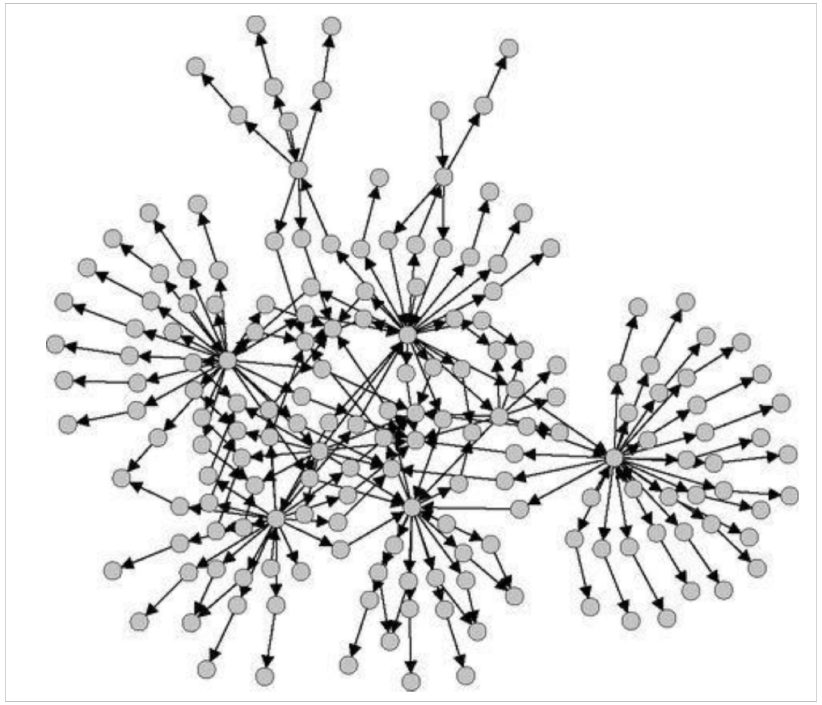
World Wide Web (WWW)

- Network of web pages
  - Each website is a node
  - Each hyperlink from a website to another is an edge
  - Dynamic pages are created when accessed
- Directed!
  - Cycles?
- Massive
  - >25B



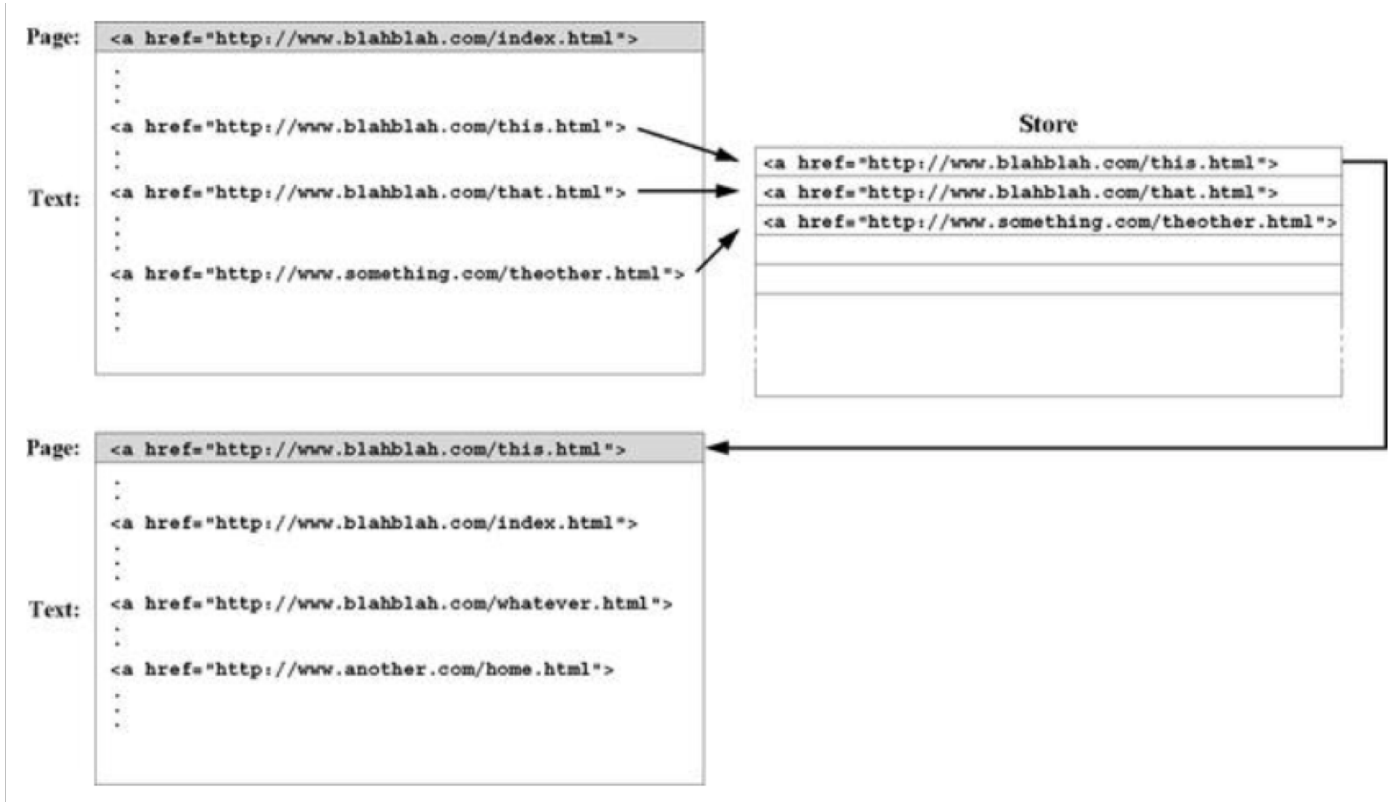
World Wide Web (WWW)

- Network of web pages
  - Each website is a node
  - Each hyperlink from a website to another is an edge
  - Dynamic pages are created when accessed
- Directed!
  - Cycles?
- Massive
  - >25B
- How to construct?



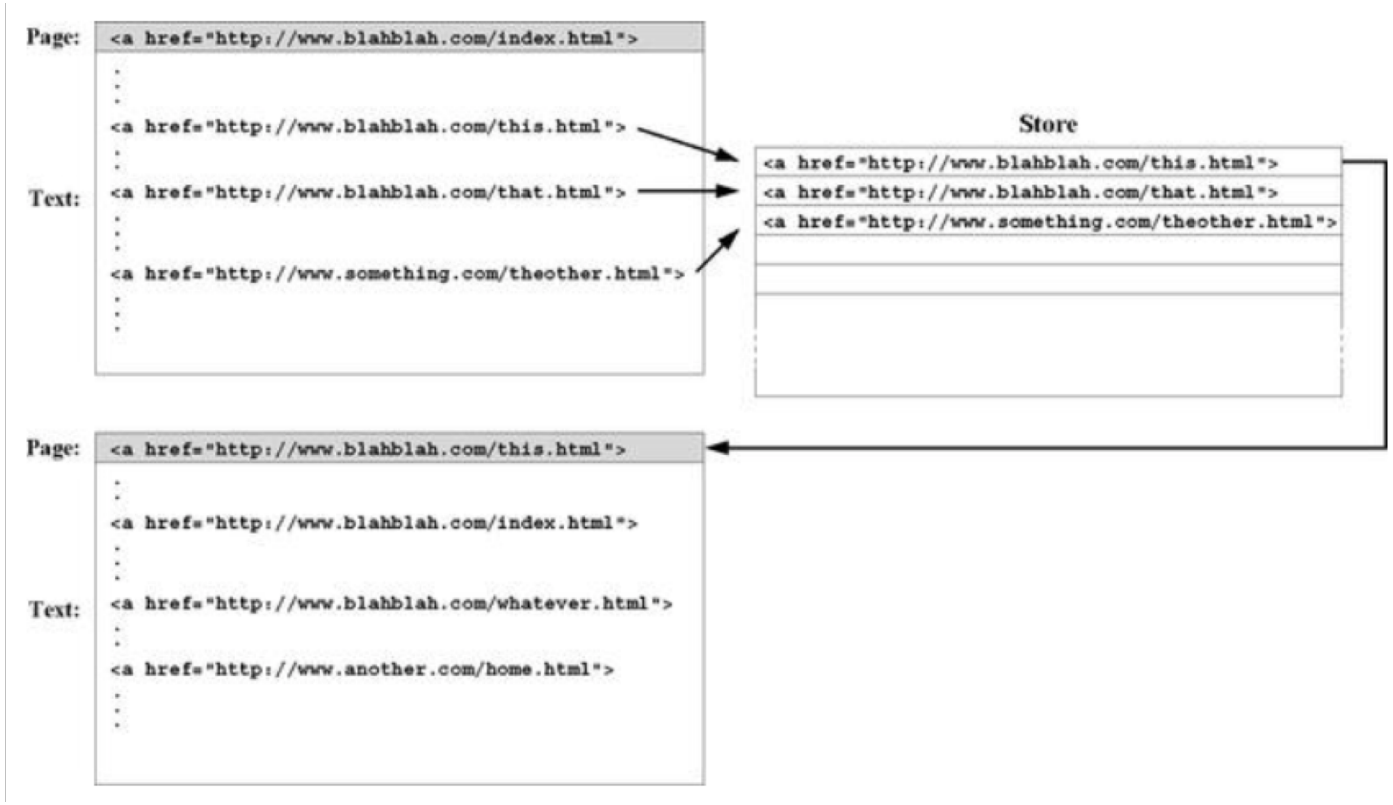
Crawling

- A website is just a text file (html)
- Start from any;
  - Find the <link> tags
  - Visit each and store
    - Track the visited ones
  - Links
    - Navigational vs Transactional
    - Like, comment …



Crawling

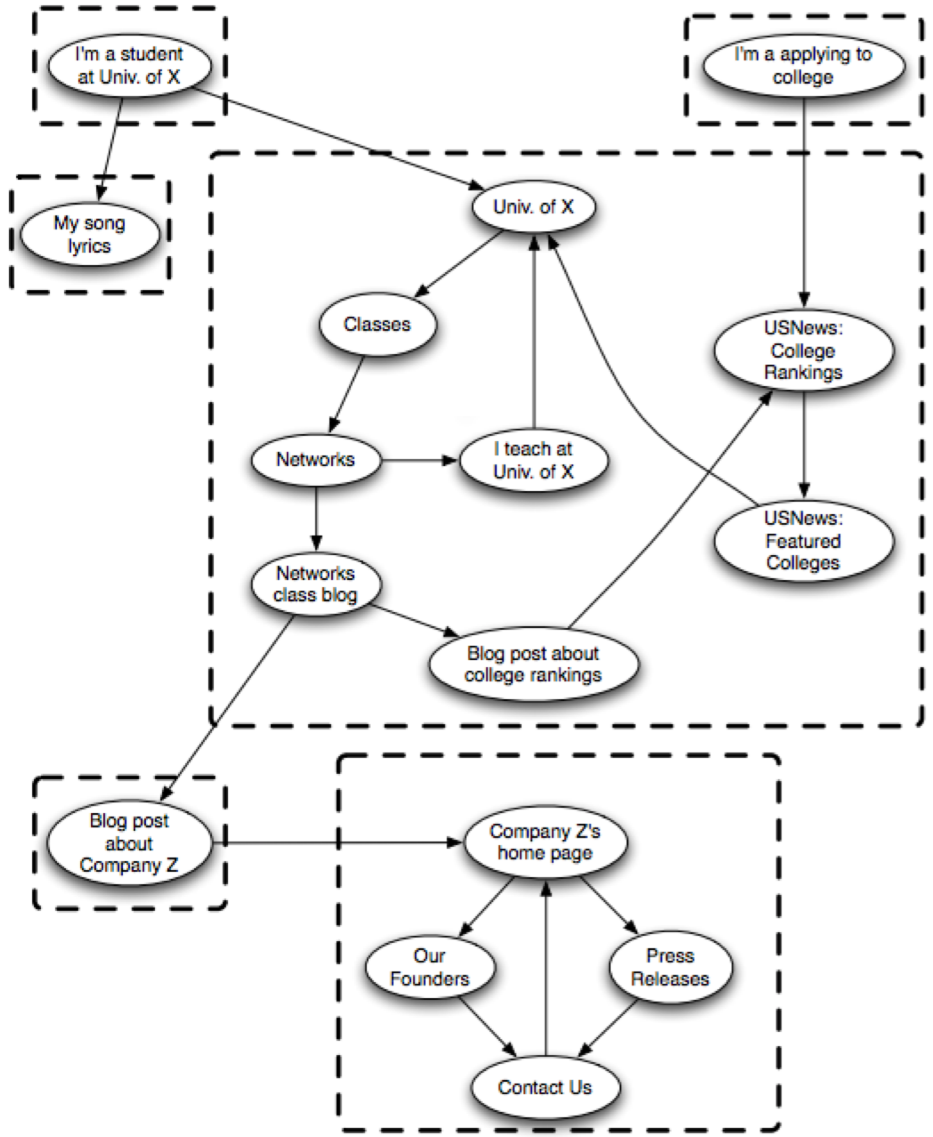
- A website is just a text file (html)
- Start from any;
  - Find the <link> tags
  - Visit each and store
    - Track the visited ones
  - Links
    - Navigational vs Transactional
    - Like, comment …
- Graph traversal!
  - BFS or DFS?
- Can you construct the entire web this way?



