Chapter 9: Social Network Analysis

- Social Network Introduction
- Social Network Generation
- Mining on Social Network
- Summary

Society

Nodes: individuals

Links: social relationship (family/work/friendship/etc.)

S. Milgram (1967)

John Guare

Six Degrees of Separation

Social networks: Many *individuals* with *diverse social interactions* between them.

Communication networks

The earth is developing an electronic nervous system, a network with diverse **nodes** and **links** are

-computers

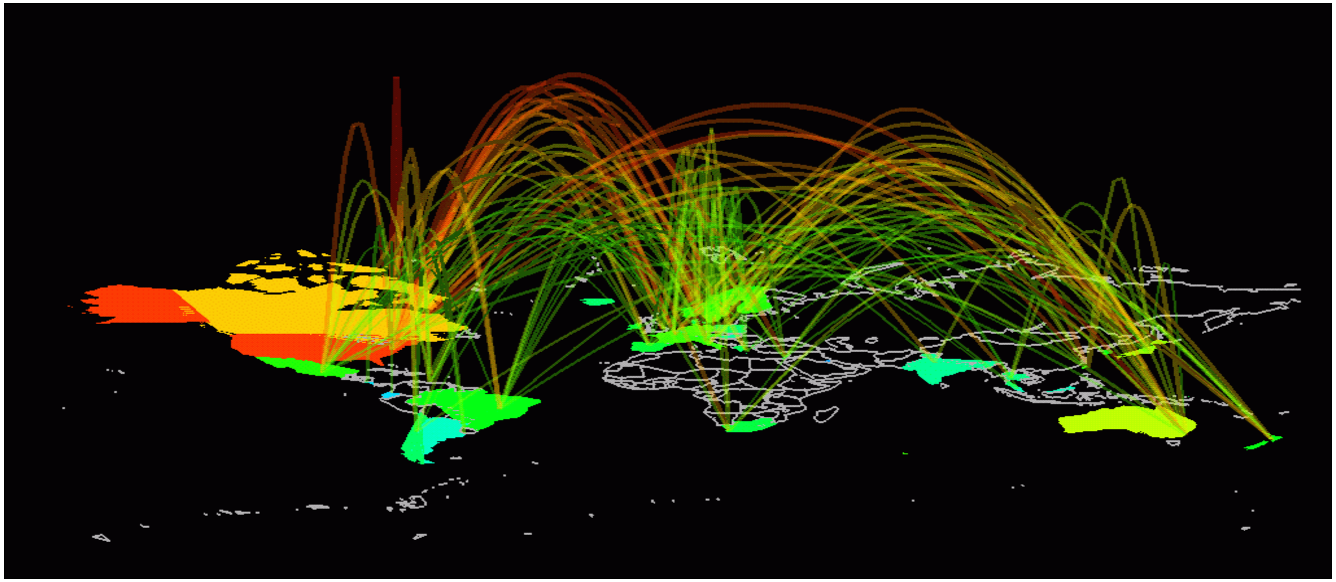
-routers

-satellites

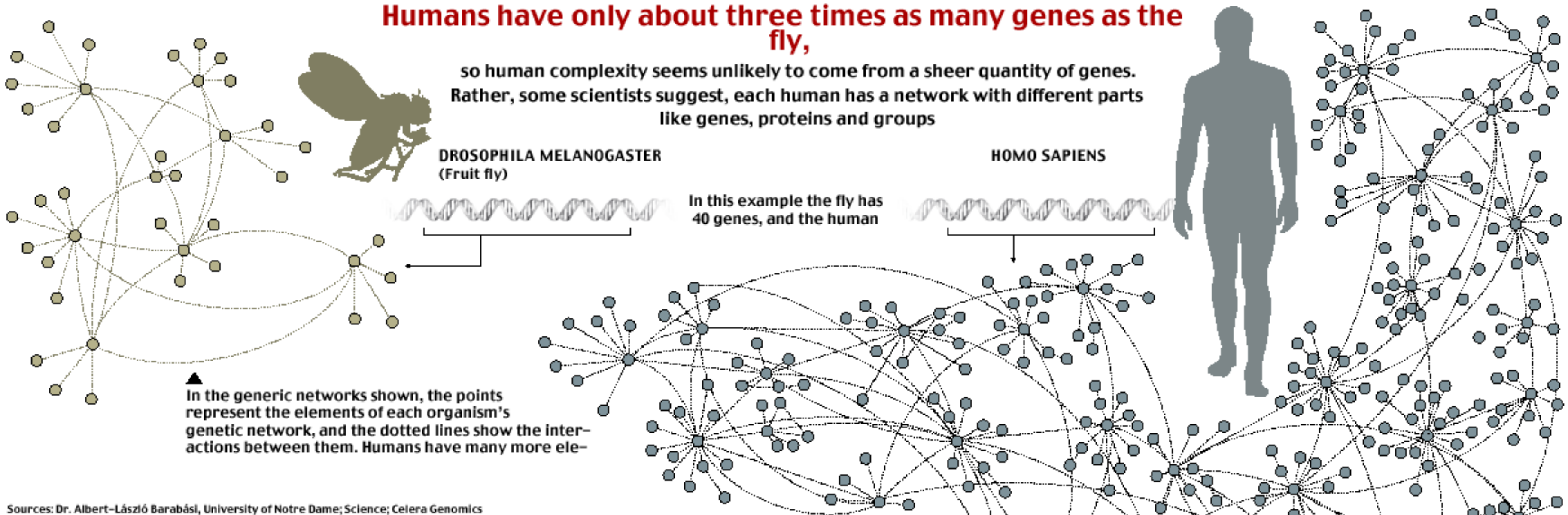
-phone lines

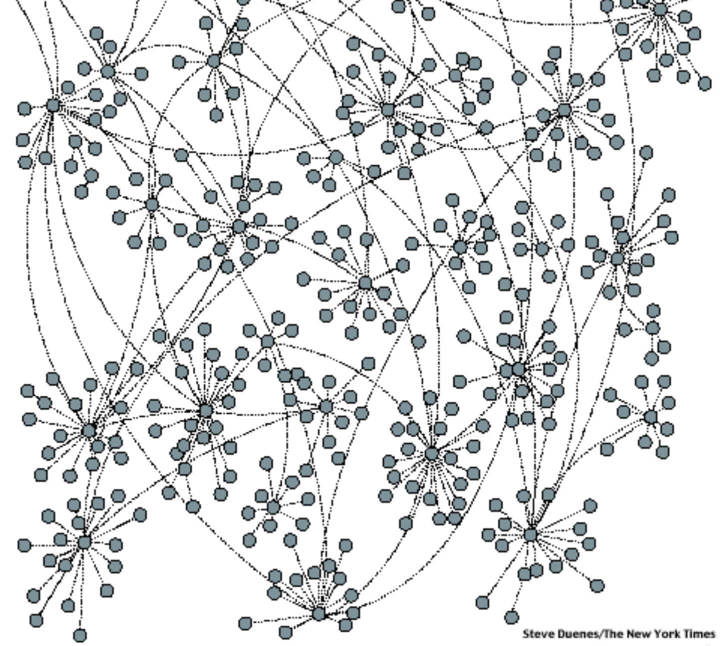
-TV cables

-communication lines



Communication networks: Many non-identical components with diverse connections between them





Complex systems

Made of
many non-identical **elements**
connected by diverse **interactions**.

NETWORK

Some Interesting Quantities

- *Connected components*:
  - how many, and how large?
- *Network diameter*:
  - maximum (worst-case) or average?
  - exclude infinite distances? (disconnected components)
  - the small-world phenomenon
- *Clustering*:
  - to what extent links tend to cluster “locally”?
  - what is the balance between local and long-distance connections?
  - what roles do the two types of links play?
- *Degree distribution*:
  - what is the typical degree in the network?
  - what is the overall distribution?

