Machine Learning meets Networks

Jure Leskovec (@jure)
Stanford University

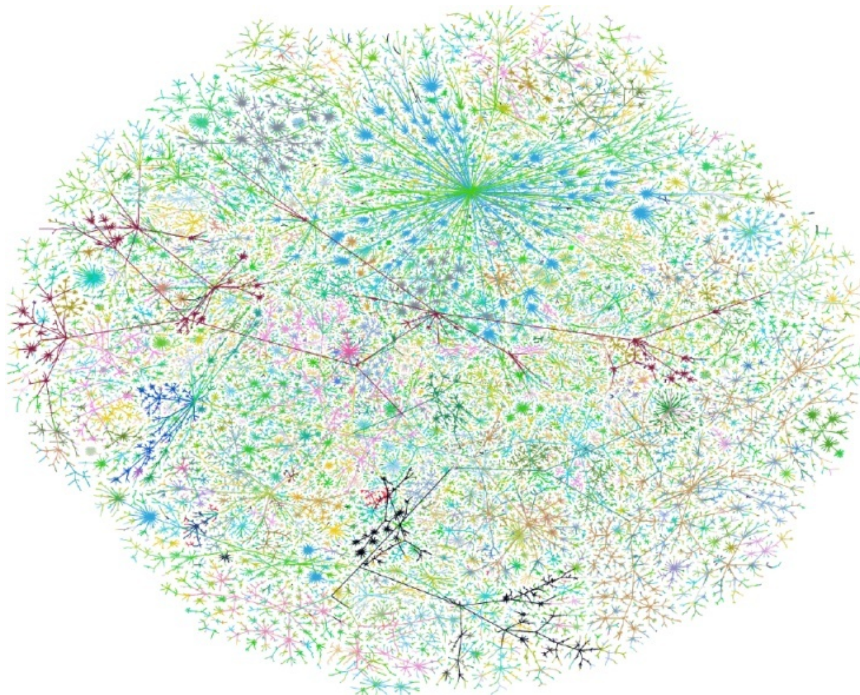
ML & Networks

- Machine Learning has rich history and methods for analyzing …
  - … tabular data
  - … textual data
  - … time series & streams
  - … market baskets

Bag of features

- What about relations and dependencies?

Network: A First Class Citizen



Tabular data:
Node / edge attributes

Time series:
Evolving network

Networks allow for modeling dependencies between parts!

Networks

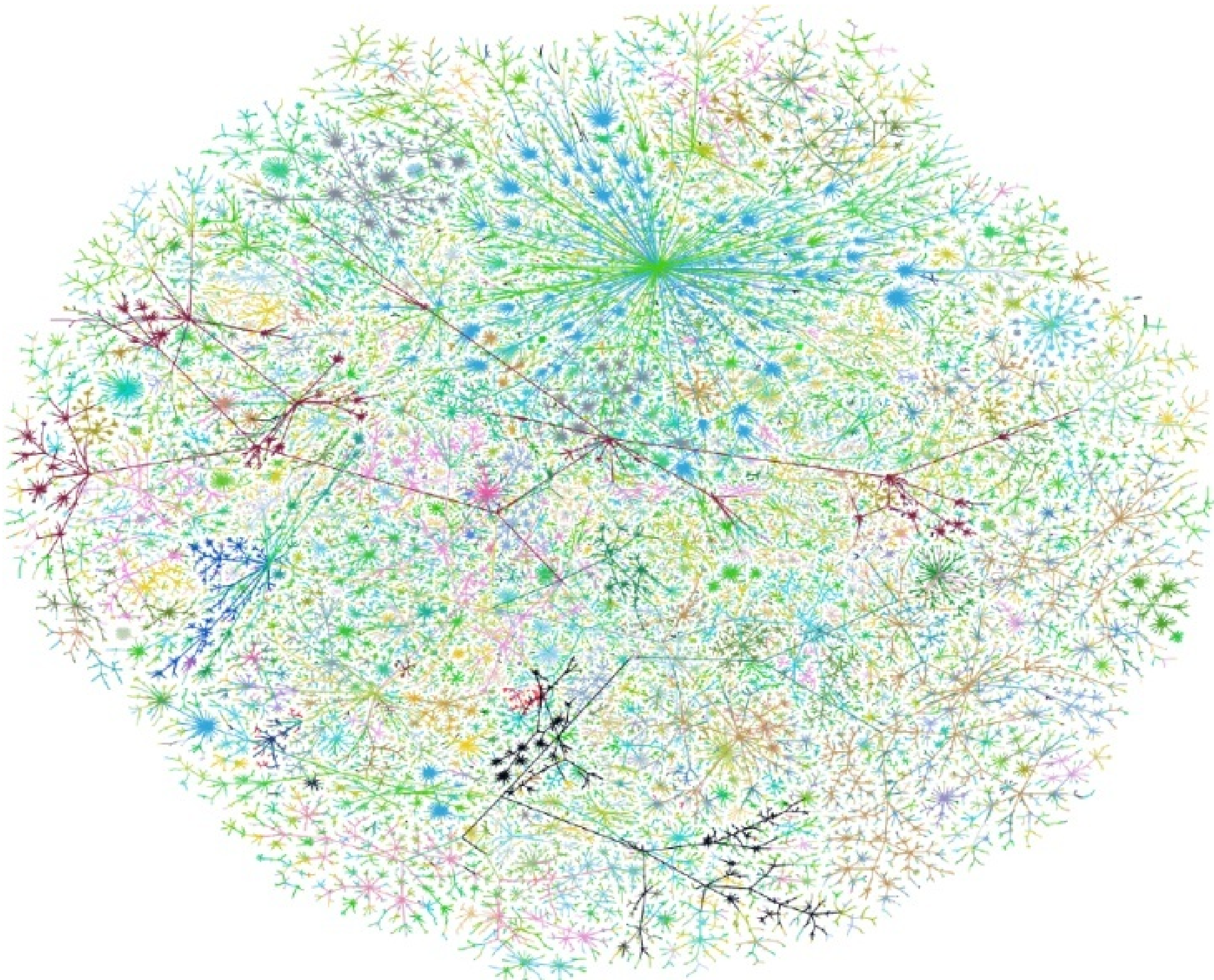
…are a general modeling language for complex data