Web as a directed graph

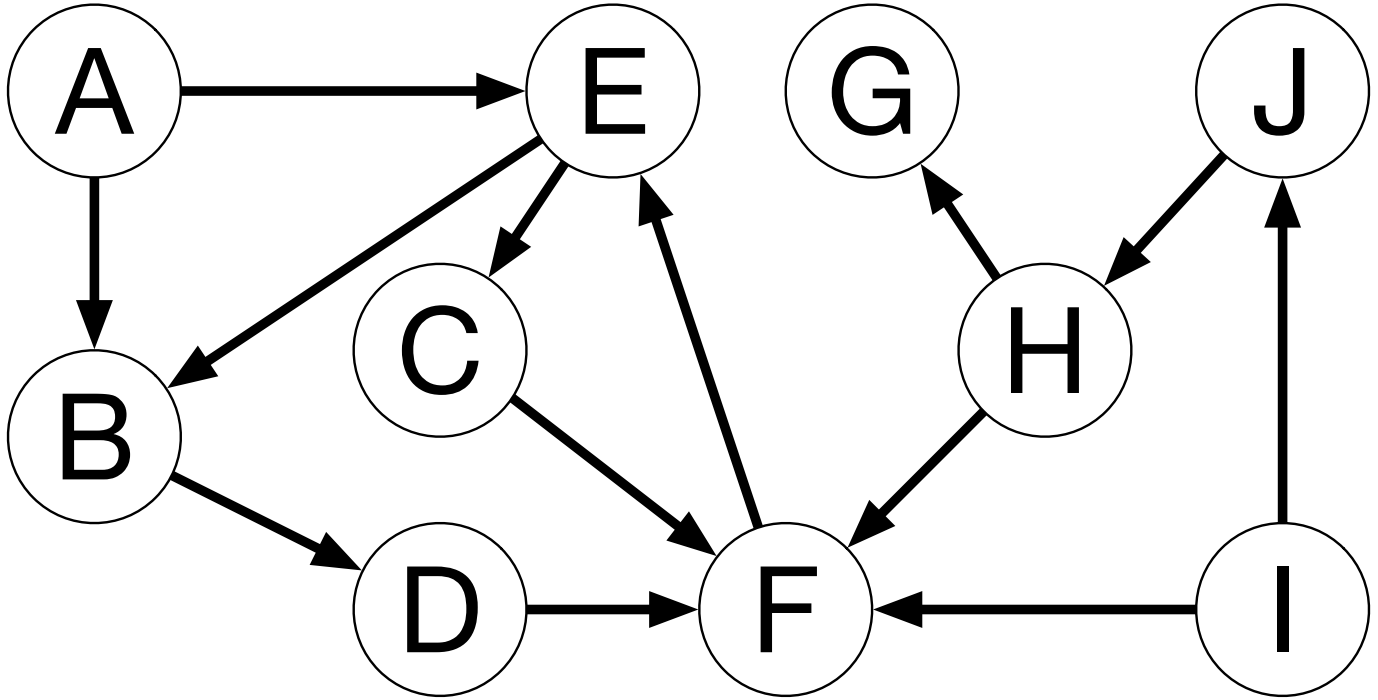
- How is the Web structured?
- Connected component
  - Straightforward for undirected

- Given a website, which other websites are reachable?
- Which others can reach to a website?



Web as a directed graph

- For node v:
  - What can it reach? IN(v)
  - What can reach to it? OUT(v)

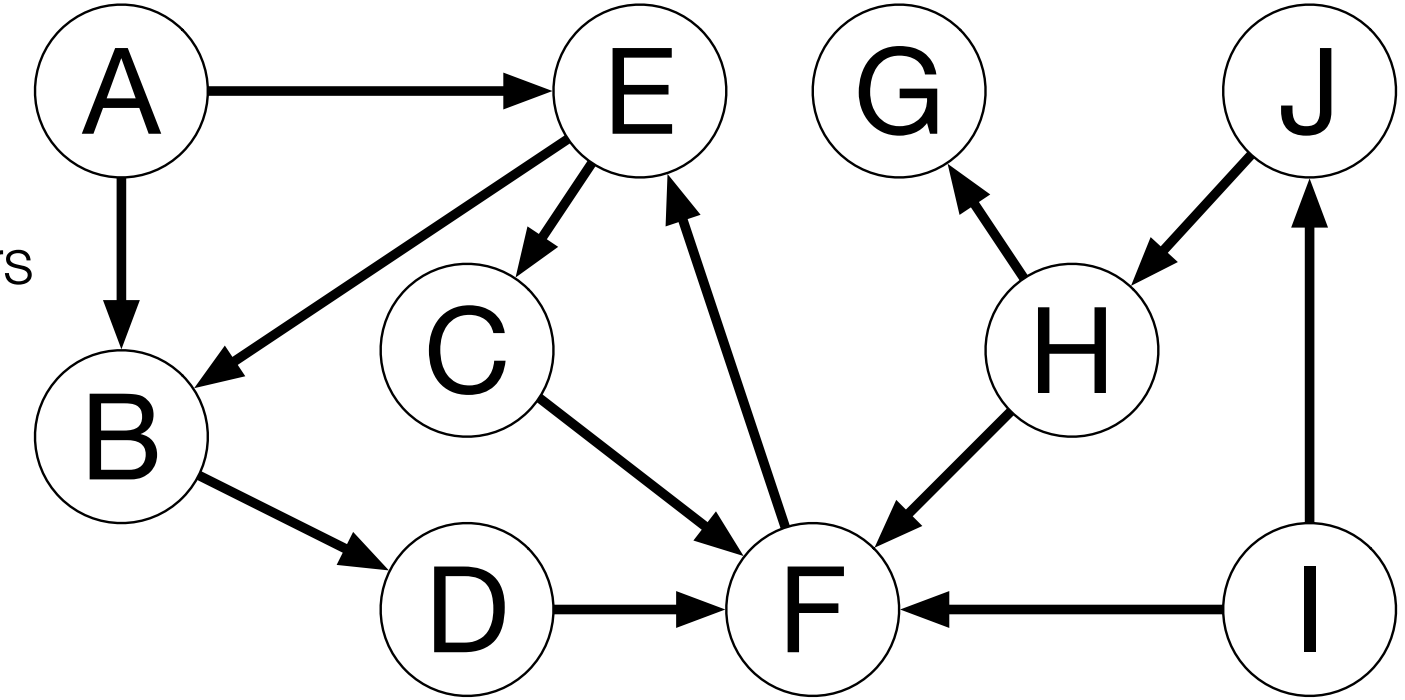


OUT(B)={B, C, D, E, F}

IN(G)={H, I, J}

Web as a directed graph

- For node v:
  - What can it reach? IN(v)
  - What can reach to it? OUT(v)
- Strongly connected (SCC)
  - Any node can reach to all others
  - {v | IN(v, G) = OUT(v, G) }
  - {B, C, D, E, F}

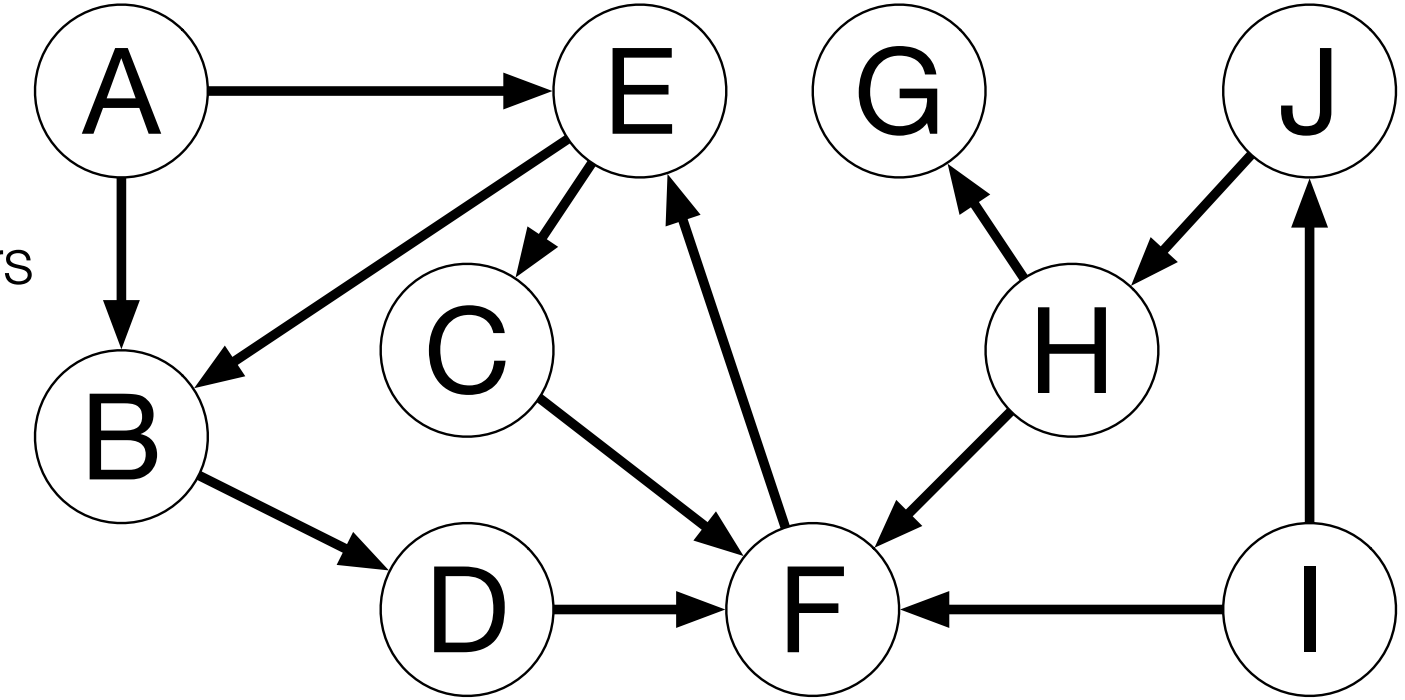


OUT(B)={B, C, D, E, F}

IN(G)={H, I, J}

Web as a directed graph

- For node v:
  - What can it reach? IN(v)
  - What can reach to it? OUT(v)
- Strongly connected (SCC)
  - Any node can reach to all others
  - {v | IN(v, G) = OUT(v, G) }
  - {B, C, D, E, F}
- DAG
  - No cycle: {F, G, H, I, J}, {A}