A "Canonical" Natural Network has…

- *A few* connected components:
  - often only 1 or a small number, *indep. of network size*
- *Small* diameter:
  - often a constant independent of network size (like 6)
  - or perhaps growing only logarithmically with a network size or even shrink?
    - typically exclude infinite distances
- A *high* degree of clustering:
  - considerably more so than for a random network
  - Related to small diameter
- A *heavy-tailed* degree distribution:
  - a small but reliable number of *high-degree vertices*
  - often of *power law* form

PART VII: Social Network Analysis

- Social Network Introduction
- Social Network Generation
- Mining on Social Network
- Summary

Models of Social Network Generation

- Random Graphs (Erdös-Rényi models)
- Scale-free Networks

Models of Social Network Generation

- Random Graphs (Erdös-Rényi models)
  - For all the nodes
  - We select randomly an (*missing*) edge each time

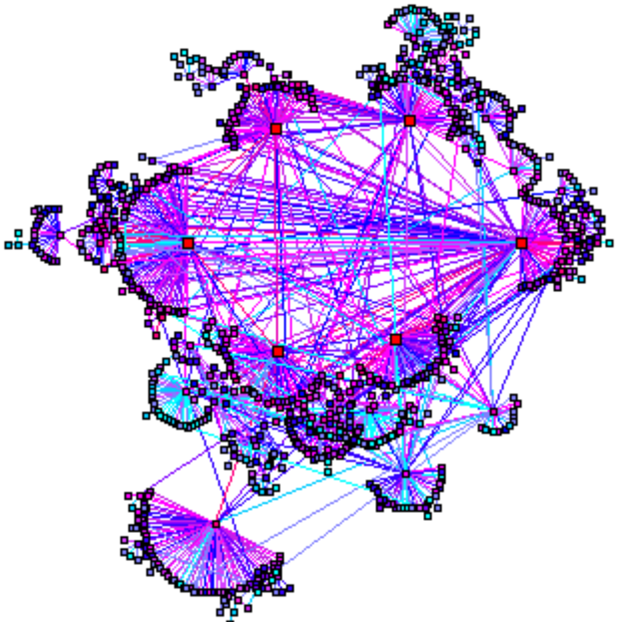
Models of Social Network Generation

- Random Graphs (Erdös-Rényi models)
- Scale-free Networks

World Wide Web

Nodes: WWW documents
Links: URL links

800 million documents
(S. Lawrence, 1999)

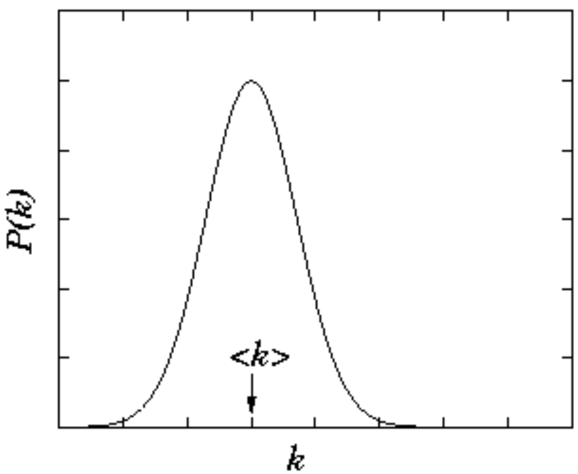


ROBOT: **collects all URL's found in a document and follows them recursively**

R. Albert, H. Jeong, A-L Barabasi, Nature, **401** 130 (1999)

World Wide Web

Expected Result



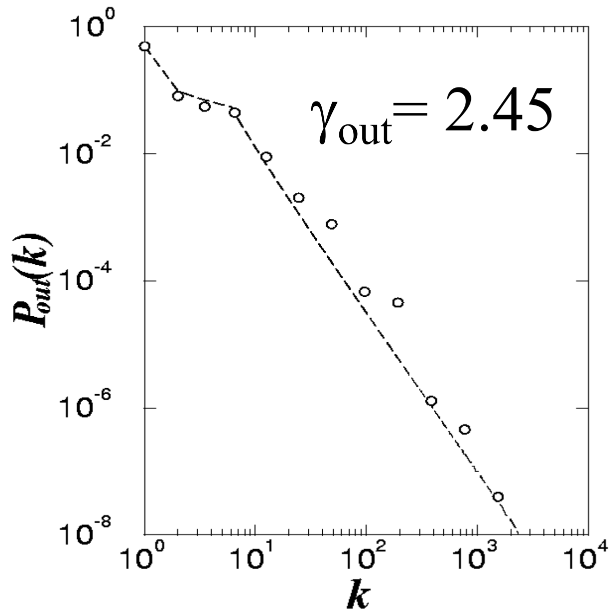
$\langle k \rangle \sim 6$

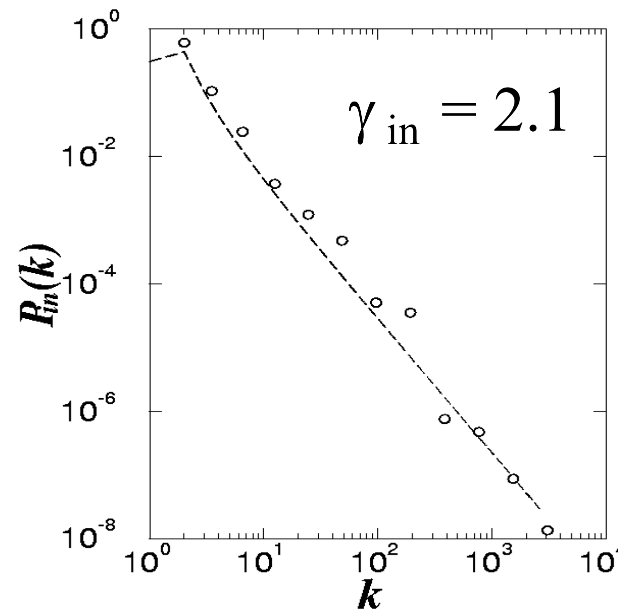
$P(k=500) \sim 10^{-99}$

$N_{WWW} \sim 10^9$

$\Rightarrow N(k=500) \sim 10^{-90}$

Real Result





$P_{out}(k) \sim k^{-\gamma_{out}}$ $P_{in}(k) \sim k^{-\gamma_{in}}$

$P(k=500) \sim 10^{-6}$ $N_{WWW} \sim 10^9$
$\Rightarrow N(k=500) \sim 10^3$

J. Kleinberg, et. al, Proceedings of the ICCC (1999)

Scale-free Networks

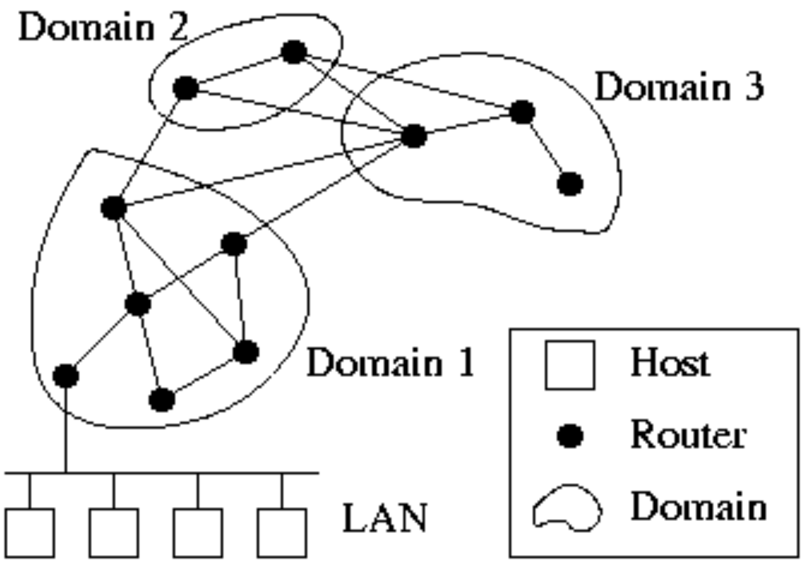
- The number of nodes (N) is not fixed
  - Networks continuously expand by additional new nodes
    - WWW: addition of new nodes
    - Citation: publication of new papers
- The attachment is not uniform (random)
  - A node is linked with higher probability to a node that already has a large number of links
    - WWW: new documents link to well known sites (CNN, Yahoo, Google)
    - Citation: Well cited papers are more likely to be cited again

