Week 10: Graph Mining (Graph Basic)

Instructor: Daejin Choi (djchoi@inu.ac.kr)

INCHEON NATIONAL UNIVERSITY

* Most slides are used from CS224w course at Stanford University. See http://web.stanford.edu/class/cs224w/

Our goal: Analyzing Large-Scale Data

Statistical analysis is not enough, what’s next?

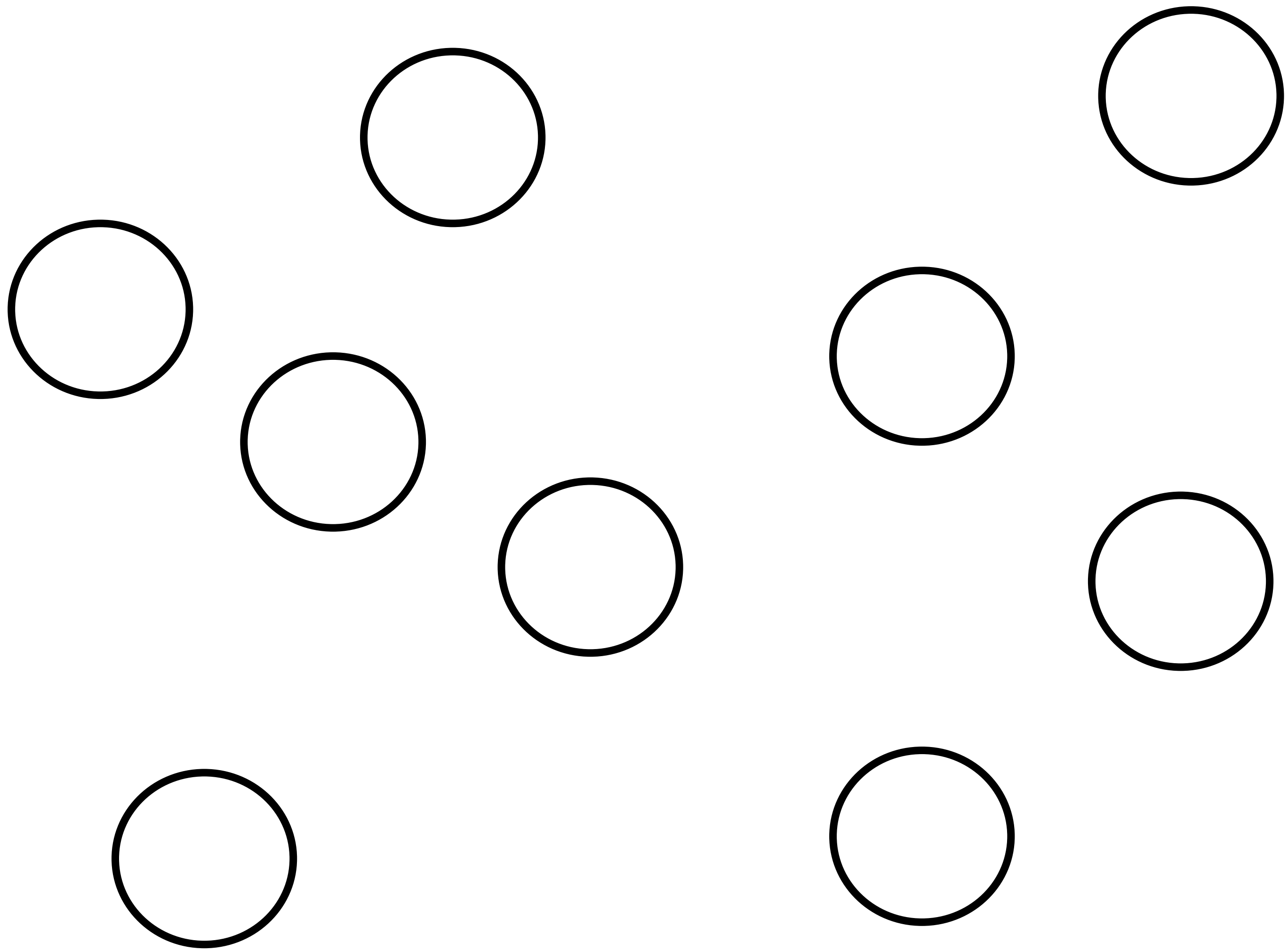
We first need to model the data
→ Graph-based model can be a solution!

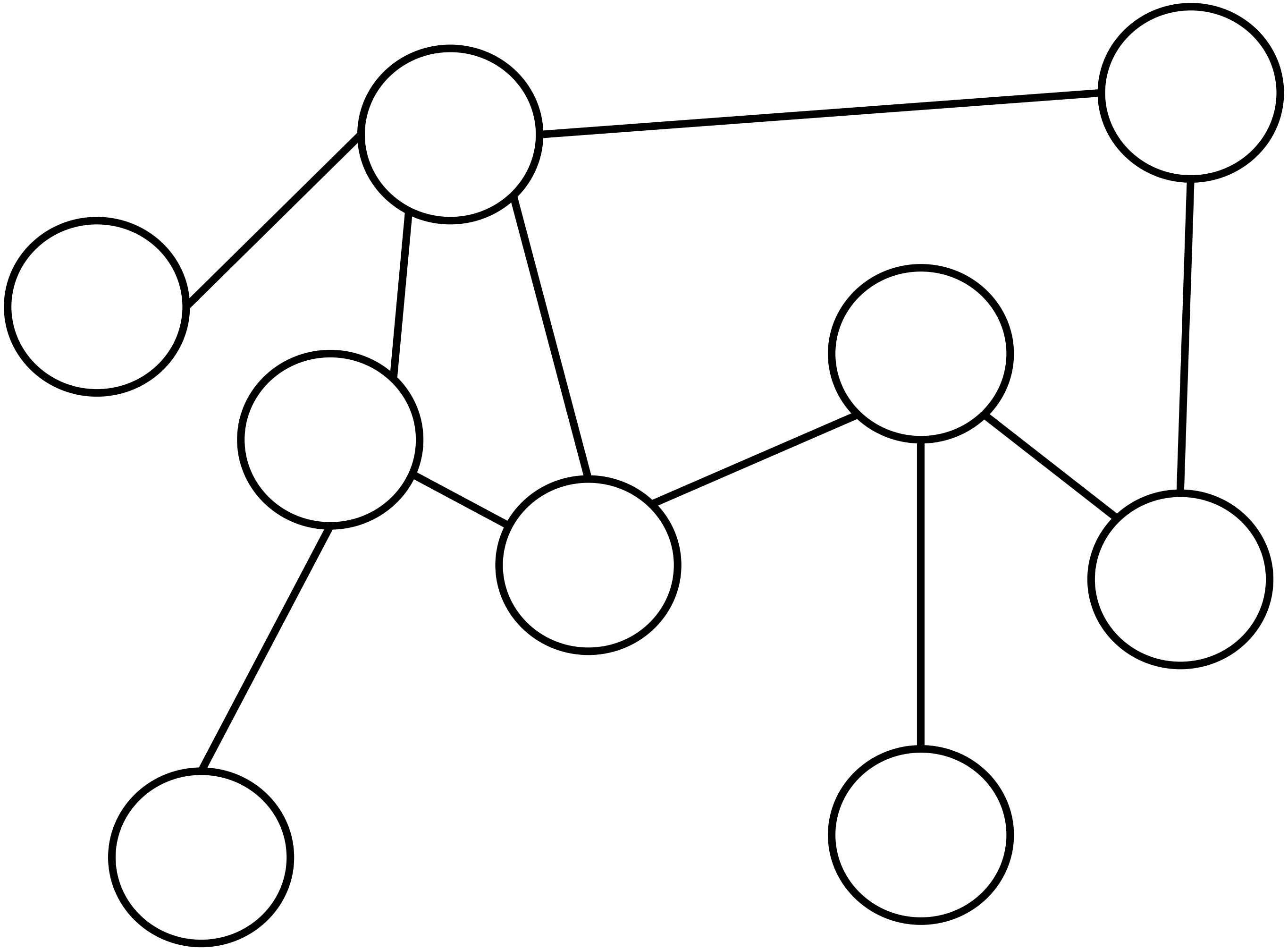
WHY GRAPH?

Most data is structured from

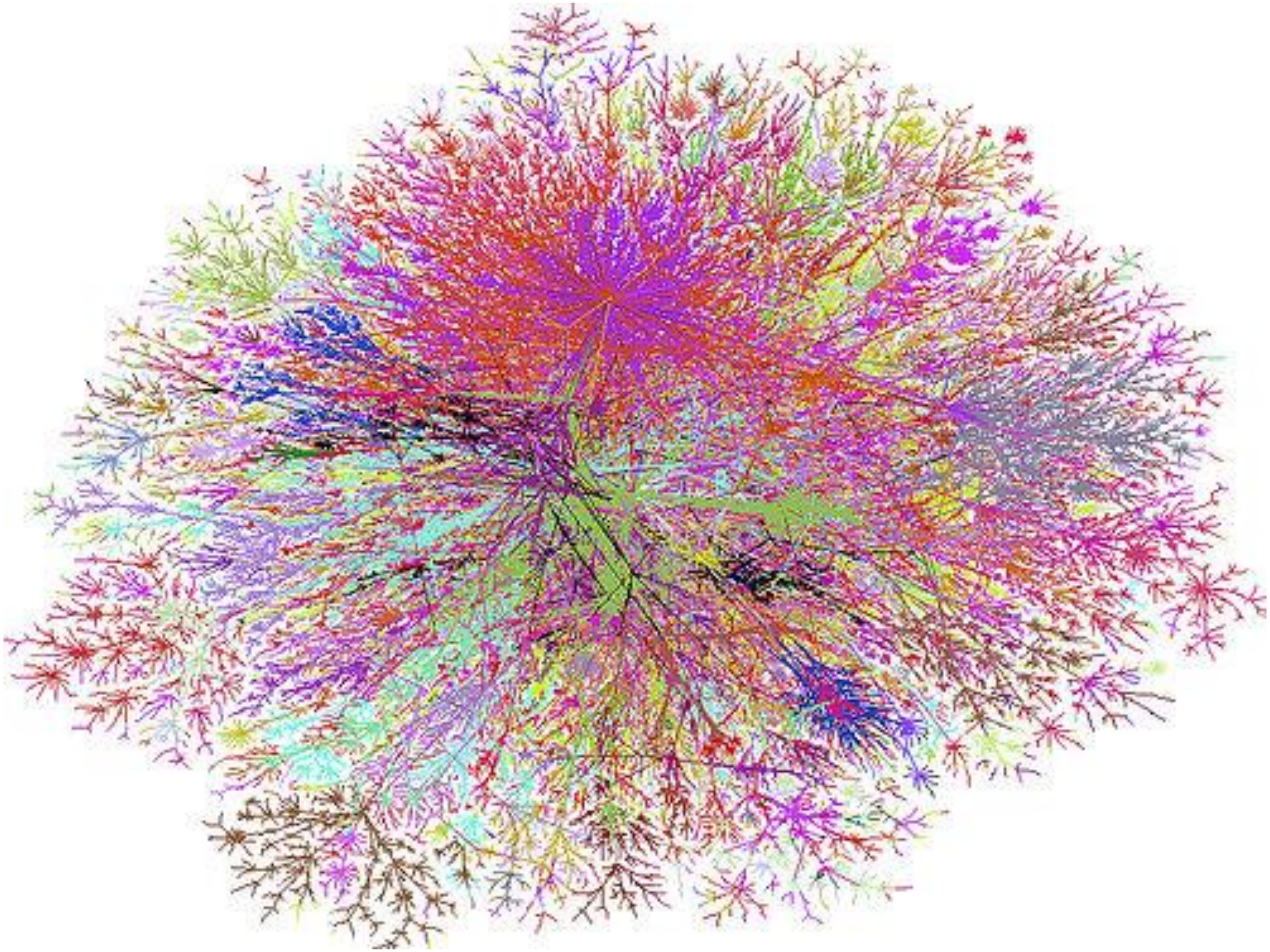
Network

A Complex System Consisting of **Interacting Entities**





The Network!