Networks: Social

Facebook social graph

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

Networks: Communication

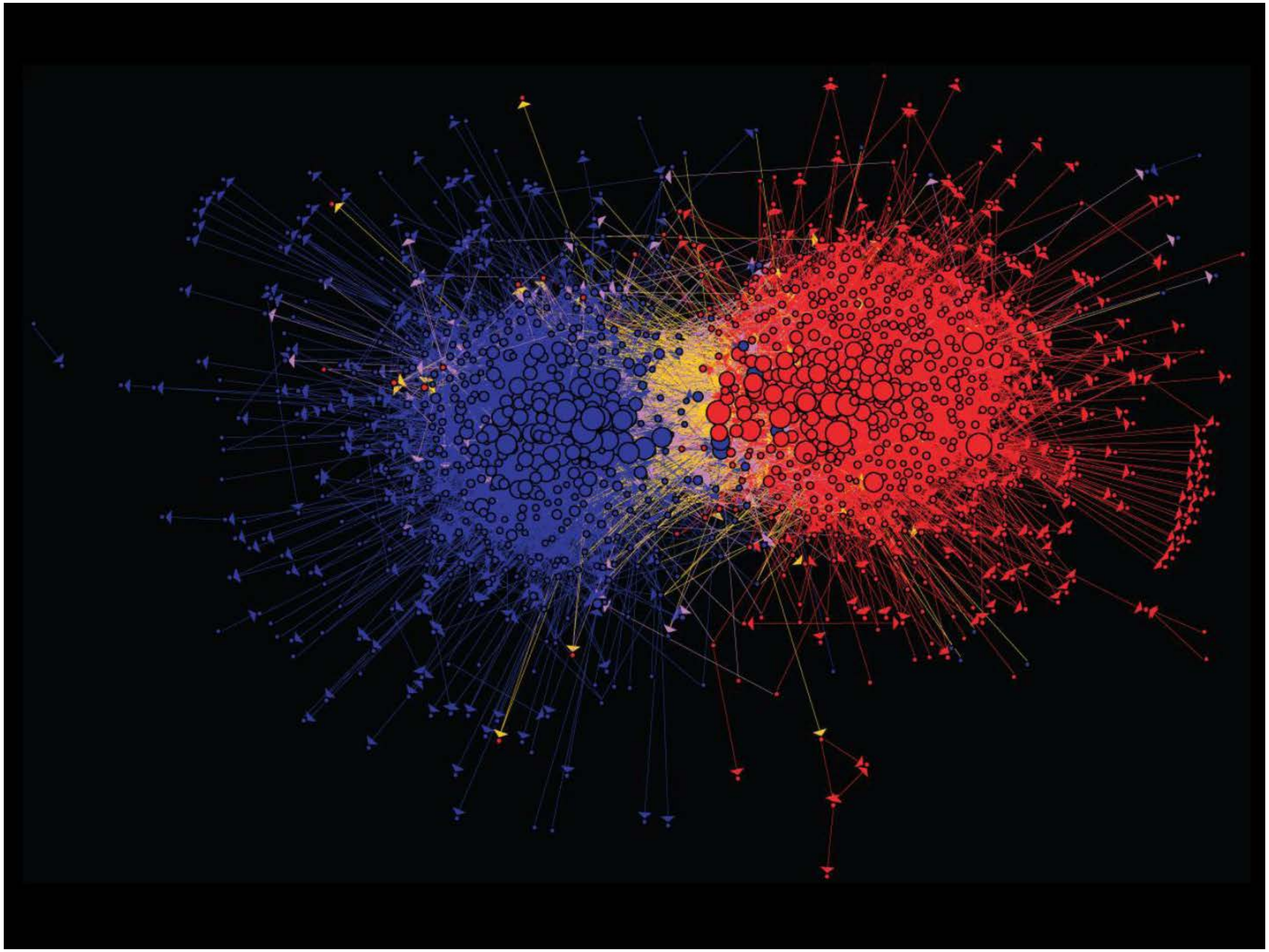


Graph of the Internet (Autonomous Systems)

Power-law degrees [Faloutsos-Faloutsos-Faloutsos, 1999]

Robustness [Doyle-Willinger, 2005]