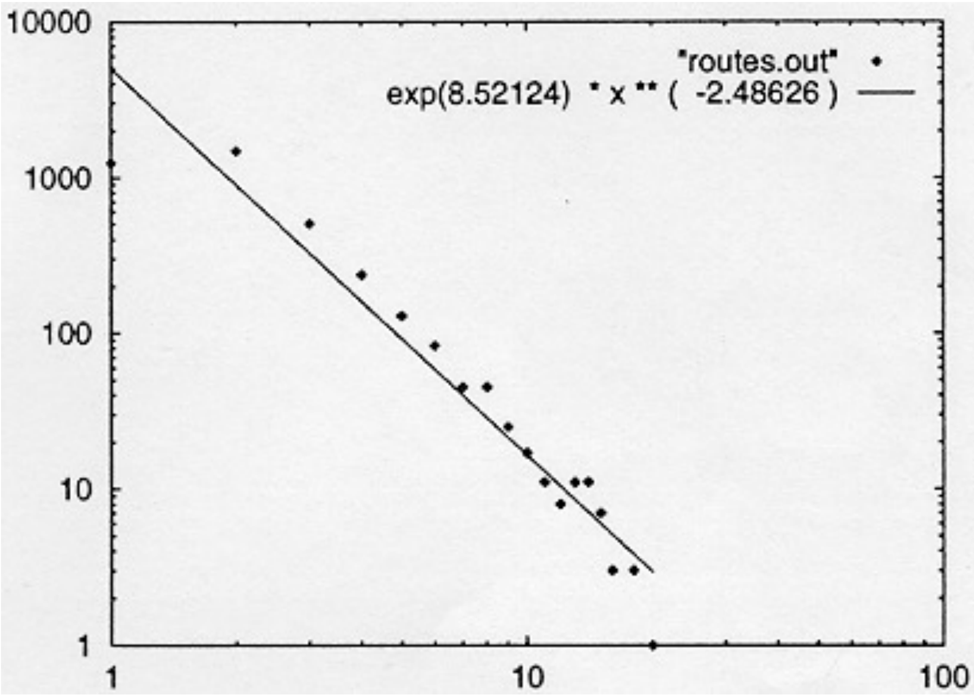
Scale-Free Networks

- Start with (say) two vertices connected by an edge
- For i = 3 to N:
  - For each 1 <= j < i, d(j) = degree of vertex j so far
  - Let Z = SUM d(j) (sum of all degrees so far)
  - Add new vertex i with *k* edges back to {1, …, i-1}:
    - i is connected back to j with probability d(j)/Z
- The rich get richer
  - Vertices j with high degree are likely to get more links!
- Natural model for many processes:
  - hyperlinks on the web
  - new business and social contacts
- Generates a power law distribution of degrees
  - exponent depends on value of k

Case1: Internet Backbone

Nodes: computers, routers
Links: physical lines





(Faloutsos, Faloutsos and Faloutsos, 1999)

Case2: Actor Connectivity

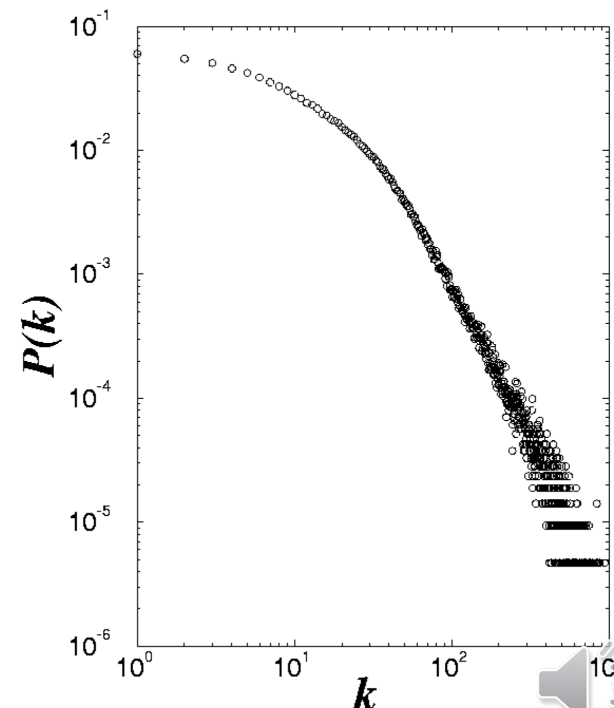
Days of Thunder (1990)
Far and Away (1992)
Eyes Wide Shut (1999)