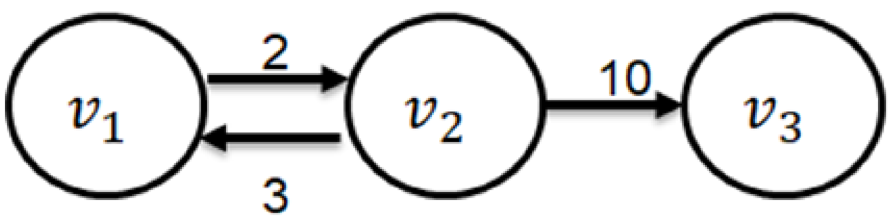
Graph Entities

A set of edges

Graph G = (V, E, W)

A set of nodes (vertex)

A set of weights

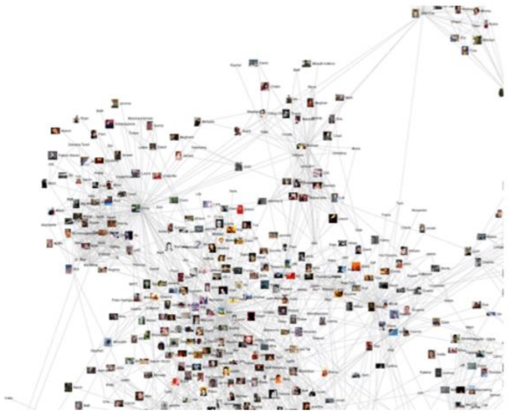


V: $\{v_1, v_2, v_3\}$

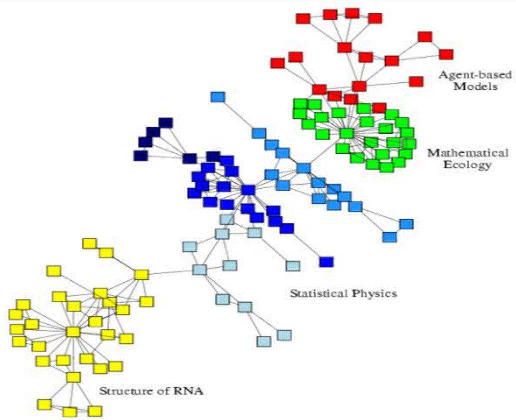
E: $\{e_{12}, e_{21}, e_{23}\}$

W: $\{w_{12}(=2), w_{21}, w_{23}\}$

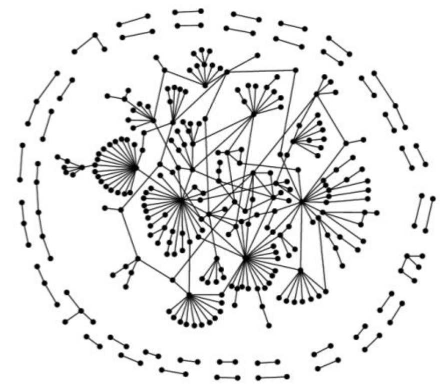
What Can Be Modeled as a Graph?



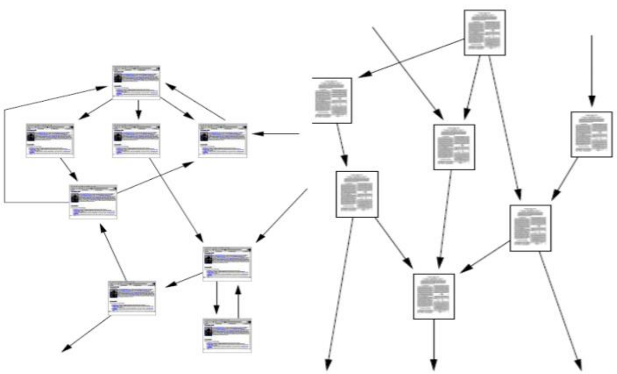
Social networks



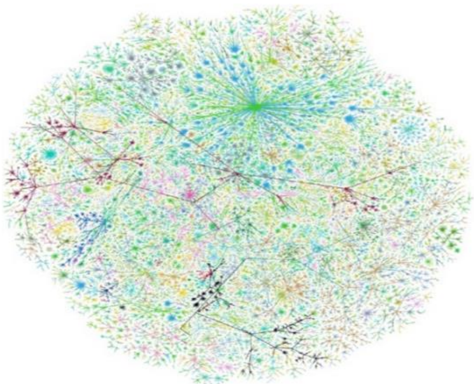
Economic networks



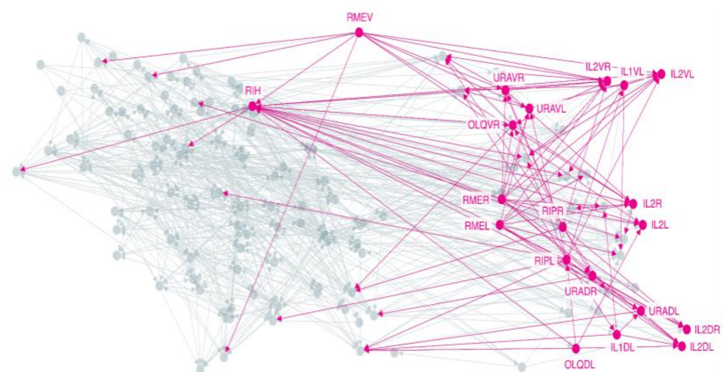
Communication networks



Information networks:
Web & citations

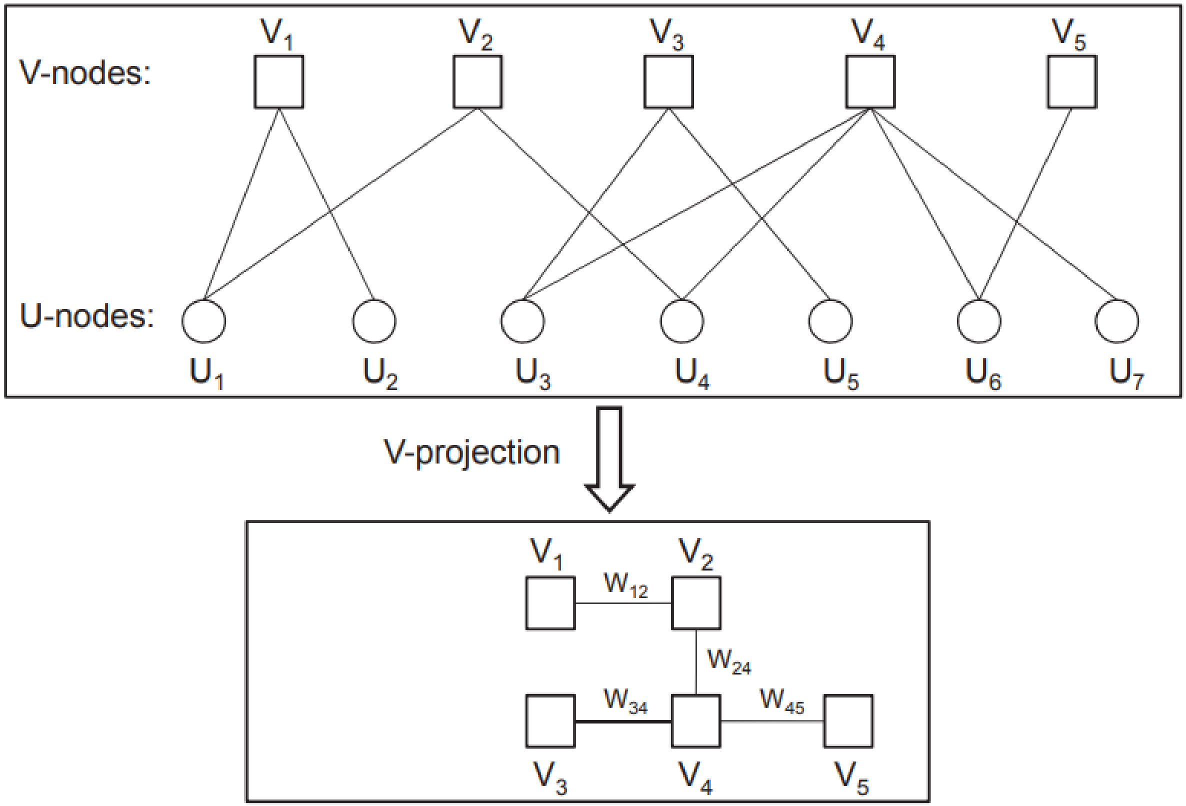


Internet



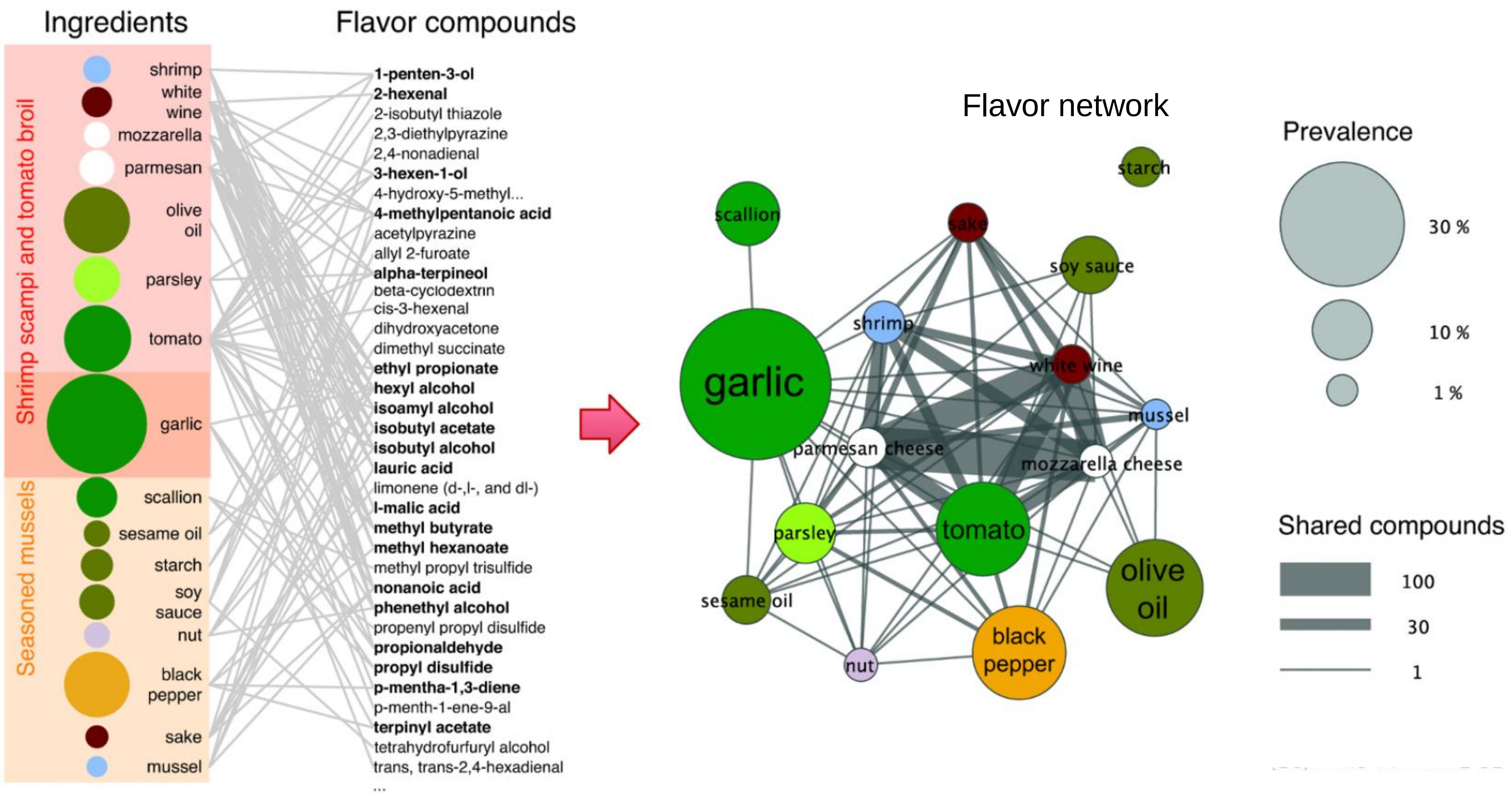
Networks of neurons

“Relations” may be inferred



"Relations" may be inferred

- Flavor network [1] is built by bipartite graph projection



[1] Flavor network and the principles of food pairing, Y.-Y. Ahn, S. Ahnert, J. P. Bagrow, and A.-L. Barabási, *Scientific Reports* **1**, 196 (2011)

Why Networks (Graph)? And… Why Now?

- 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 & computational challenges
  - Web/mobile, bio, health, and medical
- Impact!
  - Social networking, Drug design, AI reasoning

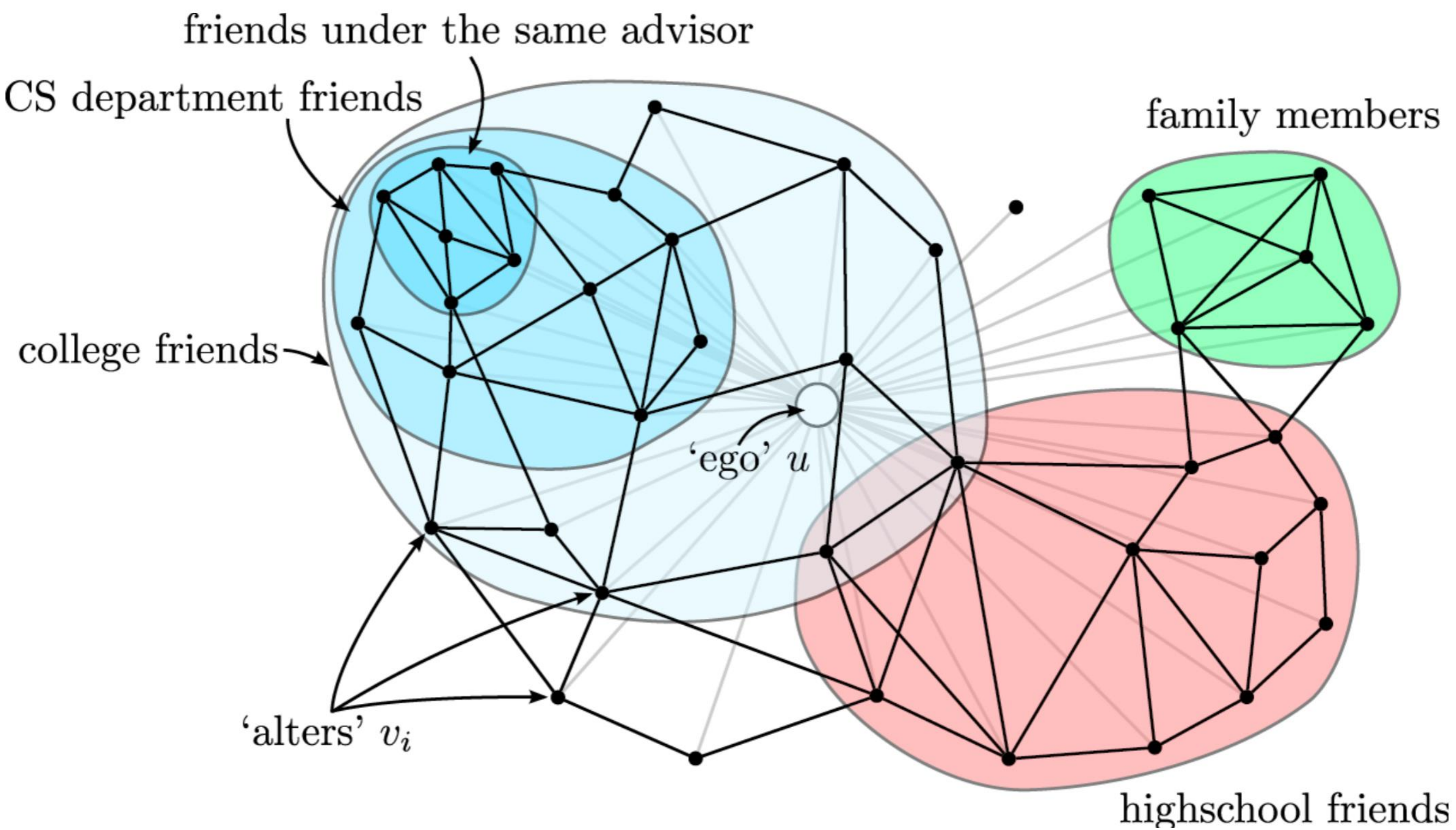
→ Okay, What Can We Do with Graph?

