Link Analysis

Lecture 7

Outline

1. Let's Build a Search Engine :)
   - The Model
   - Problem #1: Fast Text Search
   - Problem #2: Ranking Documents
   - Enter the 

ers

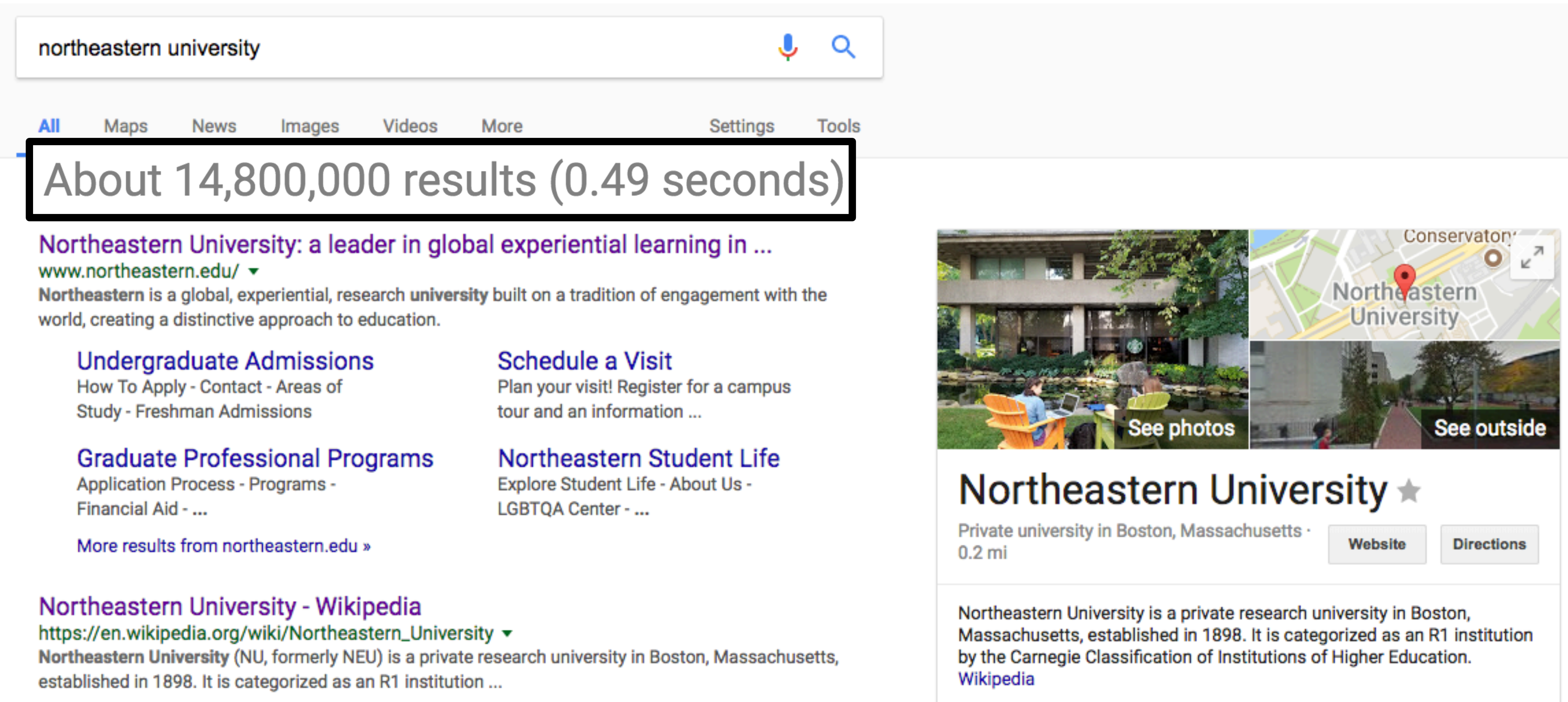
2. Bringing Order to the Web
   - Voting with Your Links
     - Two Equivalent Views
   - Problem Representation
     - Spiders Everywhere!
   - Return of the Spammer (Farms)

3. Related Approaches
   - Topic-Specific PageRank
   - SimRank
   - HITS: Hubs and Authorities

What Makes a Search Engine?

- Inputs
  - The Web (set of webpages)
    - Text, images, downloads
    - Links to other pages
  - User query
    - Simplest: set of words
- Outputs
  - Ranked list of the "best" documents related to the query
  - Desired properties
    - Fast & scalable
    - Relevant results
    - Expressive queries
    - Up-to-date

Problem #1


northeastern university
All
Maps
News
Images
Videos
More
Settings
Tools
About 14,800,000 results (0.49 seconds)
Northeastern University: a leader in global experiential learning in ...
www.northeastern.edu/
Northeastern is a global, experiential, research university built on a tradition of engagement with the world, creating a distinctive approach to education.
Undergraduate Admissions
How To Apply - Contact - Areas of Study - Freshman Admissions
Schedule a Visit
Plan your visit! Register for a campus tour and an information ...
Graduate Professional Programs
Application Process - Programs - Financial Aid - ...
Northeastern Student Life
Explore Student Life - About Us - LGBTQA Center - ...
More results from northeastern.edu »
Northeastern University - Wikipedia
https://en.wikipedia.org/wiki/Northeastern_University
Northeastern University (NU, formerly NEU) is a private research university in Boston, Massachusetts, established in 1898. It is categorized as an R1 institution ...
Northeastern University
See photos
See outside
Northeastern University
Private university in Boston, Massachusetts · 0.2 mi
Website
Directions
Northeastern University is a private research university in Boston, Massachusetts, established in 1898. It is categorized as an R1 institution by the Carnegie Classification of Institutions of Higher Education.
Wikipedia

Fast & Scalable: # of Documents

Google	> 130 Trillion Pages
facebook	~ 2 Billion Users
amazon	~ 400 Million Products

Fast & Scalable: Queries/sec

Google	> 63K
	~ 6K

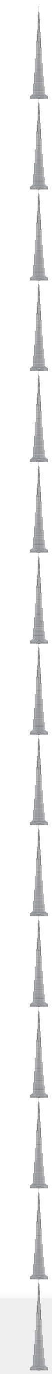
Linear-Time Google

- Assume simple query:

northeastern university

- Single repo of all pages
  - 130T * 250 w/page * 8 bytes/w **~ 260 PB**

~10.4M blu ray
~ 7.75 miles
~ 15 x Burj Khalifa



- Require 1s response time
  - 8M * 3GHz 64-bit CPU (assume 1 cycle/w)
  - (8M * 63K) CPUs * 35W/CPU * $0.14/kWH

