Web Information Processing and Applications: Web Mining

Roadmap

Web Mining

- Content Mining
  - Classification, Clustering
- Structure Mining
  - Social network analysis
- Usage Mining
  - Recommendation …

What do the following things have in common?

- World economy
- Human cell
- Railroads
- Brain
- Internet
- Friends and Family
- Media & Information
- Society

The Network!

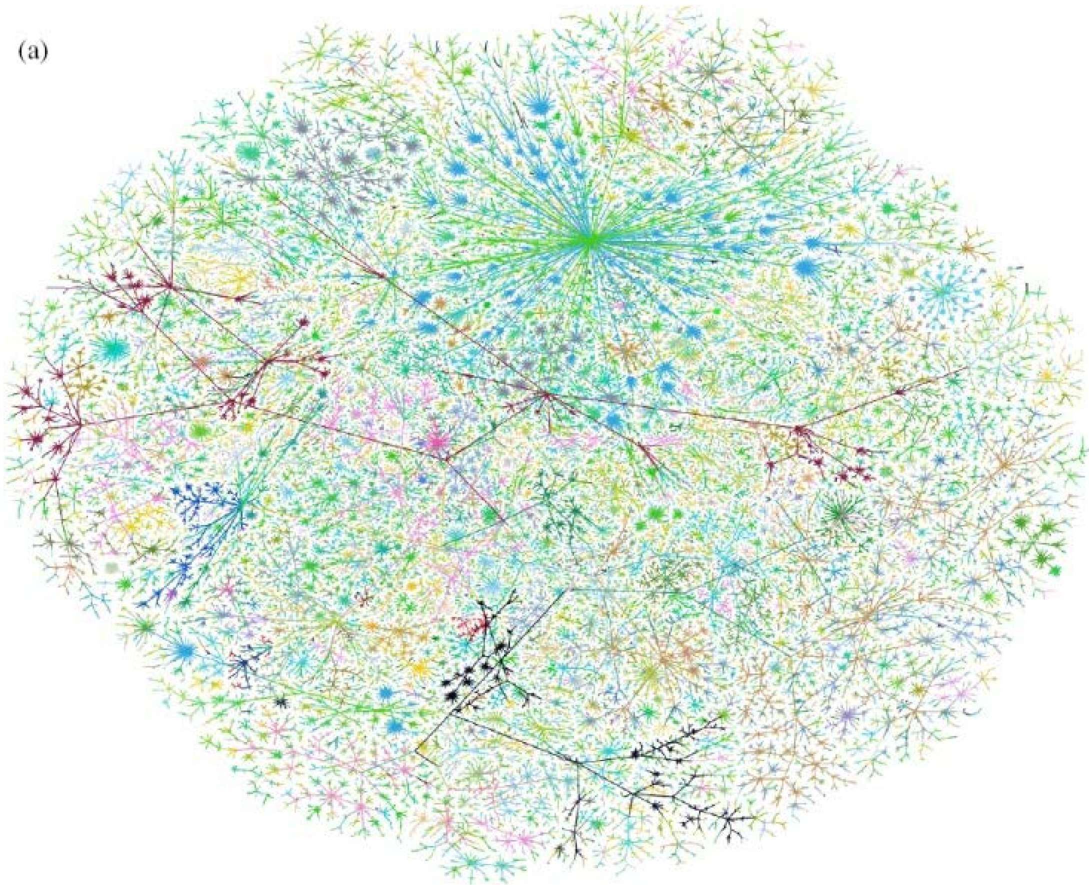
Behind each such system there is an intricate wiring diagram, **a network**, that defines the **interactions** between the components

Networks: Social

Facebook social graph

4-degrees of separation

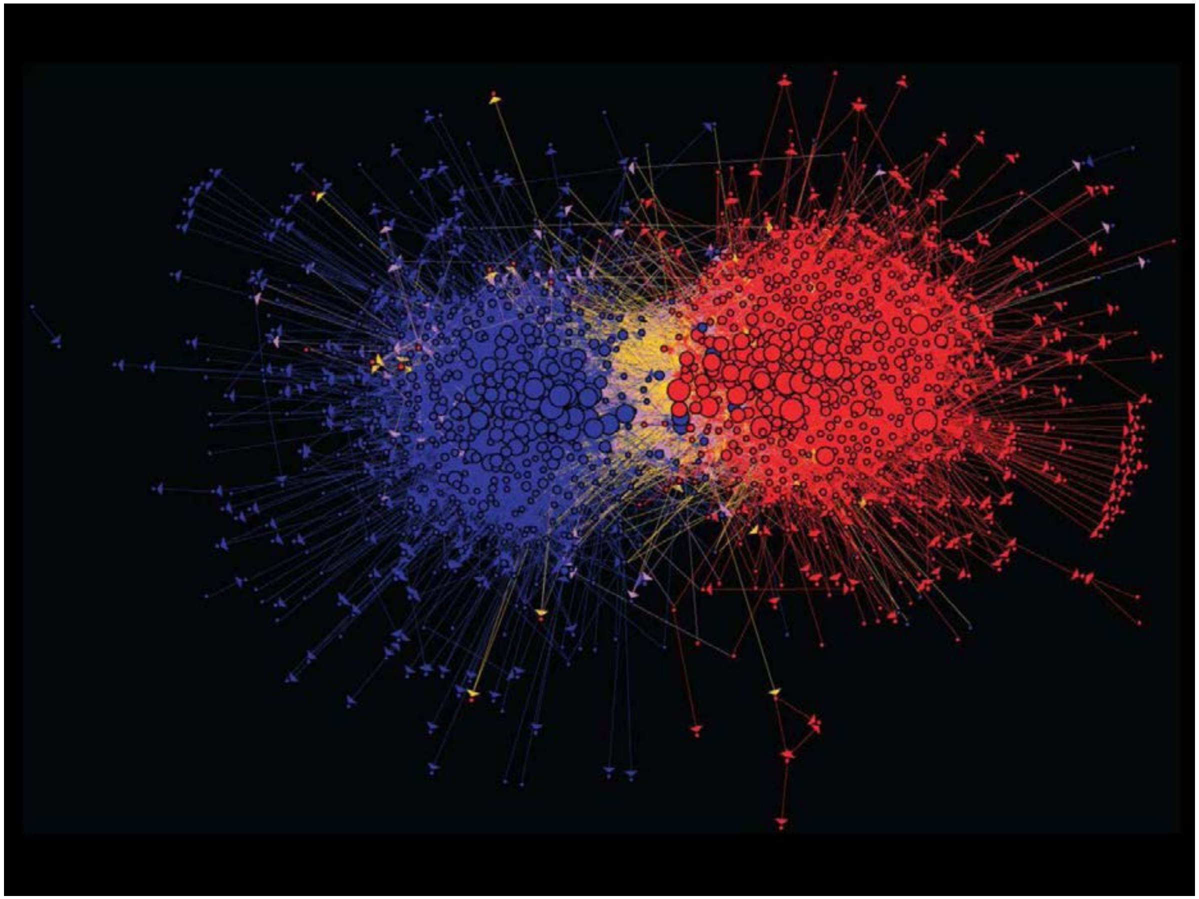
Networks: Communication



Graph of the Internet (Autonomous Systems)