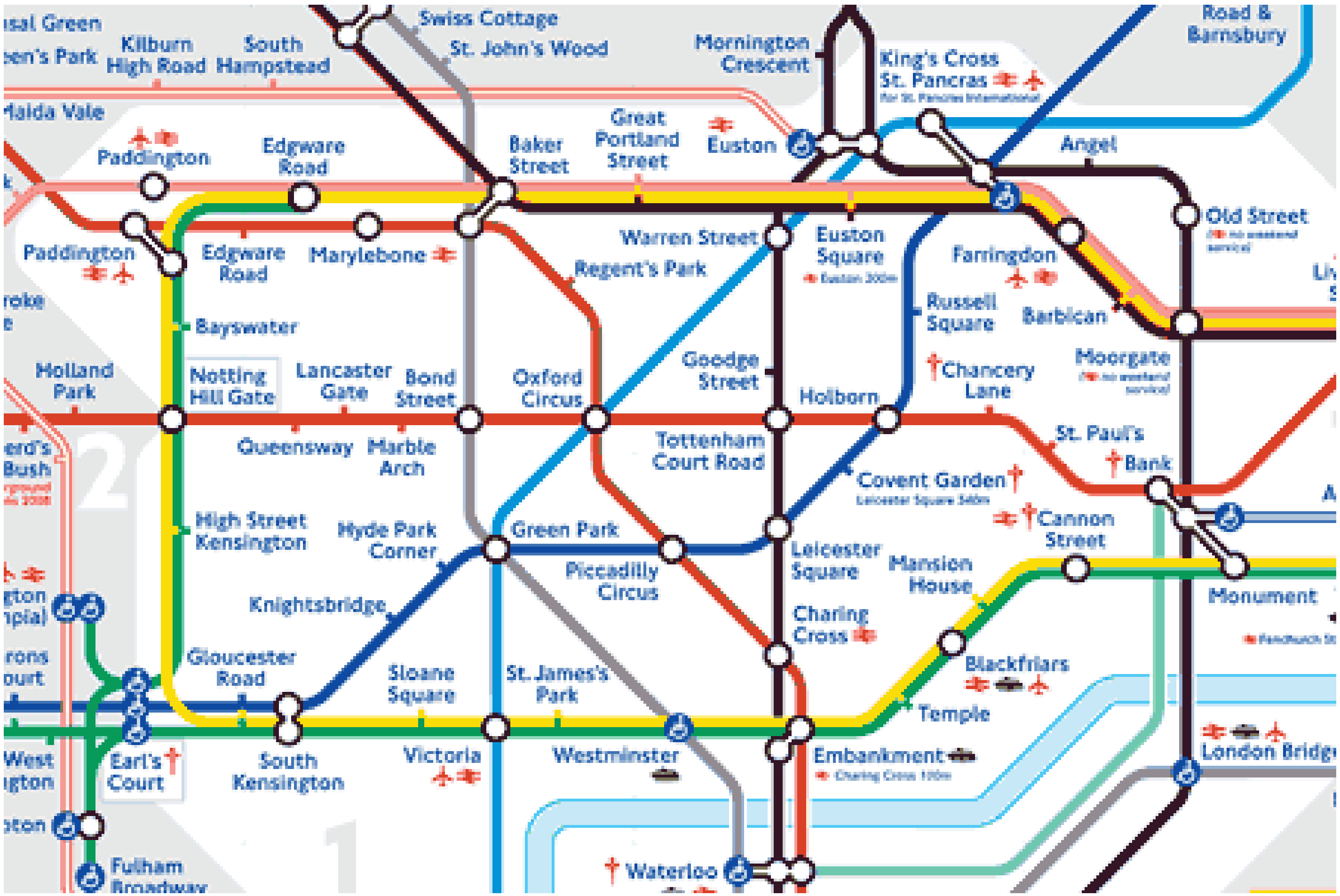
Networks: Media



Connections between political blogs