    ~ $0.7M/sec

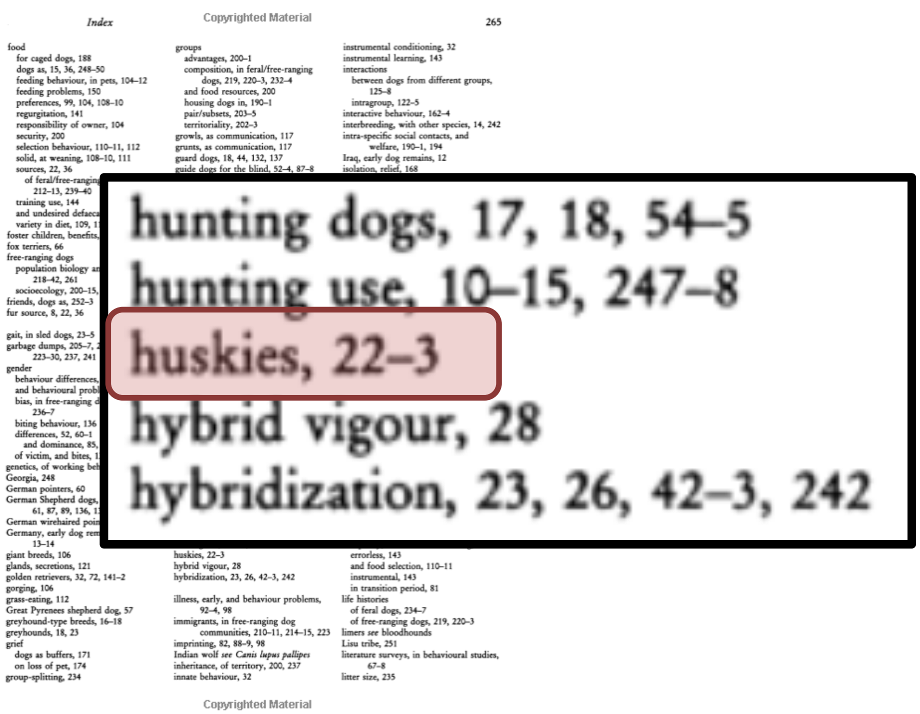
~ $21.6T/year
~ 1.25 x US GDP

~67 CPU/person

Enter: The Inverted Index

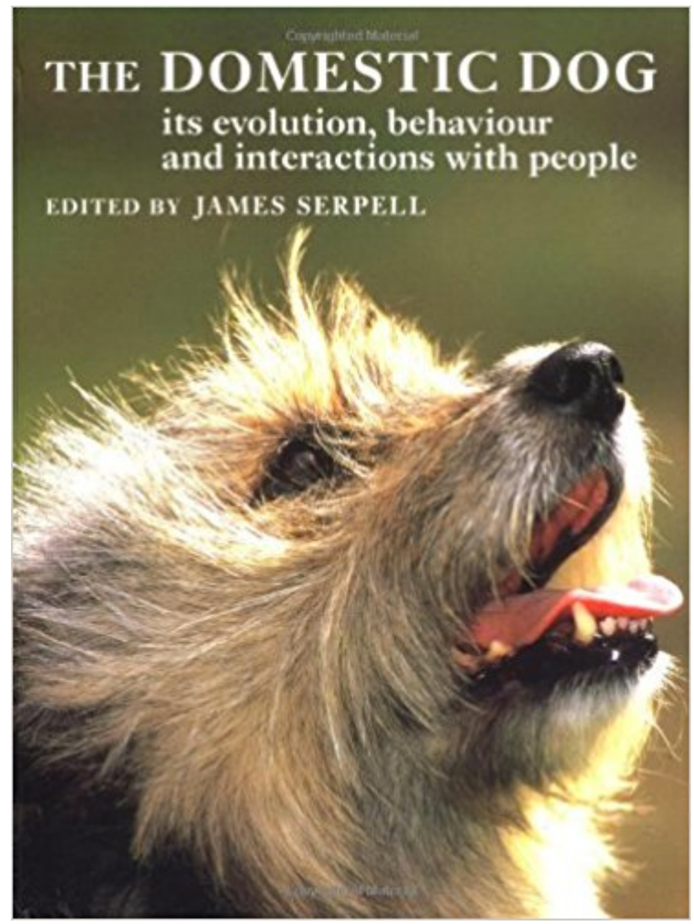
Physical Book

Find all pages that contain the word "Husky"



Index Copyrighted Material 265

hunting dogs, 17, 18, 54–5
hunting use, 10–15, 247–8
huskies, 22–3
hybrid vigour, 28
hybridization, 23, 26, 42–3, 242

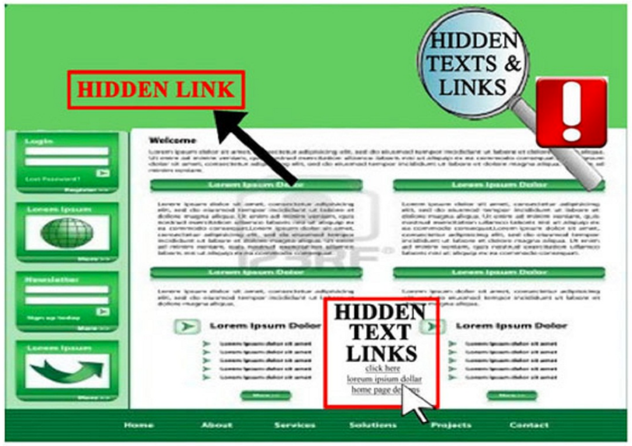


In the Beginning…

In the Beginning…

Initial Approaches

- Human-curated (e.g. Yahoo)
  - Hand-written descriptions
  - Wait time for inclusion



- Text-search (e.g. Lycos)
  - Prone to term spam

- **Core Question**: how to automatically rank pages (i.e. efficiently) in a quality way that is resistant to term spam?
  - And, at least early on, in an **unsupervised** fashion (i.e. given only the contents of the pages + structure of the web)

A Humble Academic Paper

L. Page, S. Brin, R. Motwani, T. Winograd

The PageRank Citation Ranking: Bringing Order to the Web

January 29, 1998

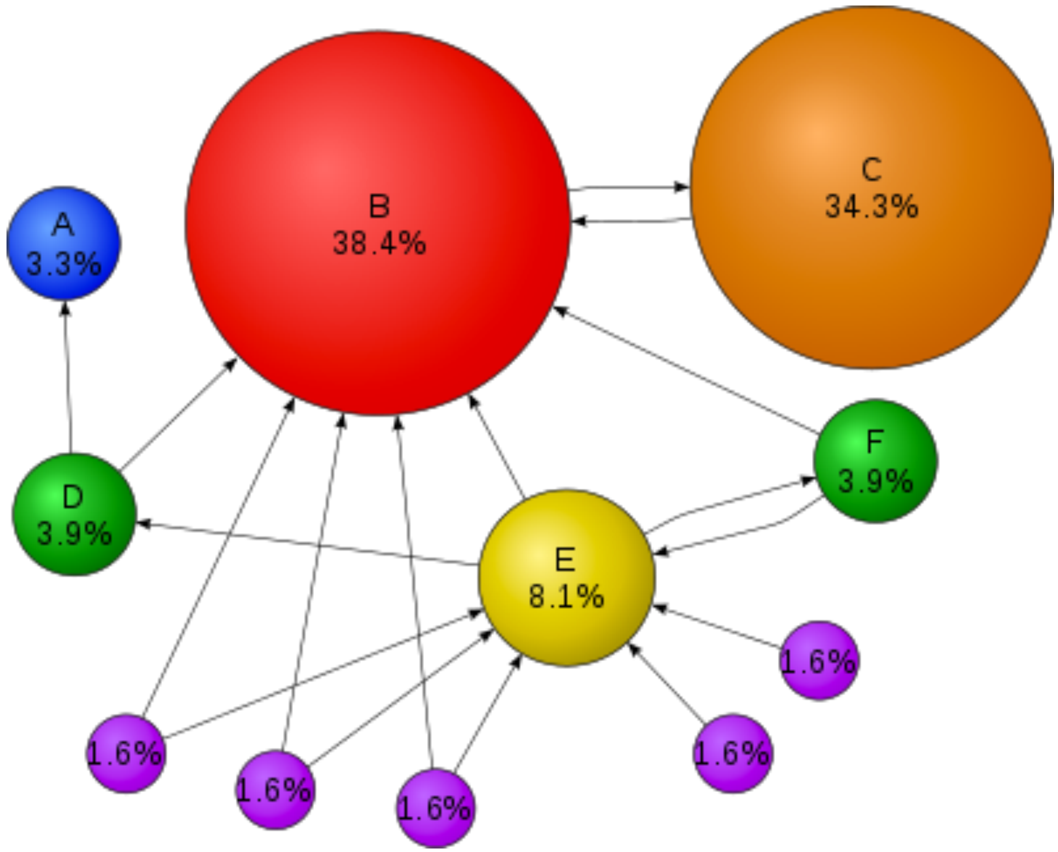
Abstract

The importance of a Web page is an inherently subjective matter, which depends on the readers interests, knowledge and attitudes. But there is still much that can be said objectively about the relative importance of Web pages. This paper describes PageRank, a method for rating Web pages objectively and mechanically, effectively measuring the human interest and attention devoted to them.

We compare PageRank to an idealized random Web surfer. We show how to efficiently compute PageRank for large numbers of pages. And, we show how to apply PageRank to search and to user navigation.