Power-law degrees

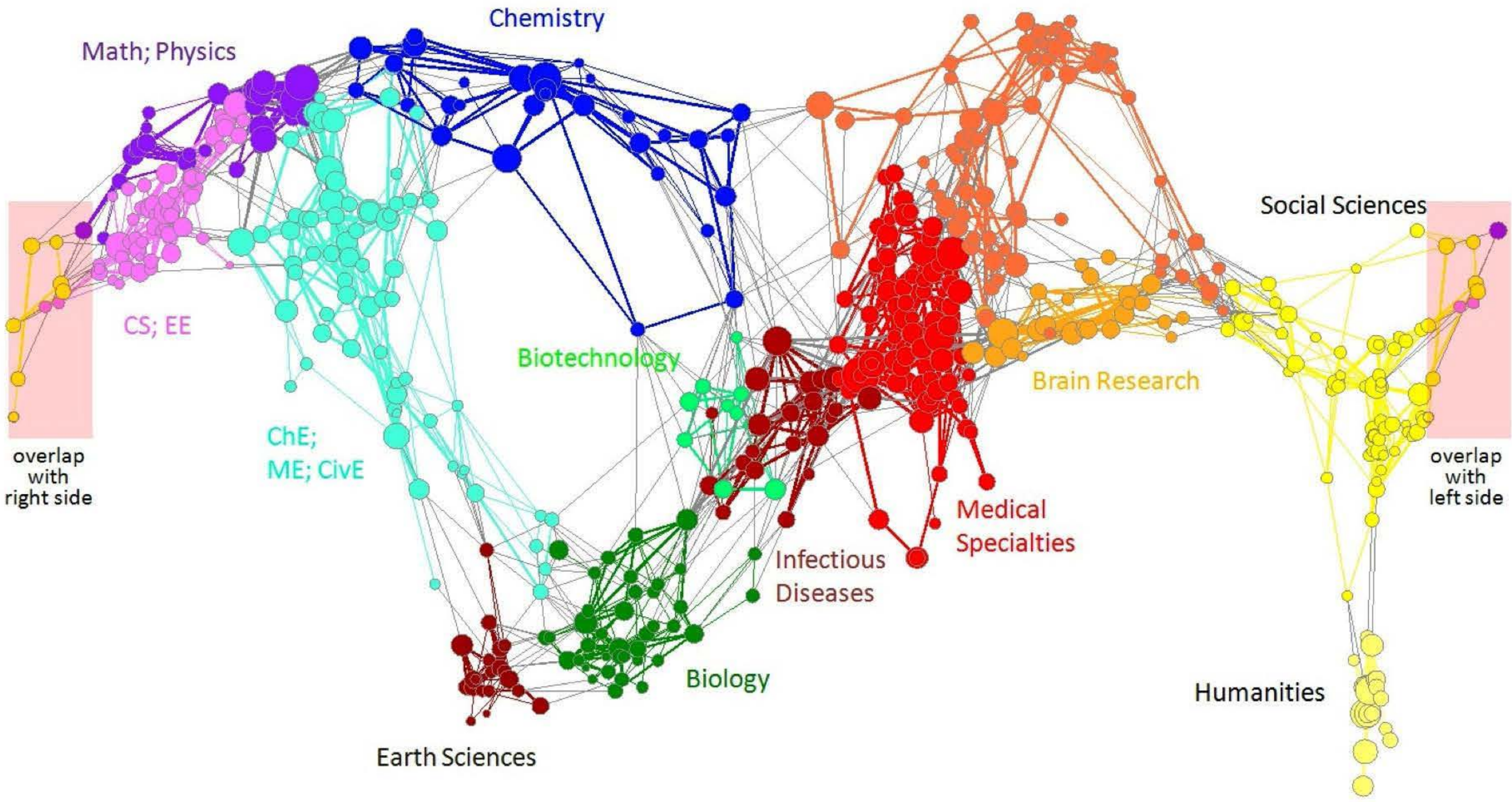
Networks: Media



Connections between political blogs

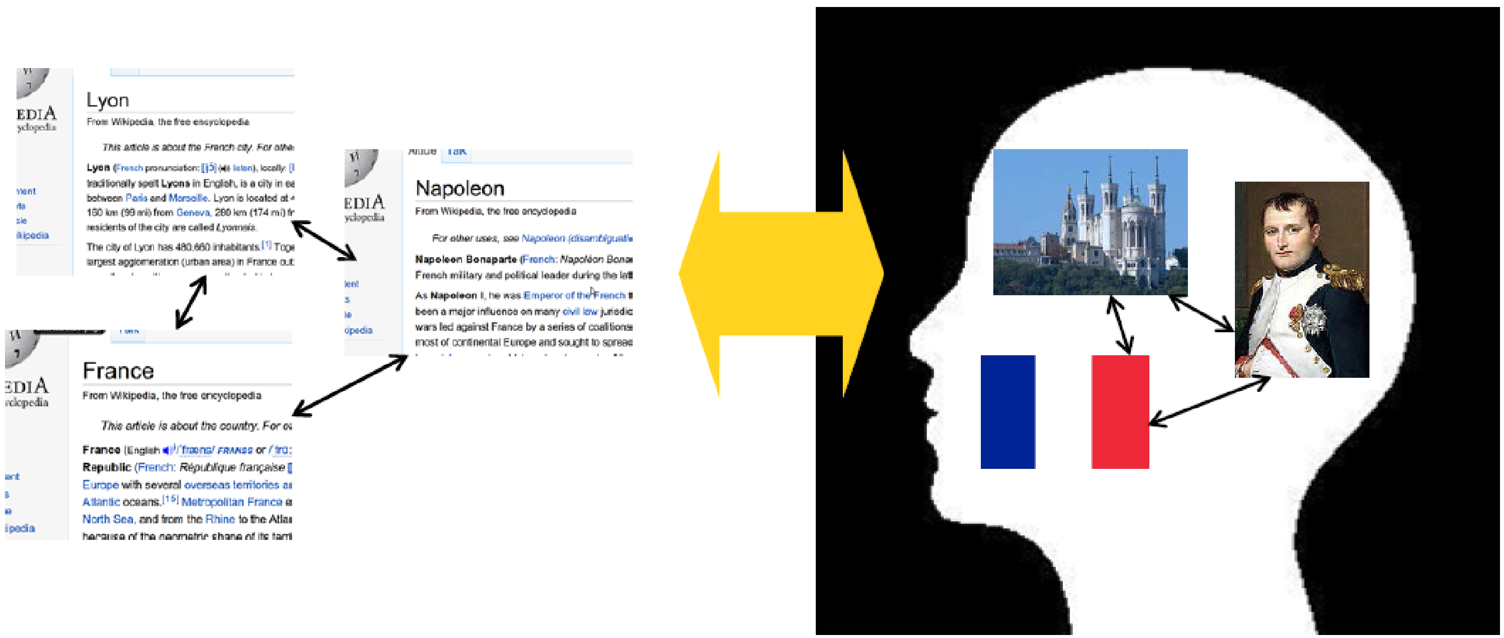
Polarization of the network

Networks: Information



Citation networks and Maps of science

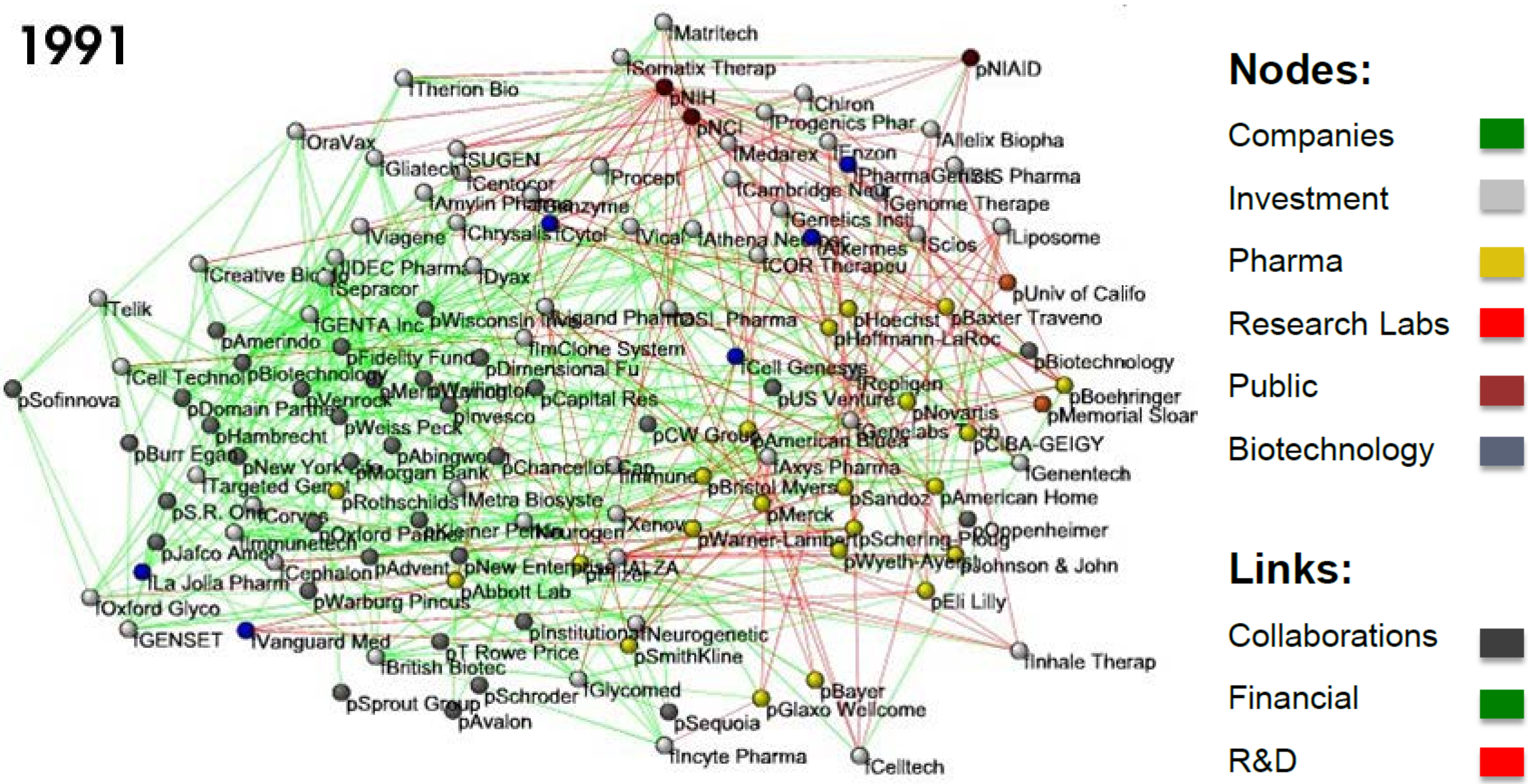
Networks: Knowledge



Understand how humans navigate Wikipedia

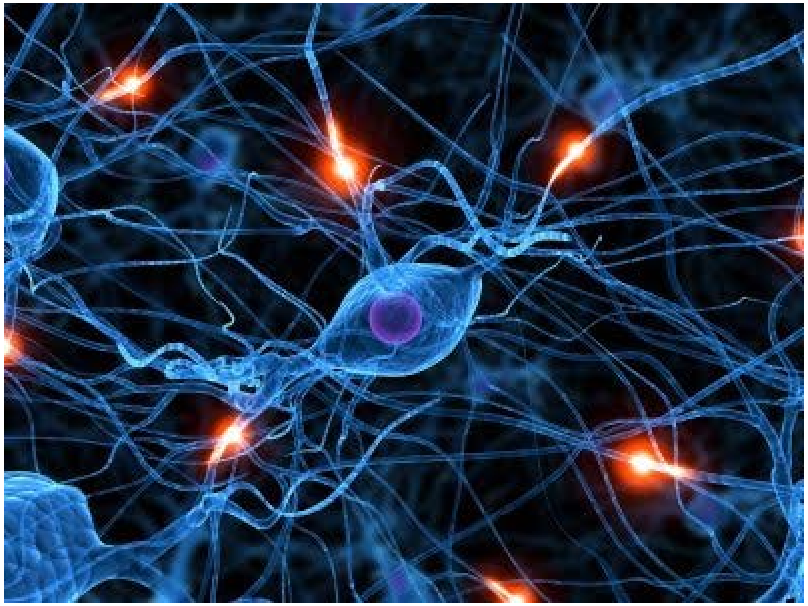
Get an idea of how people connect concepts