N = 212,250 actors
$\langle k \rangle = 28.78$

$P(k) \sim k^{-\gamma}$

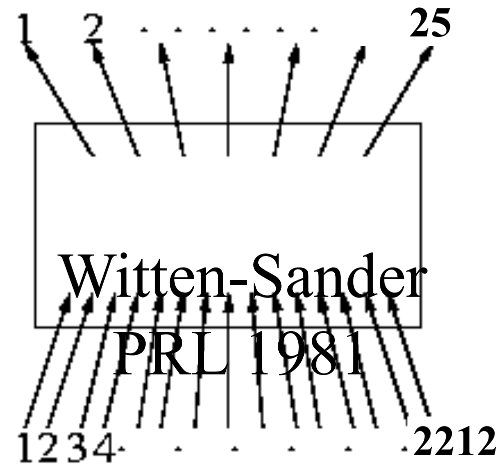
$\gamma = 2.3$

Nodes: actors
Links: cast jointly

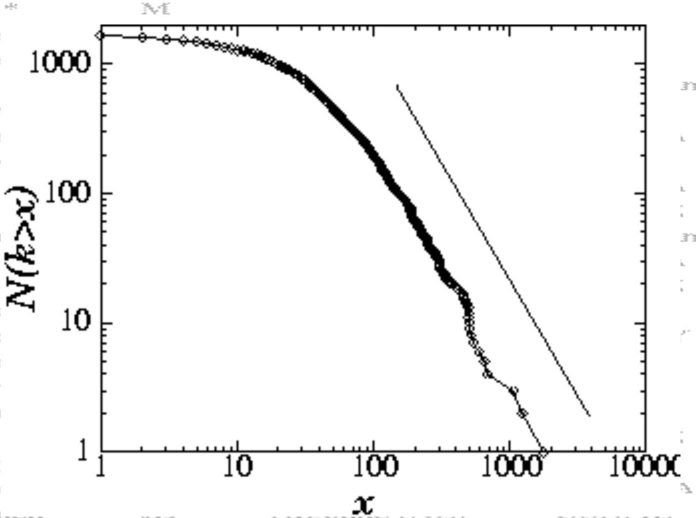


Case 3: Science Citation Index

Nodes: papers
Links: citations



1736 PRL papers (1988)

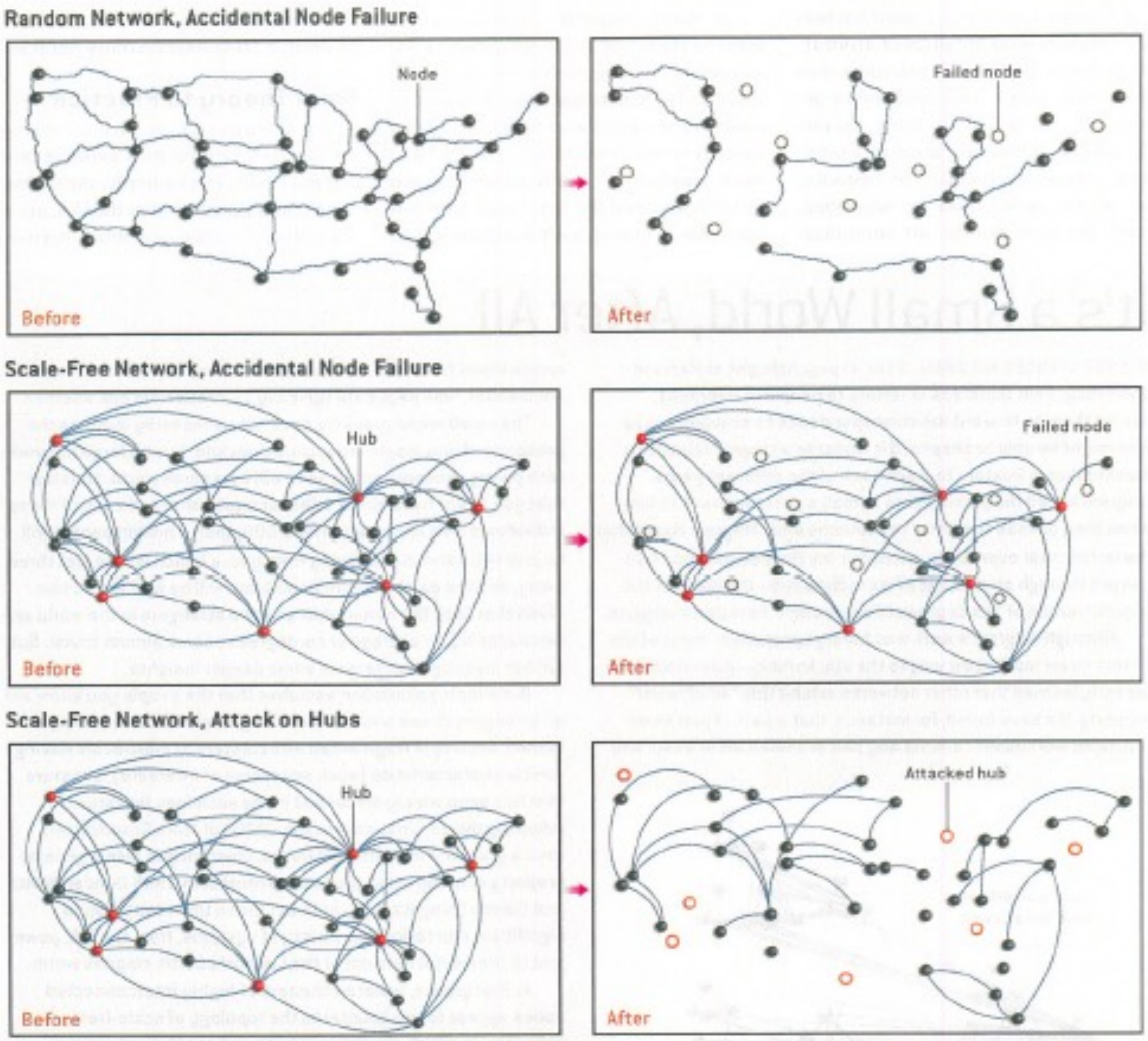


$P(k) \sim k^{-\gamma}$

$(\gamma = 3)$

(S. Redner, 1998)

Robustness of Random vs. Scale-Free Networks



- The accidental failure of a number of nodes in a random network can fracture the system into non-communicating islands.
- Scale-free networks are more robust in the face of such failures.
- Scale-free networks are highly vulnerable to a coordinated attack against their hubs.

PART VII: Social Network Analysis

- Social Network Introduction
- Social Network Generation
- Mining on Social Network
- Summary

Information on the Social Network

- Heterogeneous, multi-relational data represented as a graph or network
  - Nodes are objects
    - May have different kinds of objects
    - Objects have attributes
    - Objects may have labels or classes
  - Edges are links
    - May have different kinds of links
    - Links may have attributes
    - Links may be directed, are not required to be binary
- Links represent relationships and interactions between objects - rich content for mining

What is New for Link Mining Here

- Traditional machine learning and data mining approaches assume:
  - A random sample of homogeneous objects from single relation
- Real world data sets:
  - Multi-relational, heterogeneous, and semi-structured
- *Link mining / Social network analysis*
  - Newly emerging research area
  - At the intersection of research in social network and link analysis, hypertext and web mining, graph mining, and relational learning

A Taxonomy of Common Link Mining Tasks

- Object-Related Tasks
  - Link-based object ranking
  - Link-based object classification
  - Object clustering (group detection)
- Link-Related Tasks
  - Link prediction
- Graph-Related Tasks
  - Subgraph discovery
  - Graph classification
  - Generative model for graphs

What Is a Link in Link Mining?

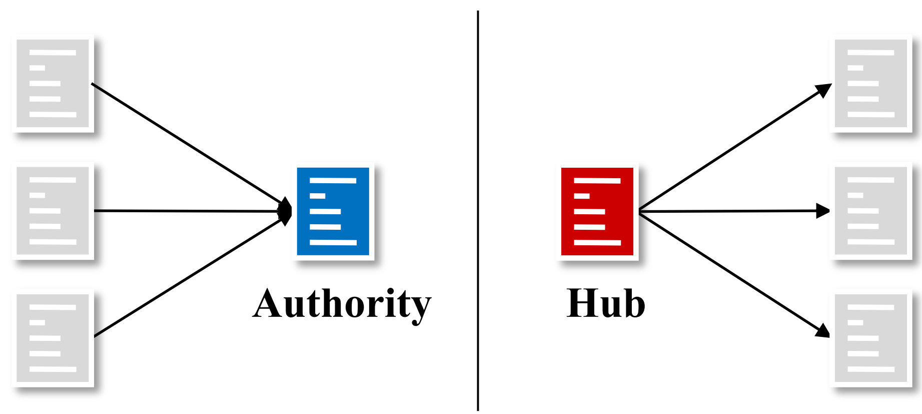
- Link: relationship among data
- Two kinds of linked networks
  - homogeneous vs. heterogeneous
- Homogeneous networks
  - Single object type and single link type
  - Single model social networks (e.g., friends)
  - WWW: a collection of linked Web pages
- Heterogeneous networks
  - Multiple object types and link types
  - Medical network: patients, doctors, disease, contacts, treatments
  - Bibliographic network: publications, authors, venues

Link-Based Object Ranking (LBR)

- LBR: Exploit the link structure of a graph to order or prioritize a set of objects within the graph
  - Focused on graphs with single object type and single link type
- This was a primary focus of link analysis community
- Web information analysis
  - *HITS* and *PageRank* are typical LBR approaches