Basic Assumptions

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



The PageRank Model

- Each out-link is an implicit conveyance of authority to the target page
  - Equal amount per link
- So the PageRank of node i, P(i), equals the sum of each incoming link's PageRank proportion

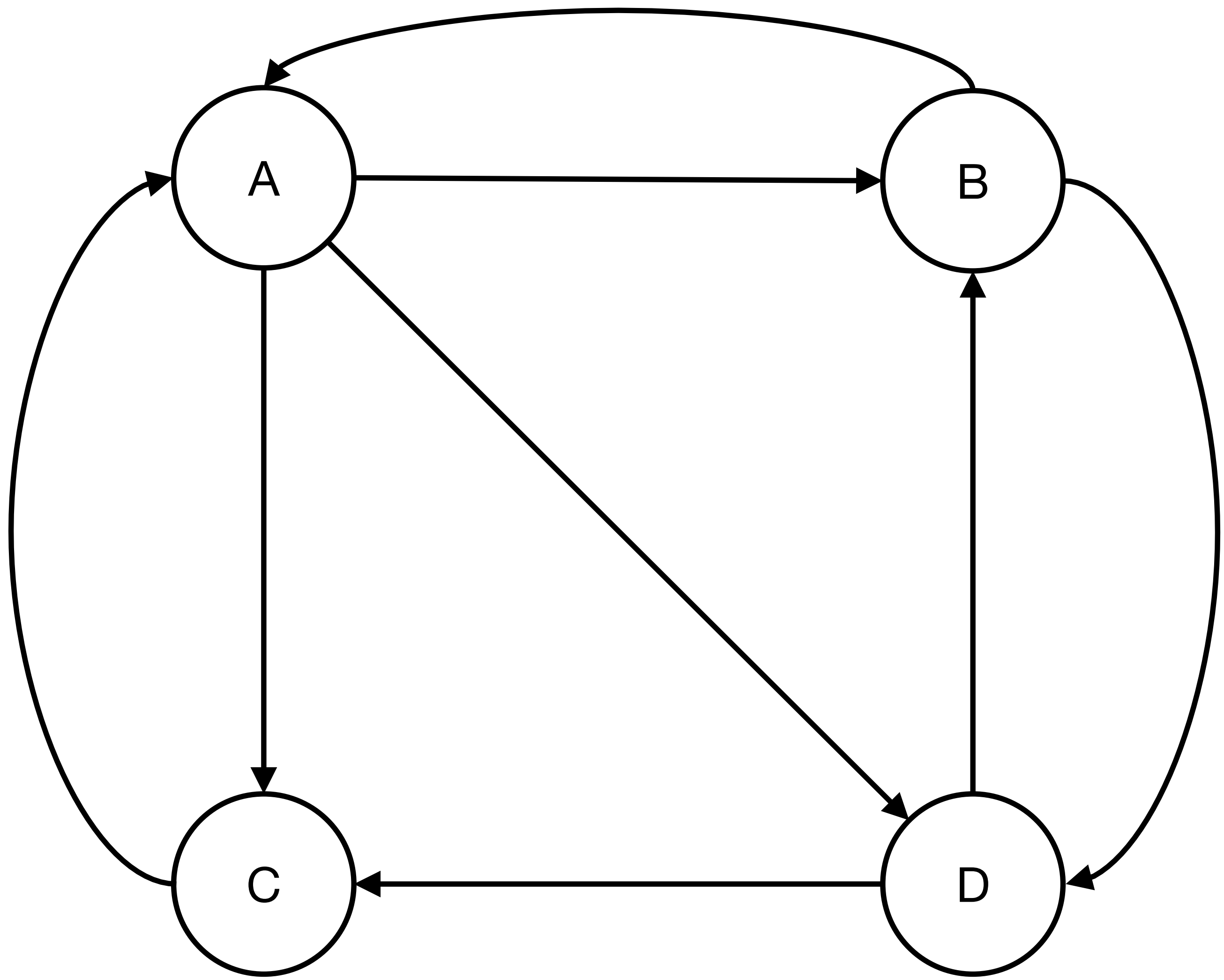
$$P(i) = \sum_{(j,i) \in E} \frac{P(j)}{O_j}$$

O_j : number of out-links of page j

An Equivalent View of PageRank

- At time t a surfer is on some page
  - Start uniformly at random
- At time t+1 the surfer follows a link to a new page at random
  - A *Markov Chain*
- Define PageRank where the surfer is likely to be after a long time

The Model as a Directed Graph



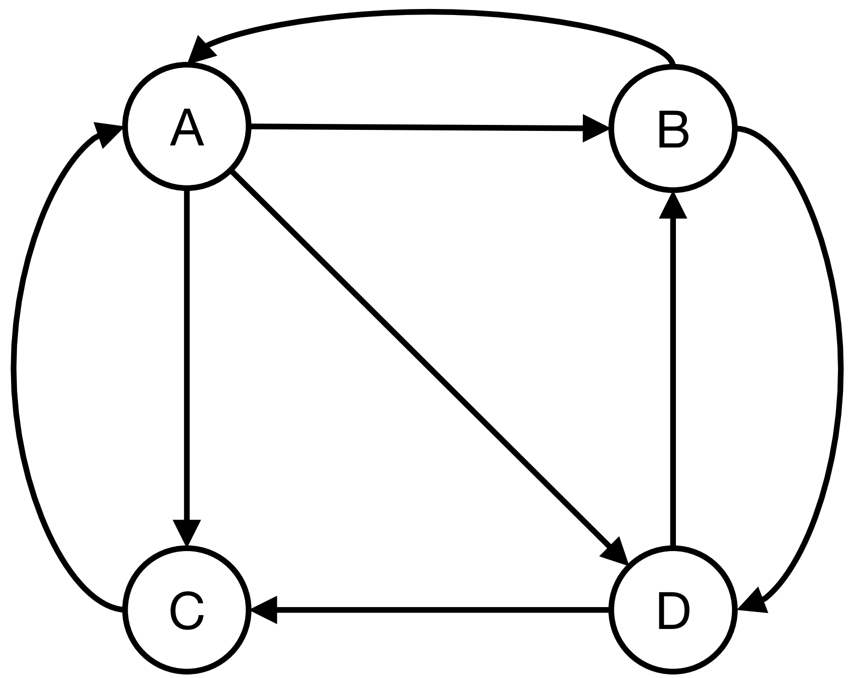
Random Surfer via Transition Matrix

- Weight each edge equally…

P(X|A)
Starting at A, probability of arriving at node X

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

P(A|X)
Starting at X, "probability" of arriving at node D



Checkup

- Assume a surfer has an equal probability of starting at any site
- What is the probability of arriving at each of the four sites at the next time step given the transition matrix?

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

Answer

$$\begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} = \begin{bmatrix} 9/24 \\ 5/24 \\ 5/24 \\ 5/24 \end{bmatrix}$$

A Few More Waves…

$$\begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \begin{bmatrix} 9/24 \\ 5/24 \\ 5/24 \\ 5/24 \end{bmatrix} \begin{bmatrix} 15/48 \\ 11/48 \\ 11/48 \\ 11/48 \end{bmatrix} \begin{bmatrix} 11/32 \\ 7/32 \\ 7/32 \\ 7/32 \end{bmatrix} \ldots \begin{bmatrix} 3/9 \\ 2/9 \\ 2/9 \\ 2/9 \end{bmatrix}$$

An Aside… Was This Surprising?

$$\begin{bmatrix} 0 & 1/2 & 1 & 0 \\ \boxed{1/3} & \boxed{0} & \boxed{0} & \boxed{1/2} \\ \boxed{1/3} & \boxed{0} & \boxed{0} & \boxed{1/2} \\ \boxed{1/3} & \boxed{1/2} & \boxed{0} & \boxed{0} \end{bmatrix}$$

Same probability of arriving at B/C/D – call each X

An Aside… Was This Surprising?

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

(½)X + X

Same probability of arriving at B/C/D – call each X

An Aside… Was This Surprising?

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

(½)X + X

Same probability of arriving at B/C/D – call each X

$$X + X + X + (3/2)X = 1$$

$$\frac{9}{2}X = 1$$

$$X = \frac{2}{9} \qquad \frac{3}{2} \cdot X = \frac{3}{2} \cdot \frac{2}{9} = \frac{3}{9}$$

Checkup

$$\begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} 3/9 \\ 2/9 \\ 2/9 \\ 2/9 \end{bmatrix} =$$

Answer

$$\begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} 3/9 \\ 2/9 \\ 2/9 \\ 2/9 \end{bmatrix} = \begin{bmatrix} 3/9 \\ 2/9 \\ 2/9 \\ 2/9 \end{bmatrix}$$

$$MP = P$$

$$MP = \lambda P$$

$\lambda = ?$

What Just Happened!!??