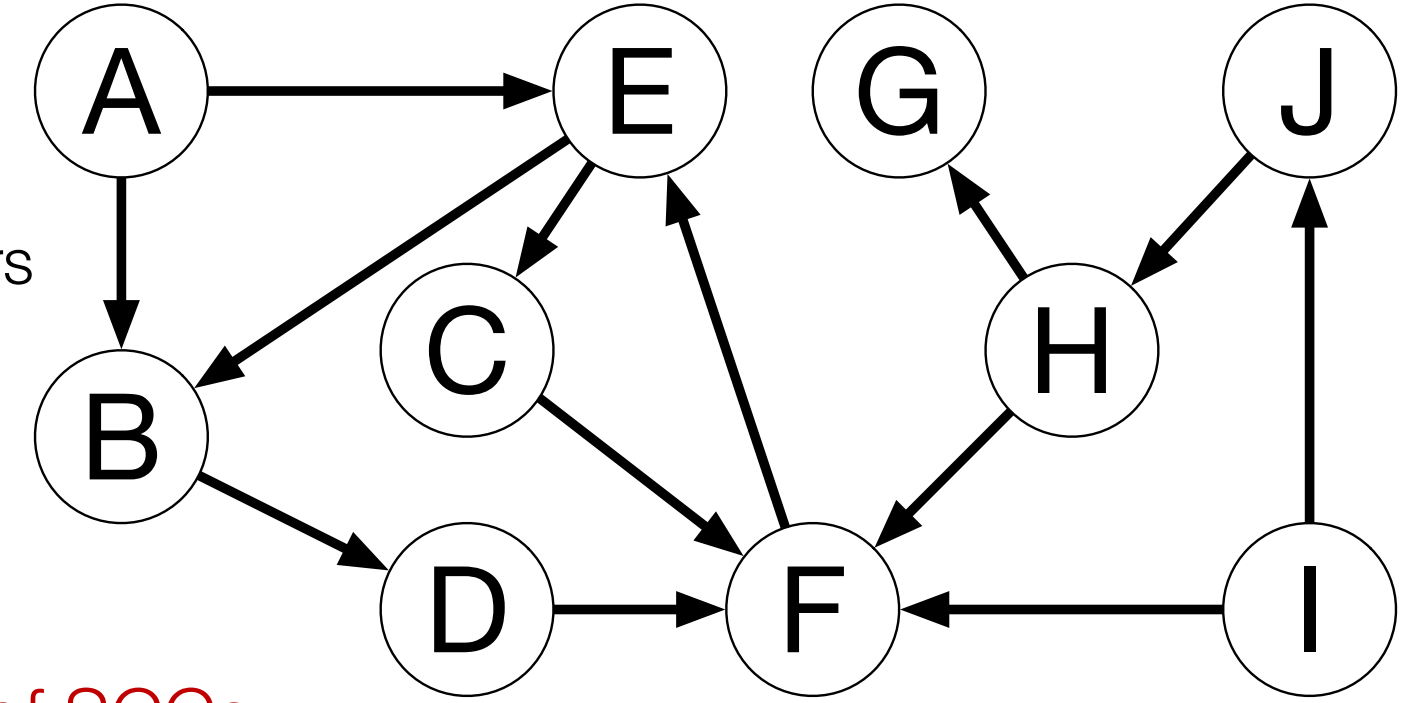


OUT(B)={B, C, D, E, F}

IN(G)={H, I, J}

Web as a directed graph

- For node v:
  - What can it reach? OUT(v)
  - What can reach to it? IN(v)
- Strongly connected (SCC)
  - Any node can reach to all others
  - {v | IN(v, G) = OUT(v, G) }
  - {B, C, D, E, F}
- DAG
  - No cycle: {F, G, H, I, J}, {A}
- Any directed graph is a DAG of SCCs
- How is the web structure? How to compute?



OUT(B)={B, C, D, E, F}

IN(G)={H, I, J}

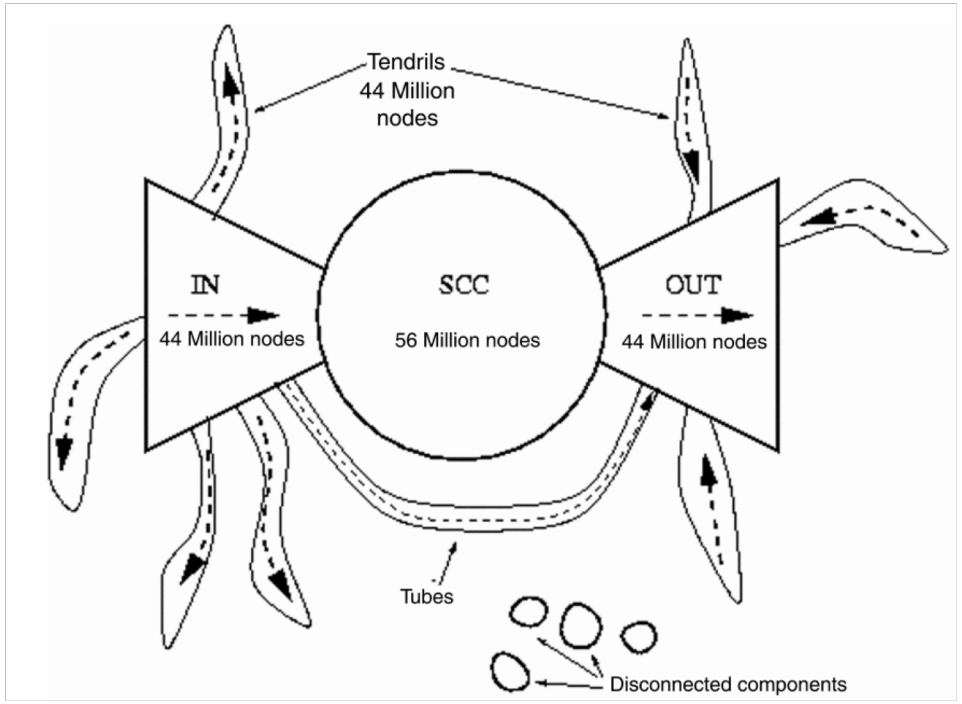
Bow-tie structure of Web

- Heard about AltaVista?
  - 203M urls, 1.5B links

Graph structure in the Web

Andrei Broder [a], Ravi Kumar [b,*], Farzin Maghoul [a], Prabhakar Raghavan [b], Sridhar Rajagopalan [b], Raymie Stata [c], Andrew Tomkins [b], Janet Wiener [c]

[a] AltaVista Company, San Mateo, CA, USA
[b] IBM Almaden Research Center, San Jose, CA, USA
[c] Compaq Systems Research Center, Palo Alto, CA, USA



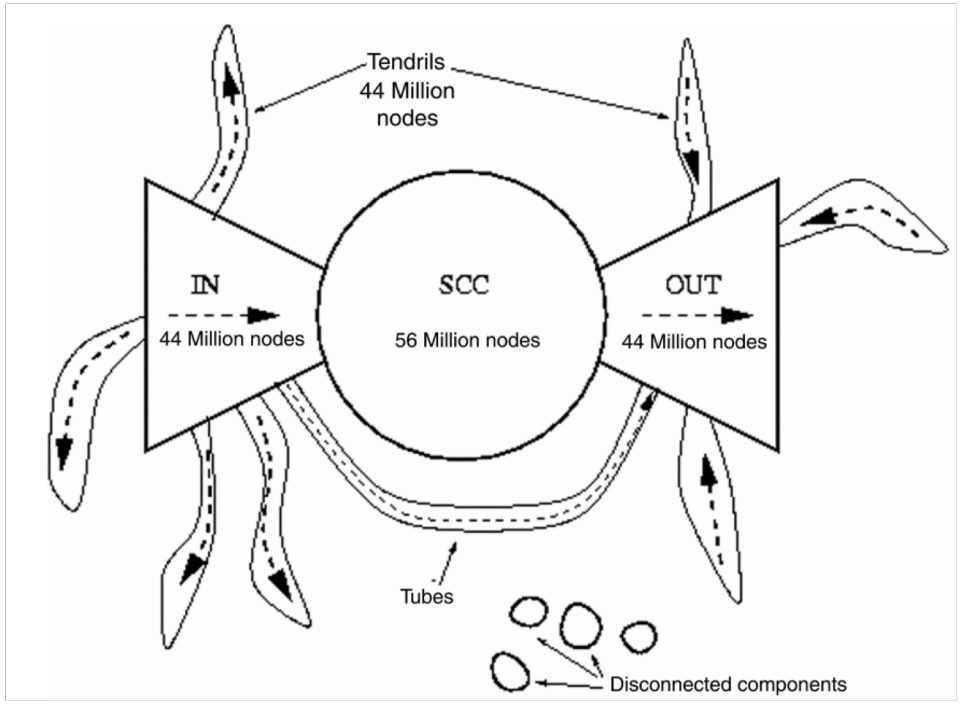
Bow-tie structure of Web

- Heard about AltaVista?
  - 203M urls, 1.5B links
- One giant strongly connected component!

Graph structure in the Web

Andrei Broder [a], Ravi Kumar [b,*], Farzin Maghoul [a], Prabhakar Raghavan [b], Sridhar Rajagopalan [b], Raymie Stata [c], Andrew Tomkins [b], Janet Wiener [c]

[a] AltaVista Company, San Mateo, CA, USA
[b] IBM Almaden Research Center, San Jose, CA, USA
[c] Compaq Systems Research Center, Palo Alto, CA, USA



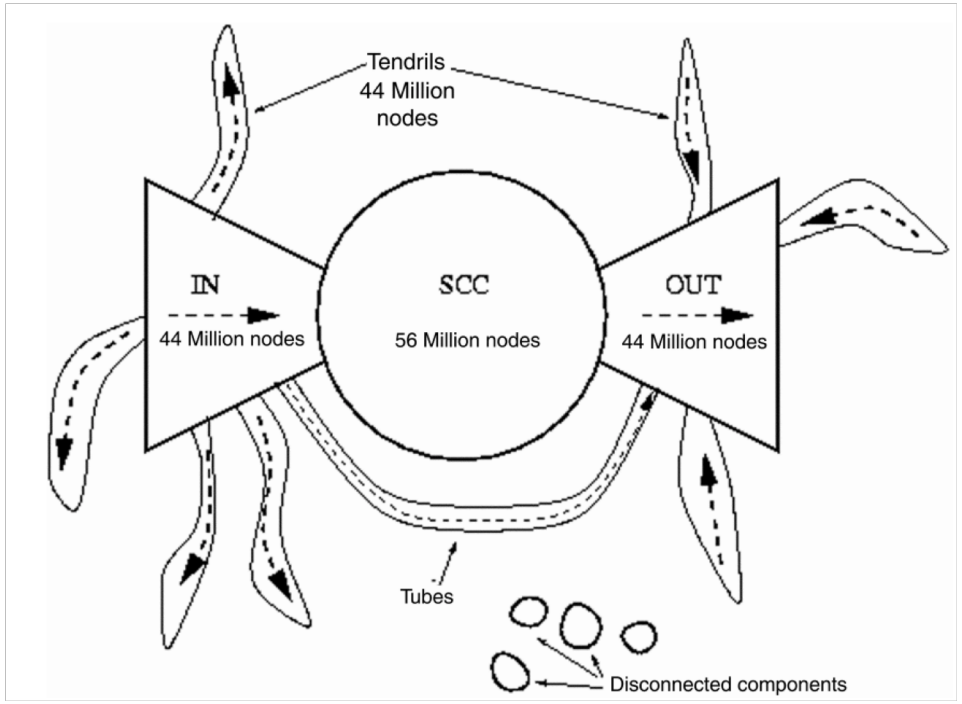
Bow-tie structure of Web

- Heard about AltaVista?
  - 203M urls, 1.5B links
- One giant strongly connected component!
  - Why?