Application Example: Degree of Separation

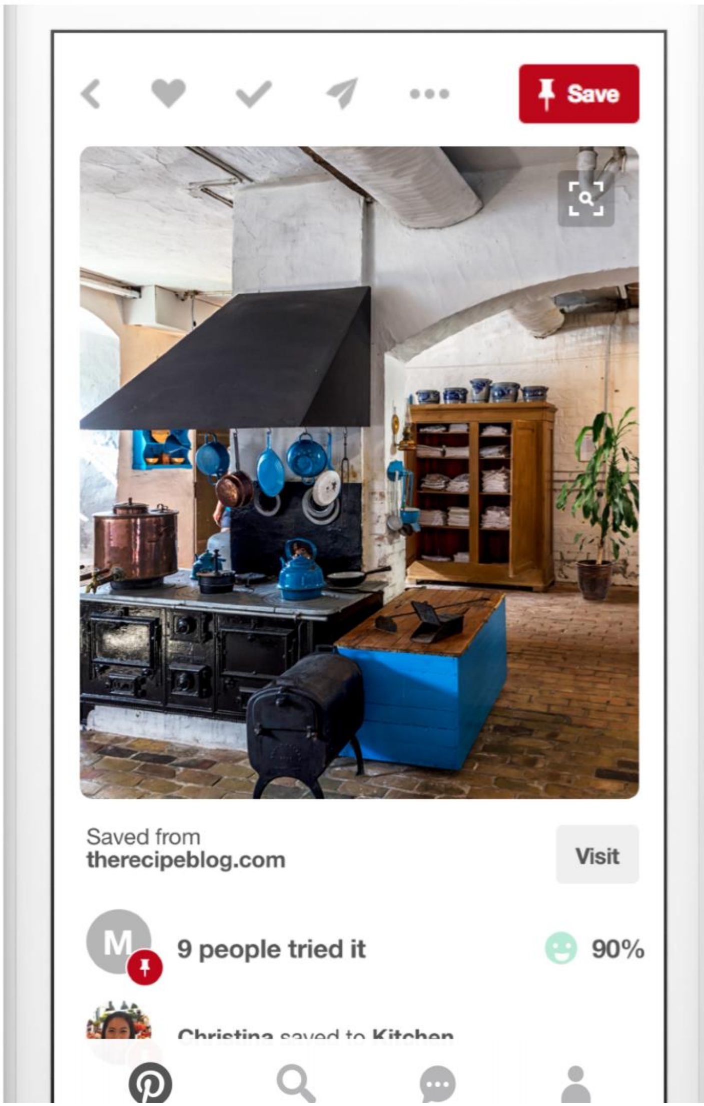
Facebook social graph

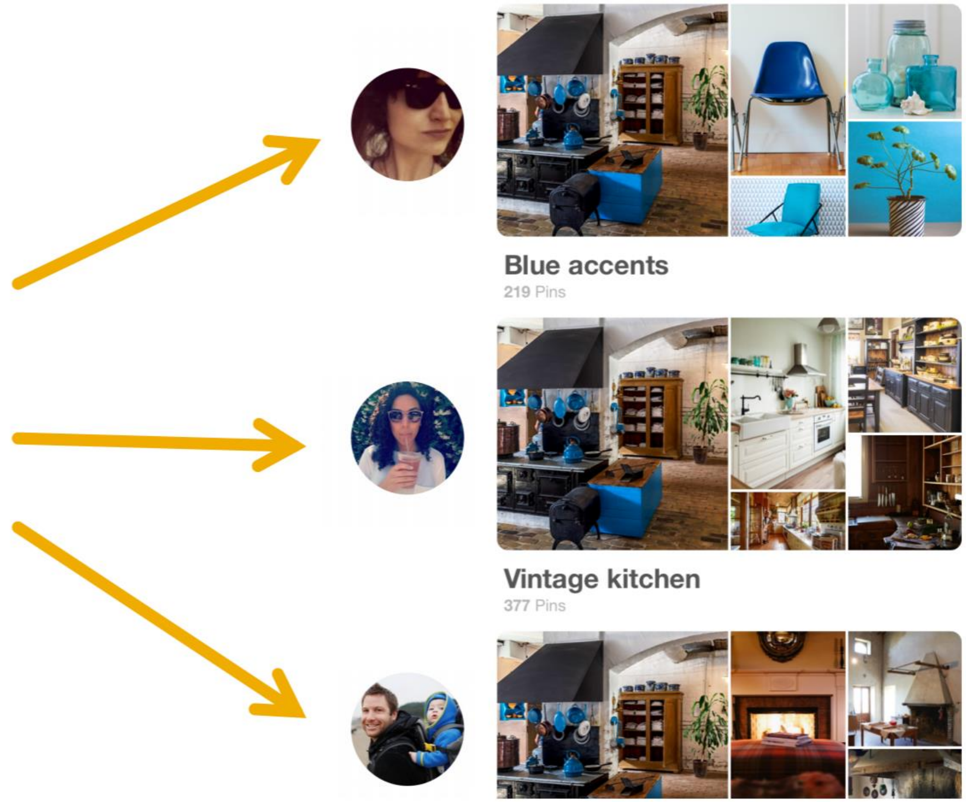
4-degrees of separation [Backstrom-Boldi-Rosa-Ugander-Vigna, 2011]

Application Example: Social Circle Detection



Application Example: Recommendation

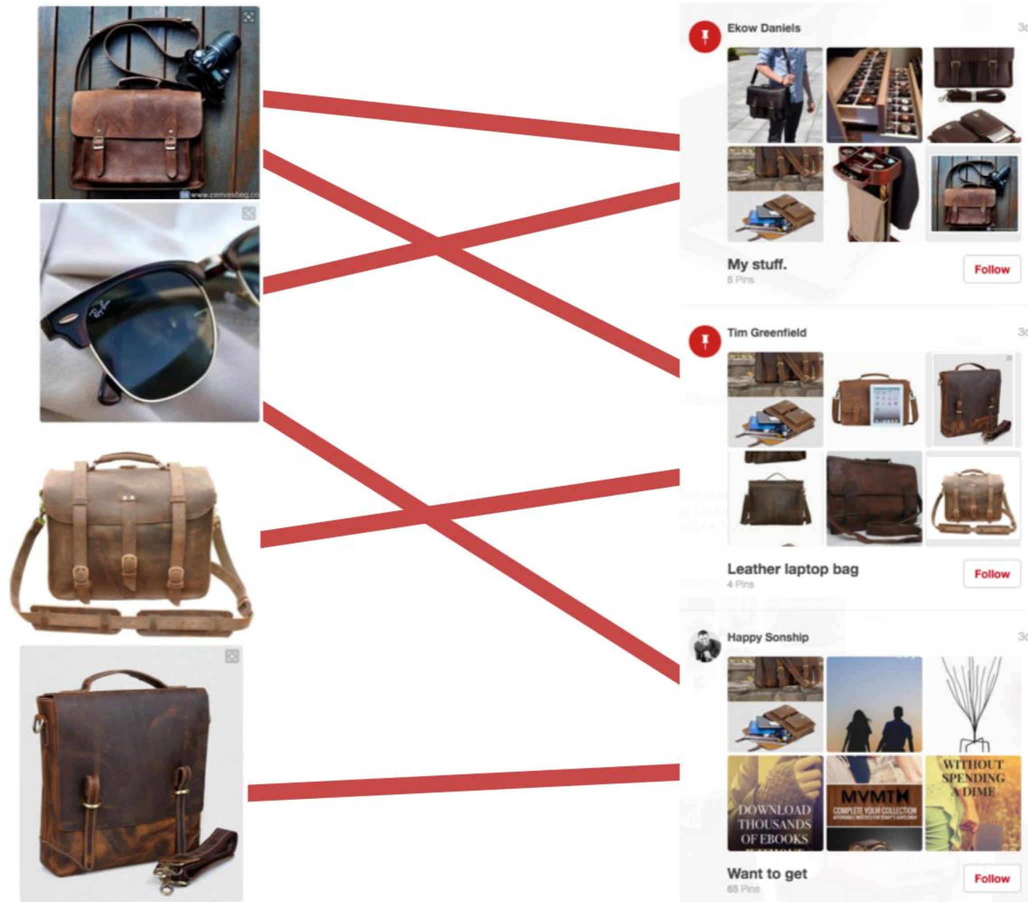




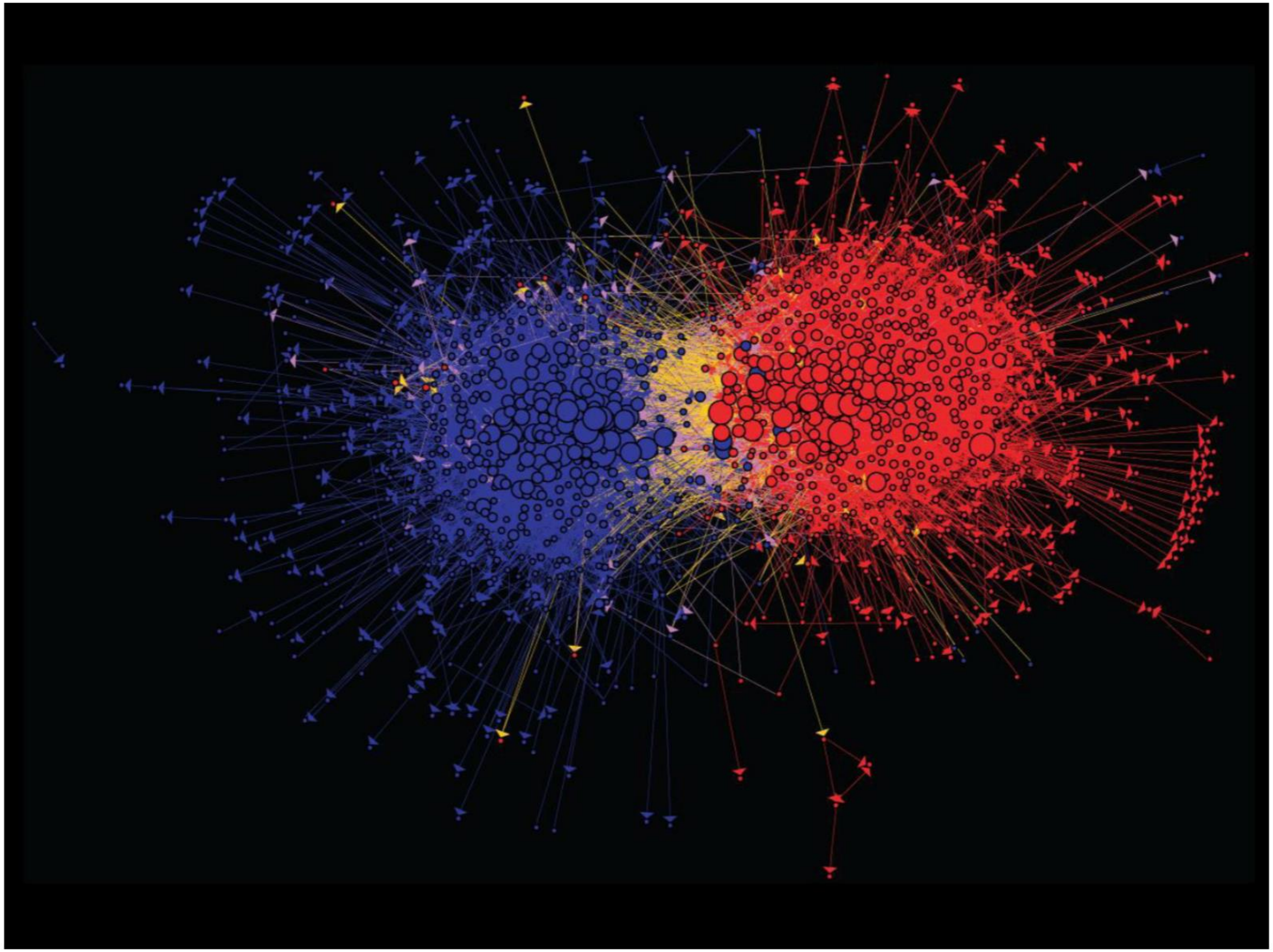
- 300M users
- 4+B pins, 2+B boards

Application Example: Recommendation (cont'd)