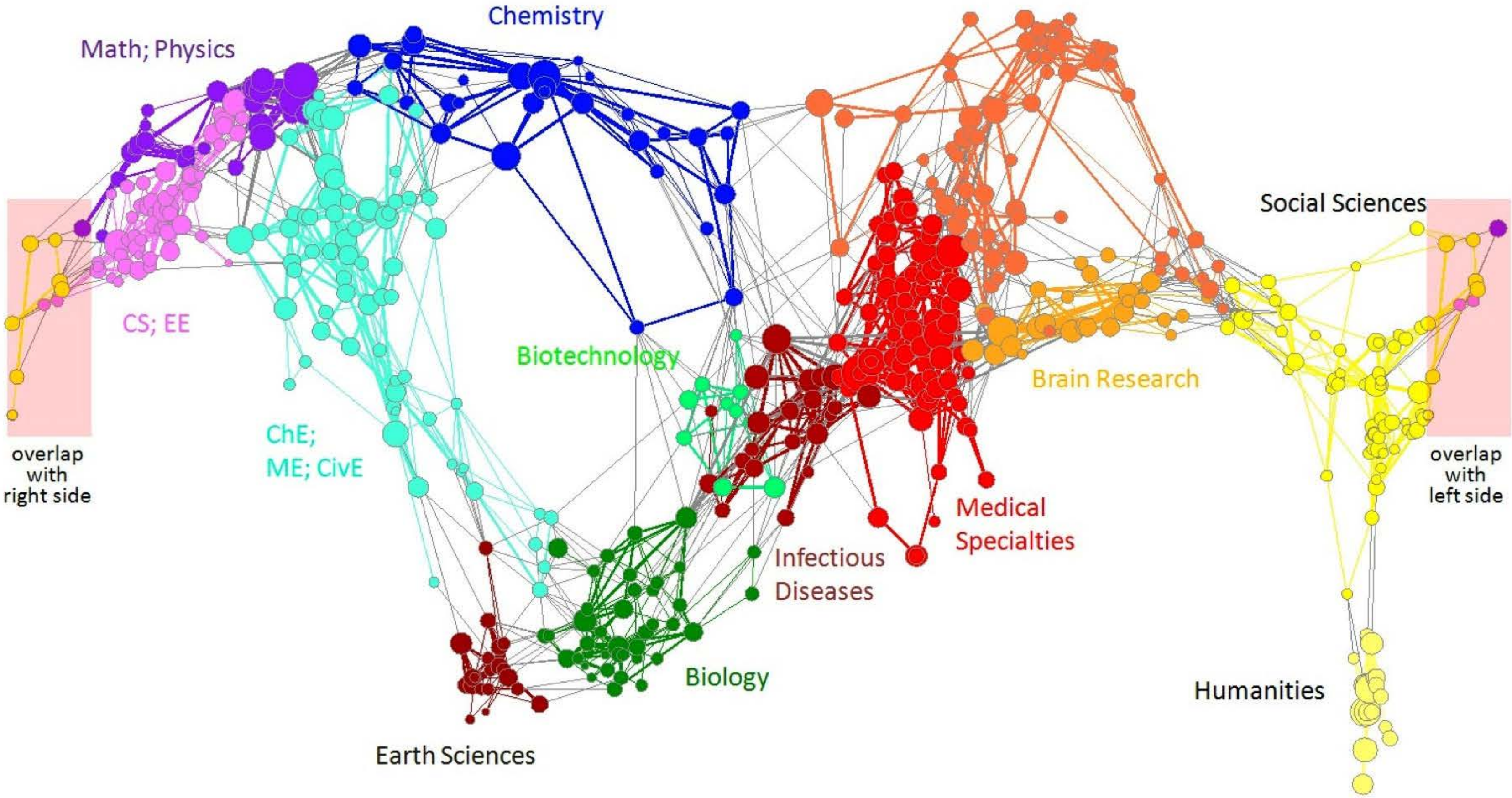
Polarization of the network [Adamic-Glance, 2005]