Networks: Economy



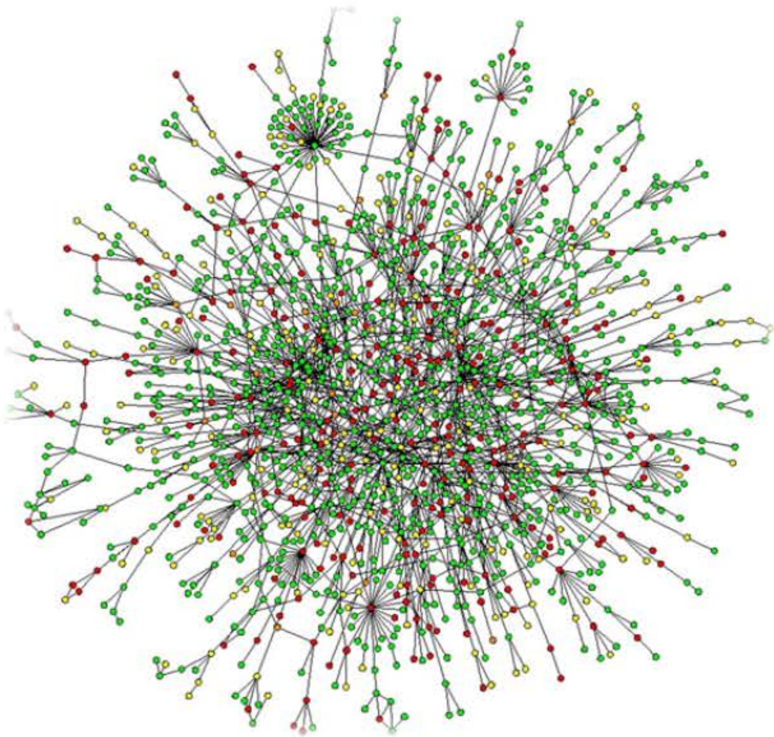
Bio-tech companies

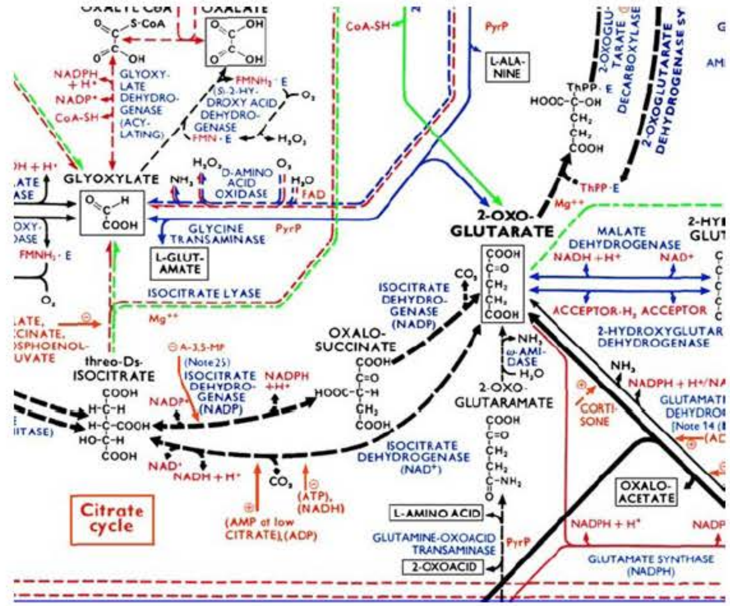
Networks: Brain



Human brain has between 10-100 billion neurons

Networks: Biology





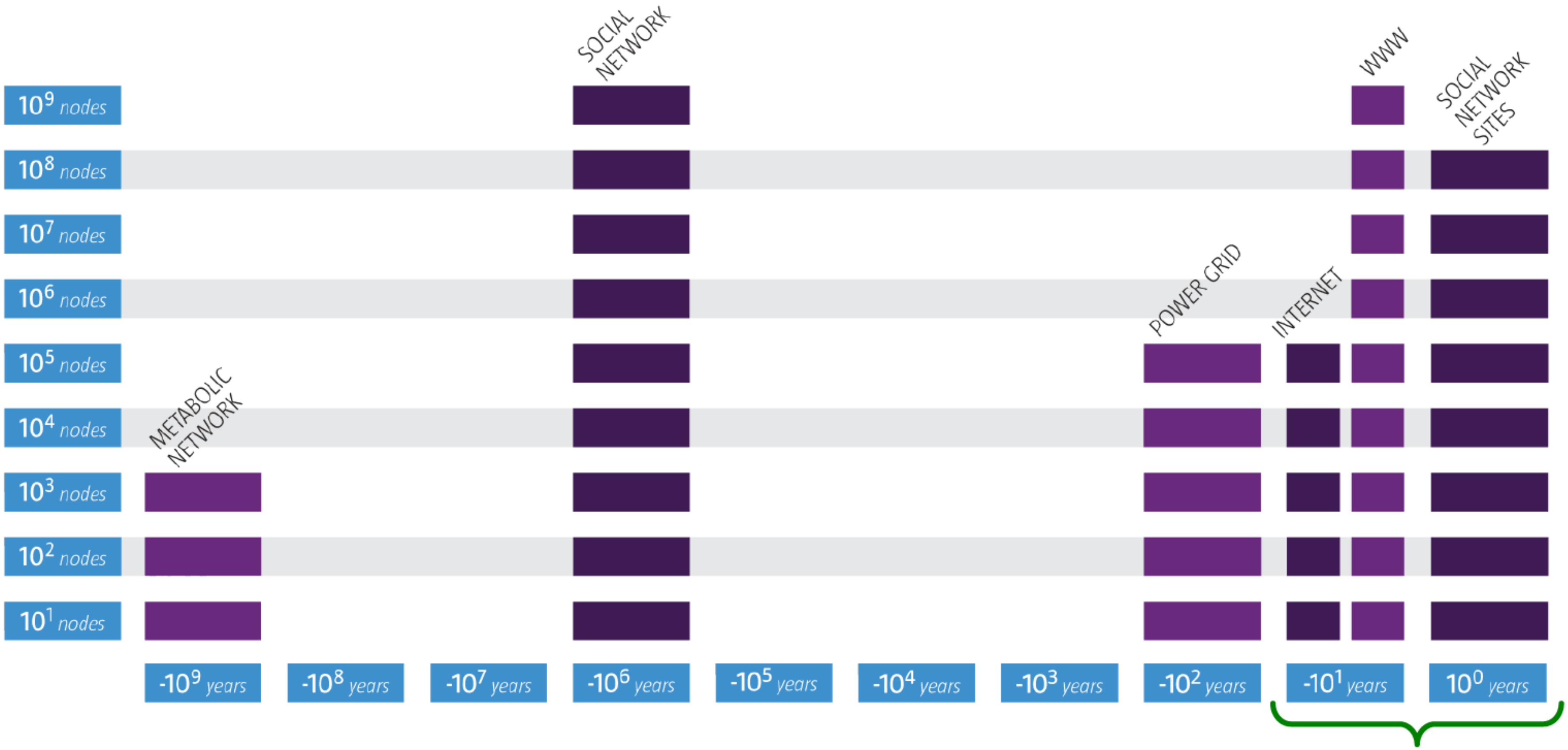
Protein-Protein Interaction Networks:
Nodes: Proteins
Edges: 'physical' interactions

Metabolic networks:
Nodes: Metabolites and enzymes
Edges: Chemical reactions

Why Networks?

- Universal language for describing complex data
  - Networks from science, nature, and technology are more similar than one would expect
- Shared vocabulary between fields
  - Computer Science, Social Science, Physics, Economics, Statistics, Biology
- Data availability
  - Web/mobile, bio, health, and medical
- Impact!
  - Social networking, social media…

Networks: Why Now?



Age and size of networks

Networks: Size Matters

- Network data: Orders of magnitude
  - 436-node network of email exchange at a corporate research lab [Adamic-Adar, SocNets '03]
  - 43,553-node network of email exchange at a university [Kossinets-Watts, Science '06]
  - 4.4-million-node network of declared friendships on a blogging community [Liben-Nowell et al., PNAS '05]
  - 240-million-node network of communication on Microsoft Messenger [Leskovec-Horvitz, WWW '08]
  - 800-million-node Facebook network [Backstrom et al. '11]

Networks: Online

- Communication networks:
  - Intrusion（入侵）/ fraud（欺诈） detection,

- Social networks:
  - Link prediction, friend recommendation
  - Social circle detection, community detection
  - Social recommendations
  - Identifying influential nodes, Viral marketing