Content recommendation is **link prediction**



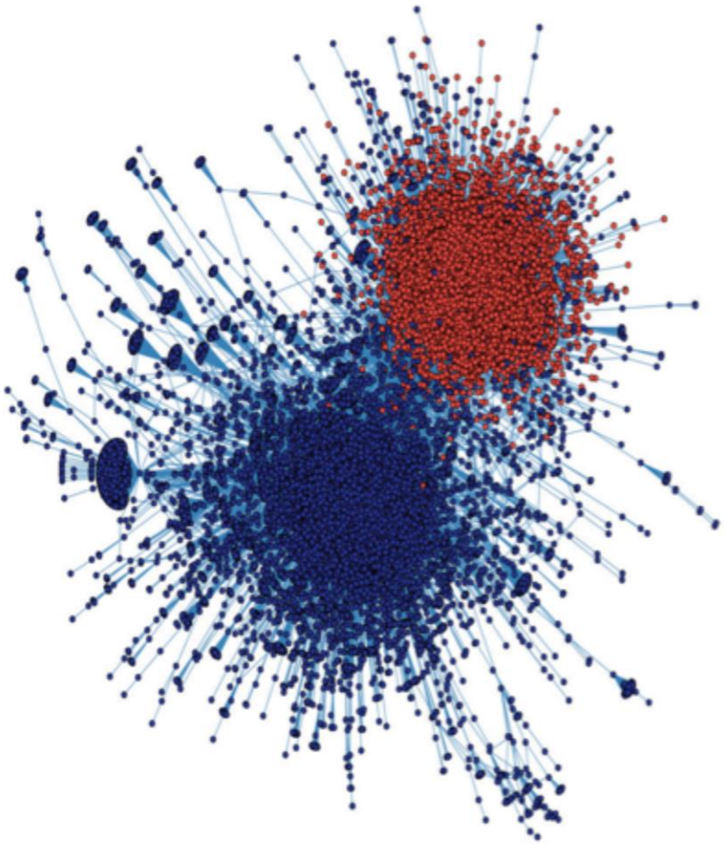
Application Example: Polarization (1/2)

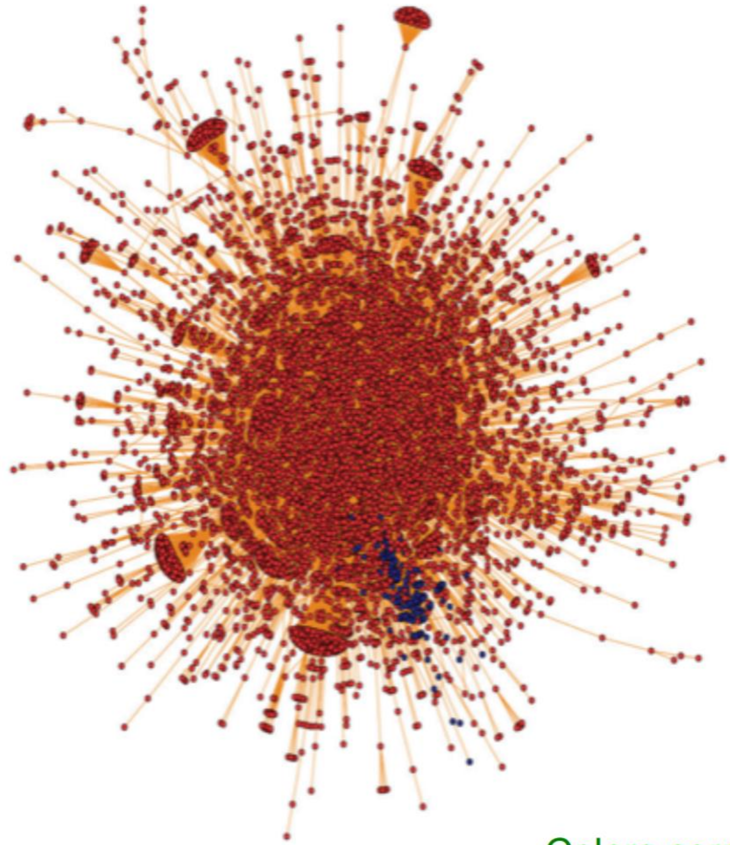


Connections between political blogs

Polarization of the network [Adamic-Glance, 2005]

Application Example: Polarization (2/2)





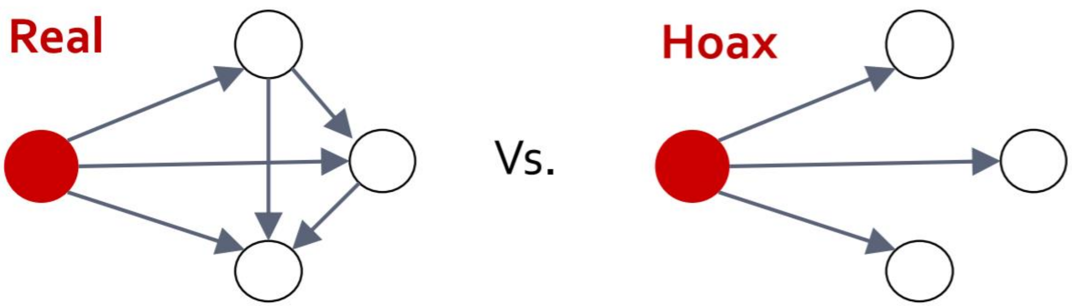
Colors correspond to clusters in the network

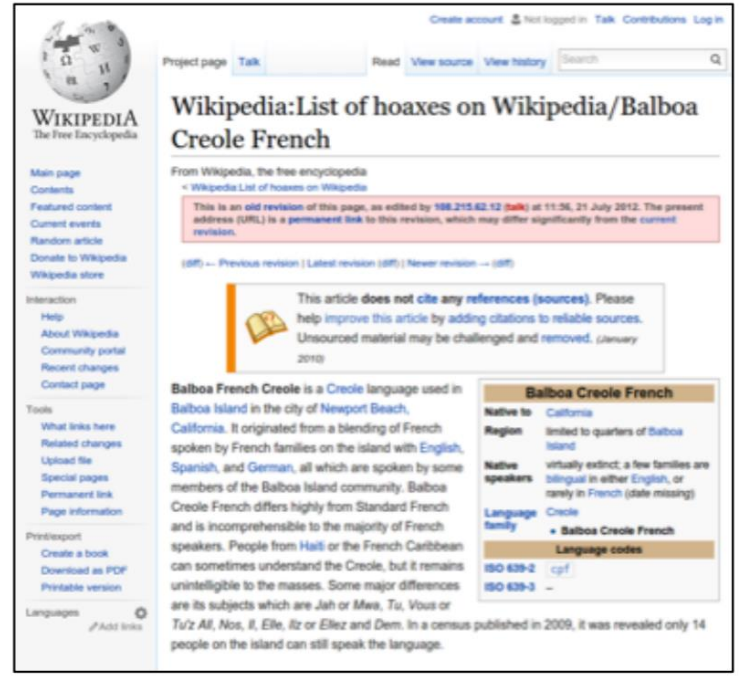
- **Retweet networks:**
  Polarized (left), Unpolarized (right)

Conover, M., Ratkiewicz, J., Francisco, M. R., Gonçalves, B., Menczer, F., & Flammini, A. "Political Polarization on Twitter." (2011)

Application Example: Misinformation

- **Q: Is a given Wikipedia article a hoax?**
  - Real articles link more coherently:



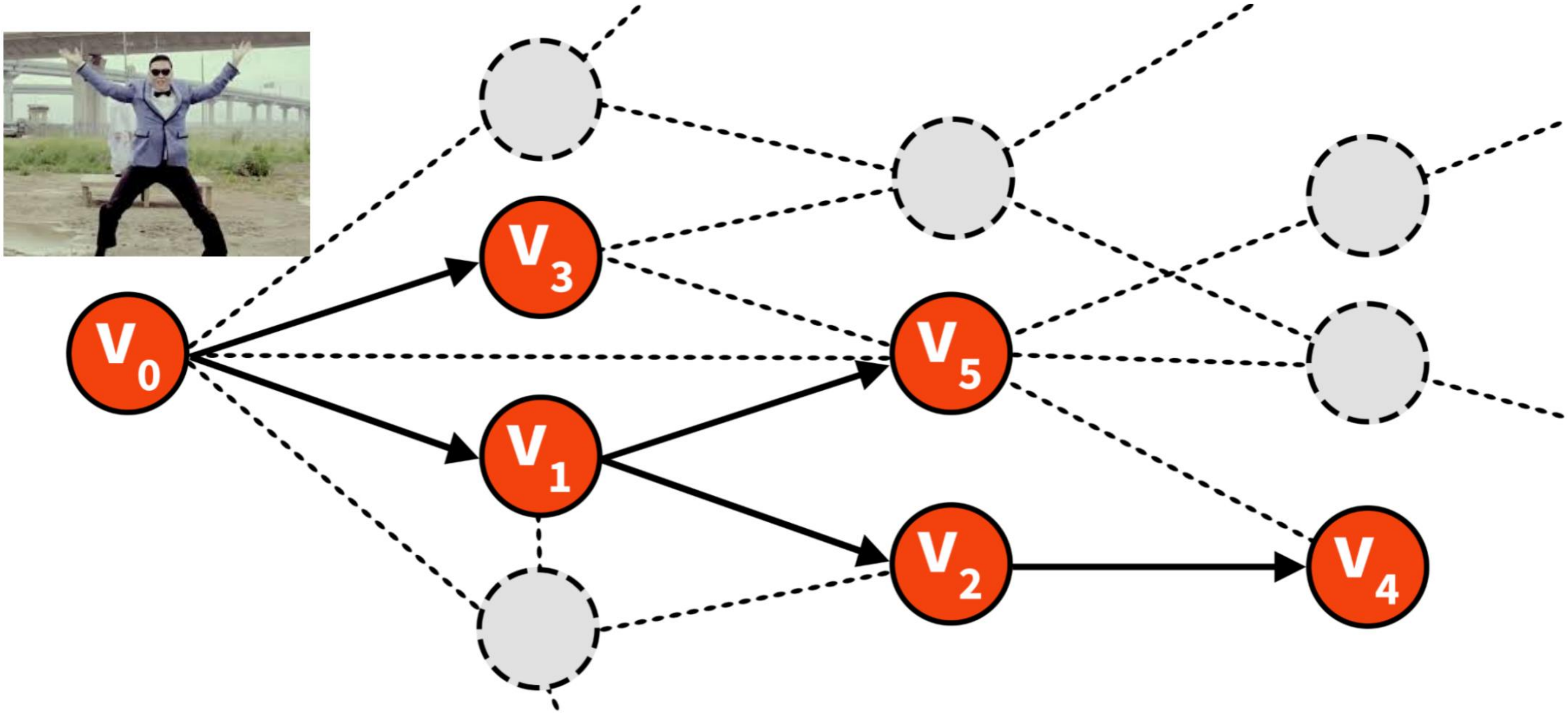


Hoax article detection performance:

50%	66%	86%
Random	Human	Network

[Disinformation on the Web: Impact, Characteristics, and Detection of Wikipedia Hoaxes](). Kumar et al. WWW '16.