Networks: Infrastructure



Infrastructure and technological networks

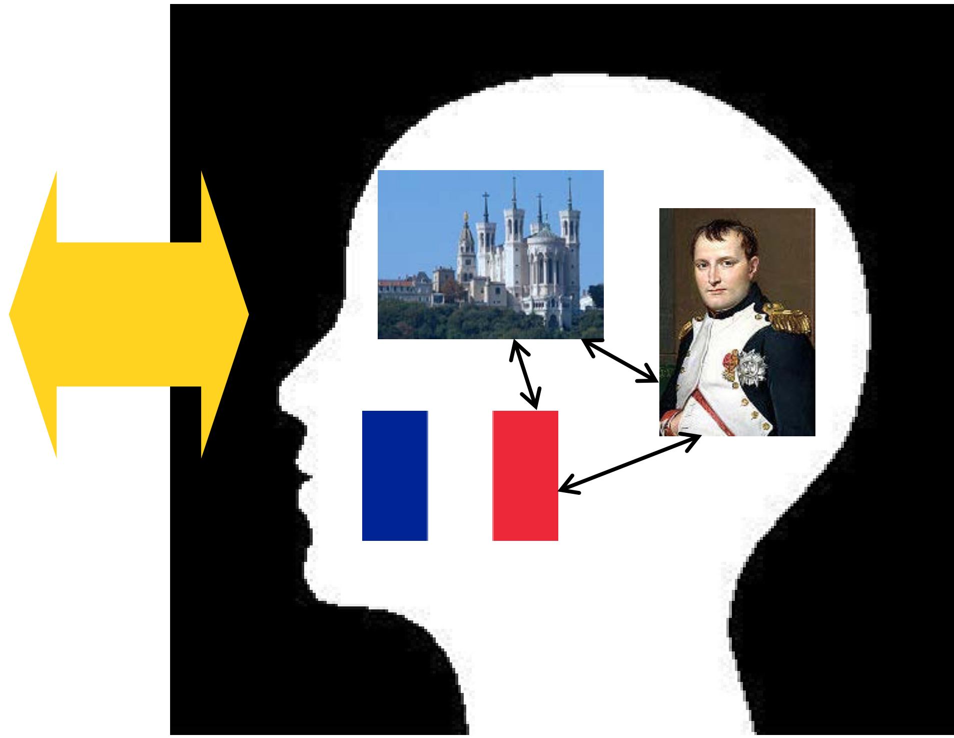
Networks: Information



Citation networks and Maps of science

[Börner et al., 2012]