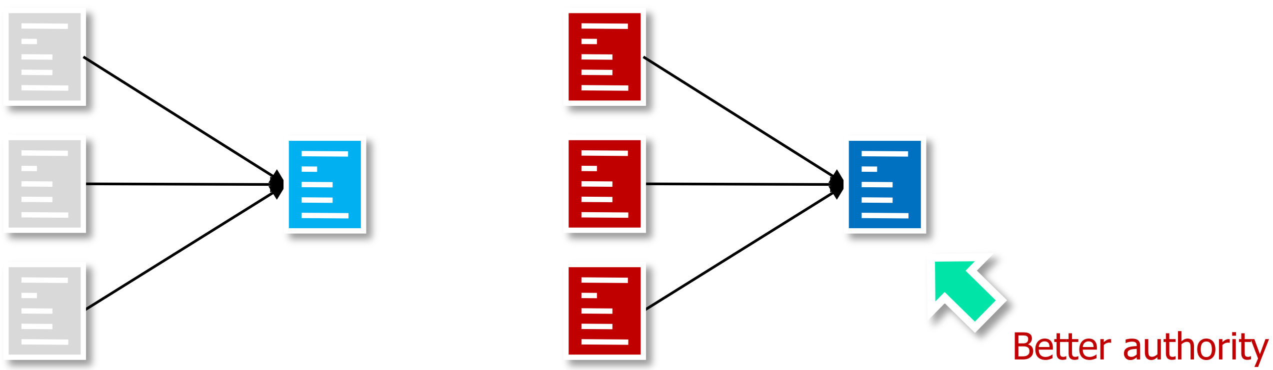
HITS: Capturing Authorities & Hubs (Kleinberg'98)

- Intuitions
  - Links are like citations in literature
  - Pages that are widely cited are good *authorities*
  - Pages that cite many other pages are good *hubs*



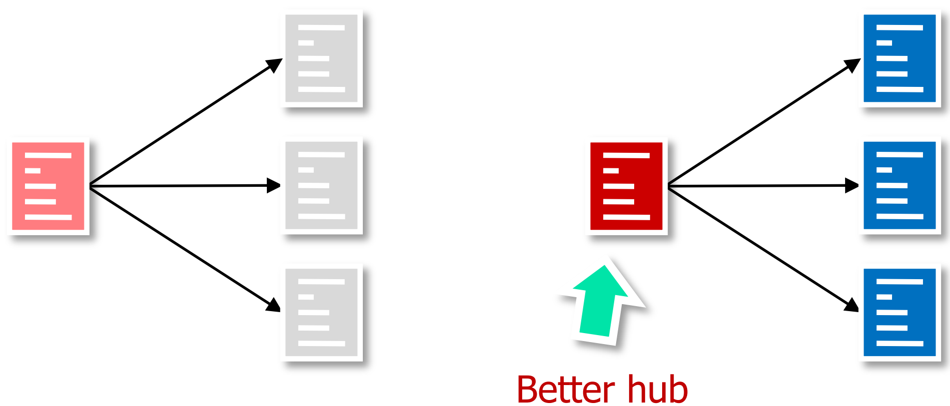
HITS: Capturing Authorities & Hubs (Kleinberg'98)

- The key idea of HITS
  - Good authorities are cited by good hubs
  - Good hubs point to good authorities
  - *Iterative mutual reinforcement* …



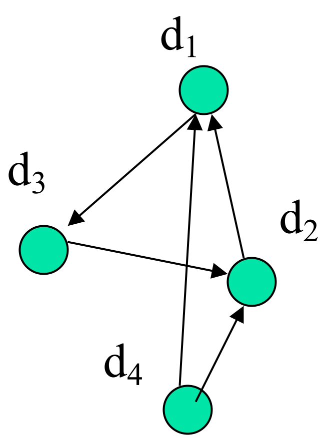
HITS: Capturing Authorities & Hubs (Kleinberg'98)

- The key idea of HITS
  - Good authorities are cited by good hubs
  - Good hubs point to good authorities
  - *Iterative mutual reinforcement* …



The HITS Algorithm (Kleinberg 98)

- Each page (d_i) has two scores:
  - Hub score ($h(d_i)$) and authority score ($a(d_i)$)
  - *Hub score* is the sum of *authority scores* from its out-link neighbors
  - *Authority score* is the sum of *hub scores* from its in-link neighbors



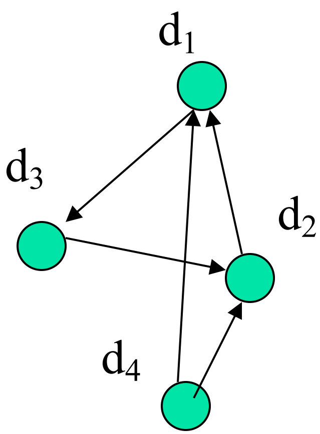
$$h(d_i) = \sum_{d_j \in OUT(d_i)} a(d_j)$$ **"Hub score"**

$$a(d_i) = \sum_{d_j \in IN(d_i)} h(d_j)$$ **"Authority score"**

The HITS Algorithm (Kleinberg 98)

$$A = \begin{array}{cc} & \begin{array}{cccc} d_1 & d_2 & d_3 & d_4 \end{array} \\ & \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \begin{array}{c} d_1 \\ d_2 \\ d_3 \\ d_4 \end{array} \end{array}$$

"Adjacency matrix"



$$h(d_i) = \sum_{d_j \in OUT(d_i)} a(d_j)$$

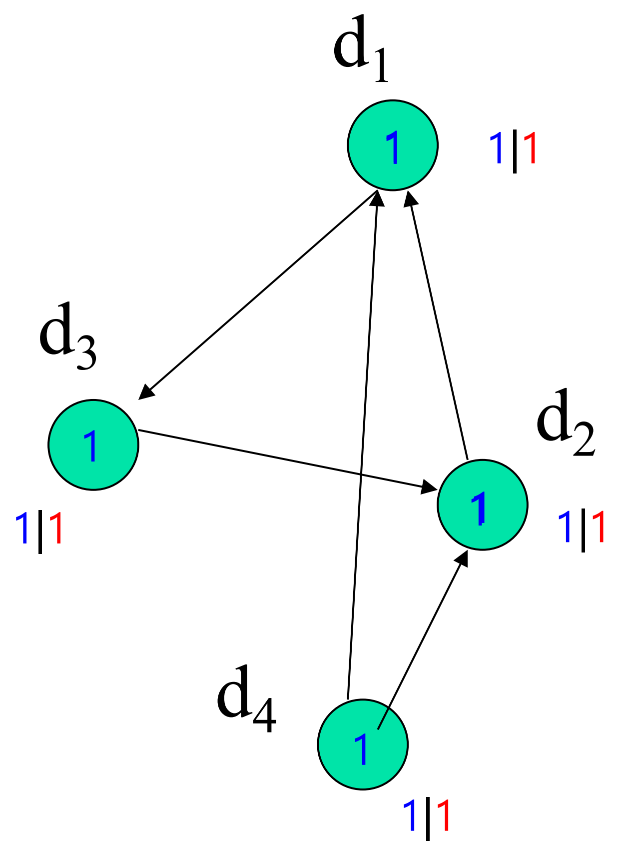
$$a(d_i) = \sum_{d_j \in IN(d_i)} h(d_j)$$