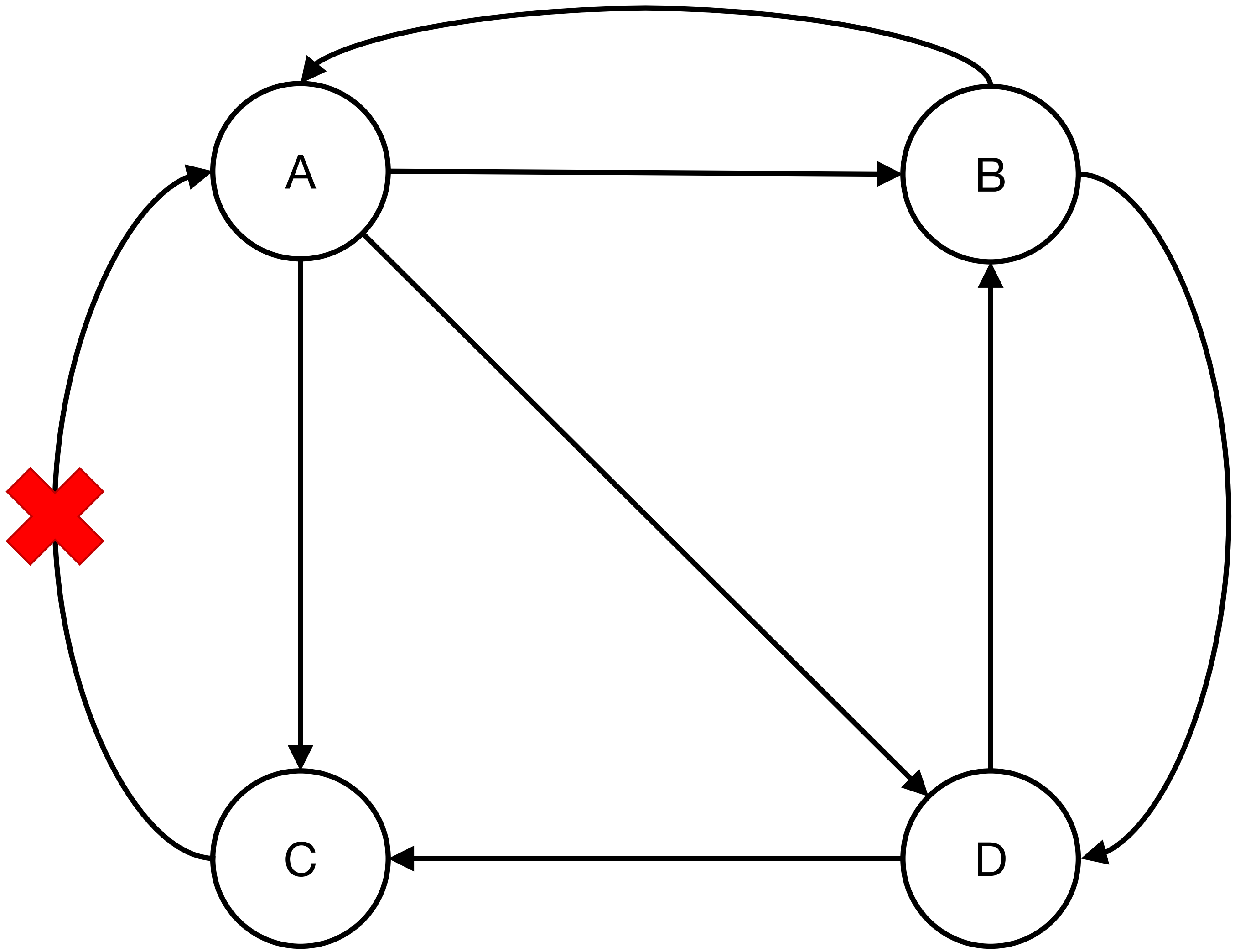
- If certain conditions hold, 1 is the largest eigenvalue and the PageRank vector is the principal eigenvector of the transition matrix
  - The conditions held for our example, but not in the general case for the web (yet!)

- We intuitively used a method called **power iteration** to compute P
  - Useful particularly when the matrix in question is large & sparse, as the method doesn't require any decomposition
  - Convergence: typically when the residual (norm of the difference between P vs P') is below a threshold; may require many iterations, typically few are good enough for the web (according to Google)

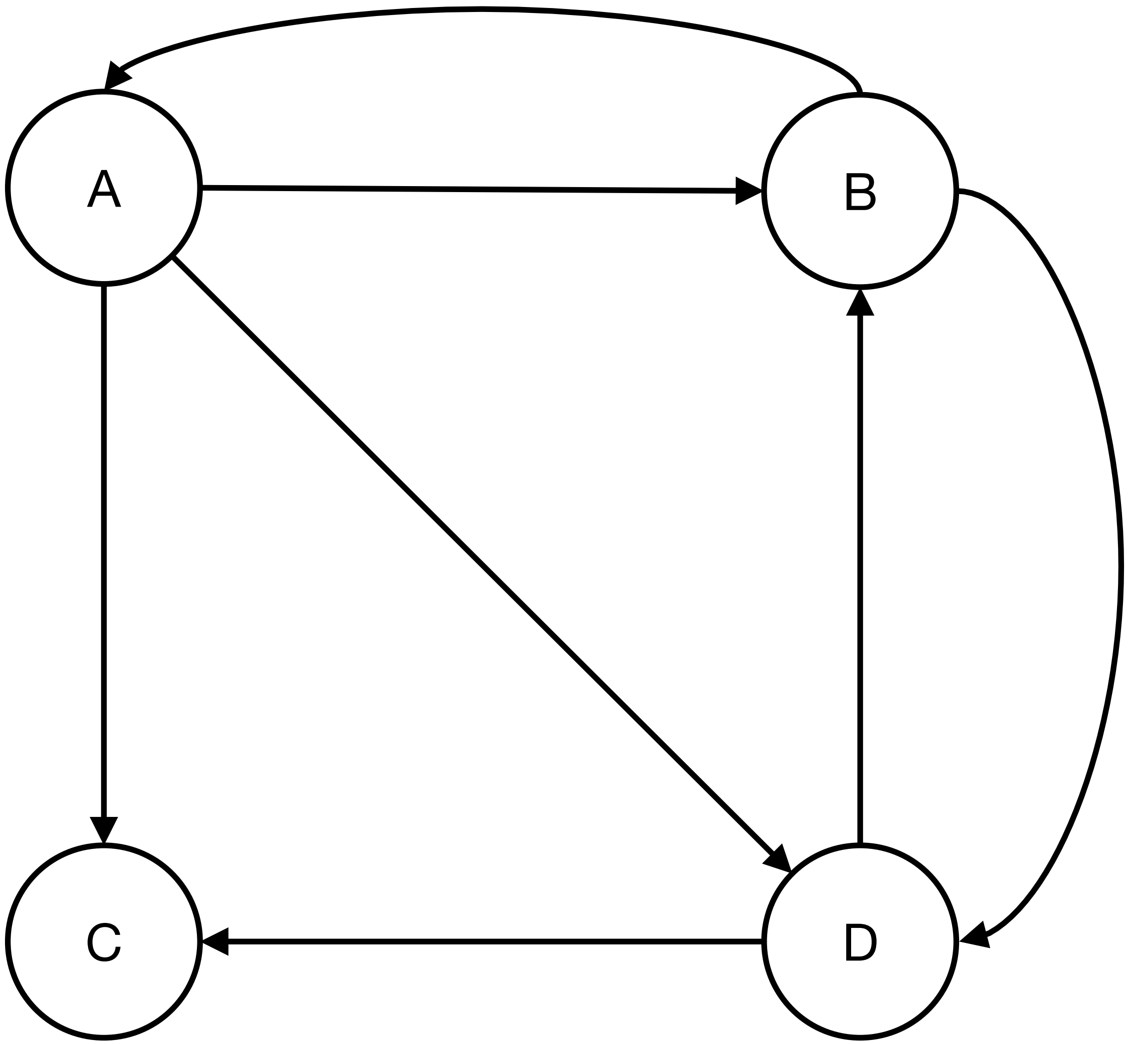
So What are these Conditions…?