Networks: Knowledge

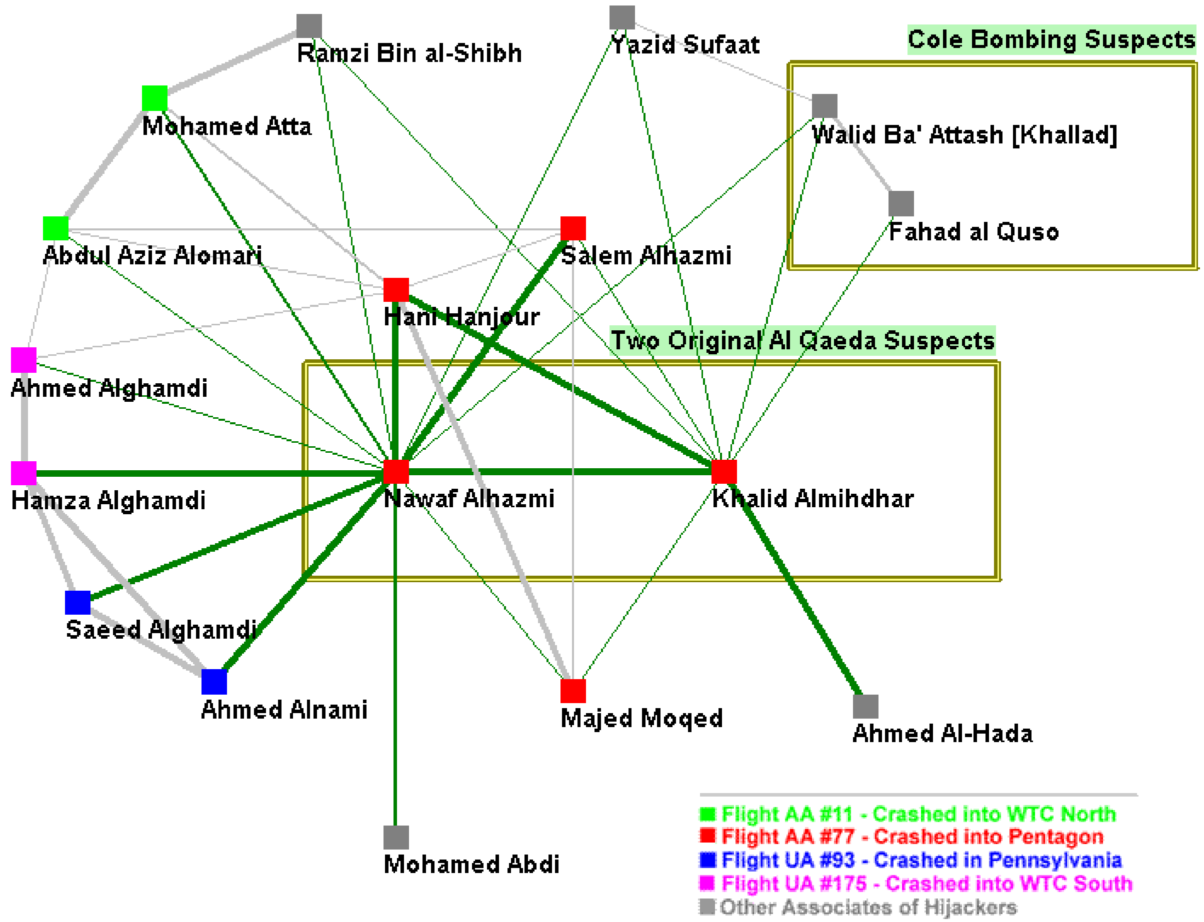


Understand how humans navigate Wikipedia

Get an idea of how people connect concepts

[West-Leskovec, 2012]