Application Example: Virality



Information cascade in social networks

Application Example: Multi-layered Graph

Economic Activity Layer
$(G_{agency},\ G_{trade})$

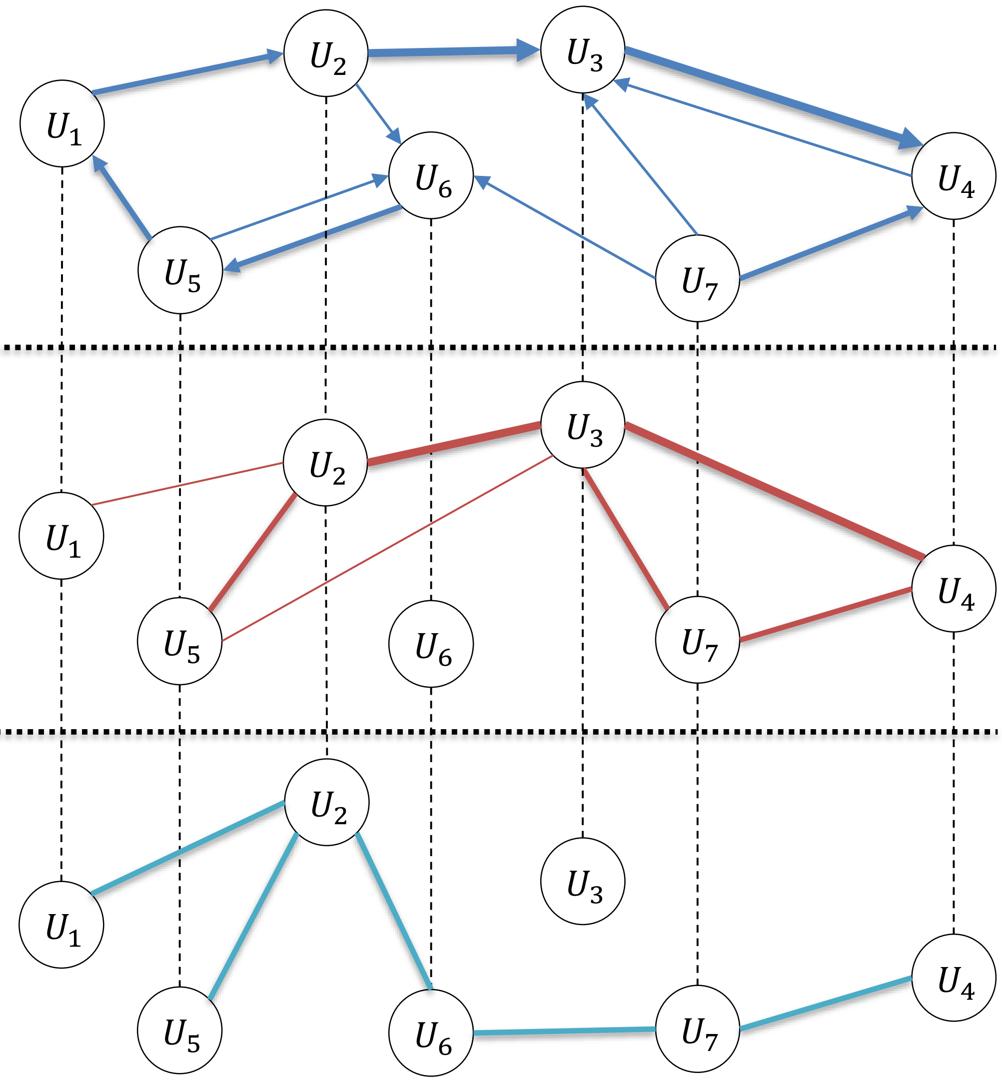
- Directed, weighted

Social Activity Layer
$(G_{pm},\ G_{party})$

- Undirected, weighted

Friendship Layer
(G_{friend})

- Undirected, unweighted



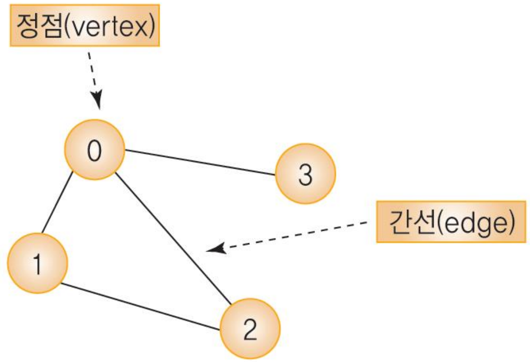
S. Chun, D. Choi, J. Han, H-K Kim and TT Kwon,
"Unveiling a Socio-Economic System in a Virtual World: A Case Study of an MMORPG",
World Wide Web Conference (WWW) 2018.

Graph Basic

Revisit: Graph Definition

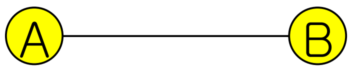
- Data Structure (or dataset) to model the relations of entity
  - E.g., Social relation, p2p network, code flow, paths on the map, …

- G = (V, E, W)
  - V: a set of entity (also called as nodes, object, …)
  - E: a set of relation (also called as link, …)
  - W: a set of weight (on the corresponding link)

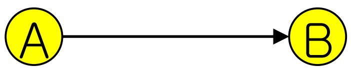


Graph Types

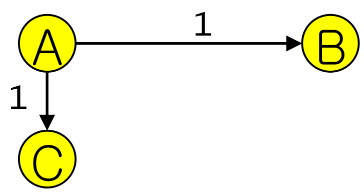
- Undirected Graph
  - E consists of only undirected links
  - E.g., Facebook friendship
  - (Usually) indicated by (A, B)
  - (A, B) = (B, A)



- Directed Graph
  - E consists of only directed links
  - E.g., Twitter following/follower
  - (Usually) indicated by <A, B>
  - <A, B> ≠ <B, A>

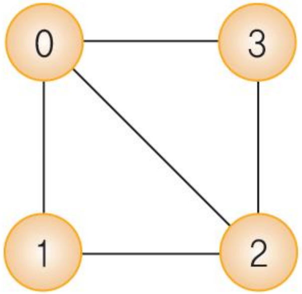


- Unweighted Graph
  - All elements in W are same
  - (Usually) weights are not indicated



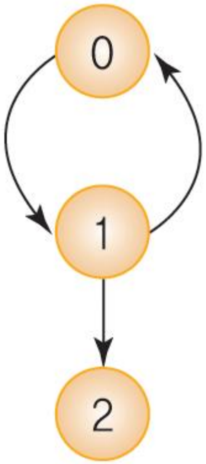
Degrees

- In undirected graph,
  - #links connected to a given node (= #neighbors)
  - In G1, degree(0) = 3



G1

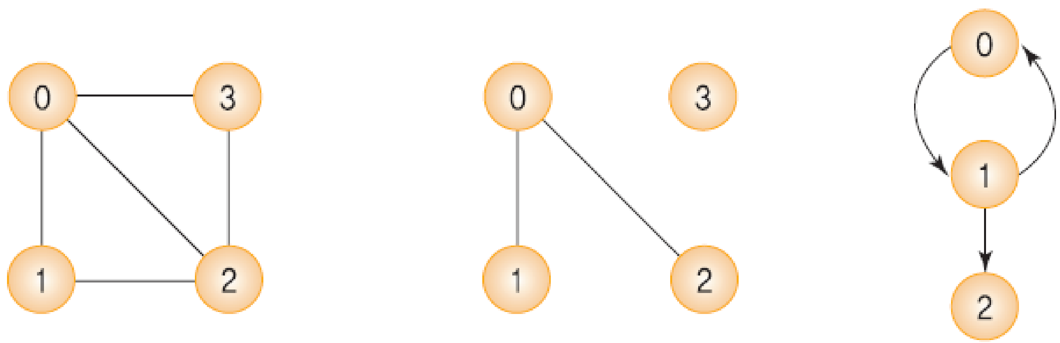
- In directed graph,
  - In-degrees: #links toward a given node (= #incoming links)
  - Out-degrees: #links from a given node (= #out-going links)
  - In G3, Indegree(1) = 1, Outdegree(1) = 2



G3

Representing a Graph: Adjacent Matrix

- For an adjacent matrix M,
  - M[i][j] = 1 if there is a link from node i to node j
  - M[i][j] = 0 otherwise
- Symmetric when undirected graph



	0	1	2	3
0	0	1	1	1
1	1	0	1	0
2	1	1	0	1
3	1	0	1	0

(a)

	0	1	2	3
0	0	1	1	0
1	1	0	0	0
2	1	0	0	0
3	0	0	0	0

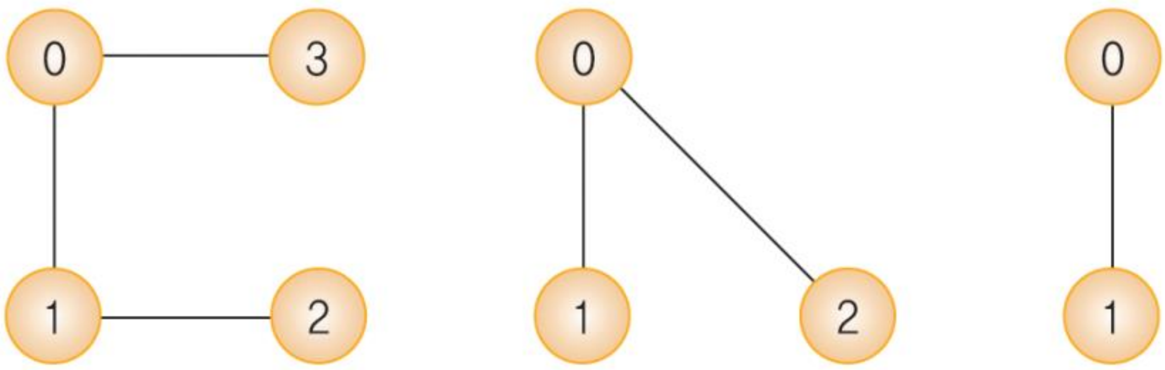
(b)

	0	1	2
0	0	1	0
1	1	0	1
2	0	0	0

(c)

Revisit: Tree

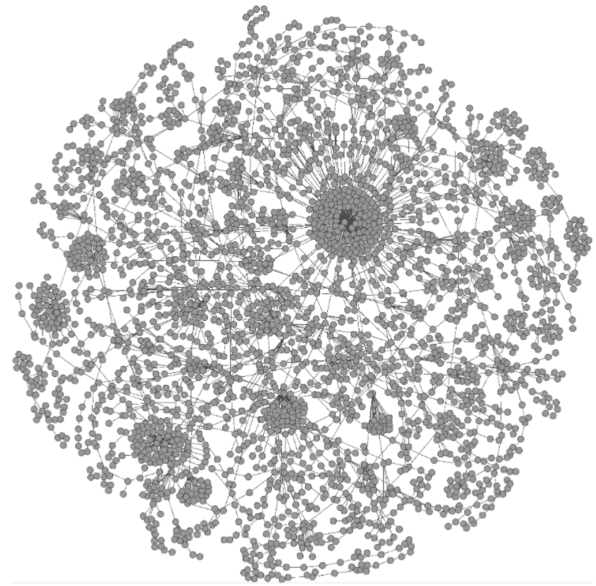
- A type of graph satisfying the following conditions
  - All nodes are connected (i.e., no isolated nodes)
  - No cycles



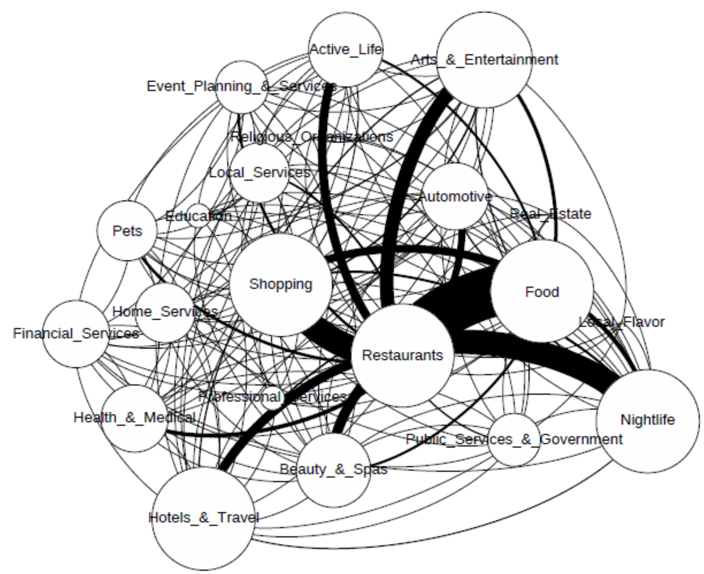
Tools for Graph Modeling & Analysis

Tools

- Graph analysis
  - igraph (Fast!)
  - NetworkX
  - Pajek
  - …



- Network visualization tools (valid for small network)
  - Gephi
  - Graphviz
  - Pajek
  - …



iGraph Example

```
import igraph
from igraph import Graph

if __name__ == '__main__':
  g = Graph.Read_Ncol('test1.csv', weights=True, directed=False)

  degrees = g.degree()

  node_count = len(g.vs)
  edge_count = len(g.es)

  print 'Number of nodes = ', node_count
  print 'Number of edges = ', edge_count

  print 'len(degrees) = ', len(degrees)
  print 'Avg.Degree = ', float(sum(degrees))/float(len(degrees))
  print degrees

  cc =  g.transitivity_undirected()
  print 'Avg.CC = ', cc

 # motif = g.motifs_randesu(3, None, None)
 # print 'Number of 3-motifs = ', len(motif)
```

Network Properties:
A first measure for graph

Overview

- Assume that you have modeled a graph, what can you do next?
- You will have to measure "coarse-grained characteristics" of a graph!
  - Can be done with network properties

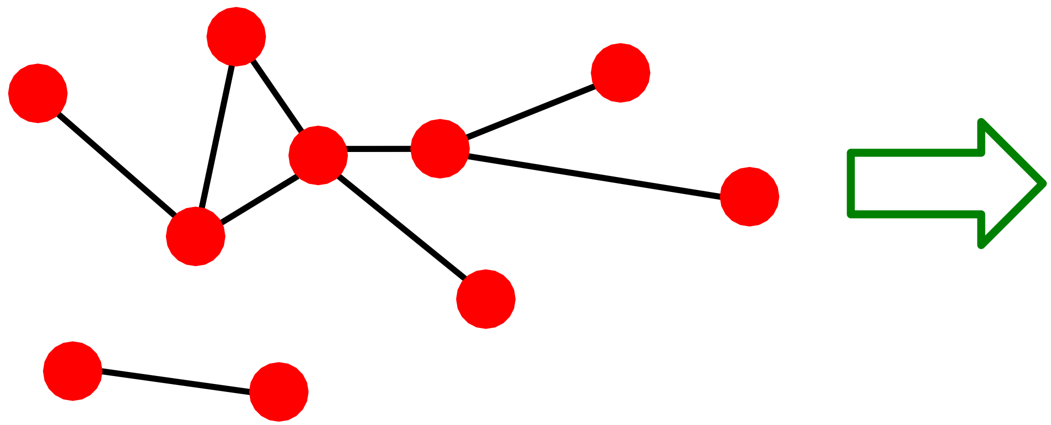
Degree distribution:	$P(k)$
Path length:	h
Clustering coefficient:	C
Connected components:	s