- Information networks:
  - Navigational aids

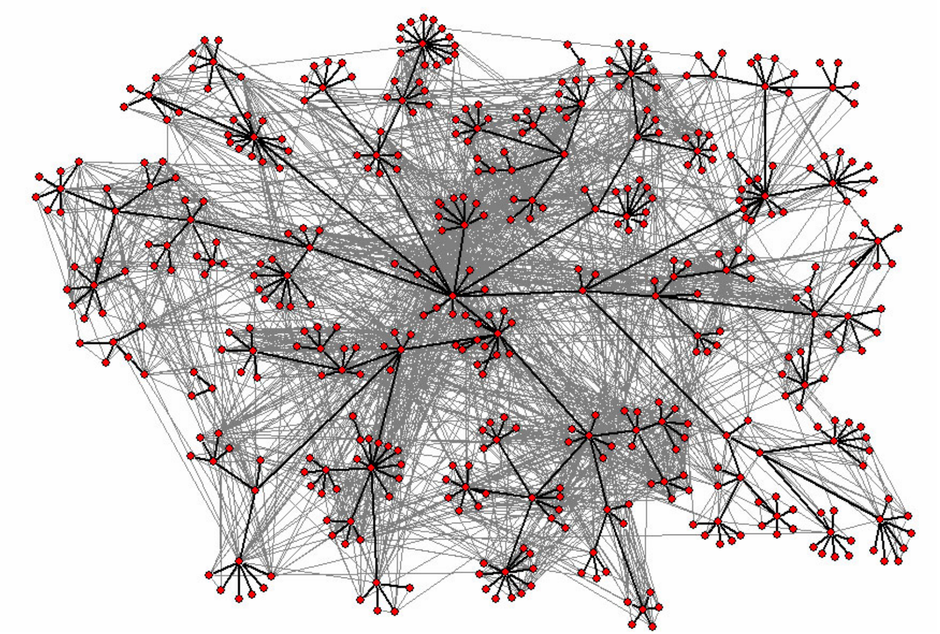
Networks: Impact

……

15万名奥巴马支持者在Facebook安装了“奥巴马2012”应用，而通过这个程序，总统竞选团队可以间接得到这些支持者数百万的Facebook好友信息。

Web/Networks Structure Mining: Structure of the Web Graph

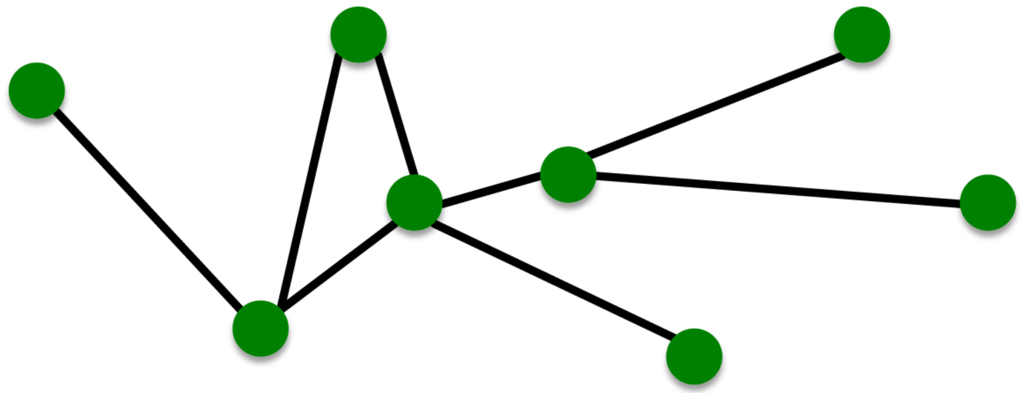
Structure of Networks?



Network is a collection of objects where some pairs of objects are connected by links

What is the structure of the network?

Components of a Network



- **Objects:** nodes, vertices V
- **Interactions:** links, edges E
- **System:** network, graph $G=(V, E)$

Networks or Graphs?

- Network often refers to real systems
  - Web, Social network, Metabolic network

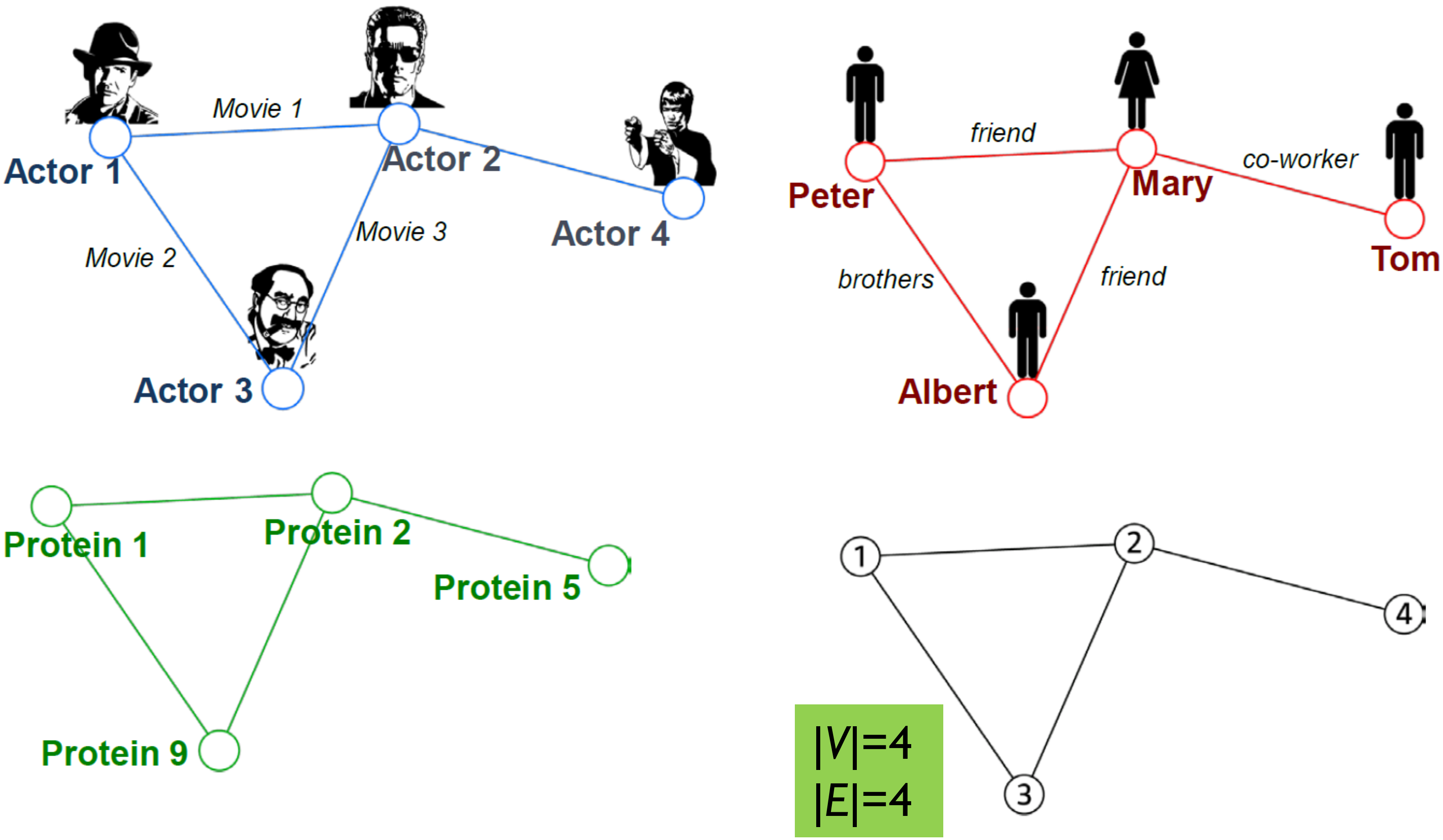
  Language: Network, node, link

- Graph is mathematical representation of a network
  - Web graph, Social graph (a Facebook term)

  Language: Graph, vertex, edge

In most cases we will use the two terms interchangeably

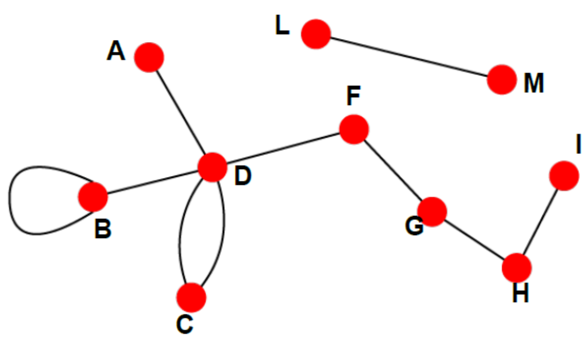
Networks: Common Language



Undirected vs. Directed Networks

Undirected

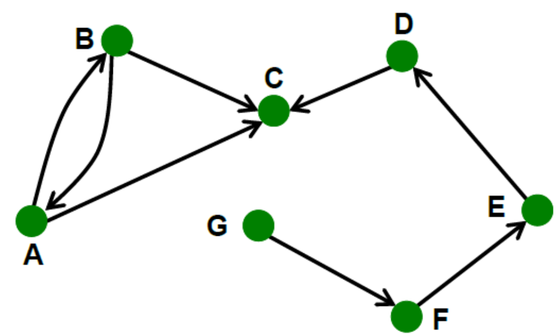
- Links: undirected (symmetric, reciprocal)



- Examples:
  - Collaborations
  - Friendship on Facebook

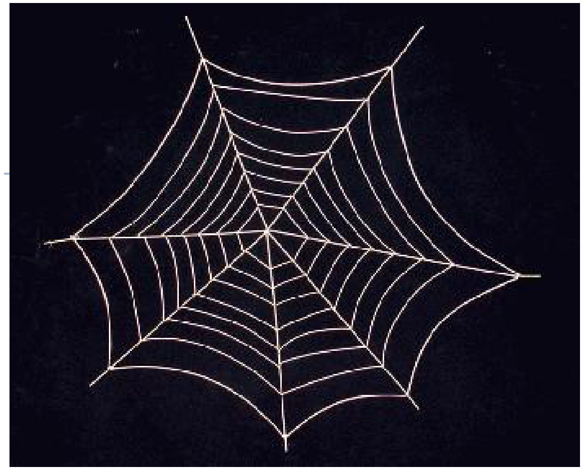
Directed

- Links: directed (arcs)