- Stochastic: columns sum to 1
  - Done… right?
- Strongly connected: possible to get from any node to any other node
  - Might be very unlikely, but must be possible!

What Would Happen If…



What Would Happen If…

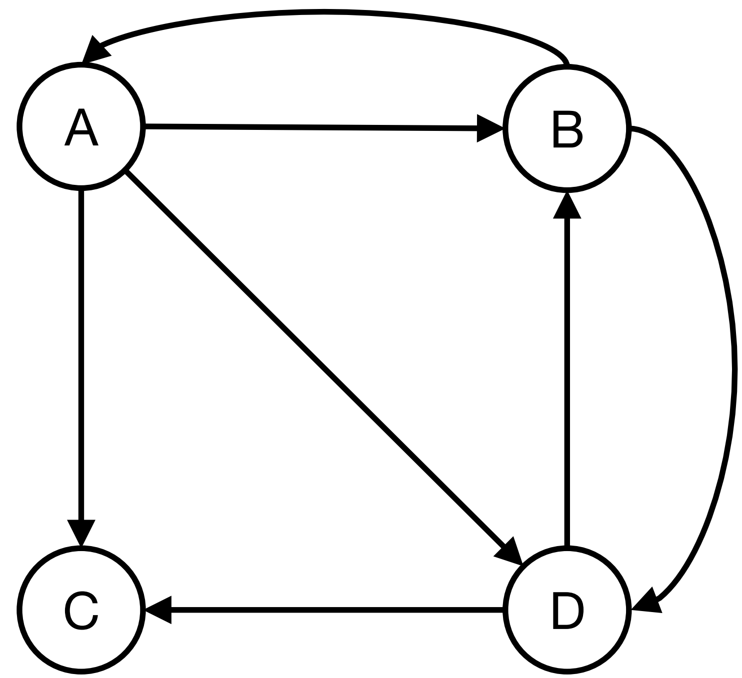


C is termed a **dead end**

New Transition Matrix

- Weight each edge equally…

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



Now **substochastic**
(columns sum to at most 1)

Checkup

- Assume a surfer has an equal probability of starting at any site
- What is the probability of arriving at each of the four sites at the next time step given the transition matrix?

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

Answer

$$\begin{bmatrix} 0 & 1/2 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} = \begin{bmatrix} 3/24 \\ 5/24 \\ 5/24 \\ 5/24 \end{bmatrix}$$

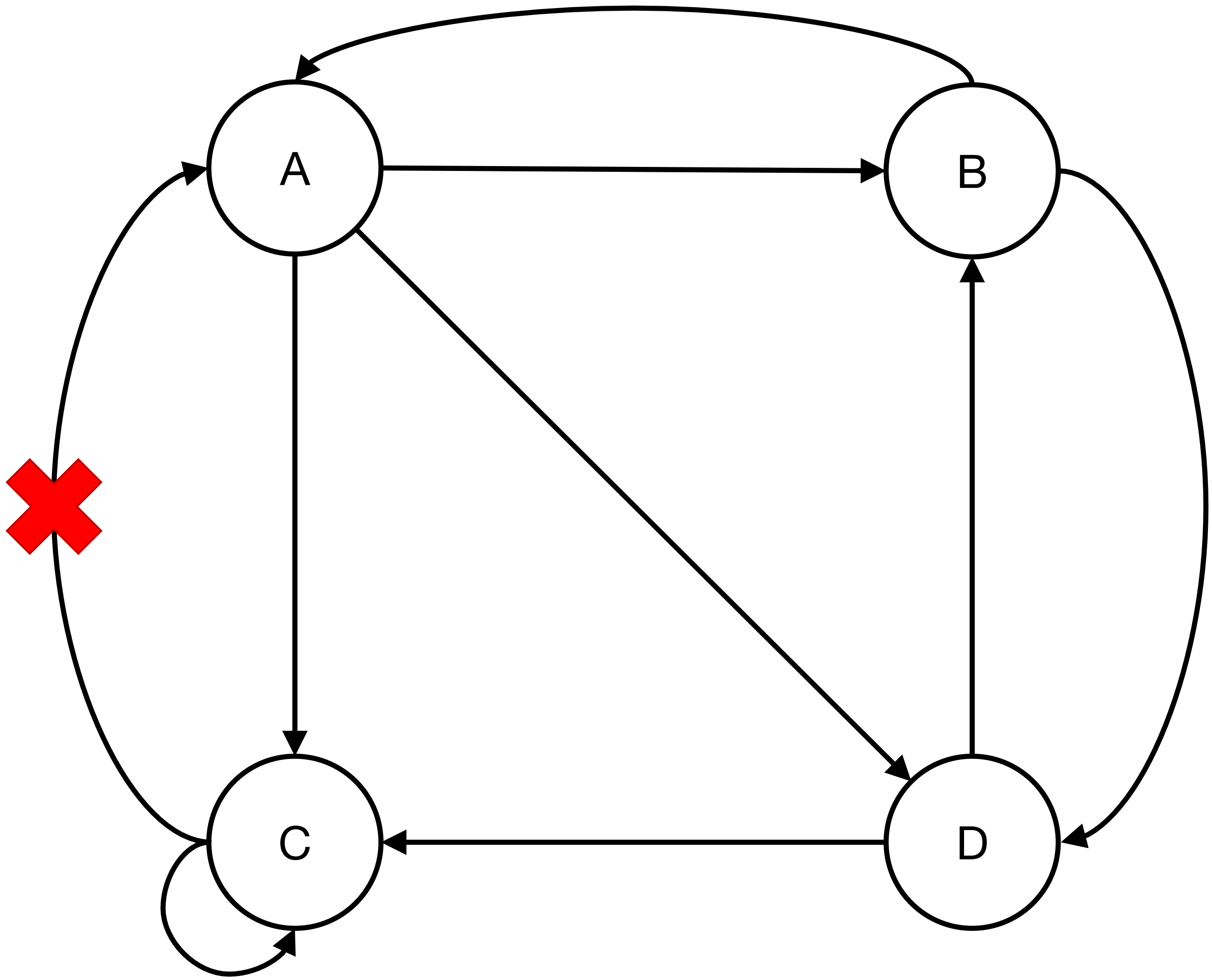
25% loss!