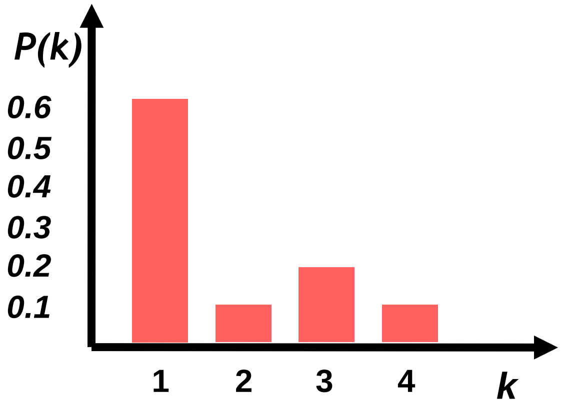
Definitions will be presented for undirected graphs, sometimes we will explicitly mention extensions to directed graphs, and sometimes extensions will be obvious

Network Properties: Degree Distribution

- Degree distribution $P(k)$: Probability that a randomly chosen node has degree k
  - N_k = # nodes with degree k

- Normalized histogram:
  - $P(k) = N_k / N$





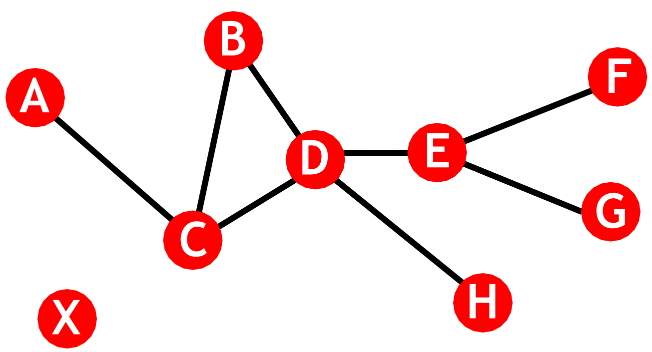
For directed graphs we have separate in- and out-degree distributions.

Paths in a Graph

- A **path** is a sequence of nodes in which each node is linked to the next one

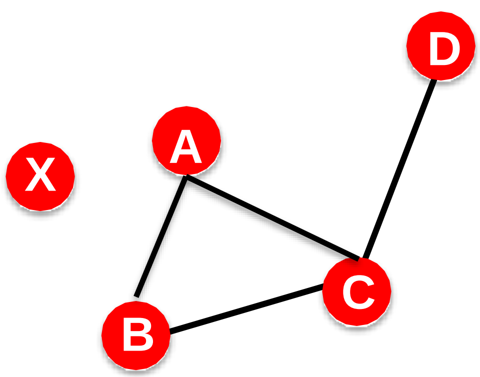
$$P_n = \{i_0, i_1, i_2, \ldots, i_n\} \qquad P_n = \{(i_0, i_1), (i_1, i_2), (i_2, i_3), \ldots, (i_{n-1}, i_n)\}$$

- NOTE: A path can intersect itself and pass through the same edge multiple times
  - i.e., the path length between two nodes can be infinite
  - E.g.: ACBD**CD**EG

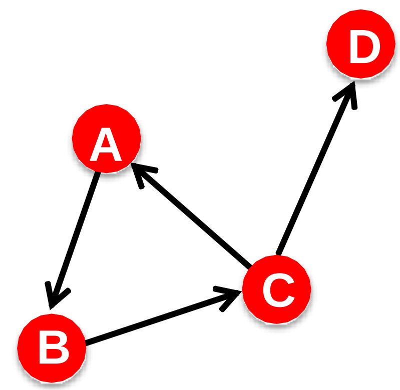


Network Properties: Distance (Shortest Path Length)

- The number of edges along the shortest path connecting the nodes
  - If the two nodes are not connected, the distance is usually defined as infinite (or zero)



$h_{B,D} = 2$, $h_{A,X} = \infty$



$h_{B,C} = 1$, $h_{C,B} = 2$

- In directed graphs, paths need to follow the direction of the arrows
  - i.e., Distance is not symmetric: $h_{B,C} \neq h_{C,B}$

Network Properties: Diameter & APL

- **Diameter**: The maximum (shortest path) distance between any pair of nodes in a graph

- **Average path length** (for a **connected** graph or a **strongly connected** directed graph)

$$\bar{h} = \frac{1}{2E_{\max}} \sum_{i,\, j \neq i} h_{ij}$$

- h_{ij} is the distance from node i to node j
- E_{max} is the max number of edges (total n umber of node pairs) = $n(n-1)/2$

- Many times we compute the average only over the connected pairs of nodes (that is, we ignore "infinite" length paths) → **Isolated nodes are not considered!**

Network Properties: Clustering Coefficient

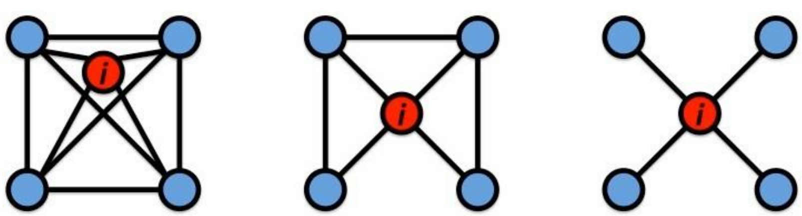
- Clustering coefficient (for undirected graphs):
  - How connected are ***i***’s neighbors to each other?
  - Node ***i*** with degree $\boldsymbol{k_i}$

$$C_i = \frac{2e_i}{k_i(k_i - 1)}$$

where e_i is the number of edges between the neighbors of node i

$$C_i \in [0,1]$$

$k_i(k_i - 1)$ is max number of edges between the k_i neighbors



$C_i = 1$ $\quad$ $C_i = 1/2$ $\quad$ $C_i = 0$

Clustering coefficient is undefined (or defined to be 0) for nodes with degree 0 or 1

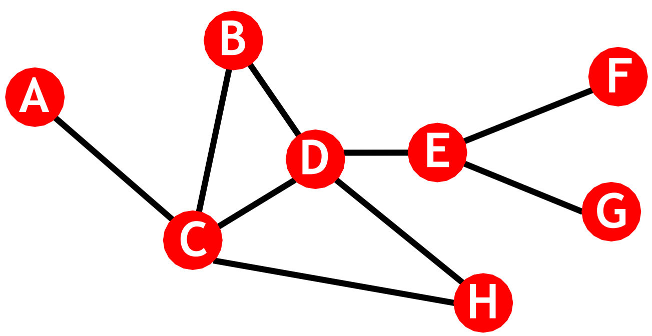
- Average clustering coefficient: $C = \frac{1}{N}\sum_{i}^{N} C_i$

Computing Clustering Coefficient

- Clustering coefficient (for undirected graphs):
  - How connected are ***i***'s neighbors to each other?
  - Node ***i*** with degree $\boldsymbol{k_i}$

$$C_i = \frac{2e_i}{k_i(k_i-1)}$$

where e_i is the number of edges between the neighbors of node i



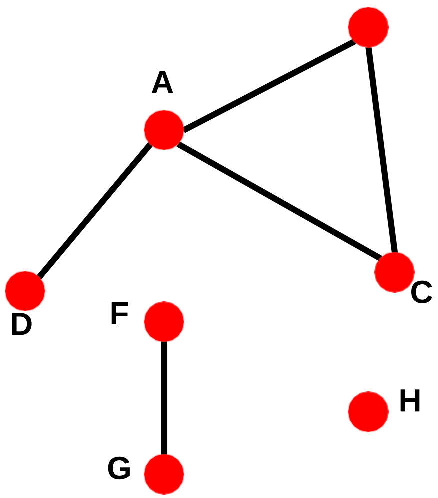
$k_B=2,\ e_B=1,\ C_B=2/2=1$

$k_D=4,\ e_D=2,\ C_D=4/12=1/3$

Avg. clustering: $C=0.33$

Network Properties: Connectivity

- Size of the largest connected component
  - Largest set where any two vertices can be joined by a path



- Finding connected components:
  - Start from random node and perform Breadth First Search (BFS)
  - Label the nodes that BFS visits
  - If all nodes are visited, the network is connected
  - Otherwise find an unvisited node and repeat BFS

Summary: Key Network Properties

Degree distribution:	$P(k)$
Path length:	h
Clustering coefficient:	C
Connected components:	s

Network Properties in Real-world Networks

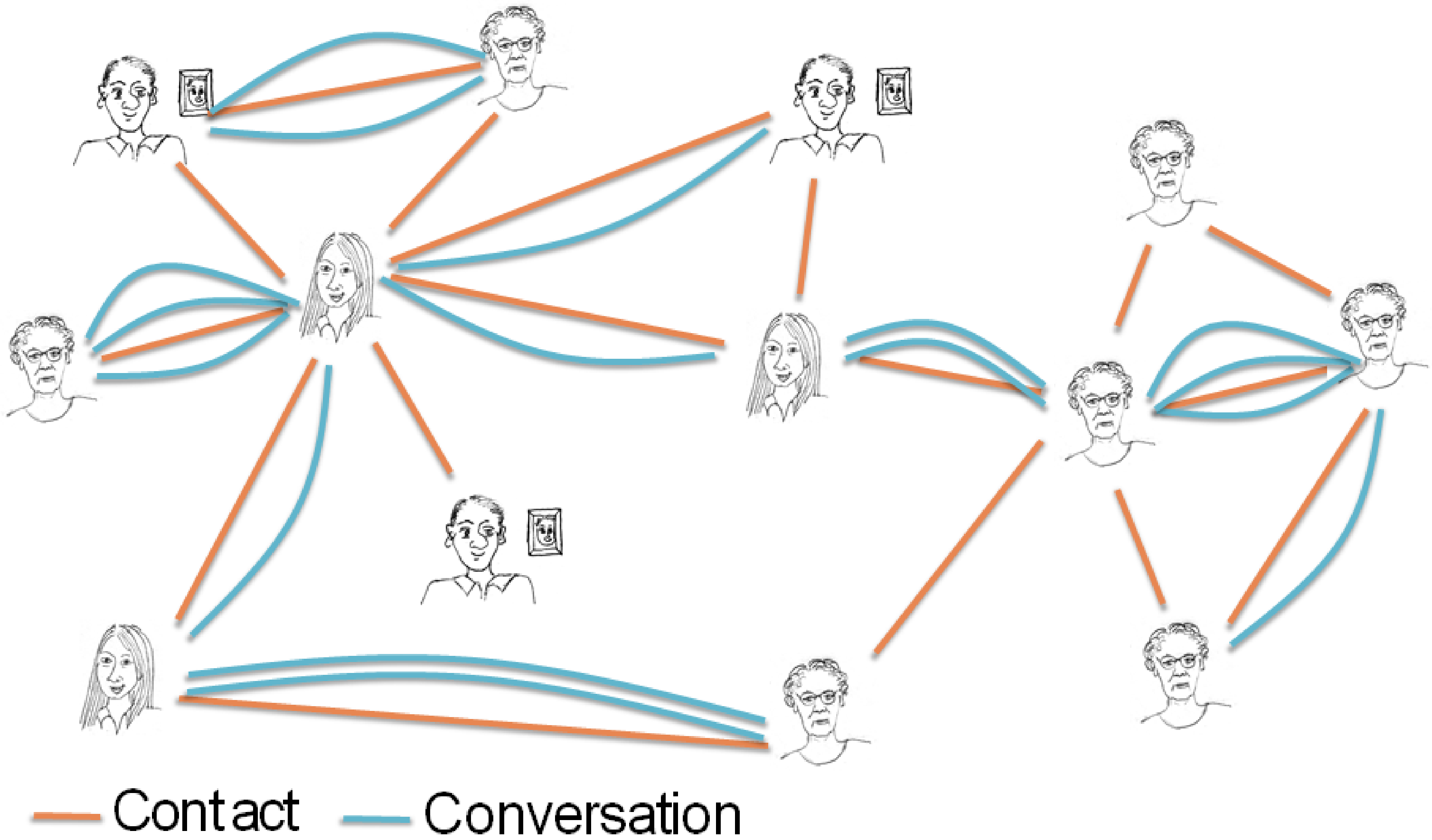
MSN Messenger

- MSN Messenger: 1 month of activity

- 245 million users logged in
- 180 million users engaged in conversations
- More than 30 billion conversations
- More than 255 billion exchanged messages