Initial values:
$a = h = 1$

Iterate until converge

Normalize: $\sum_i a(d_i)^2 = \sum_i h(d_i)^2 = 1$

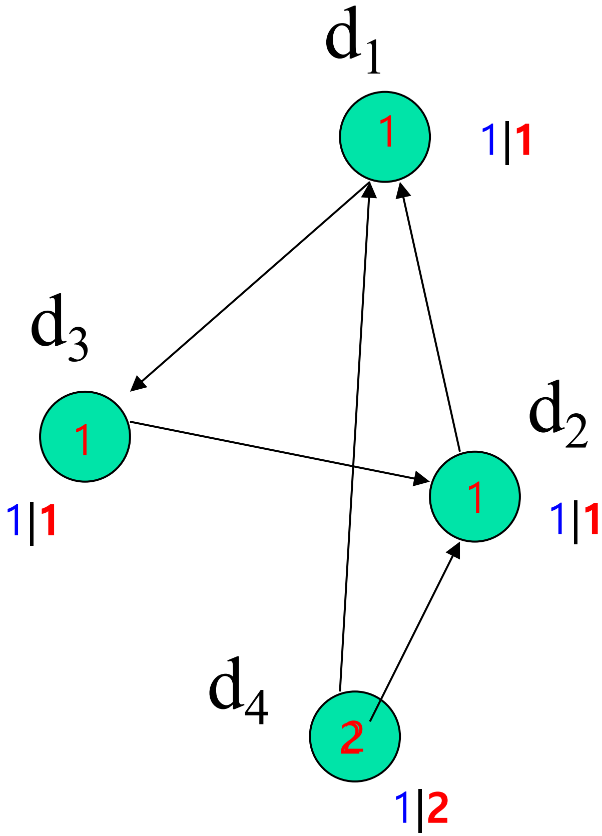
The HITS Algorithm (Kleinberg 98)



$$h(d_i) = \sum_{d_j \in OUT(d_i)} a(d_j)$$

$$a(d_i) = \sum_{d_j \in IN(d_i)} h(d_j)$$

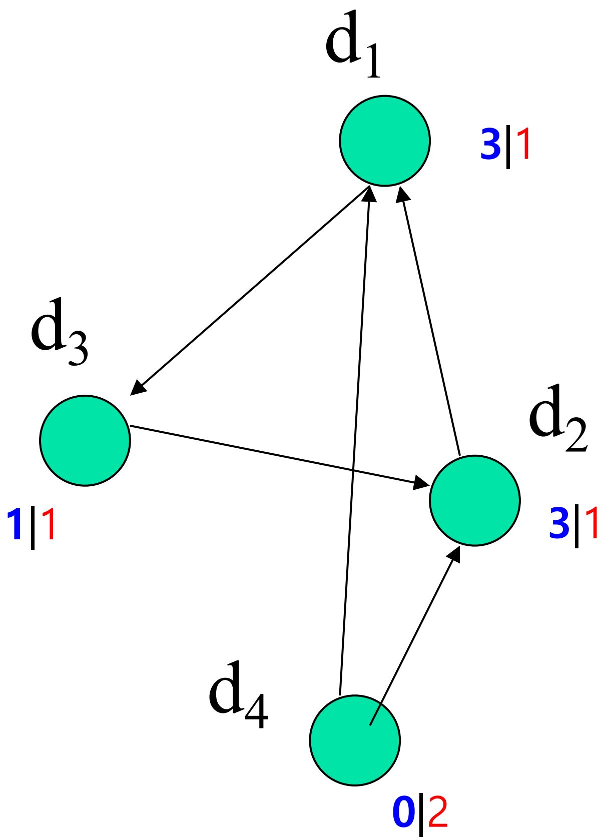
The HITS Algorithm (Kleinberg 98)



$$h(d_i) = \sum_{d_j \in OUT(d_i)} a(d_j)$$

$$a(d_i) = \sum_{d_j \in IN(d_i)} h(d_j)$$

The HITS Algorithm (Kleinberg 98)



$$h(d_i) = \sum_{d_j \in OUT(d_i)} a(d_j)$$

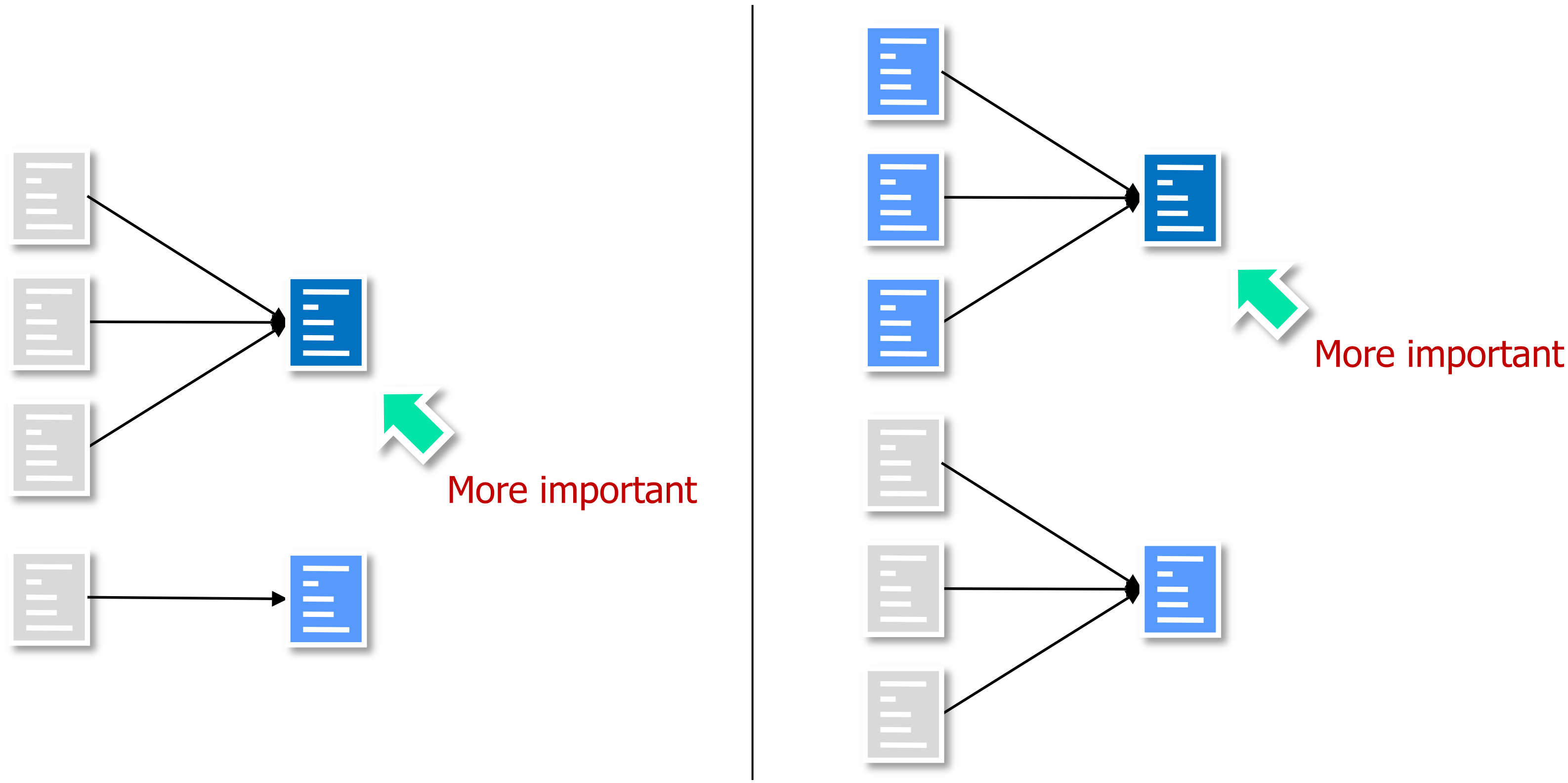
$$a(d_i) = \sum_{d_j \in IN(d_i)} h(d_j)$$