A Few More Waves…

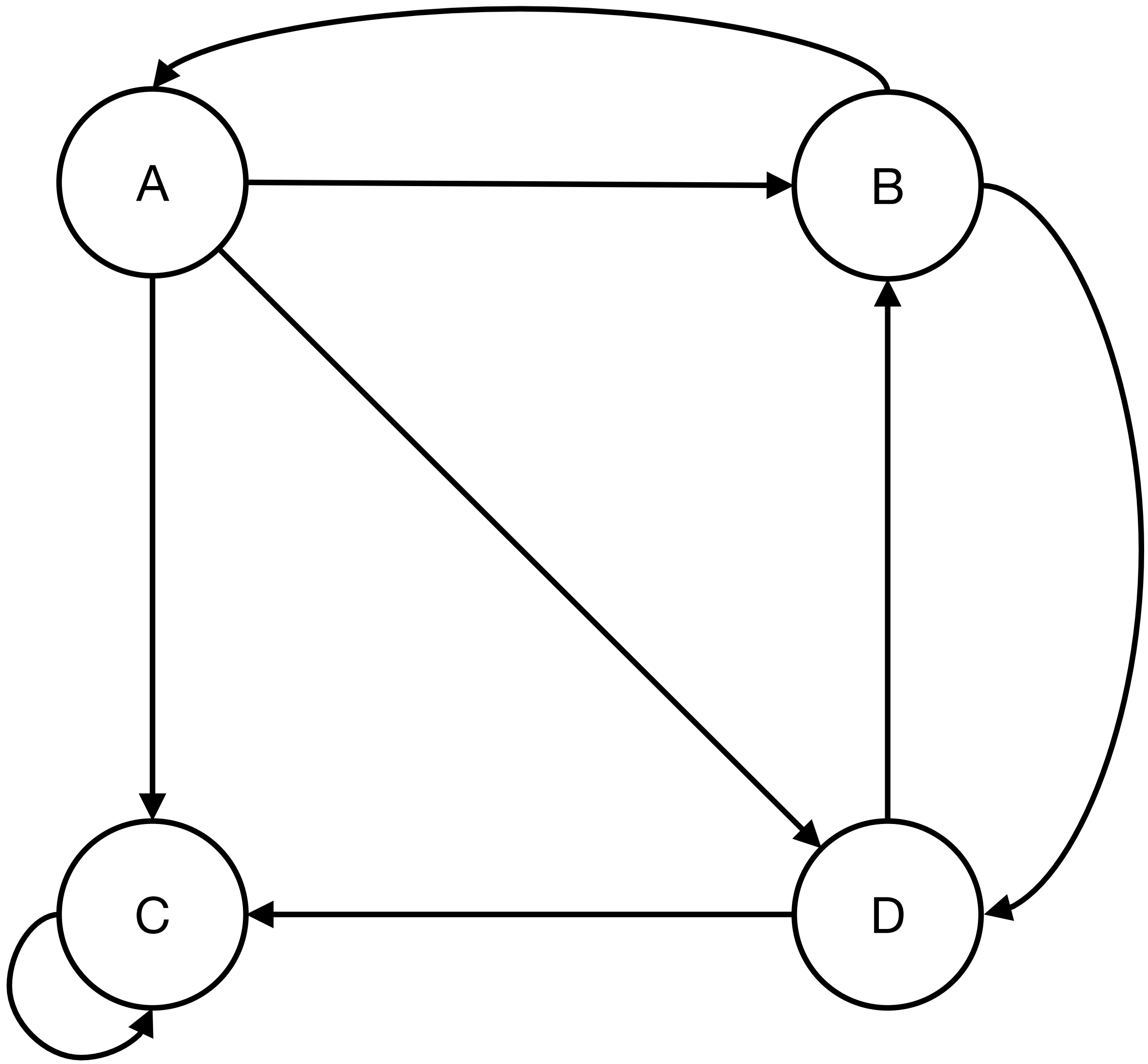
$$\begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \begin{bmatrix} 3/24 \\ 5/24 \\ 5/24 \\ 5/24 \end{bmatrix} \begin{bmatrix} 5/48 \\ 7/48 \\ 7/48 \\ 7/48 \end{bmatrix} \begin{bmatrix} 21/288 \\ 31/288 \\ 31/288 \\ 31/288 \end{bmatrix} \ldots \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

1	0.75	0.54	0.29		0

What Would Happen If…



What Would Happen If…

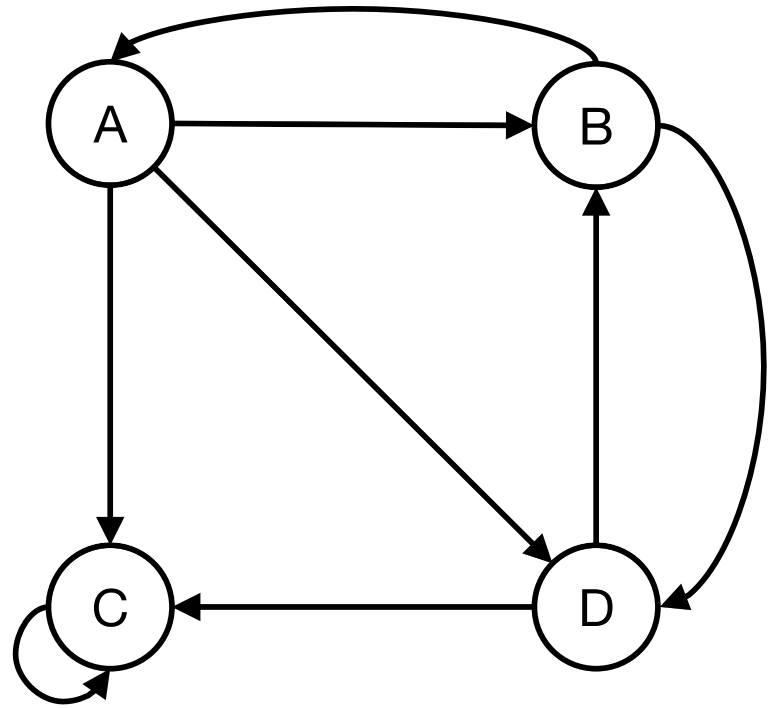


Simplest form of a **spider trap** (could involve multiple nodes)

Sticky Transition Matrix

- Weight each edge equally…

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



Stochastic – yes!
But strongly connected?

Checkup

- Assume a surfer has an equal probability of starting at any site
- What is the probability of arriving at each of the four sites at the next time step given the transition matrix?

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

Answer

$$\begin{bmatrix} 0 & 1/2 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 1 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} = \begin{bmatrix} 3/24 \\ 5/24 \\ 11/24 \\ 5/24 \end{bmatrix}$$

A Few More Waves…

$$\begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \begin{bmatrix} 3/24 \\ 5/24 \\ 11/24 \\ 5/24 \end{bmatrix} \begin{bmatrix} 5/48 \\ 7/48 \\ 29/48 \\ 7/48 \end{bmatrix} \begin{bmatrix} 21/288 \\ 31/288 \\ 205/288 \\ 31/288 \end{bmatrix} \ldots \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Solution: Damping/Taxation

- Allow a relatively small probability of hopping from any page to any other page (teleporting!)
  - Typical: 10-15%