Graph structure in the Web

Andrei Broder [a], Ravi Kumar [b,*], Farzin Maghoul [a], Prabhakar Raghavan [b], Sridhar Rajagopalan [b], Raymie Stata [c], Andrew Tomkins [b], Janet Wiener [c]

[a] AltaVista Company, San Mateo, CA, USA
[b] IBM Almaden Research Center, San Jose, CA, USA
[c] Compaq Systems Research Center, Palo Alto, CA, USA



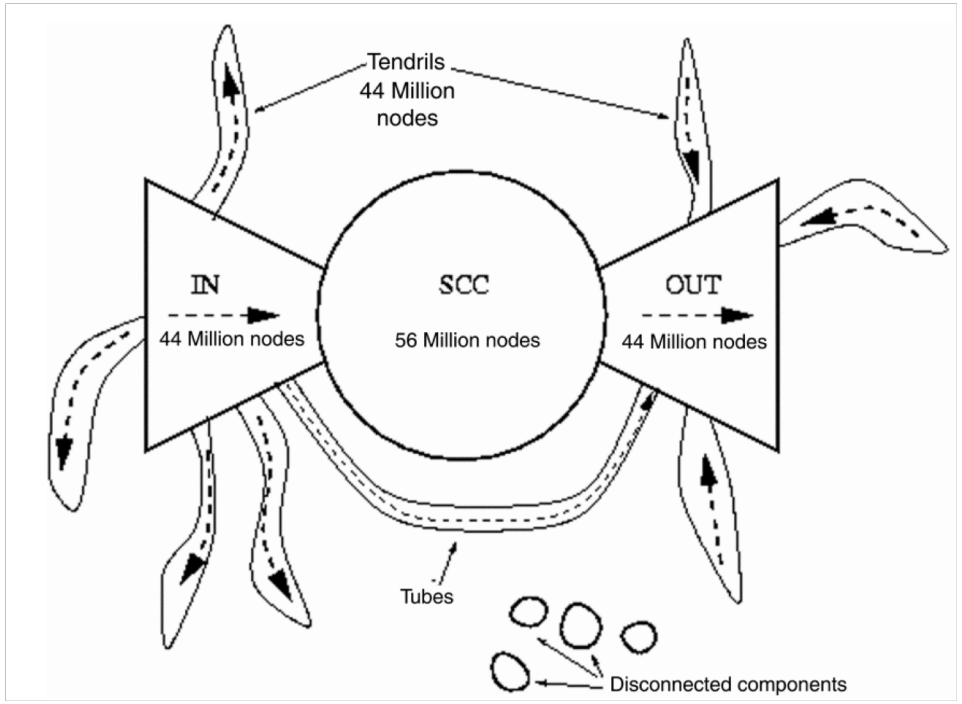
Bow-tie structure of Web

- Heard about AltaVista?
  - 203M urls, 1.5B links
- One giant strongly connected component!
  - Why?
- Two others
  - IN: Can reach SCC
    - SCC cannot reach IN
  - OUT: SCC can reach
    - Cannot reach SCC
- Tendrils, tubes, disconnected components

Graph structure in the Web

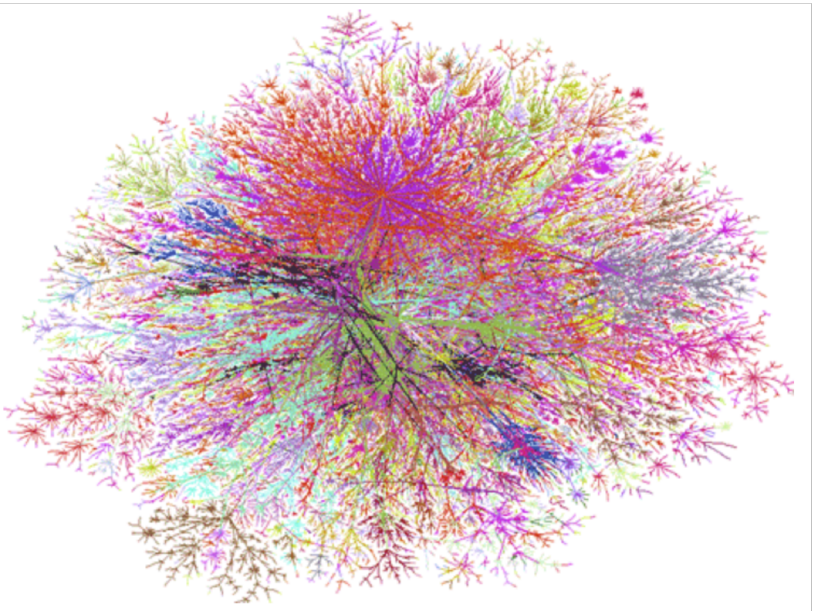
Andrei Broder [a], Ravi Kumar [b,*], Farzin Maghoul [a], Prabhakar Raghavan [b], Sridhar Rajagopalan [b], Raymie Stata [c], Andrew Tomkins [b], Janet Wiener [c]

[a] AltaVista Company, San Mateo, CA, USA
[b] IBM Almaden Research Center, San Jose, CA, USA
[c] Compaq Systems Research Center, Palo Alto, CA, USA



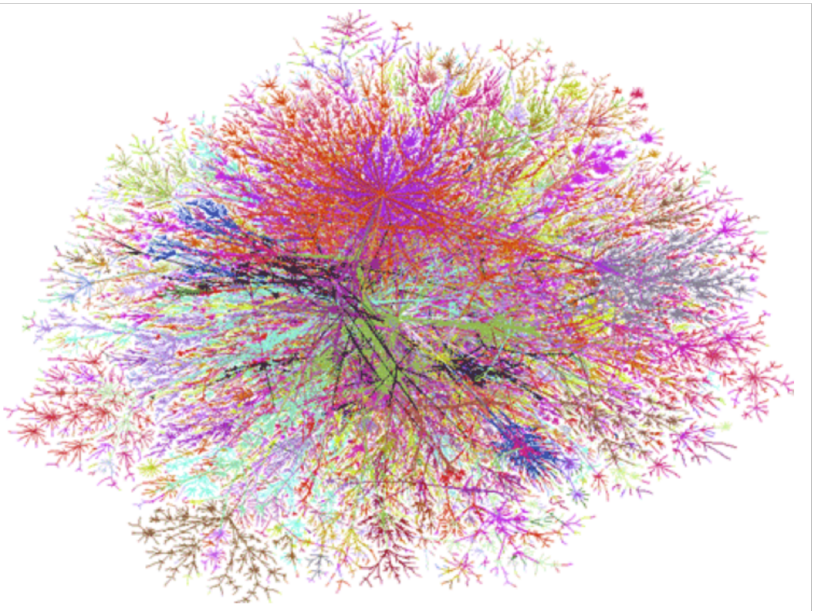
How to make sense of WWW?

- Organizing principle to put order among websites
  - Some matters more than others
- Ranking by using the web structure



How to make sense of WWW?

- Organizing principle to put order among websites
  - Some matters more than others
- Ranking by using the web structure



- In-links and out-links
- Large degree does not imply importance
  - And easy to manipulate

Two measures: PageRank and HITS

- Introduced at around the same time, different outcomes
- PageRank assigns a single score for each node
  - Iterative computation
  - By Google Founders (Page & Brin, 98)

- HITS assigns two scores (Hyper-text induced search)
  - Iterative computation
  - By J. Kleinberg (SODA, 98)

- Idea: Links as votes

PageRank

- A page with more links is more important
- In-degree? Out-degree?

PageRank

- A page with more links is more important
- In-degree? Out-degree?

- But some links are more important (?)
- From a webpage with large degree
- Recursive meaning

PageRank

- A page with more links is more important
- In-degree? Out-degree?

- But some links are more important (?)
- From a webpage with large degree
- Recursive meaning

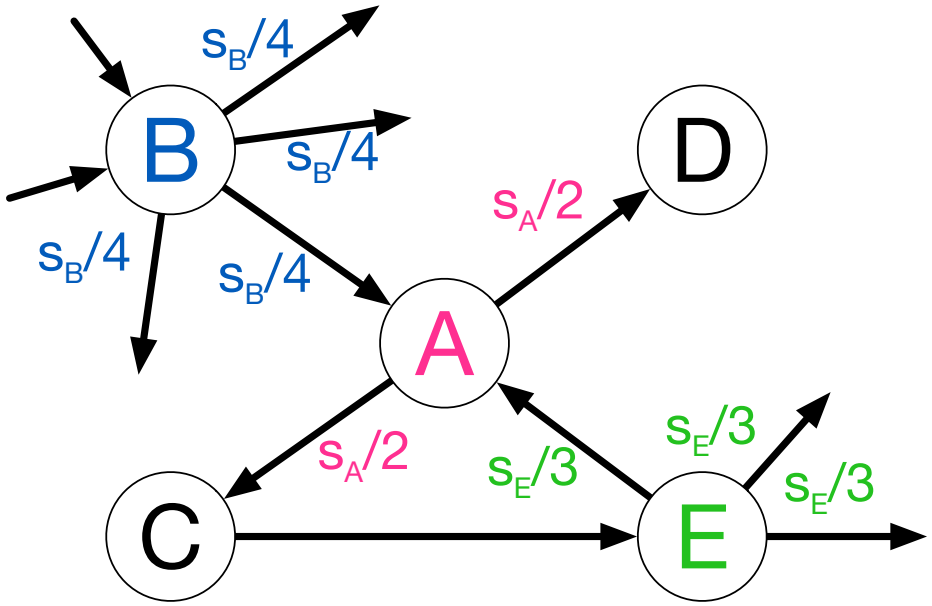
- A vote from important page is worth more

Votes determine PageRank score

- Each link's importance is determined by the frequency

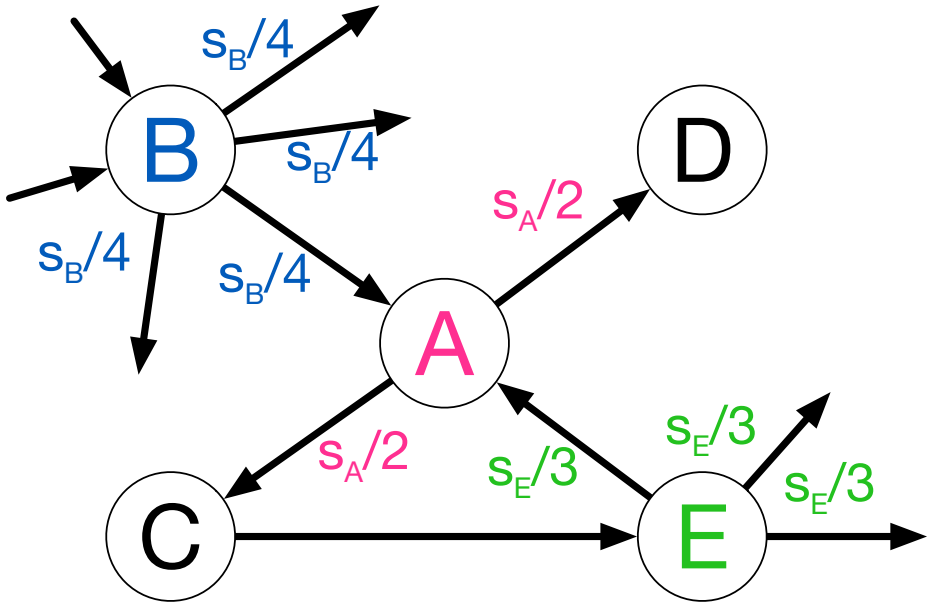
Votes determine PageRank score

- Each link's importance is determined by the frequency
  - PageRank score $\boldsymbol{s}$ for node **A**
  - Each outgoing link takes has $\boldsymbol{s/d_o(A)}$