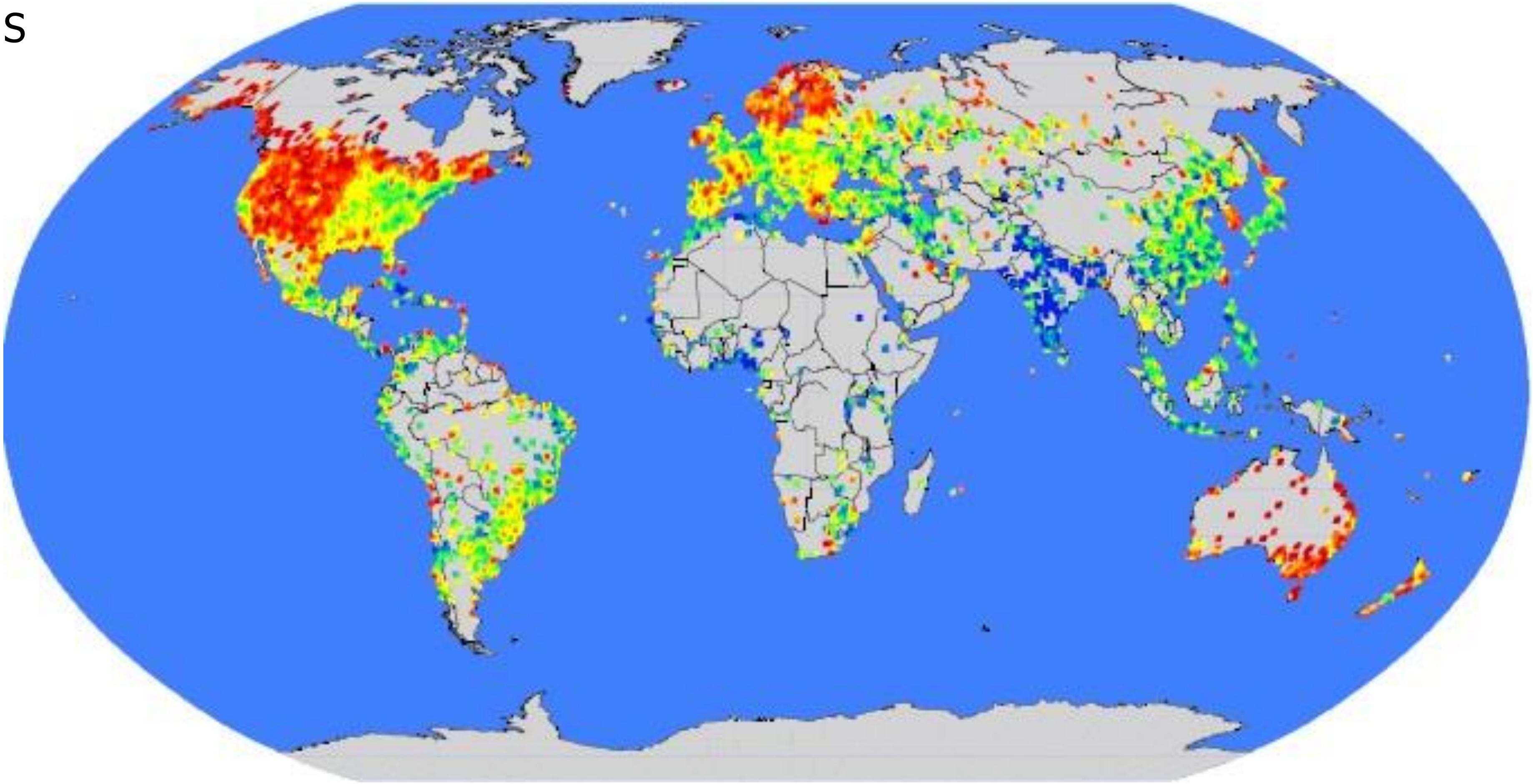
Modeling Interactions as a Graph

- Messaging as an undirected graph
  - Edge (u,v) if users u and v exchanged at least 1 msg