- Ensures the requirements of the model

PageRank Model

Simplified

$$MP = P$$

Damped

$$((1-d)\frac{E}{n} + dM)P = P$$

$$(1-d)\frac{\mathbf{e}}{n} + dMP = P$$

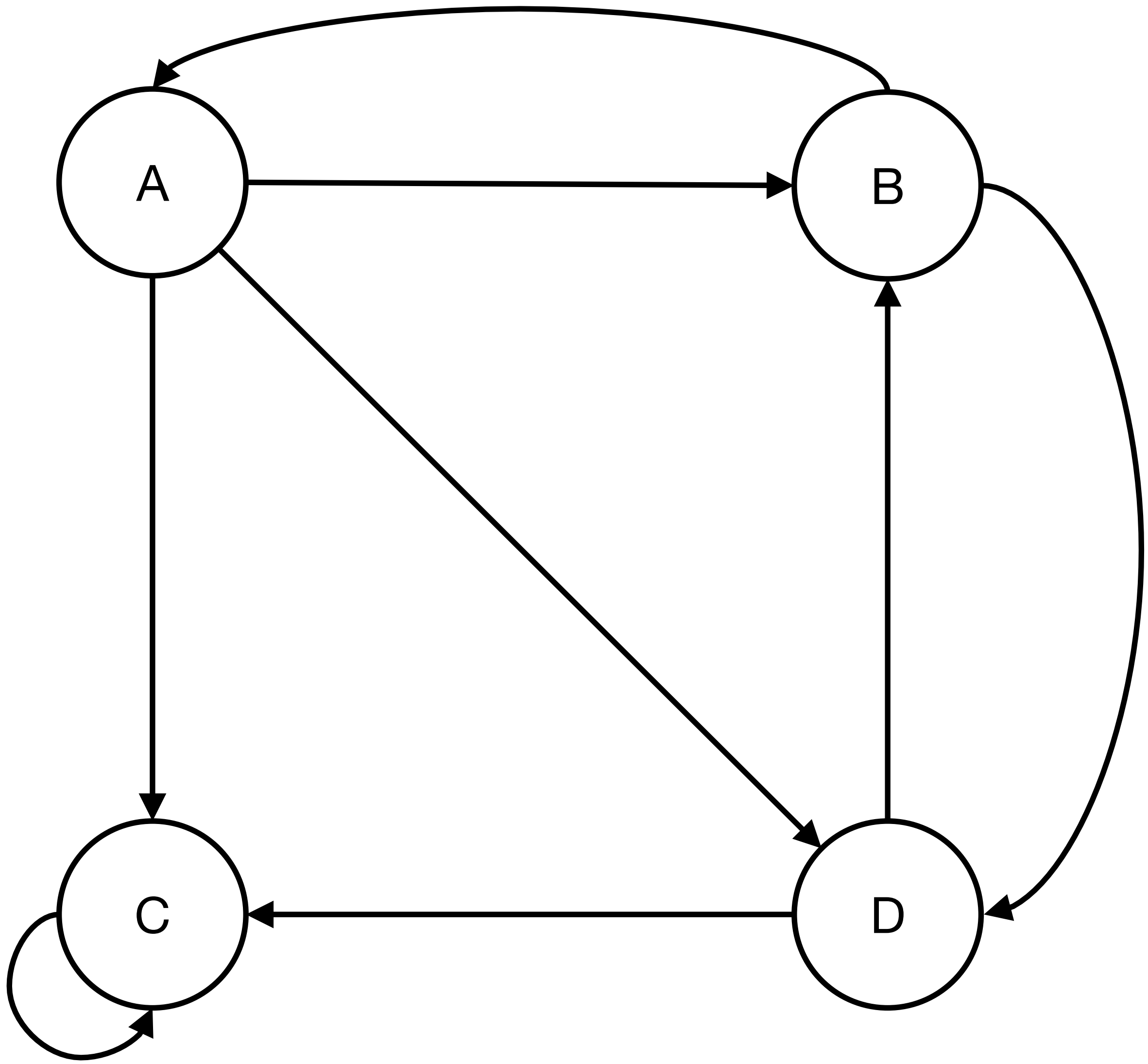
n: number of nodes

E: ee^T

- e: n-length column of 1's

Think back to Naïve Bayes
Anything seem *smoother?*
Think about extreme values of d…

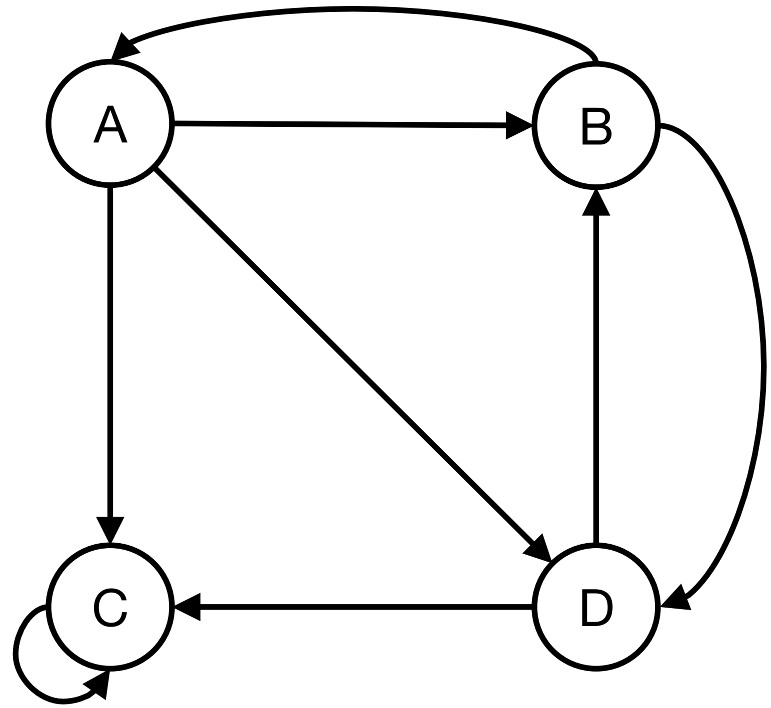
Recall the Spider Trap



New Model: d=0.8

- Weight each edge equally…

$$\begin{bmatrix} 0 & 2/5 & 0 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 4/5 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix}$$



Checkup

- Assume a surfer has an equal probability of starting at any site
- What is the probability of arriving at each of the four sites at the next time step given the transition and damping matrices?

$$\begin{bmatrix} 0 & 2/5 & 0 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 4/5 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1/20 \\ 1/20 \\ 1/20 \\ 1/20 \end{bmatrix}$$

Answer

$$\begin{bmatrix} 0 & 2/5 & 0 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 4/5 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} + \begin{bmatrix} 1/20 \\ 1/20 \\ 1/20 \\ 1/20 \end{bmatrix} = \begin{bmatrix} 9/60 \\ 13/60 \\ 25/60 \\ 13/60 \end{bmatrix}$$

A Few More Waves…

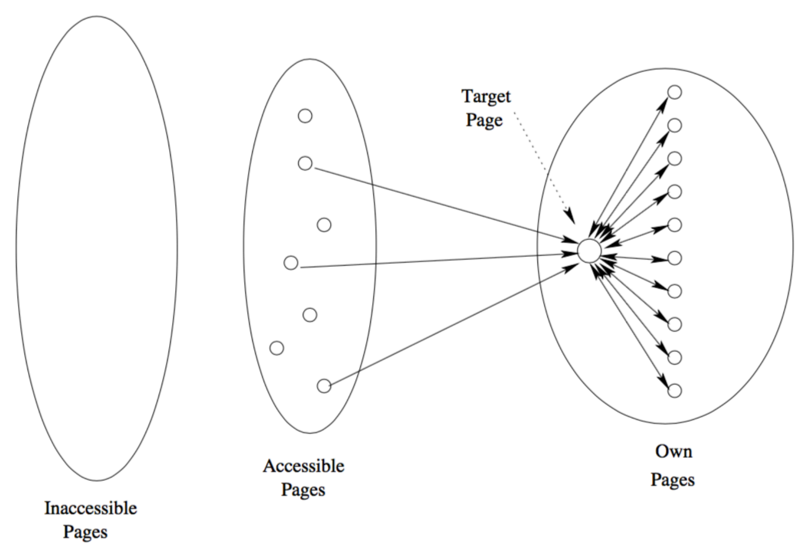
$$\begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \begin{bmatrix} 9/60 \\ 13/60 \\ 25/60 \\ 13/60 \end{bmatrix} \begin{bmatrix} 41/300 \\ 53/300 \\ 153/300 \\ 53/300 \end{bmatrix} \dots \begin{bmatrix} 15/148 \\ 19/148 \\ 95/148 \\ 19/148 \end{bmatrix}$$

Compare to…

$$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

PageRank in Practice

- Efficient implementation via MapReduce
  – See LRU
- Still have to deal with Spam Farms
  – See LRU