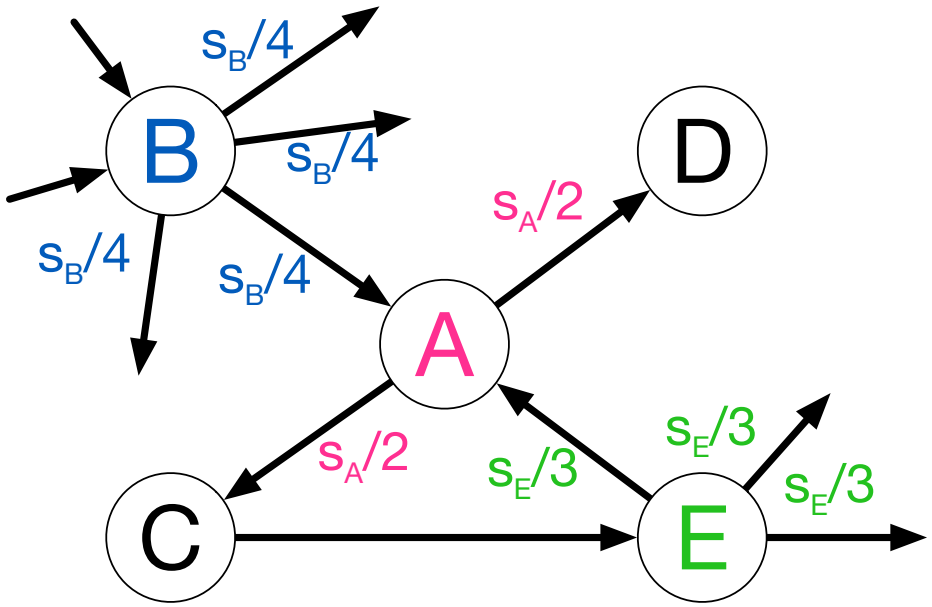
Votes determine PageRank score

- Each link's importance is determined by the frequency
  - PageRank score $\boldsymbol{s}$ for node A
  - Each outgoing link takes has $\boldsymbol{s/d_o(A)}$
- Each page's score is the sum of incoming votes
  - $\boldsymbol{s_A = s_B/4 + s_E/3}$



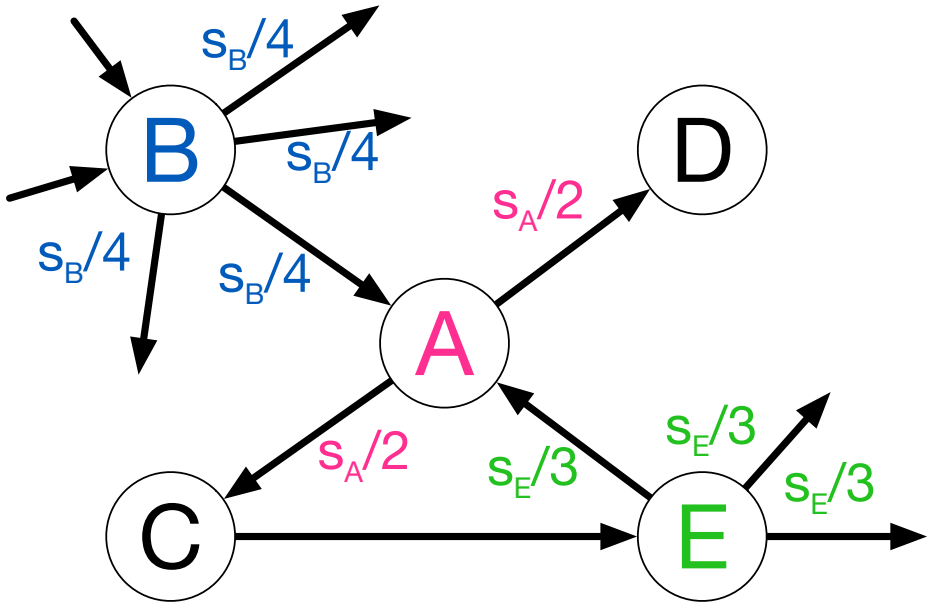
Votes determine PageRank score

- Each link's importance is determined by the frequency
  - PageRank score $\boldsymbol{s}$ for node A
  - Each outgoing link takes has $\boldsymbol{s/d_o(A)}$
- Each page's score is the sum of incoming votes
  - $\boldsymbol{s_A = s_B/4 + s_E/3}$
- More important if gets vote from important pages
- $\boldsymbol{s_j = \sum_{i \to j} \frac{s_i}{d_o(i)}}$



The equation

- Set of linear equations
  - $s_A = s_B/4 + s_E/3$
  - $s_E = s_C$
  - $s_D = s_A$
  - …
- |V| equations
- Gaussian elimination?



Matrix-vector multiplication

- Consider matrix $\boldsymbol{M}$
  - For the link $\boldsymbol{j \rightarrow i}$, $\boldsymbol{M_{ij} = 1/d_o(j)}$

		1/3
		1/3
		1/3

$\boldsymbol{M}$

Matrix-vector multiplication

- Consider matrix $\boldsymbol{M}$
  - For the link $\boldsymbol{j \rightarrow i}$, $\boldsymbol{M_{ij} = 1/d_o(j)}$
  - Each columns sums to 1 (column-stochastic)

		1/3
		1/3
		1/3

$\boldsymbol{M}$

Matrix-vector multiplication

- Consider matrix $\boldsymbol{M}$
  - For the link $\boldsymbol{j \rightarrow i}$, $\boldsymbol{M_{ij} = 1/d_o(j)}$
  - Each columns sums to 1 (column-stochastic)
- For all j: $\boldsymbol{s_j = \sum_{i \rightarrow j} \frac{s_i}{d_o(i)}}$

		1/3
		1/3
		1/3

$\boldsymbol{M}$

Matrix-vector multiplication

- Consider matrix $\boldsymbol{M}$
  - For the link $\boldsymbol{j \rightarrow i}$, $\boldsymbol{M_{ij} = 1/d_o(j)}$
  - Each columns sums to 1 (column-stochastic)
- For all j: $\boldsymbol{s_j = \sum_{i \rightarrow j} \frac{s_i}{d_o(i)}}$
- Find score vector $\boldsymbol{s}$ s.t.
  - $\boldsymbol{\sum_i s_i = 1}$ **and** $\boldsymbol{s = M.s}$

		1/3
		1/3
		1/3

$\boldsymbol{M}$

Random-walk interpretation

- Random web surfer:

Random-walk interpretation

- Random web surfer:
  - At time $\boldsymbol{t}$, on page a
  - At time $\boldsymbol{t+1}$, goes to one of a's neighbors, a', with probability $\boldsymbol{1/d_o(i)}$

Random-walk interpretation

- Random web surfer:
  - At time $\boldsymbol{t}$, on page a
  - At time $\boldsymbol{t+1}$, goes to one of a's neighbors, a', with probability $\boldsymbol{1/d_o(i)}$
  - Do the same on page a'
  - Repeat indefinitely

Random-walk interpretation

- Random web surfer:
  - At time $\boldsymbol{t}$, on page a
  - At time $\boldsymbol{t+1}$, goes to one of a's neighbors, a', with probability $\boldsymbol{1/d_o(i)}$
  - Do the same on page a'
  - Repeat indefinitely
- Say $\boldsymbol{p(t)}$ is the prob. distribution vector over pages (size |V|)
  - $\boldsymbol{i^{th}}$ number is the probability that surfer is at page $\boldsymbol{i}$ at time $\boldsymbol{t}$

Random-walk interpretation

- Random web surfer:
  - At time $\boldsymbol{t}$, on page a
  - At time $\boldsymbol{t+1}$, goes to one of a's neighbors, a', with probability $1/d_o(i)$
  - Do the same on page a'
  - Repeat indefinitely
- Say $\boldsymbol{p(t)}$ is the prob. distribution vector over pages (size |V|)
  - $\boldsymbol{i^{th}}$ number is the probability that surfer is at page $\boldsymbol{i}$ at time $\boldsymbol{t}$
  - $\boldsymbol{p(t+1) = M.p(t)}$

Random-walk interpretation

- Random web surfer:
  - At time $\boldsymbol{t}$, on page a
  - At time $\boldsymbol{t+1}$, goes to one of a's neighbors, a', with probability $1/d_o(i)$
  - Do the same on page a'
  - Repeat indefinitely
- Say $\boldsymbol{p(t)}$ is the prob. distribution vector over pages (size |V|)
  - $\boldsymbol{i^{th}}$ number is the probability that surfer is at page $\boldsymbol{i}$ at time $\boldsymbol{t}$
  - $\boldsymbol{p(t+1) = M.p(t)}$
- When it reaches the state $\boldsymbol{p(t+1) = M.p(t) = p(t)}$
  - Stationary distribution
  - Corresponds to vector $\boldsymbol{s}$!

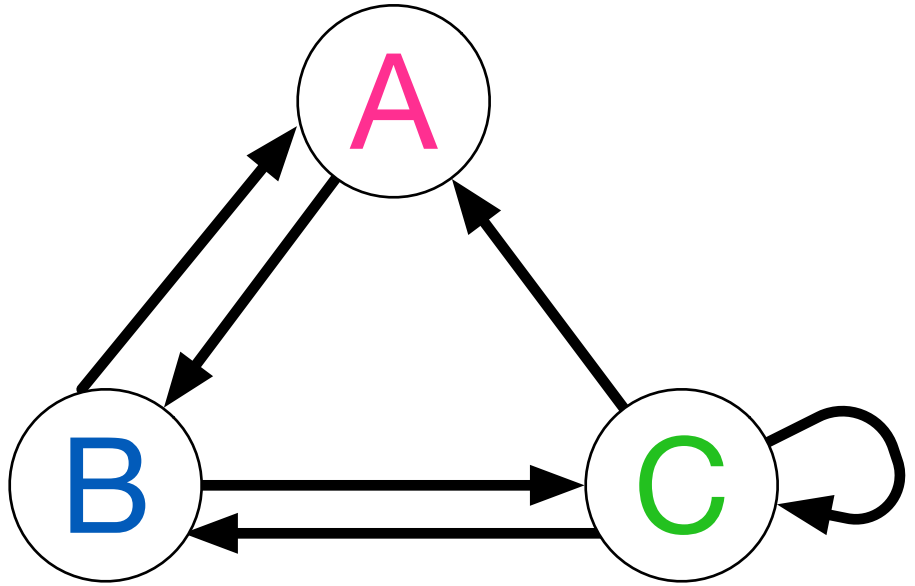
How to compute PageRank?

- Given a directed graph $\boldsymbol{G}$ with $\boldsymbol{n}$ nodes
  - Assign each node an initial score $\frac{\boldsymbol{1}}{\boldsymbol{n}}$
  - Calculate $\boldsymbol{s_j^{(t+1)} = \sum_{i \to j} \frac{s_i^{(t)}}{d_o(i)} \; \forall \, j \epsilon G}$
    - Until convergence $\boldsymbol{\sum_i |s_i^{(t+1)} - s_i^{(t)}| < \epsilon}$

Example

- Power iteration
  - Set $s_i = \frac{1}{n}$
  - $s'_i = \sum_{i \to j} \frac{s_i}{d_o(i)}$
  - If $|s' - s| \geq \epsilon$
    - $s \leftarrow s'$
    - Repeat
  - Else
    - Done

	A	B	C
A	0	0.5	0.33
B	1	0	0.33
C	0	0.5	0.33



s_A	0.33	0.27	0.31	0.28	0.29	…	0.3
s_B	0.33	0.44	0.36	0.41	0.37	…	0.4
s_C	0.33	0.27	0.31	0.28	0.29	…	0.3