PageRank: Capturing Page Popularity (Brin & Page'98)

- Intuitions
  - A page that is cited often can be expected to be more *important* (or authoritative) in general
- PageRank is essentially "citation counting", but improves over simple counting
  - Consider "indirect citations" (being cited by a highly cited paper counts a lot…)
  - Smoothing of citations (every page is assumed to have a non-zero citation count)
- PageRank can also be interpreted as *random surfing* (thus capturing popularity)

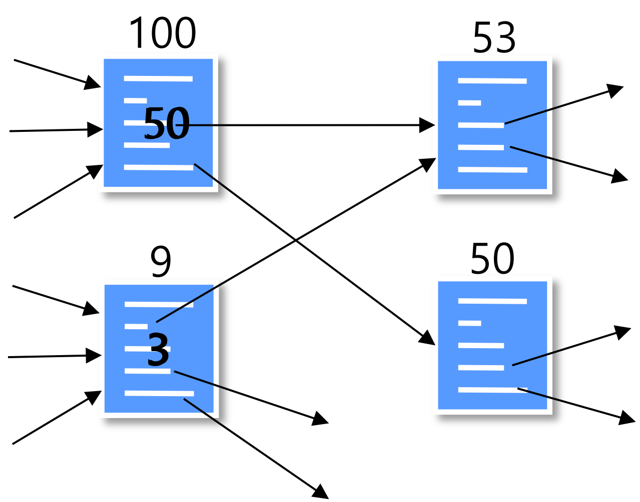
PageRank: Indirect Citations

- A page that is *important* is…
  - Cited often by other pages
  - Cited often by other *important* pages



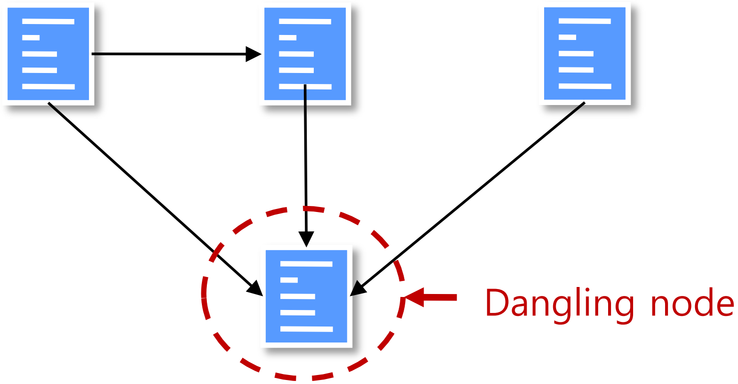
PageRank: Simple Version

- Calculate importance score (authority score)
  - Initially, assign the same score to every page (e.g. 1)
  - For each page:
    - Transfer its score (divided equally) to its neighbors through out-links
    - Sum up the scores transferred from its neighbors through in-links
  - Iterate until…

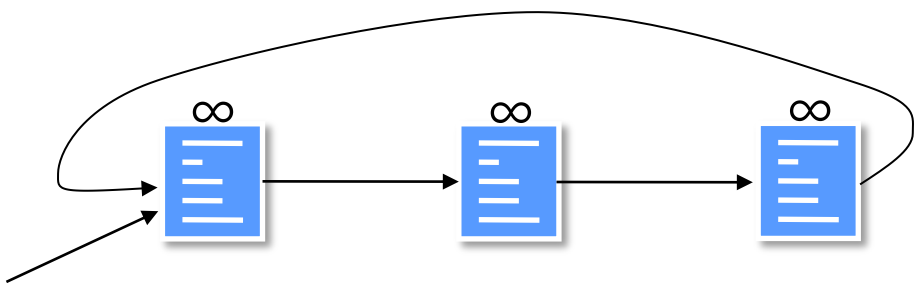


PageRank: Simple Version

- Problems of the simple version
  - Dangling nodes

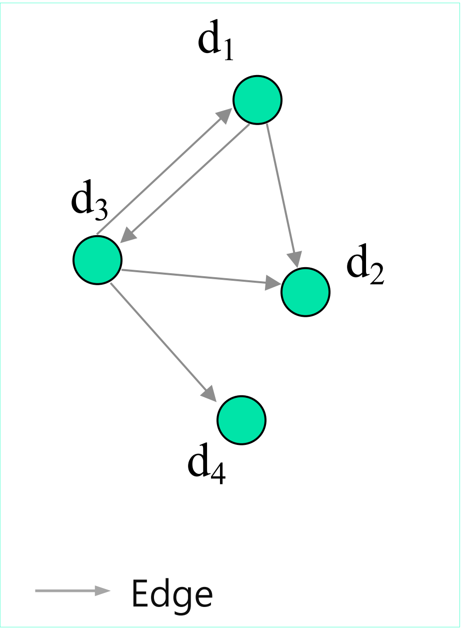


  - Cyclic citation



PageRank: Random Surfer Model

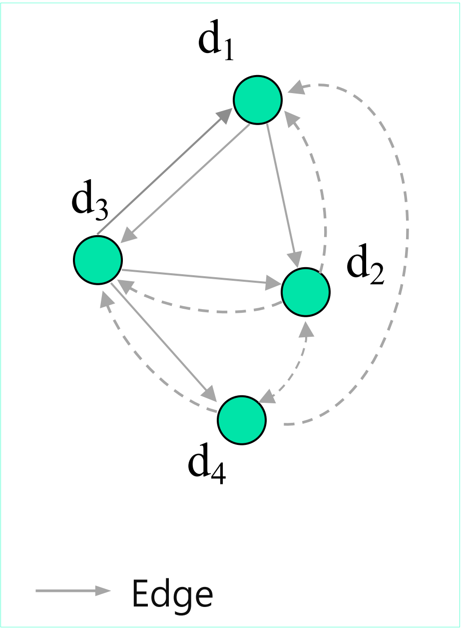
- Random Surfer
  - Surfing the web by clicking on hyperlinks randomly
  - Or jump to a random page and *restart* surfing
- At any page,
  - With prob. α, randomly picking a link to follow (random walk)
  - With prob. $(1 - \alpha)$, randomly jumping to a page (restart)



Movement of a virtual web surfer

PageRank: Random Surfer Model

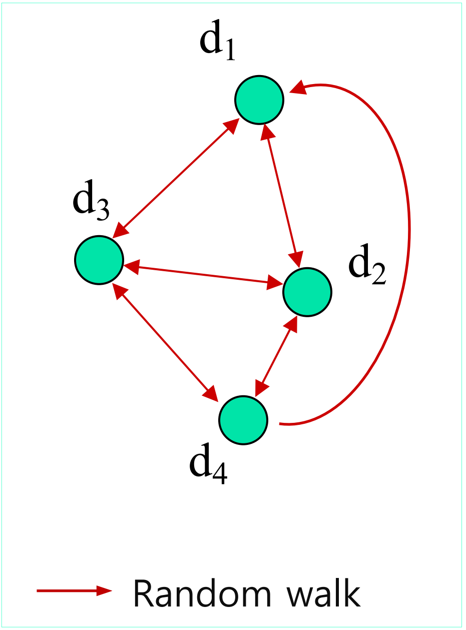
- Random Surfer
  - Surfing the web by clicking on hyperlinks randomly
  - Or jump to a random page and *restart* surfing
- At any page,
  - With prob. α, randomly picking a link to follow (random walk)
  - With prob. $(1 - \alpha)$, randomly jumping to a page (restart)