- Examples:
  - Phone calls
  - Following on Twitter

Web as a Graph



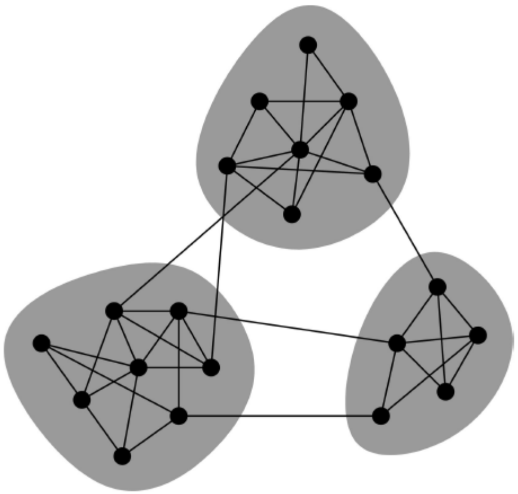
- Q: What does the Web "look like"?
- Here is what we will do next:
  - We will take a real system (i.e., the Web)
  - We will collect lots of Web data
  - We will represent the Web as a graph
  - We will use language of graph theory to reason about the structure of the graph
  - Do a computational experiment on the Web graph
  - **Learn something about the structure of the Web!**

Web/Networks Structure Mining: Communities

One of the most important structural properties in networks

Network Communities（社区）

- Granovetter's theory (and common sense) suggest that networks are composed of tightly connected sets of nodes



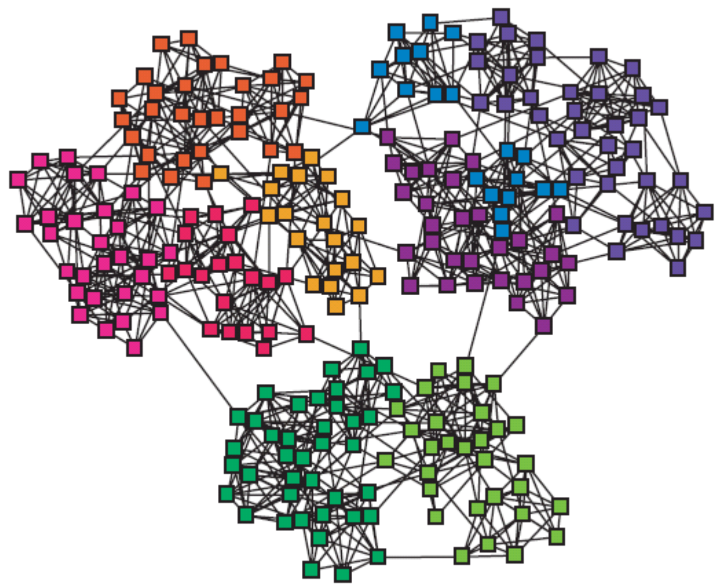
- Network communities:
  - Sets of nodes with lots of connections inside and few to outside (the rest of the network)

Communities, Clusters, Groups, Modules

Finding Network Communities

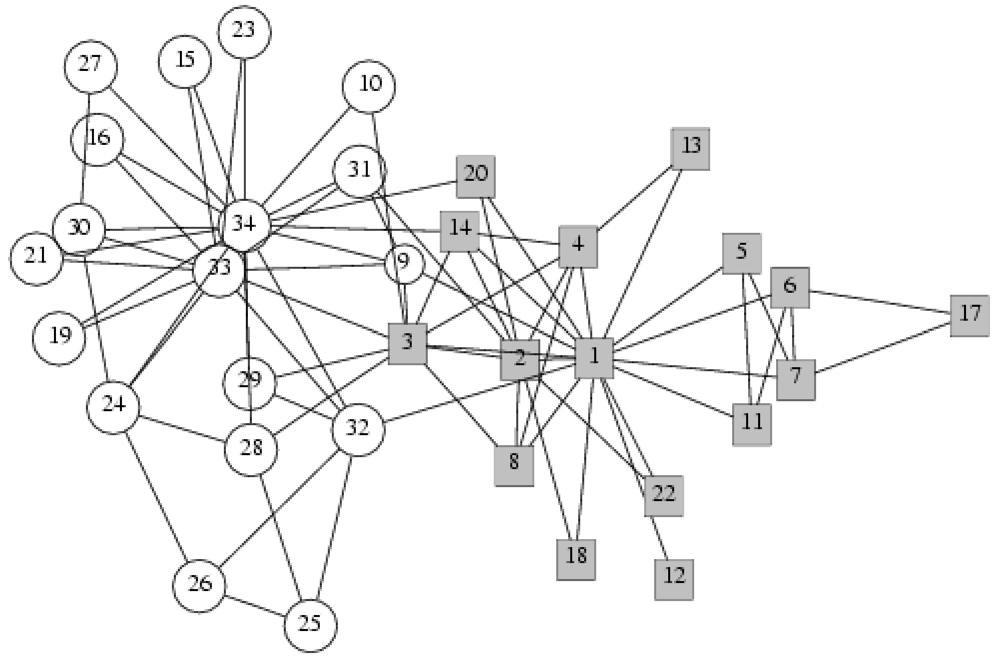
- How to automatically find such densely connected groups of nodes?

- Ideally such automatically detected clusters would then correspond to real groups