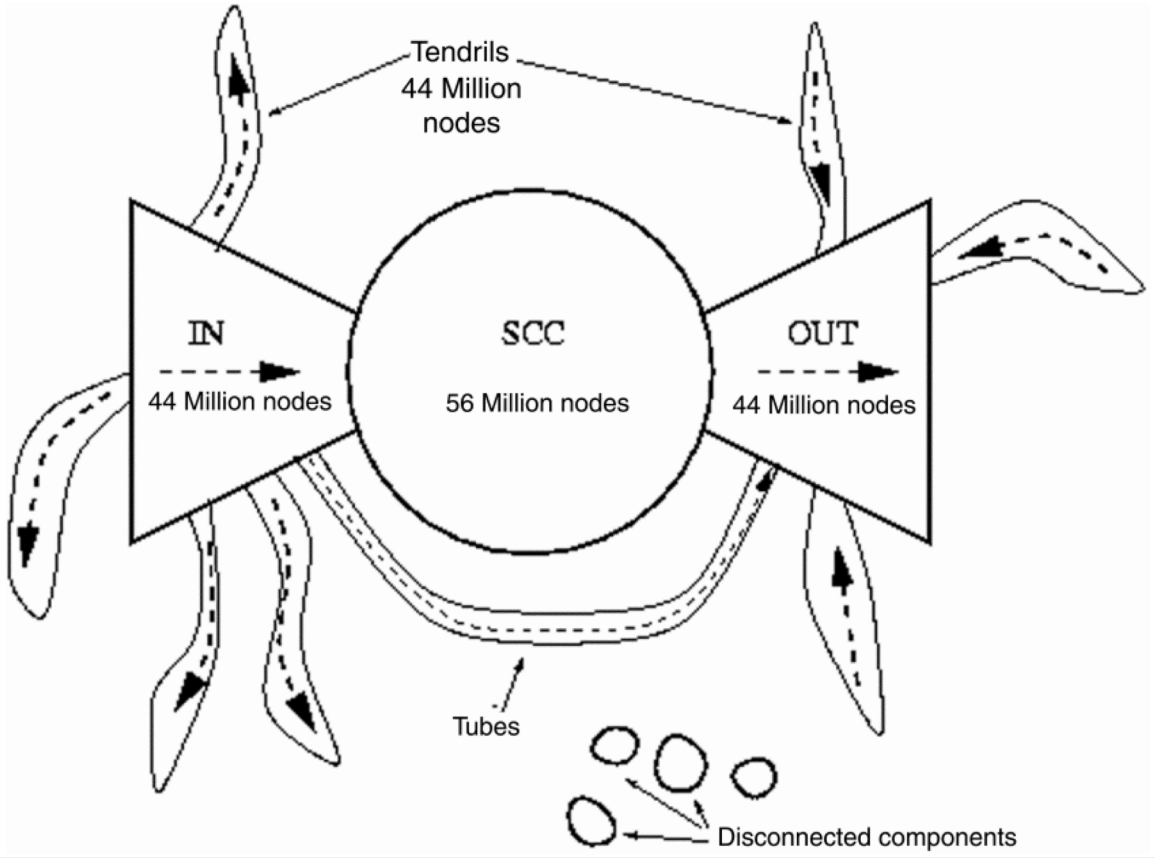
$$s_A = s_B/2 + s_C/3$$

$$s_B = s_A + s_C/3$$

$$s_C = s_B/2 + s_C/3$$

All looks good so far

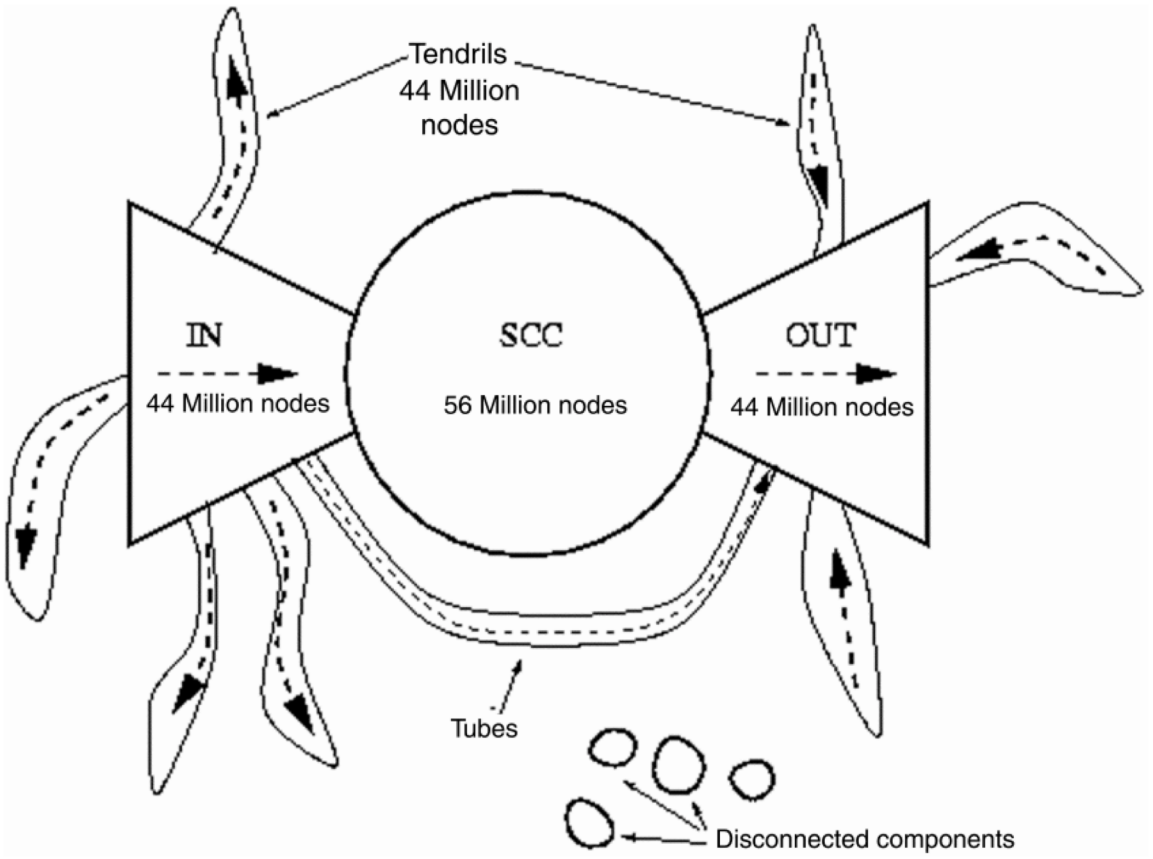
- $\boldsymbol{s} = \boldsymbol{M}.\boldsymbol{s}$ is the PageRank
  - Principal eigen-vector



All looks good so far

- $\boldsymbol{s} = \boldsymbol{M}.\boldsymbol{s}$ is the PageRank
  - Principal eigen-vector

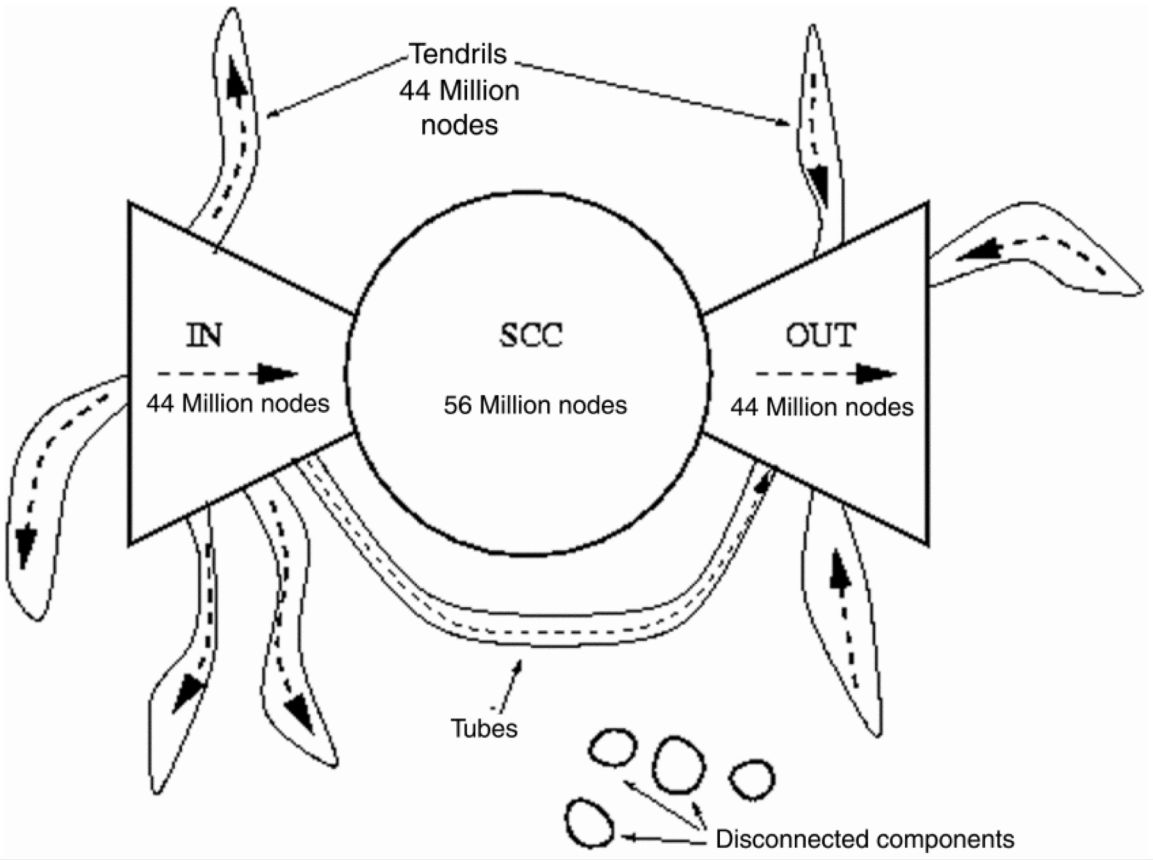
- Does this always converge?



All looks good so far

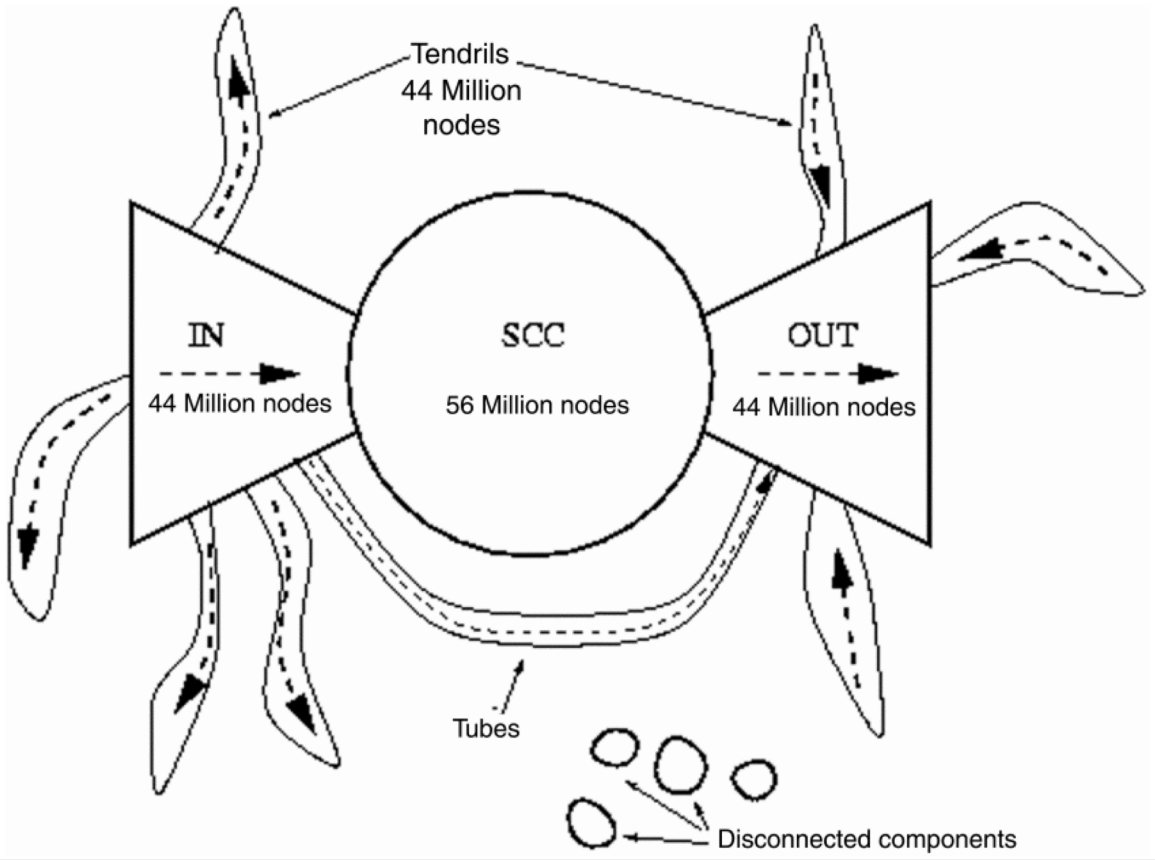
- $\boldsymbol{s} = \boldsymbol{M}.\boldsymbol{s}$ is the PageRank
  - Principal eigen-vector