Movement of
a virtual web surfer

PageRank: Random Surfer Model

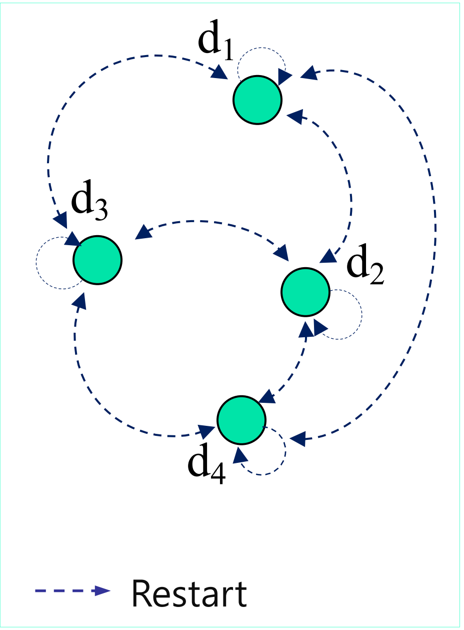
- Random Surfer
  - Surfing the web by clicking on hyperlinks randomly
  - Or jump to a random page and *restart* surfing
- At any page,
  - With prob. α, randomly picking a link to follow (random walk)
  - With prob. $(1 - \alpha)$, randomly jumping to a page (restart)



Movement of
a virtual web surfer

PageRank: Random Surfer Model

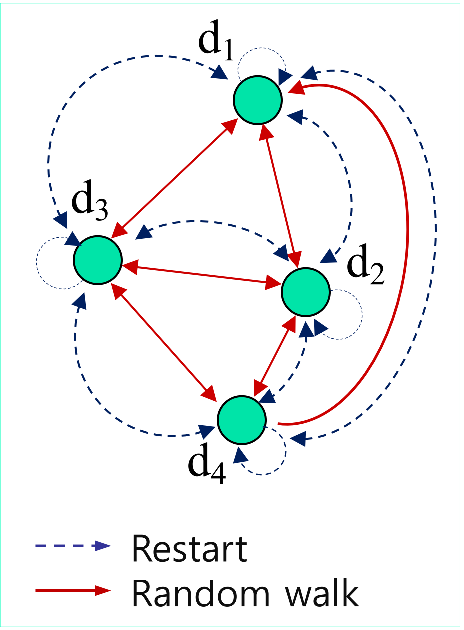
- Random Surfer
  - Surfing the web by clicking on hyperlinks randomly
  - Or jump to a random page and *restart* surfing
- At any page,
  - With prob. α, randomly picking a link to follow (random walk)
  - With prob. $(1 - \alpha)$, randomly jumping to a page (restart)



Movement of
a virtual web surfer

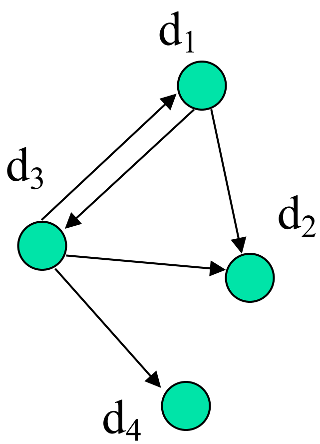
PageRank: Random Surfer Model

- Random Surfer
  - Surfing the web by clicking on hyperlinks randomly
  - Or jump to a random page and *restart* surfing
- At any page,
  - With prob. α, randomly picking a link to follow (random walk)
  - With prob. $(1 - \alpha)$, randomly jumping to a page (restart)



Movement of
a virtual web surfer

The PageRank Algorithm (Brin & Page'98)



$$M = \begin{array}{c} \begin{array}{cccc} d_1 & d_2 & d_3 & d_4 \end{array} \\ \begin{bmatrix} 0 & 0 & \frac{1}{3} & 0 \\ \frac{1}{2} & 0 & \frac{1}{3} & 0 \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 \end{bmatrix} \end{array} \qquad d = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \qquad w = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \begin{array}{c} d_1 \\ d_2 \\ d_3 \\ d_4 \end{array}$$

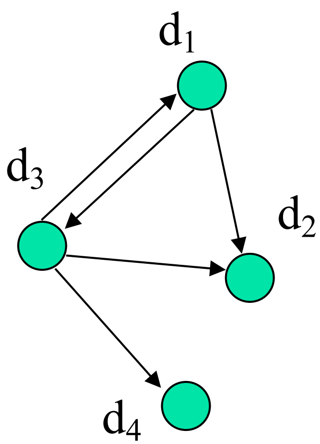
"Transition matrix"

Initial value *r*(*d*)=1/*N*

$$r_{i+1} = \underbrace{(1-\alpha)(M + w \times d^T) \times r_i}_{\text{"Random Walk"}} + \underbrace{\alpha w}_{\text{"Restart"}}$$

Iterate until converge

The PageRank Algorithm (Brin & Page'98)



$$M + w \times d^T = \begin{array}{c} \begin{array}{cccc} d_1 & d_2 & d_3 & d_4 \end{array} \\ \begin{bmatrix} 0 & \frac{1}{4} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{4} & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & \frac{1}{3} & \frac{1}{4} \end{bmatrix} \end{array}$$

No dangling node

Initial value *r*(*d*)=1/*N*

$$r_{i+1} = (1-\alpha)(M + w \times d^T) \times r_i + \alpha w$$

Iterate until converge

Solves dangling nodes problem

Solves cyclic citation problem

Link-Based Object Classification (LBC)

- Predicting the category of an object based on its attributes, *its links, and the attributes of linked objects*
  - **Web**: Predict the category of a web page, based on words that occur on the page, links between pages, anchor text, html tags, etc.
  - **Citation**: Predict the topic of a paper, based on word occurrence, citations, co-citations
  - **Epidemics**: Predict disease type based on characteristics of the patients infected by the disease

Group Detection

- Cluster the nodes in the graph into groups that share common characteristics
  - **Web:** identifying communities
  - **Citation:** identifying research communities
- Methods
  - Hierarchical clustering
  - Blockmodeling of SNA
  - Spectral graph partitioning
  - Stochastic blockmodeling
  - Multi-relational clustering

Link Cardinality Estimation

- Predicting the number of links to an object
  - **Web**: predict the authority of a page based on the number of in-links; identifying hubs based on the number of out-links
  - **Citation**: predicting the impact of a paper based on the number of citations
  - **Epidemics**: predicting the number of people that will be infected based on the infectiousness of a disease
- Predicting the number of objects reached along a path from a *specific object*
  - **Web**: predicting number of pages retrieved by crawling a site
  - **Citation**: predicting the number of citations of a particular author in a specific journal

Link Prediction

- Predict whether a link exists between two entities, based on attributes and other observed links
- Applications
  - **Web**: predict if there will be a link between two pages
  - **Citation**: predicting if a paper will cite another paper
  - **Epidemics**: predicting who a patient's contacts are
- Methods
  - Often viewed as a binary classification problem
  - Local conditional probability model, based on structural and attribute features
  - Difficulty: sparseness of existing links
  - Collective prediction, e.g., Markov random field model

Subgraph Discovery

- Find characteristic subgraphs
  - Focus of graph-based data mining
- Applications
  - **Biology:** protein structure discovery
  - **Communications:** legitimate vs. illegitimate groups
  - **Chemistry:** chemical substructure discovery
- Methods
  - Subgraph pattern mining
- Graph classification
  - Classification *based on subgraph pattern analysis*

Summary: Link Mining in Social Networks

- Object-Related Tasks
  - Link-based object ranking
  - Link-based object classification
  - Object clustering (group detection)
  - Object identification (entity resolution)
- Link-Related Tasks
  - Link prediction
- Graph-Related Tasks
  - Subgraph discovery
  - Graph classification
  - Generative model for graphs