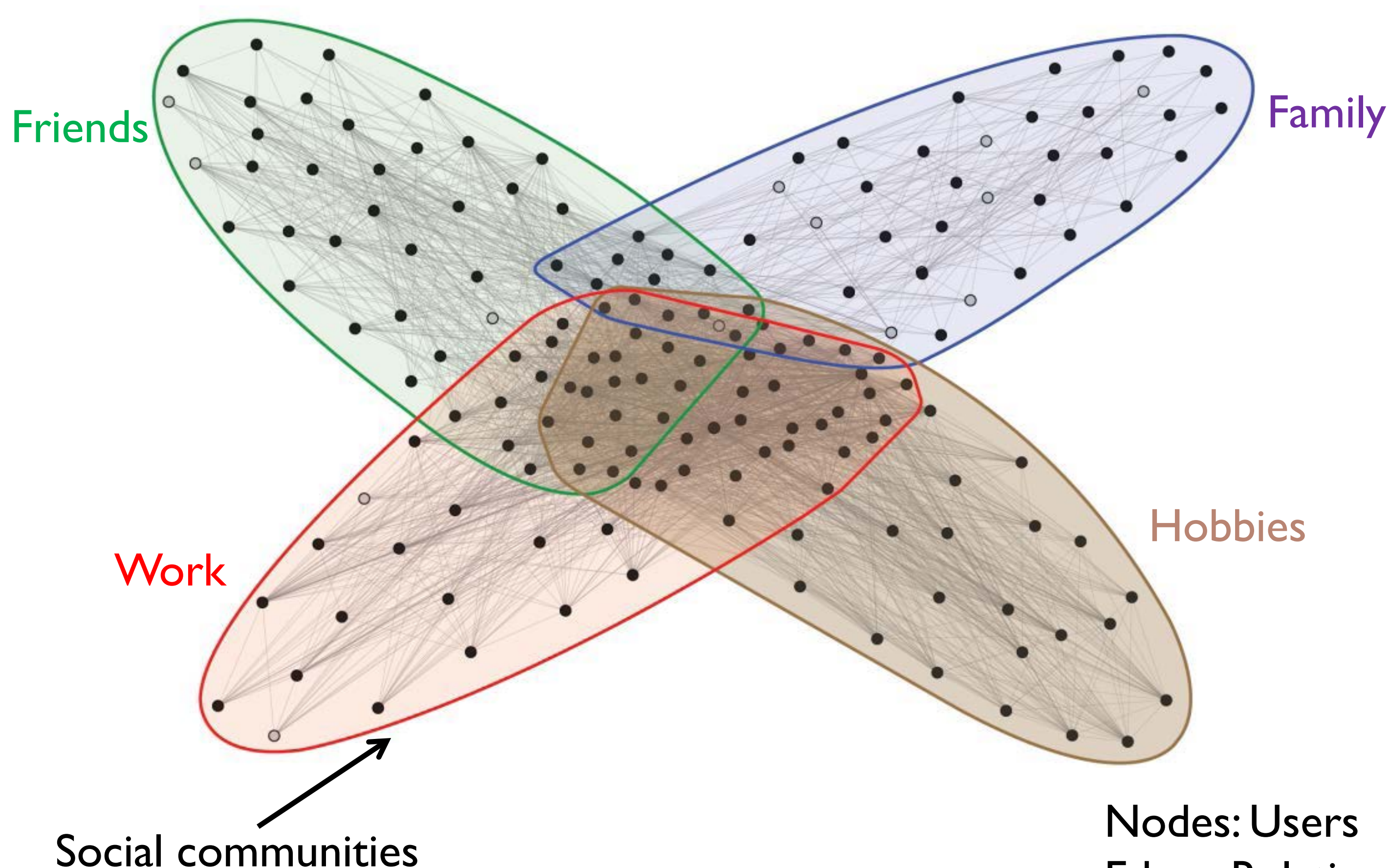
Communities, Clusters, Groups, Modules

Zachary's Karate Club Network



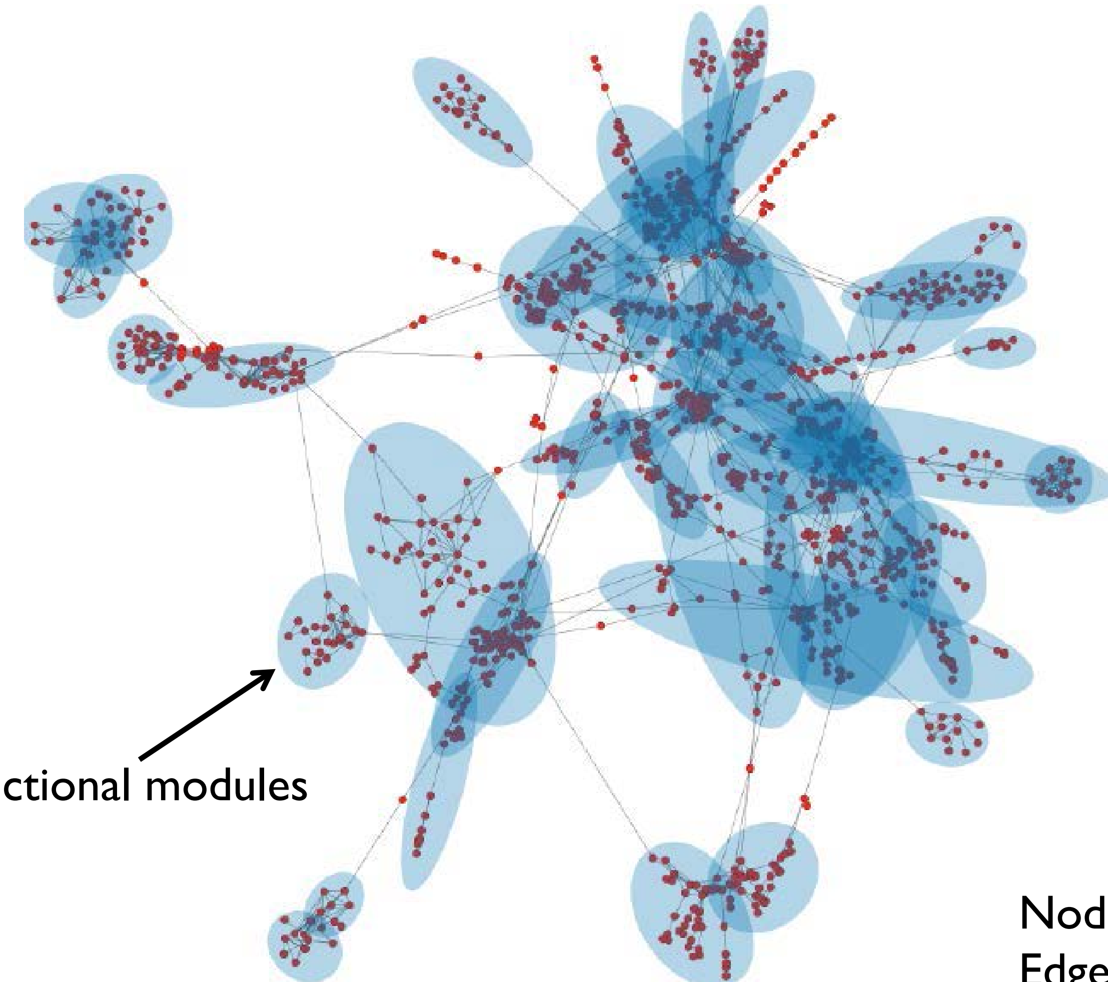
- Social ties and rivalries in a university karate club
- Two conflicting groups
- Split could be explained by a minimum cut in the network

Social Communities



Nodes: Users
Edges: Relationship

Protein-Protein Interactions

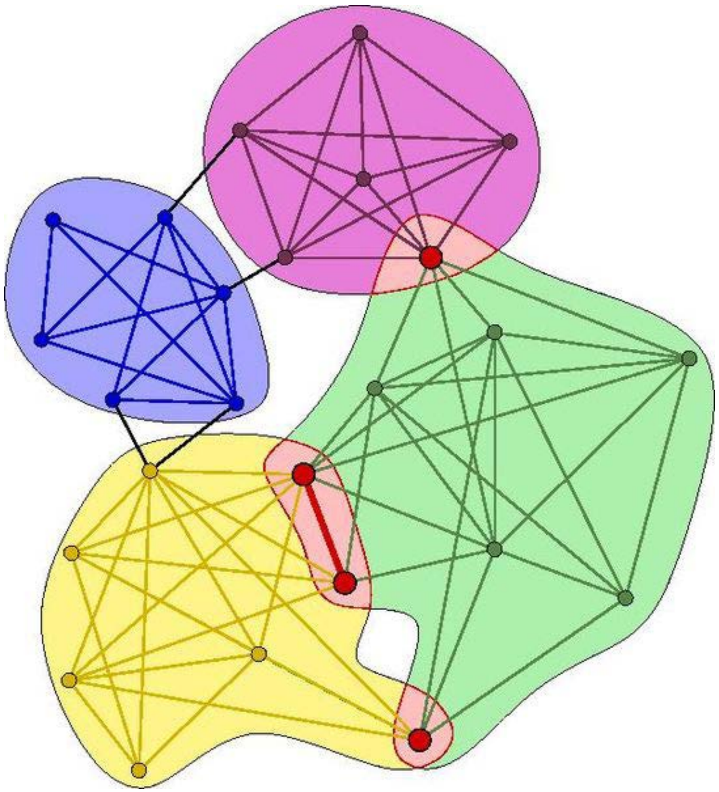


Functional modules

Nodes: Proteins
Edges: Interactions

Community Detection

- How to find communities?



We will work with **undirected** networks

Community Detection Methods

Connection between community detection and clustering

- Agglomerative hierarchical clustering
- Partitional clustering
  - K-means

Based on Structural Similarity

Vertex Similarity

A: adjacency matrix of undirected *G*

- A_{ij}=1 if (*i,j*) is an edge, else 0

- Structural dissimilarity measure

$$d_{ij} = \sqrt{\sum_{k \neq i,j} (A_{ik} - A_{jk})^2}$$

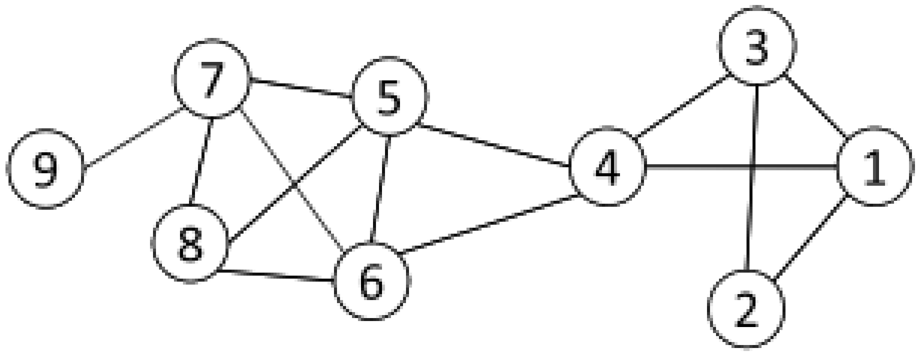
- Jaccard similarity

$$Jaccard(\mathbf{v}_i, \mathbf{v}_j) = \frac{|N_i \cap N_j|}{|N_i \cup N_j|}$$

- Cosine similarity

$$cosine(\mathbf{v}_i, \mathbf{v}_j) = \frac{\sum_k A_{ik} A_{jk}}{\sqrt{\sum_s A_{is}^2}\sqrt{\sum_t A_{jt}^2}}$$

Vertex Similarity



$$
\begin{aligned}
Jaccard(4,6) &= \frac{|\{5\}|}{|\{1,3,4,5,6,7,8\}|} = \frac{1}{7} \\
cosine(4,6) &= \frac{1}{\sqrt{4 \cdot 4}} = \frac{1}{4}
\end{aligned}
$$

Community Detection Methods

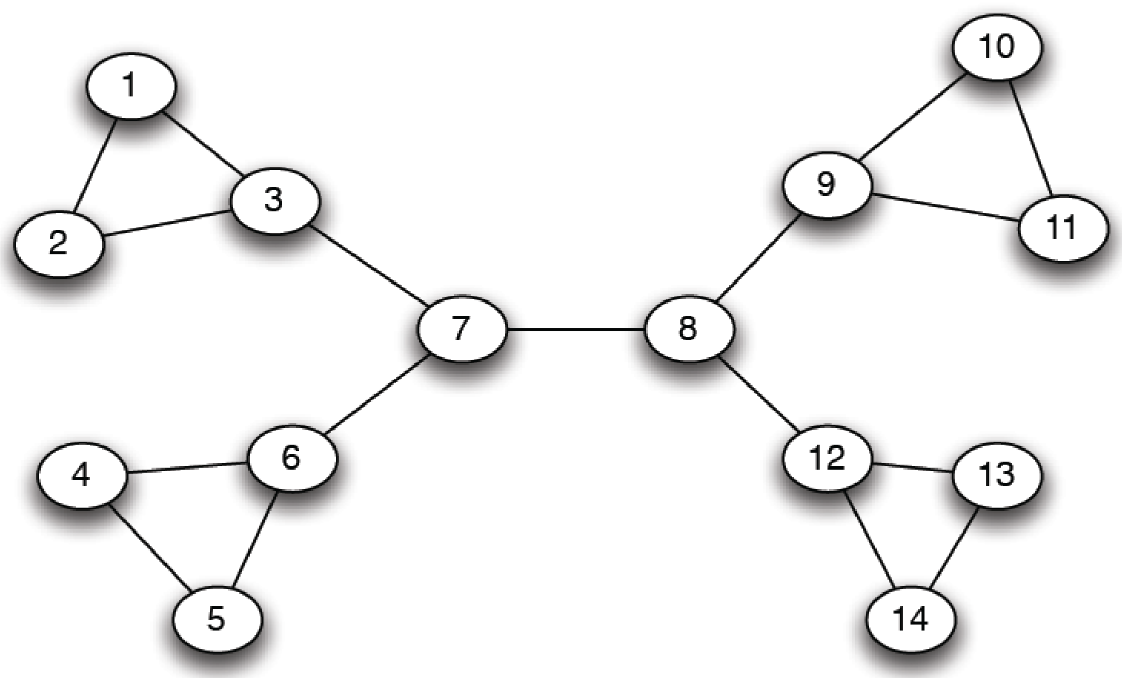
Connection between community detection and clustering