Some Related Approaches

- Topic-Sensitive PageRank
- SimRank
- HITS

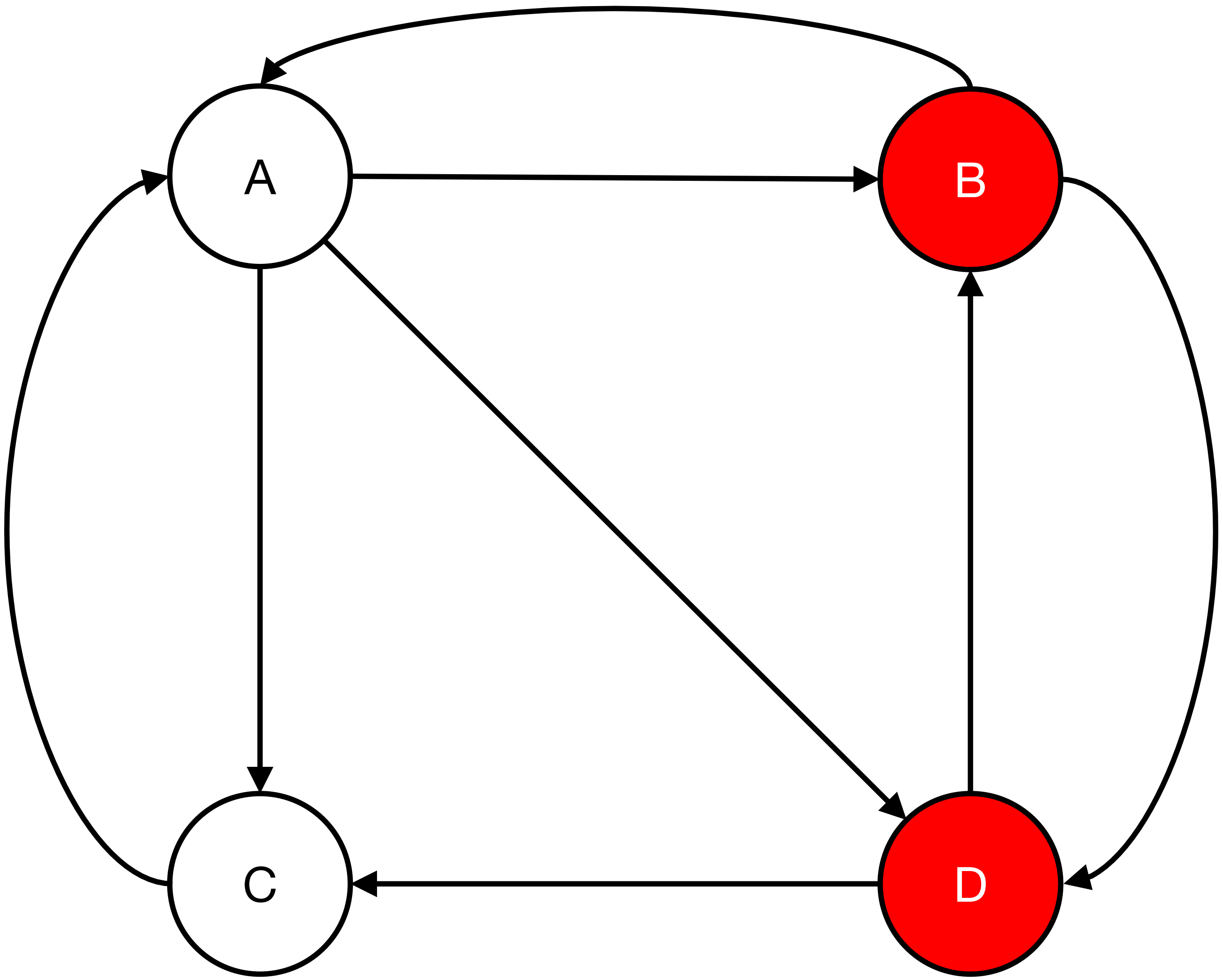
Biased Random Walks

- Suppose we wish to create a ranking for "sports" (or some other topic)

- We can modify PageRank via a **teleport set** (representative topic-related pages)
  - Could look to a known directory/authority (e.g. http://dmoztools.net)

- Start randomly from within this set & only include them within the damping options

Creating a RED PageRank



Checkup

- Assume a surfer has an equal probability of starting at any RED site (B, D)
- What is the probability of arriving at each of the four sites at the next time step given the transition and damping matrices?

$$\begin{bmatrix} 0 & 2/5 & 4/5 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ 1/10 \\ 0 \\ 1/10 \end{bmatrix}$$

(1 – 0.8) / 2 nodes

Answer

$$\begin{bmatrix} 0 & 2/5 & 4/5 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0/2 \\ 1/2 \\ 0/2 \\ 1/2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/10 \\ 0 \\ 1/10 \end{bmatrix} = \begin{bmatrix} 2/10 \\ 3/10 \\ 2/10 \\ 3/10 \end{bmatrix}$$

Application Steps

1. Choose a topic set
2. Choose a teleport set for each topic
   – Solve for PageRank vector
3. For a user/query, choose the topics that are most relevant
   – Hard task in-and-of itself
4. Weight results via combined PageRank vectors

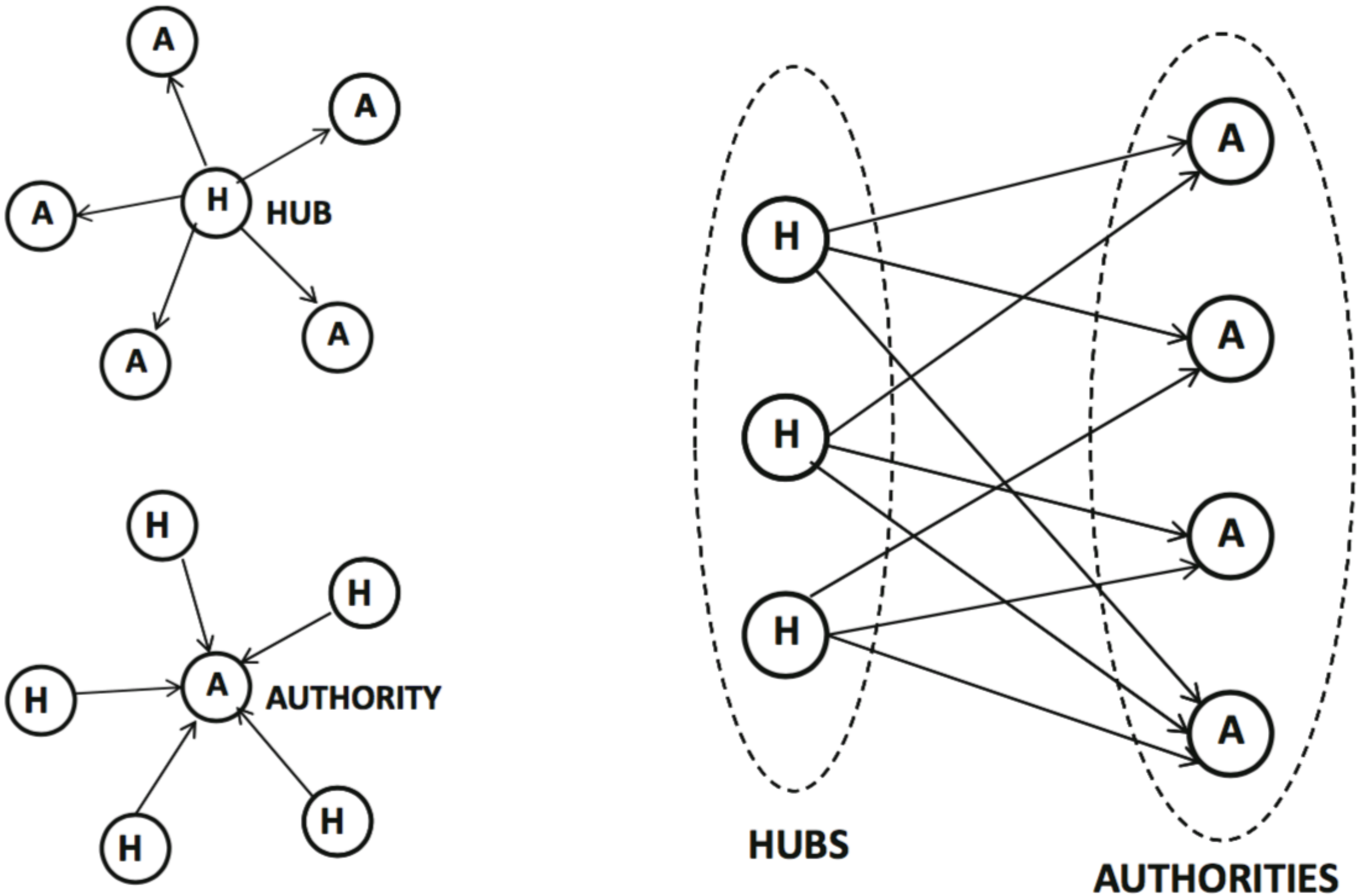
Structural Similarity Between Nodes

- One possibility: Topic-Sensitive PageRank with teleport set of 1 (node in question)
  - Provides an *asymmetric* ranking of nodes that are structurally close

- SimRank(i, j)
  - Basic idea: what would the expected distance be if two random surfers walked from nodes i/j

$$\mathrm{SimRank}(i,j) = \frac{C}{|\mathrm{In}(i)| \cdot |\mathrm{In}(j)|} \sum_{p \in \mathrm{In(i)}} \sum_{q \in \mathrm{In(i)}} \mathrm{SimRank}(p,q)$$

Hubs and Authorities



Hypertext Induced Topic Search

1. Collect top-r (e.g. 200) most relevant results to a query (e.g. via Google), R
2. Produce base set S as nodes that are in/out-links of nodes in R, edges (A) within S
   – Commonly need to limit size
3. $h^{0(i)} = a^0(i) = 1/\sqrt{|S|}$
4. Iterate

for each $i \in S$ set $a^{t+1}(i) \Leftarrow \sum_{j:(j,i)\in A} h^t(j)$;
for each $i \in S$ set $h^{t+1}(i) \Leftarrow \sum_{j:(i,j)\in A} a^{t+1}(j)$;
Normalize L_2-norm of each of hub and authority vectors to 1;

HITS

- Because it is query dependent, runs at query time, not indexing time
- Produces two scores per document
  – And only looks at a subset relevant to the query
- Not commonly used by search engines
  – Might be used by Ask
- Not affected by dead ends/spider traps!
  – So no need for damping