- Does this always converge?
  - Yes



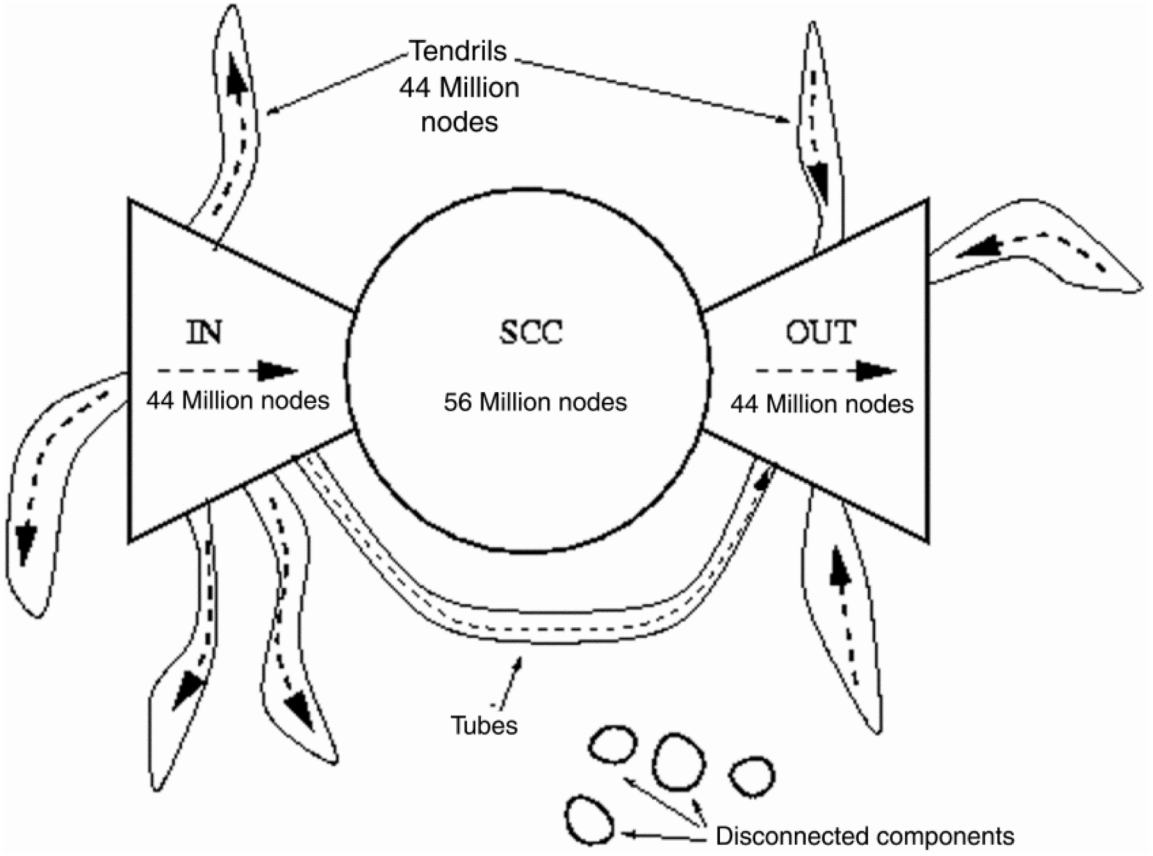
All looks good so far

- $\boldsymbol{s} = \boldsymbol{M}.\boldsymbol{s}$ is the PageRank
  - Principal eigen-vector



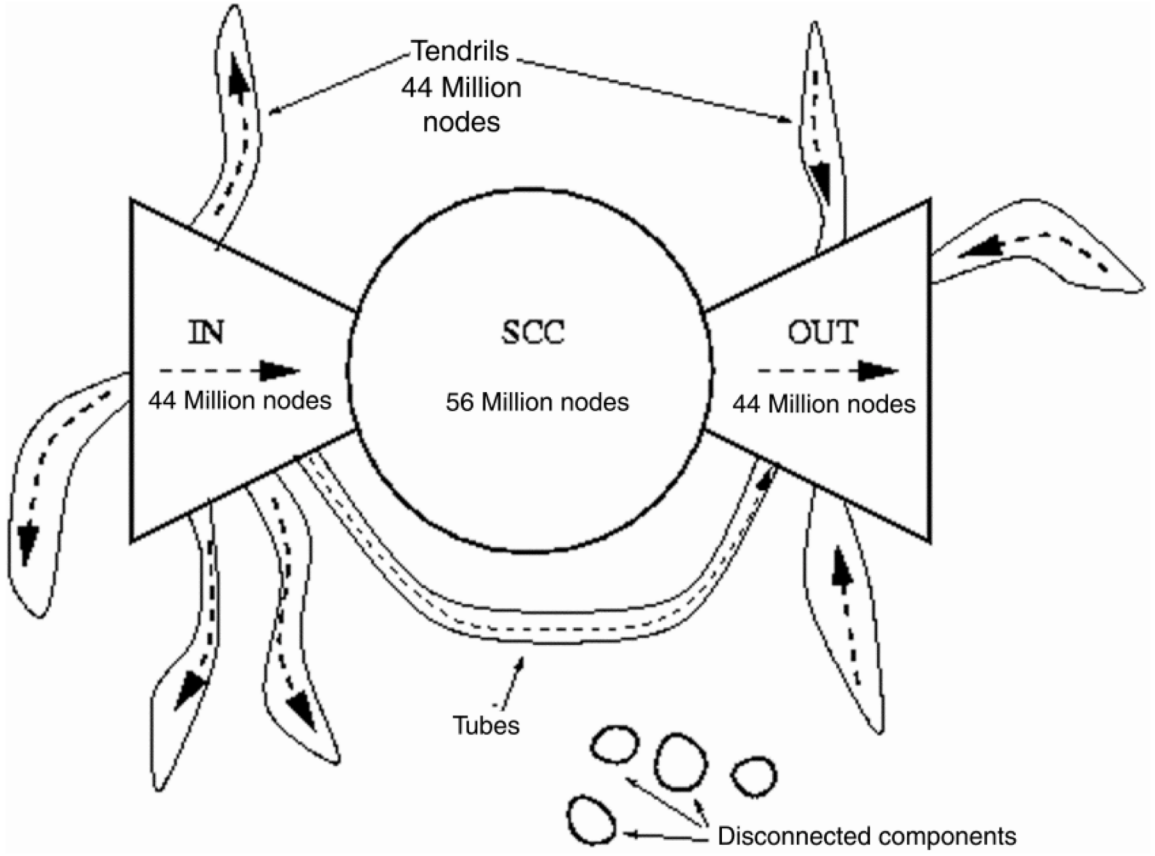
- Does this always converge?
  - Yes
- But, does this converge to a reasonable metric?
  - What structures are problematic?

Dead ends and spider traps



Dead ends and spider traps

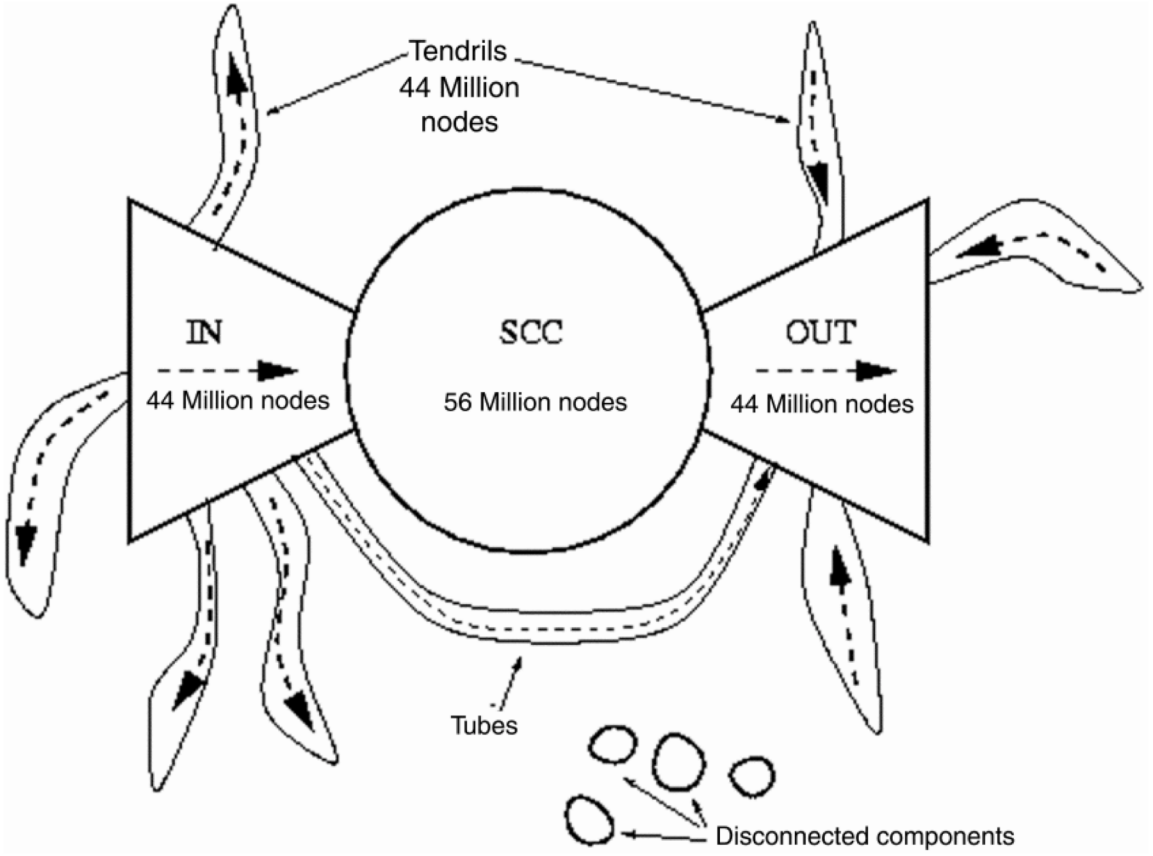
- Pages with no outgoing links!
  - Dead-end
  - Importance leaks out



Dead ends and spider traps

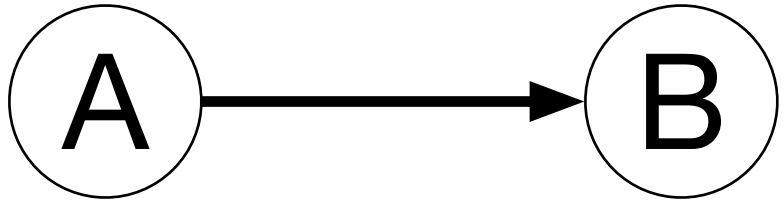
- Pages with no outgoing links!
  - Dead-end
  - Importance leaks out

- What if all out-going links are within a certain group?
  - Spider-trap
  - Absorbs all importance



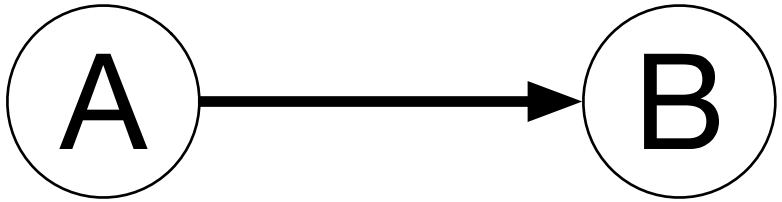
Examples

- Dead-end
  - s_A=1/2, 0, 0
  - s_B=1/2, 1/2, 0
  - Sum of scores is not 1!
    - Leak

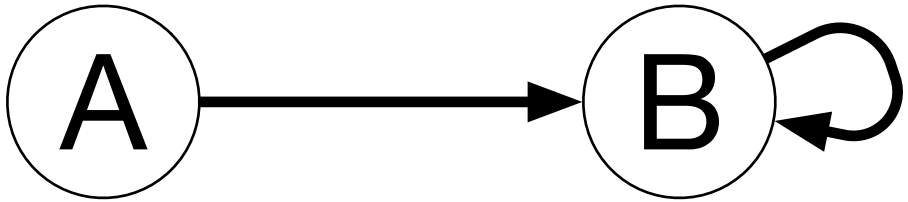


Examples

- Dead-end
  - s_A=1/2, 0, 0
  - s_B=1/2, 1/2, 0
  - Sum of scores is not 1!
    - Leak



- Spider-trap
  - s_A=1/2, 0,
  - s_B=1/2, 1,
  - A is ignored at all!