- Agglomerative hierarchical clustering
- Partitional clustering
  - K-means

- Divisive hierarchical algorithm – Girvan and Newman
- Spectral graph cut
- Modularity maximization

Divisive Removal of Weak Ties/Bridges

- Bridges:
  - Form part of the shortest path between pairs of nodes in different parts of the network
- Simple idea:
  - Remove bridges and local bridges

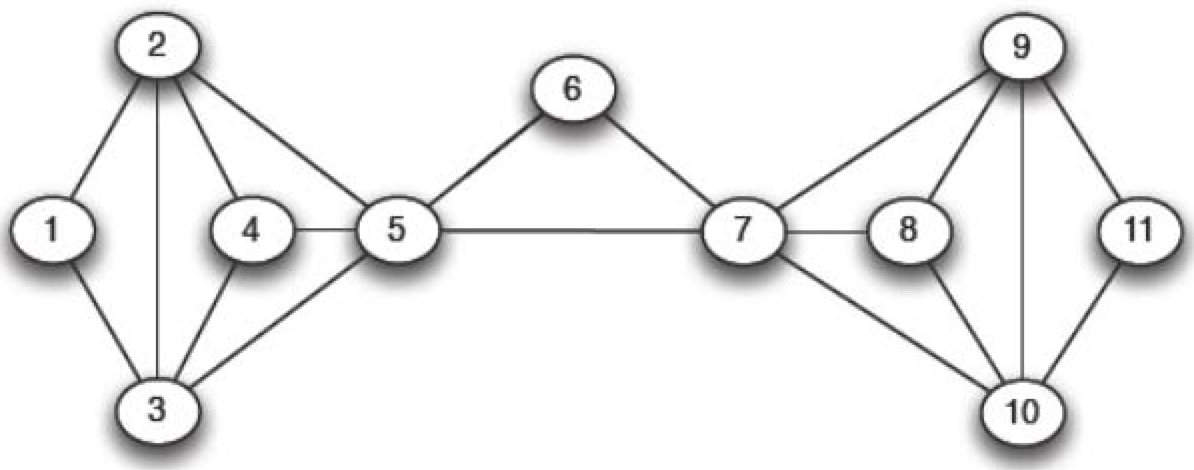


Generalize the Role of Bridges

- Look for the edges that carry the most of "traffic" in a network
  - Without the edge, paths between many pairs of nodes may have to be "re-routed" a longer way
  - Edges to link different densely-connected regions
  - Good candidates for removal in a divisive method
  - Generalize the (local) bridges

Traffic in a Network

- For nodes A and B connected by a path assume 1 unit of "flow"
  - (If A and B in different connected components, flow = 0)
- Divide flow evenly along all possible shortest paths from A to B
  - if k shortest paths from A and B, then 1/k units of flow pass along each
- Eg: 2 shortest paths from 1 to 5, each with 1/2 units of flow

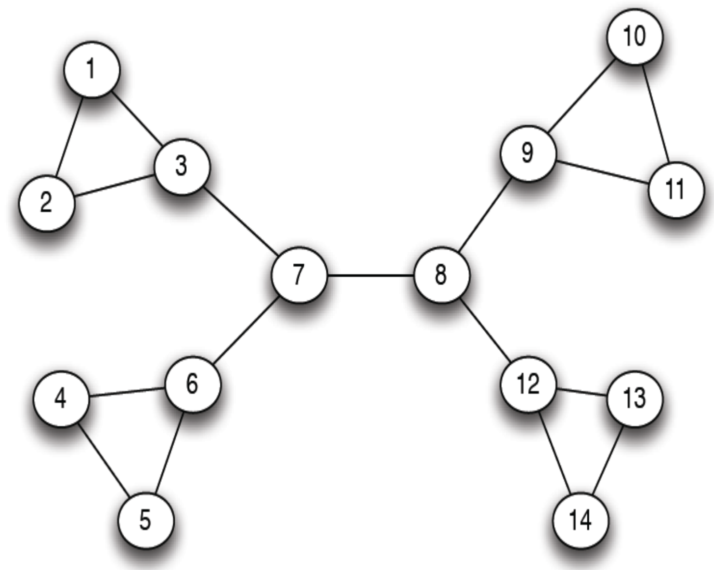


Edge Betweenness

- Betweenness of an edge: the total amount of flow it carries
  - counting flow between all pairs of nodes using this edge

Eg:

- Edge 7-8: each pair of nodes between [1-7] and [8-14]; each pair with traffic = 1; total 7 x 7 = 49
- Edge 3-7: each pair of nodes between [1-3] and [4-14]; each pair with traffic = 1; total 3 x 11 = 33
- Edge 1-3: each pair of nodes between [1] and [3-14] (not node 2); each pair with traffic = 1; total 1 x 12 = 12
  - similar for edges 2-3, 4-6, 5-6, 9-10, 9-11, 12-13, and 12-14
- Edge 1-2: each pair of nodes between [1] and [2] (no other); each pair with traffic = 1; total 1 x 1 = 1
  - similar for edges 4-5, 10-11, and 13-14



Girvan-Newman

- Divisive hierarchical clustering based on the notion of edge betweenness:
  - Number of shortest paths passing through the edge
- Girvan-Newman Algorithm:
    - Undirected unweighted networks

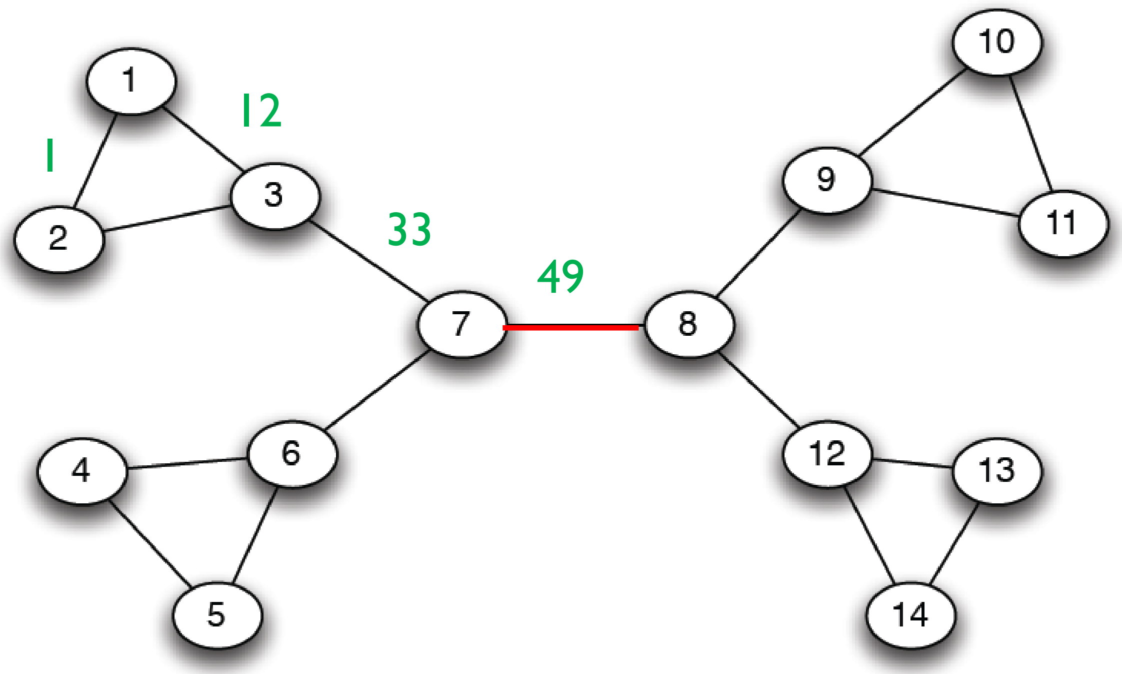
Repeat until no edges are left:

Homework

  - Calculate betweenness of edges ($O(mn)$, or $O(n^2)$ on a sparse graph, with breadth-first-search)
  - Remove edges with highest betweenness

- Connected components are communities
- Gives a hierarchical decomposition of the network

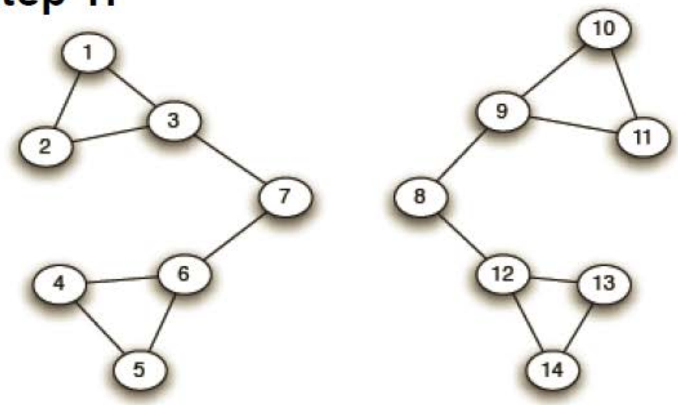
Girvan-Newman: Example



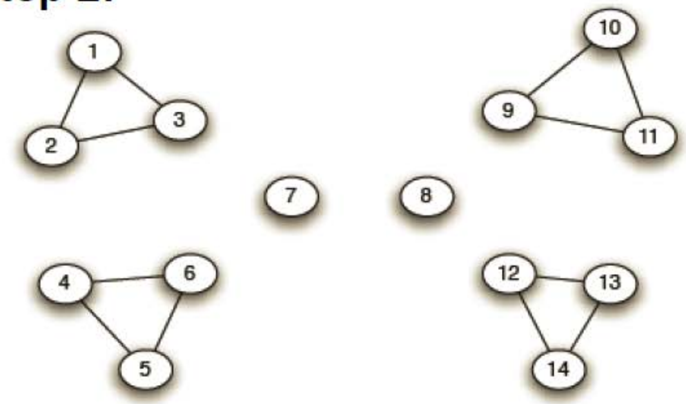
Need to re-compute betweenness at every step