Geography of Communication

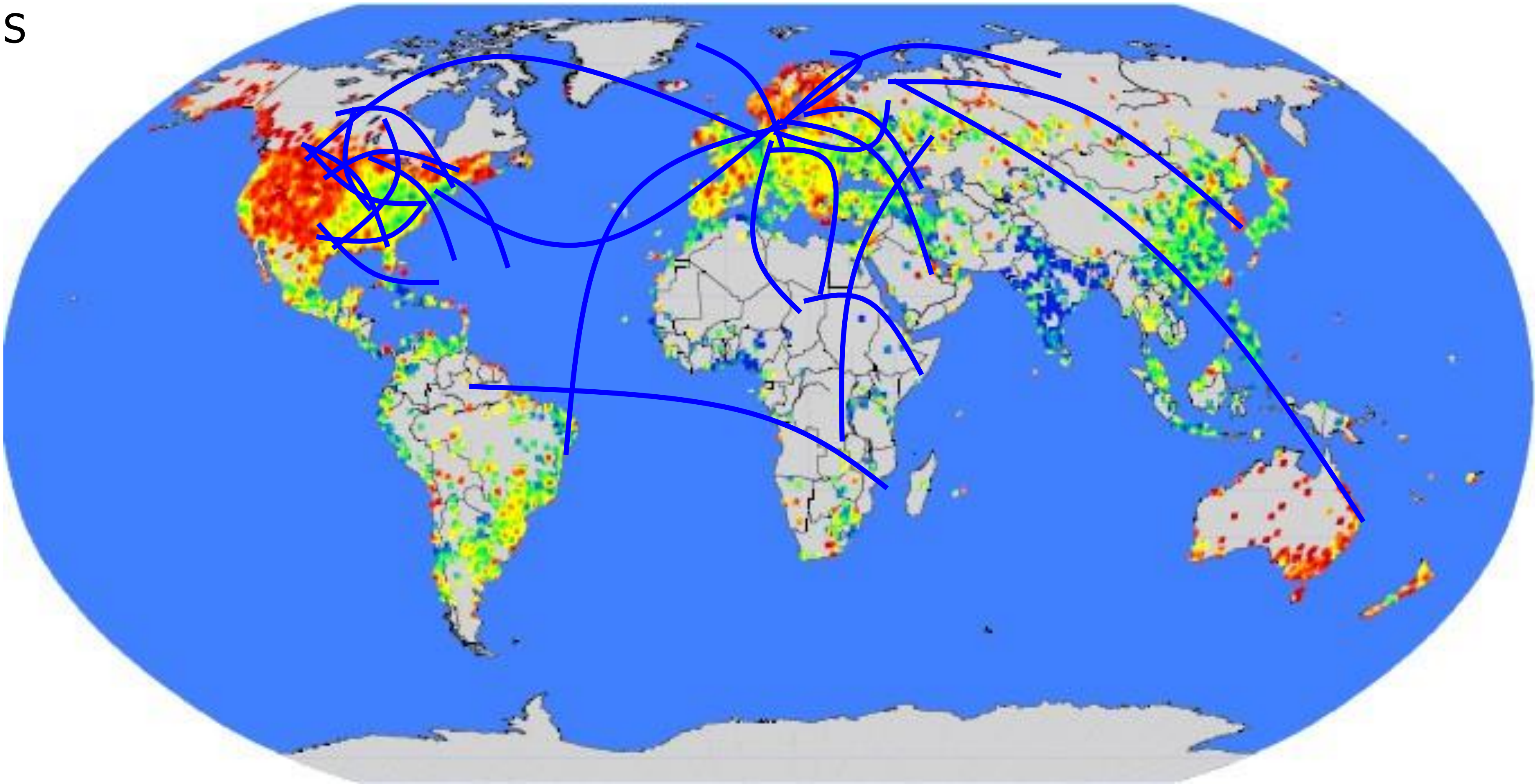
S



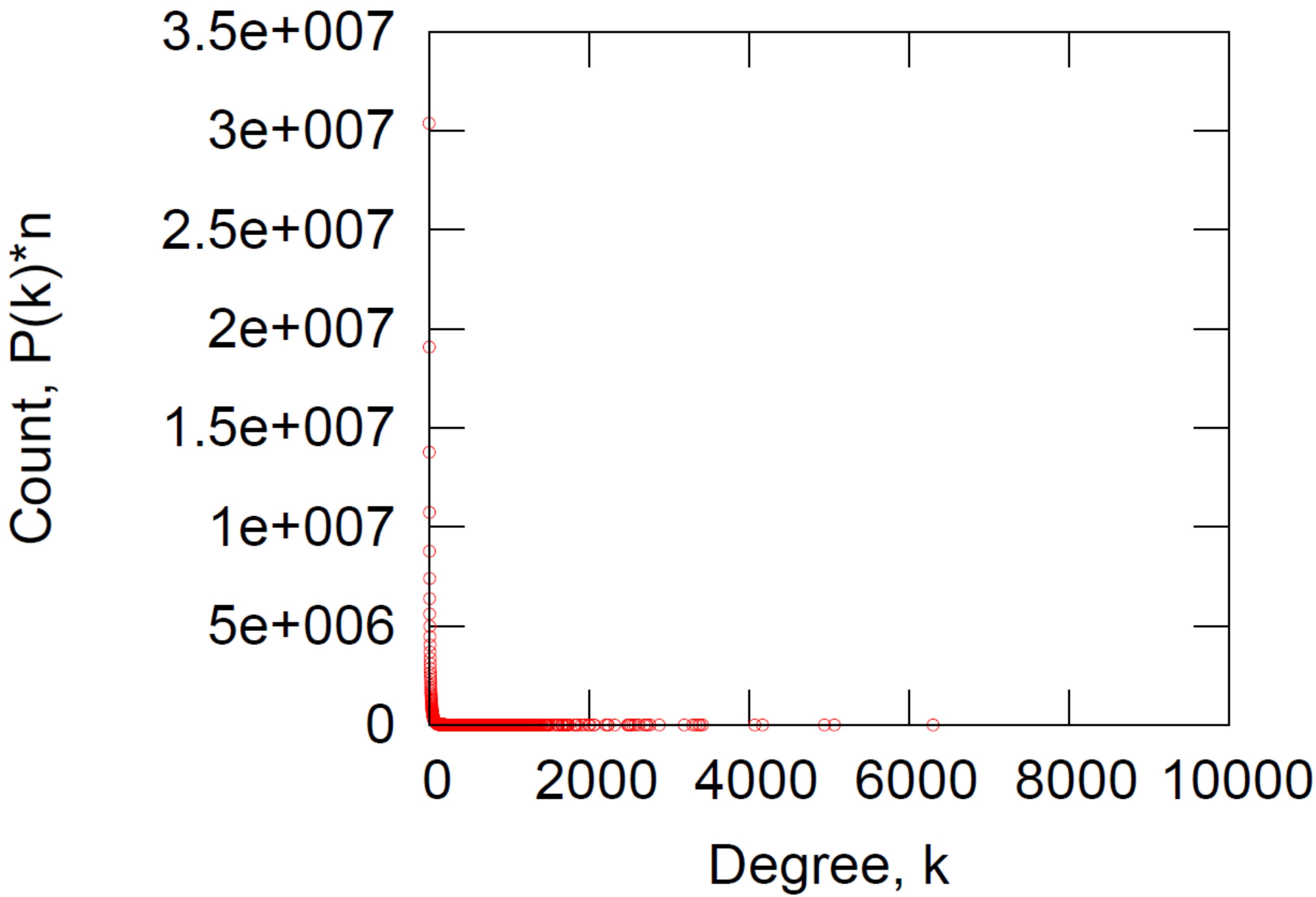
Geography of Communication

- Network: 180M people, 1.3B edges

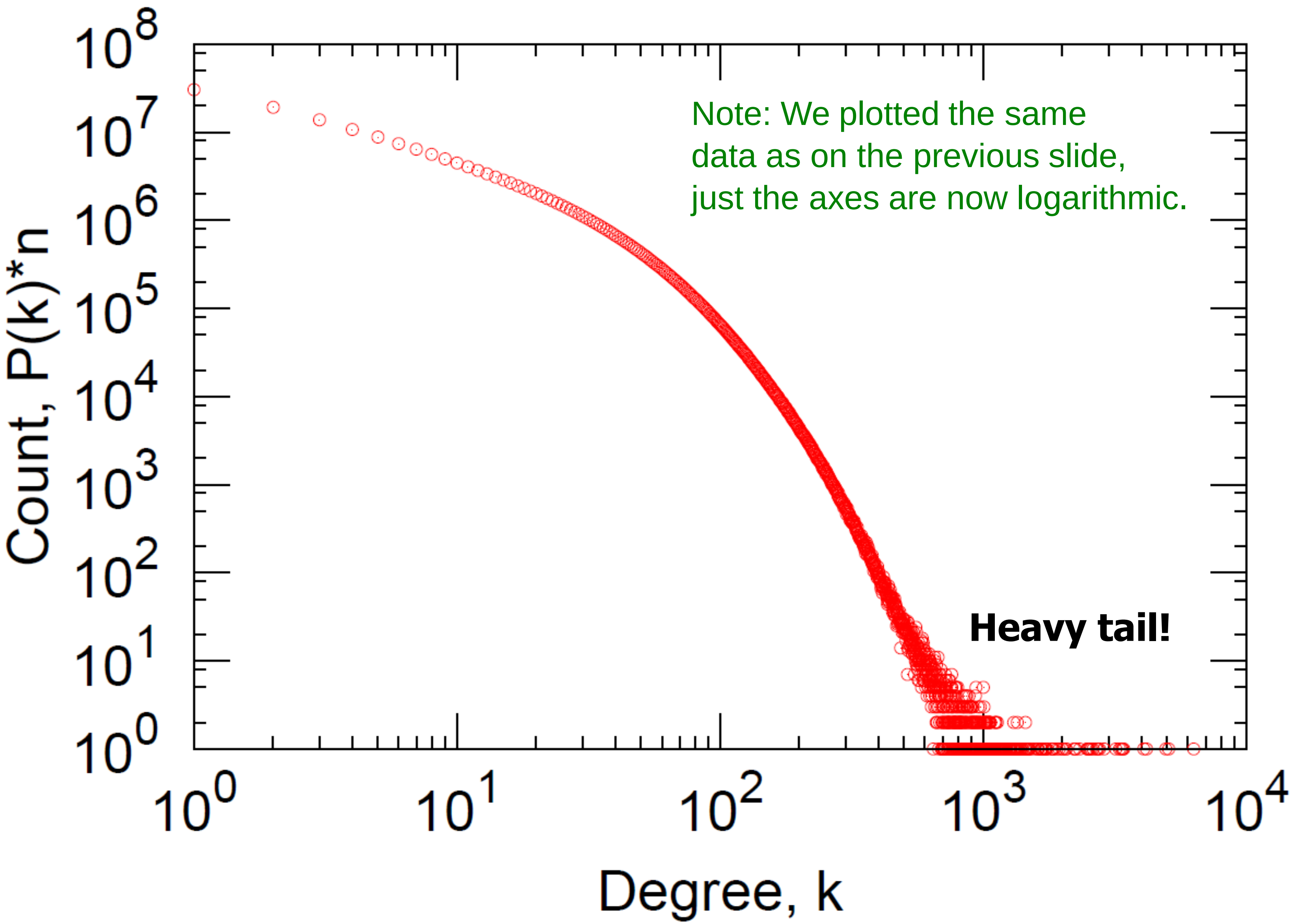
S



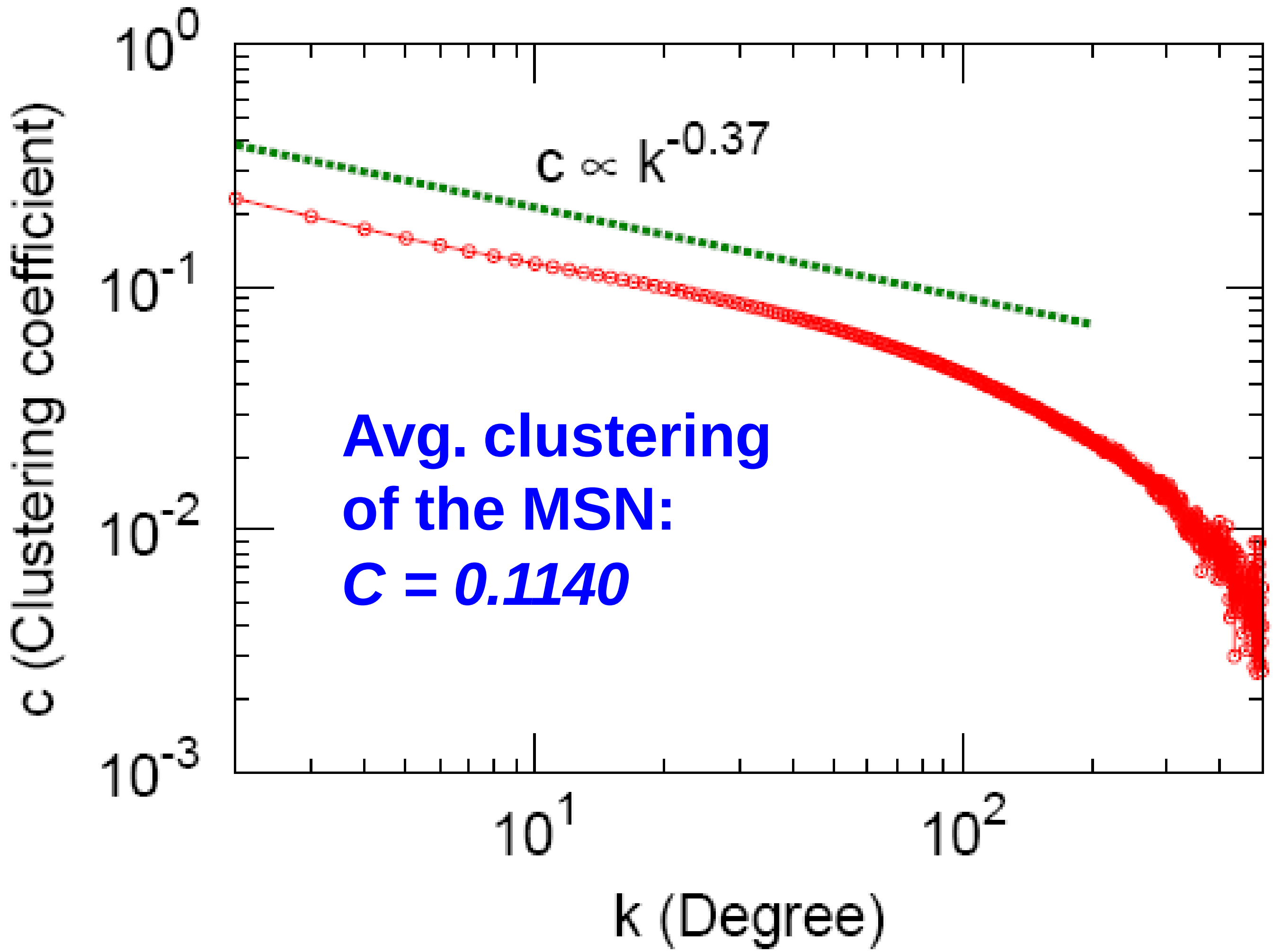
Network Properties: Degree Distribution



Degree Distribution: Log-Log Scale

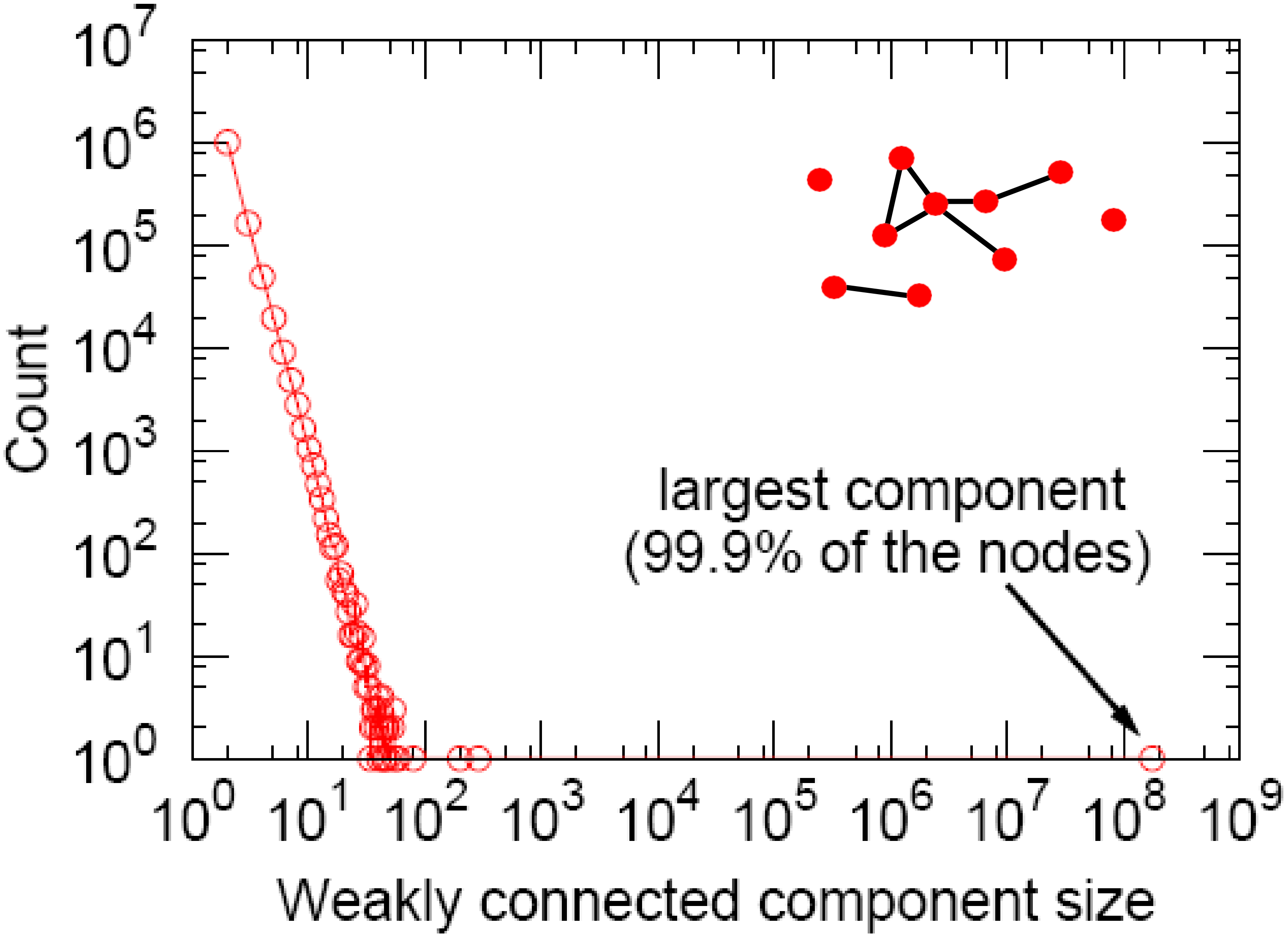


Network Properties: Clustering Coefficient

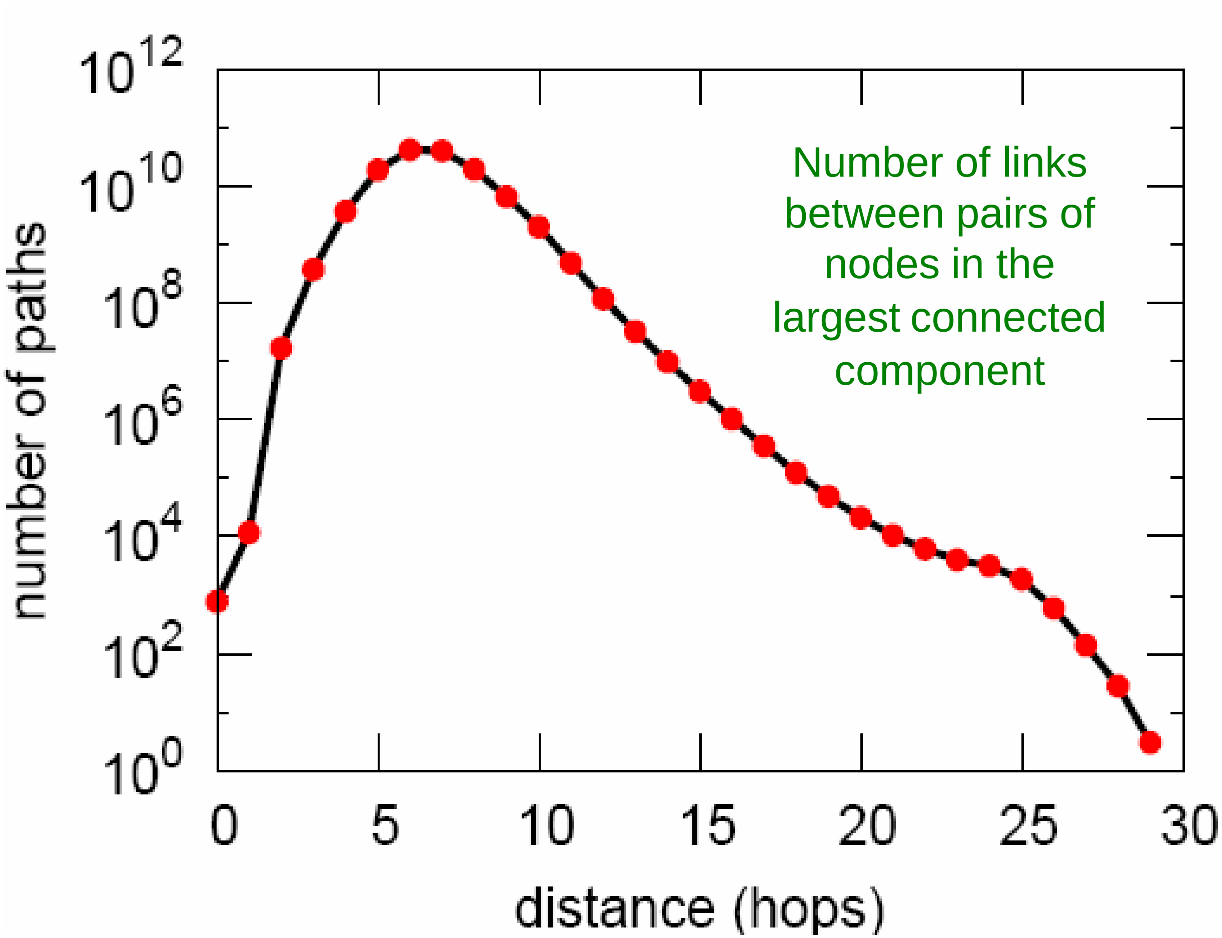


C_k: average C_i of nodes i of degree k: $C_k = \frac{1}{N_k} \sum_{i:k_i=k} C_i$

Network Properties: Connected Components



Network Properties: Diameter of WCC



Avg. path length **6.6**

90% of the nodes can be reached in < 8 hops

\# nodes as we do BFS out of a random node

Steps	#Nodes
0	1
1	10
2	78
3	3,96
4	8,648
5	3,299,252
6	28,395,849
7	79,059,497
8	52,995,778
9	10,321,008
10	1,955,007
11	518,410
12	149,945
13	44,616
14	13,740
15	4,476
16	1,542
17	536
18	167
19	71
20	29
21	16
22	10
23	3
24	2
25	3

Summary: Properties of MSN Network

Degree distribution: *Heavily skewed; avg. degree = 14.4*

Path length: *6.6*

Clustering coefficient: *0.11*

Connected components: *giant component*

Are these values "expected"?
Are they "surprising"?

Let's generate **a random model** and compare!

Thank you!

Instructor: Daejin Choi (djchoi@inu.ac.kr)