How to fix?

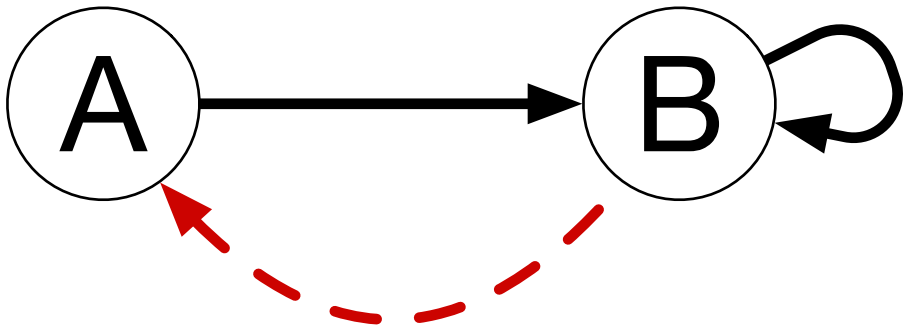
- Consider the web surfer

How to fix?

- Consider the web surfer
- What to do when get stuck?

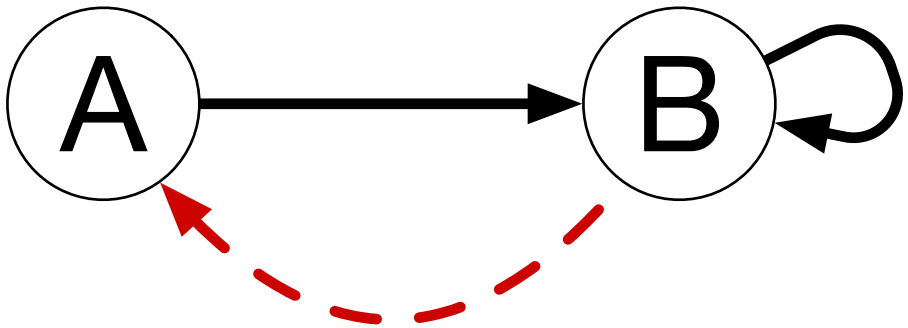
How to fix?

- Consider the web surfer
- What to do when get stuck?
  - Teleport!
  - Even proactively



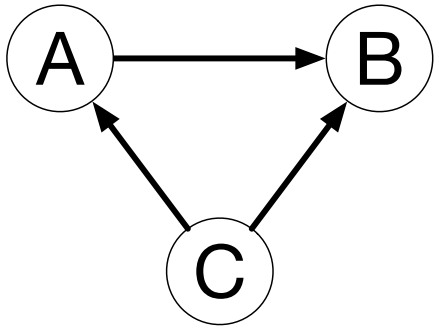
How to fix?

- Consider the web surfer
- What to do when get stuck?
  - Teleport!
  - Even proactively
- At each time, there are two options:
  - Follow a random out-going link with probability $\boldsymbol{\beta}$
  - Teleport to a random node with probability $\mathbf{1} - \boldsymbol{\beta}$
- Common values for $\boldsymbol{\beta}$ are in the range of 0.8-0.9



Adjusting computation

- Remember $\boldsymbol{M}$
- If a node has no outgoing link
  - Randomly jumps to a random node
  - With equal probability



	A	B	C
A	0	0.33	0.5
B	1	0.33	0.5
C	0	0.33	0

- Final equation: $\boldsymbol{s_j = \sum_{i \to j} \beta \frac{s_i}{d_o(i)} + (1 - \beta)\frac{1}{n}}$
  - Assuming $\boldsymbol{M}$ is edited for dead-end nodes

Final algorithm

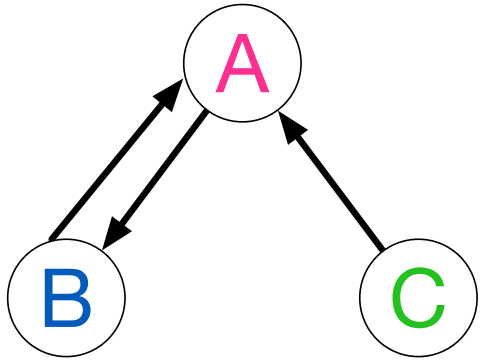
- Given a directed graph $\boldsymbol{G}$ with $\boldsymbol{n}$ nodes
  - Assign each node an initial score $\frac{\mathbf{1}}{\boldsymbol{n}}$
  - Calculate $\boldsymbol{a_j^{(t+1)} = \sum_{i \to j} \beta \frac{s_i^{(t)}}{d_o(i)} \; \forall \; j \epsilon G}$
    - $\boldsymbol{a_j^{(t+1)} = 0}$ if $\boldsymbol{d_i(i) = 0}$ (no incoming links)
  - Re-insert the leaked scores
    - $\boldsymbol{s_j^{(t+1)} = a_j^{(t+1)} + \frac{1-T}{n} \; \forall \; j \epsilon G}$ where $\boldsymbol{T = \sum_j a_j}$
  - Repeat until convergence $\boldsymbol{\sum_i |s_i^{(t+1)} - s_i^{(t)}| < \epsilon}$

Clarification

- That example did not converge
- Like A←→B
- M must be stochastic, aperiodic, irreducible (SCC)

		A	B	C		
2/3	A	0	1	1		1/3
1/3 =	B	1	0	0	·	2/3
0	C	0	0	0		0

		A	B	C		
1/3	A	0	1	1		2/3
2/3 =	B	1	0	0	·	1/3
0	C	0	0	0		0



s_A	1/3	2/3	1/3	2/3	1/3	…
s_B	1/3	1/3	2/3	1/3	2/3	…
s_C	1/3	0	0	0	0	…

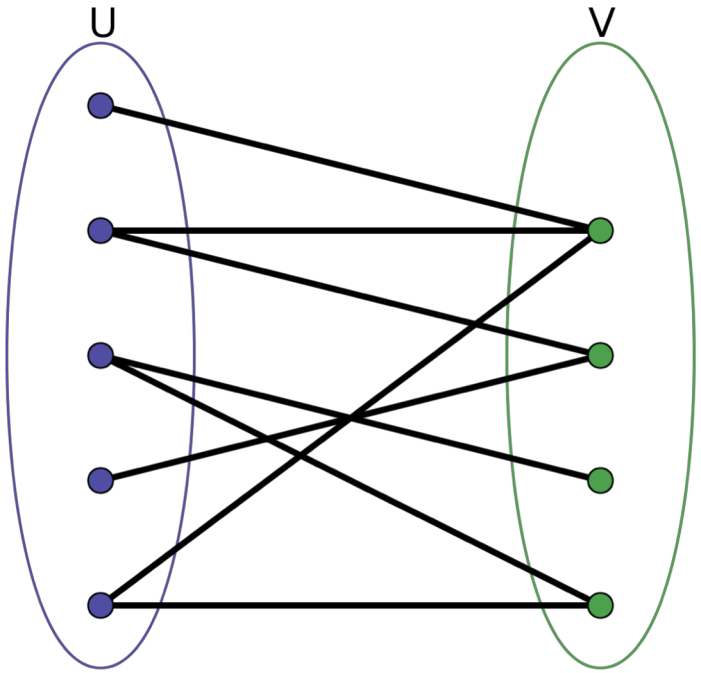
HITS (Hyper-text induced search)

- Measures of pages
- Another solution to same problem

- Consider newspapers, what is useful?
  - Content (Authorities)
    - World news, not Pawnee-IN news
  - Expertise (Hubs)
    - Followed by many, popular
- Links as votes

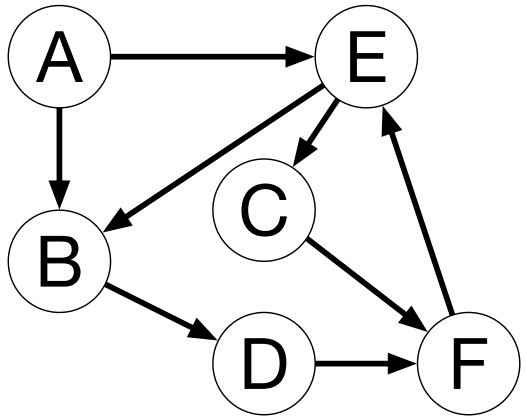
Hubs and Authorities

- Two scores for each node
  - Hub (Popularity)
    - Links to good authorities
  - Authority (Content)
    - Links to good hubs
  - Motivated by web directories (Yahoo!)
    - List of newspapers, game websites
- Iterative computation until convergence!
  - Hub scores feed authorities (sum)
  - Authority scores feed hubs (sum)



Computation

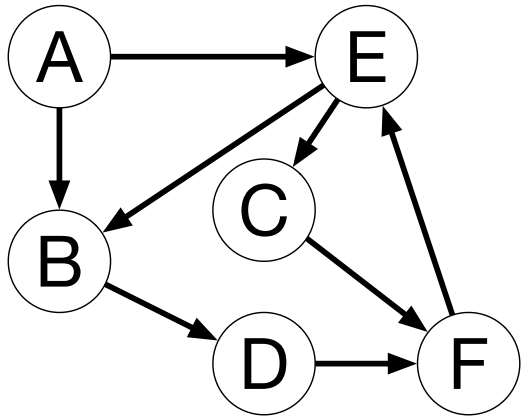
- a_i , h_i scores
- Initialize $a_i = h_i = 1$
- Iterate
  - $a_i = \sum_{j \to i} h_j \quad \forall\, i$
  - $h_i = \sum_{i \to j} a_j \quad \forall\, i$
- Normalize
  - $a_i = a_i / \sum_i a_i \quad \forall\, i$
  - $h_i = h_i / \sum_i h_i \quad \forall\, i$
- Until convergence



	A	B	C	D	E	F
hub	1	1	1	1	1	1
authority	0	0.25	0.12	0.12	0.25	0.25
hub	0.29	0.07	0.14	0.14	0.22	0.14
authority	0	0.33	0.14	0.05	0.29	0.19
…						
hub	0.44	0	0	0	0.36	0.20
authority	0	0.44	0.20	0	0.36	0

Matrix notation

- $\boldsymbol{h} = \boldsymbol{A}.\boldsymbol{a}$
- $\boldsymbol{a} = \boldsymbol{A}^{\boldsymbol{T}}.\boldsymbol{h}$
- $\boldsymbol{A}$ is adjacency matrix
- $\boldsymbol{a}$ is authority vector
- $\boldsymbol{h}$ is hub vector
- Scale to normalize
- Hub/authority of u is proportional to the sum of authority/hub scores of its out/in neighbors



	A	B	C	D	E	F
hub	1	1	1	1	1	1
authority	0	0.25	0.12	0.12	0.25	0.25
hub	0.29	0.07	0.14	0.14	0.22	0.14
authority	0	0.33	0.14	0.05	0.29	0.19
	…					
hub	0.44	0	0	0	0.36	0.20
authority	0	0.44	0.20	0	0.36	0

Existence and uniqueness

- $\boldsymbol{h} = \boldsymbol{\lambda}.\boldsymbol{A}.\boldsymbol{a}$ where $\boldsymbol{\lambda} = \mathbf{1}/\sum \boldsymbol{h_i}$
- $\boldsymbol{a} = \boldsymbol{\mu}.\boldsymbol{A}^{\boldsymbol{T}}.\boldsymbol{h}$ where $\boldsymbol{\mu} = \mathbf{1}/\sum \boldsymbol{a_i}$

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

- Converged $\boldsymbol{h}$ is the principal eigen-vector of $\boldsymbol{A}.\boldsymbol{A}^{\boldsymbol{T}}$
- Converged $\boldsymbol{a}$ is the principal eigen-vector of $\boldsymbol{A}^{\boldsymbol{T}}.\boldsymbol{A}$

Both applicable beyond web

- PageRank and HITS are useful for any directed network
- Comprehensive survey
- *PageRank Beyond the Web, by D. Gleich*
  - *SIAM Rev.*, 57(3), 321–363
  - https://epubs.siam.org/doi/abs/10.1137/140976649