Girvan-Newman: Example

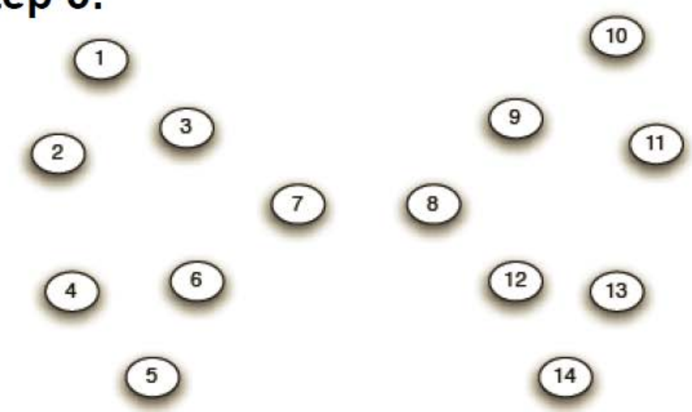
Step 1:



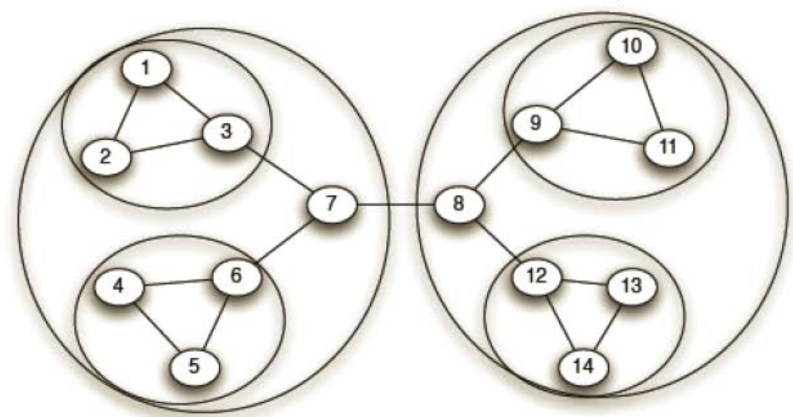
Step 2:



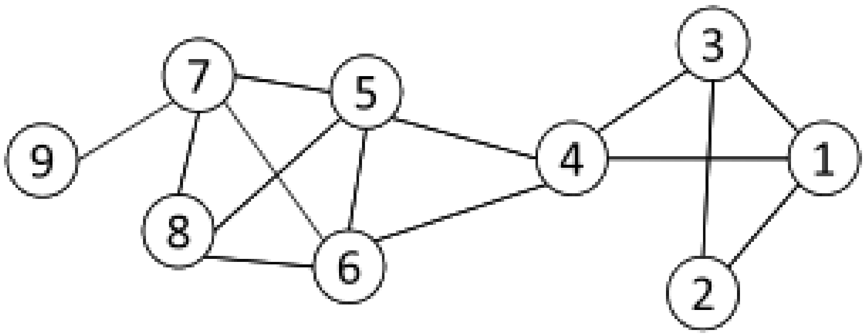
Step 3:



Hierarchical network decomposition:



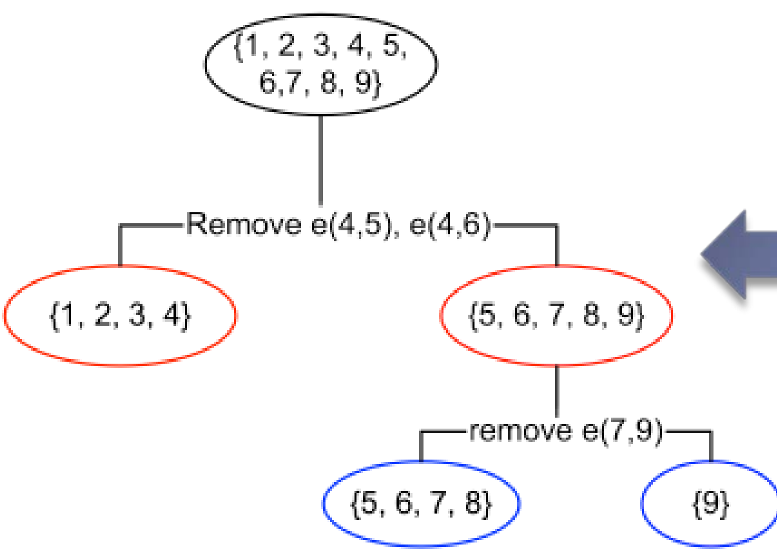
Divisive clustering based on edge betweenness



Initial betweenness value

Table 3.3: Edge Betweenness

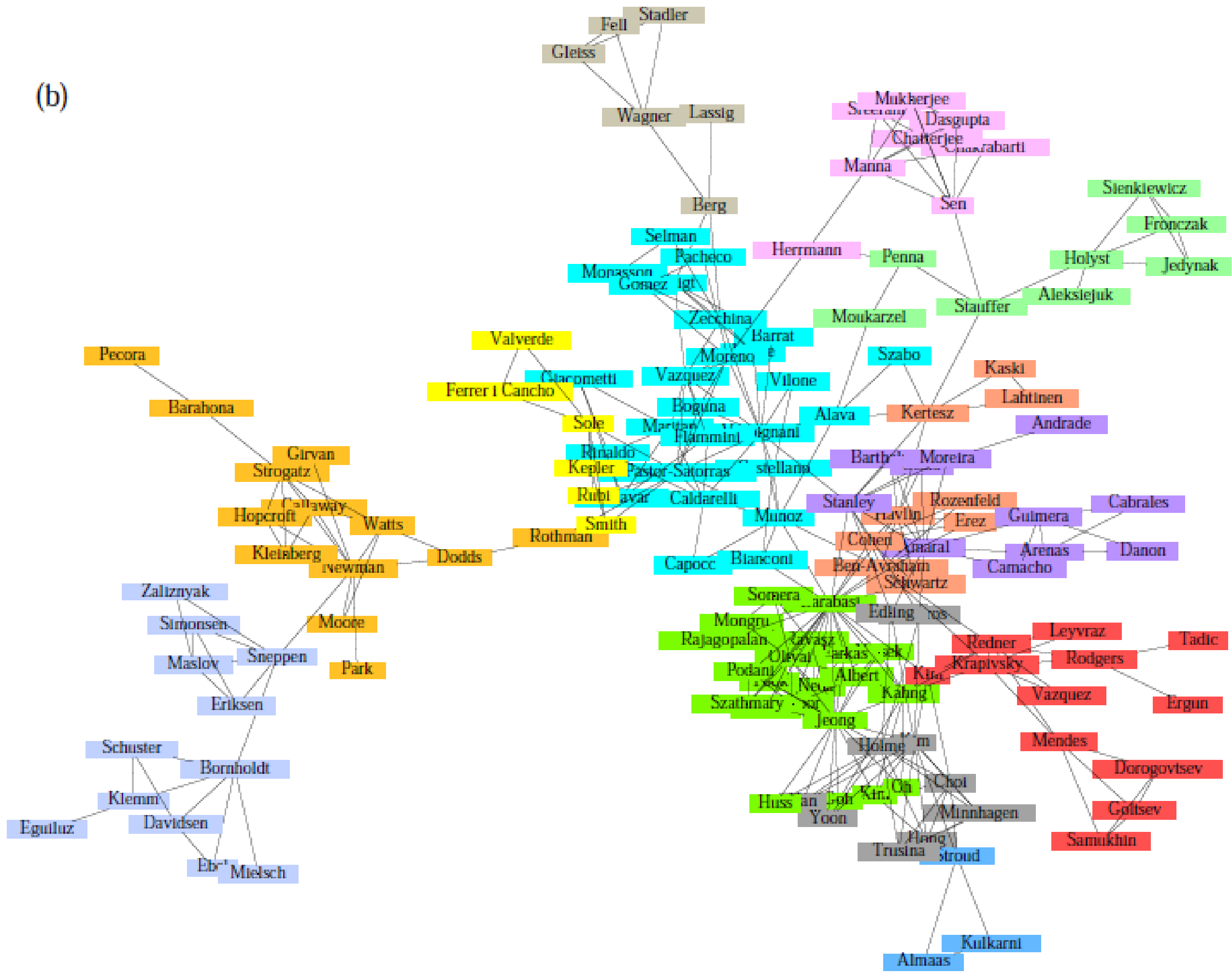
	1	2	3	4	5	6	7	8	9
1	0	4	1	9	0	0	0	0	0
2	4	0	4	0	0	0	0	0	0
3	1	4	0	9	0	0	0	0	0
4	9	0	9	0	10	10	0	0	0
5	0	0	0	10	0	1	6	3	0
6	0	0	0	10	1	0	6	3	0
7	0	0	0	0	6	6	0	2	8
8	0	0	0	0	3	3	2	0	0
9	0	0	0	0	0	0	8	0	0



After remove e(4,5), the betweenness of e(4, 6) becomes 20, which is the highest;

After remove e(4,6), the edge e(7,9) has the highest betweenness value 4, and should be removed.

Girvan-Newman: Results



Communities in physics collaborations