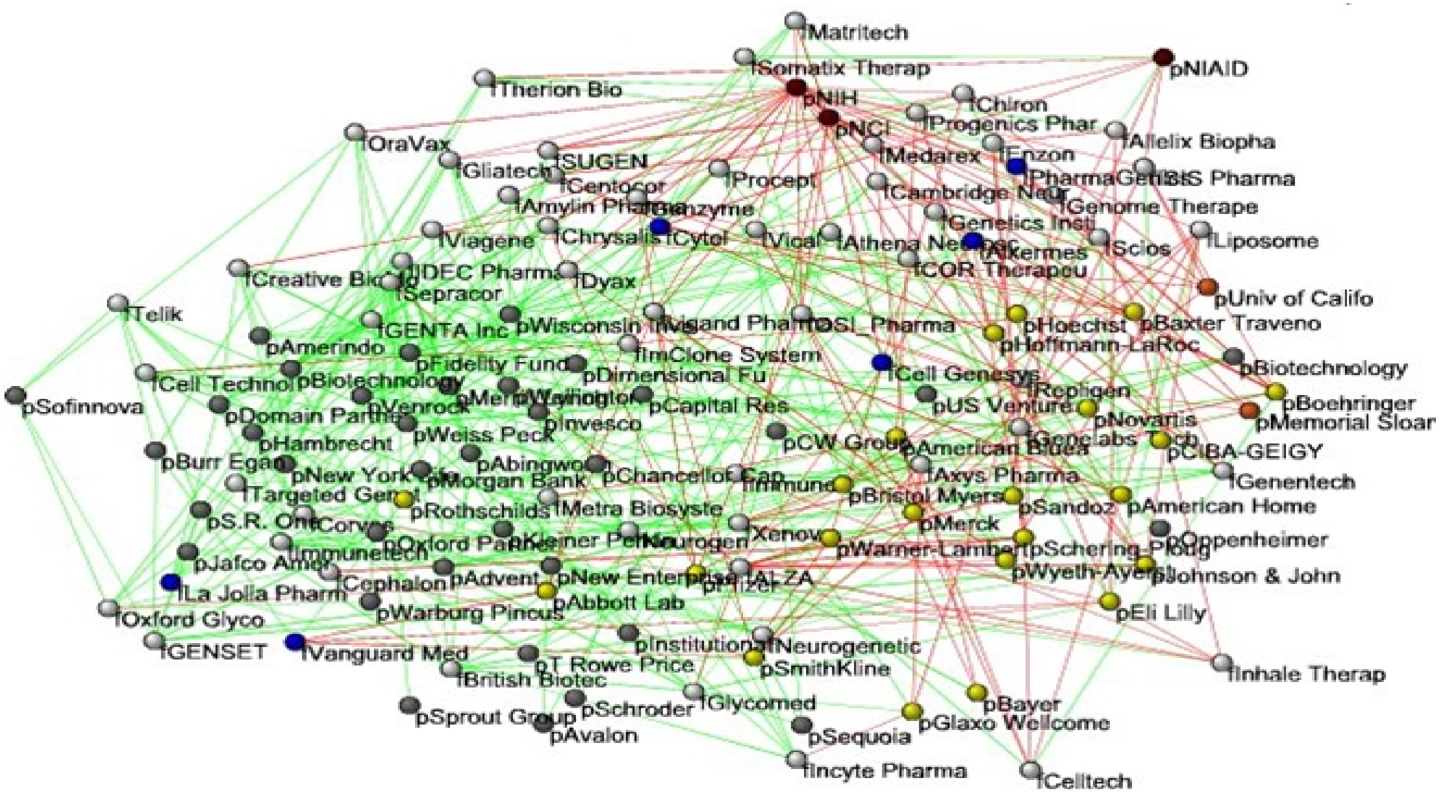
Networks: Organizations



9/11 terrorist network

[Krebs, 2002]

Networks: Economy



Nodes:

- Companies
- Investment
- Pharma
- Research Labs
- Public
- Biotechnology

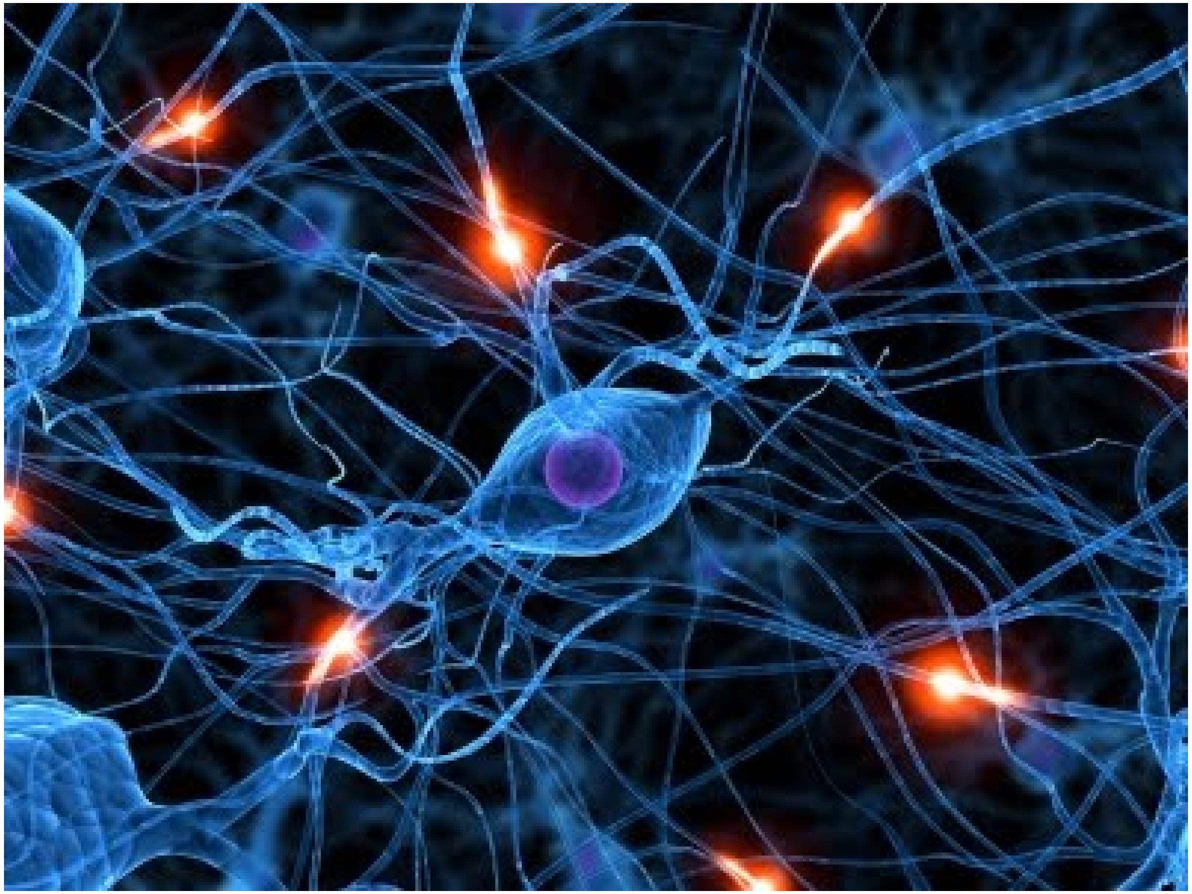
Links:

- Collaborations
- Financial
- R&D

Bio-tech companies

[Powell-White-Koput, 2002]

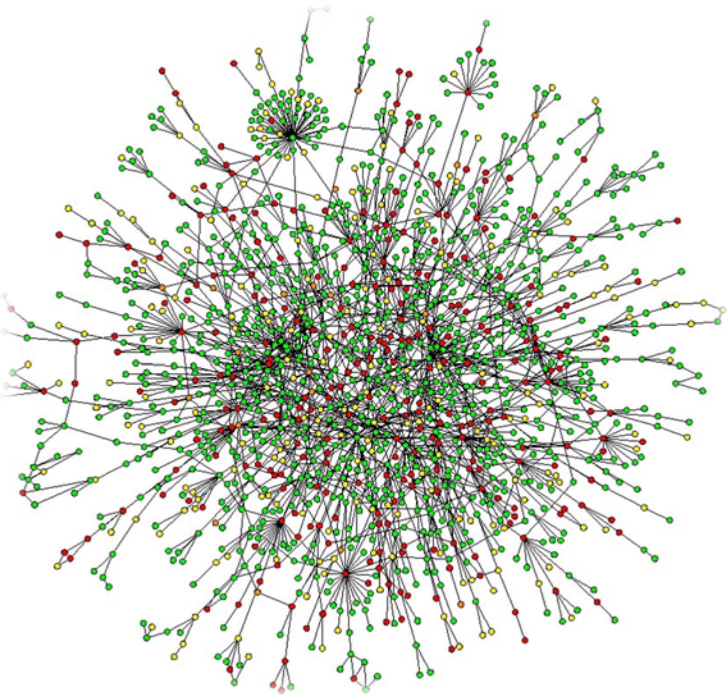
Networks: Brain

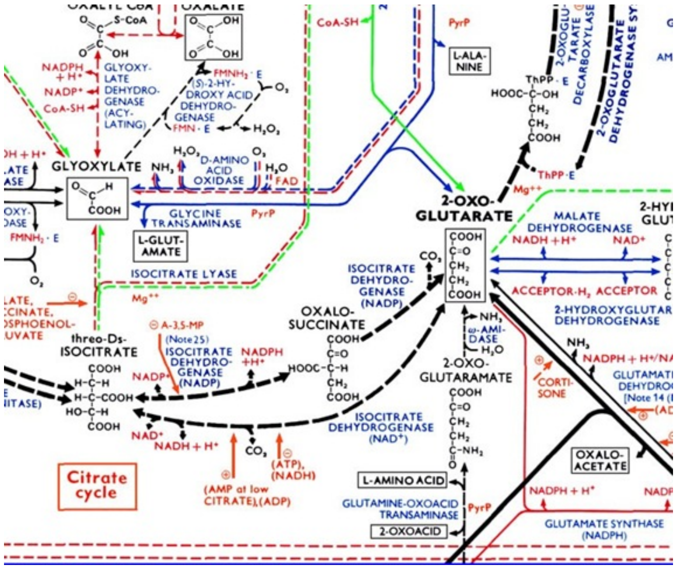


Human brain has between 10-100 billion neurons

[Sporns, 2011]

Networks: Biology



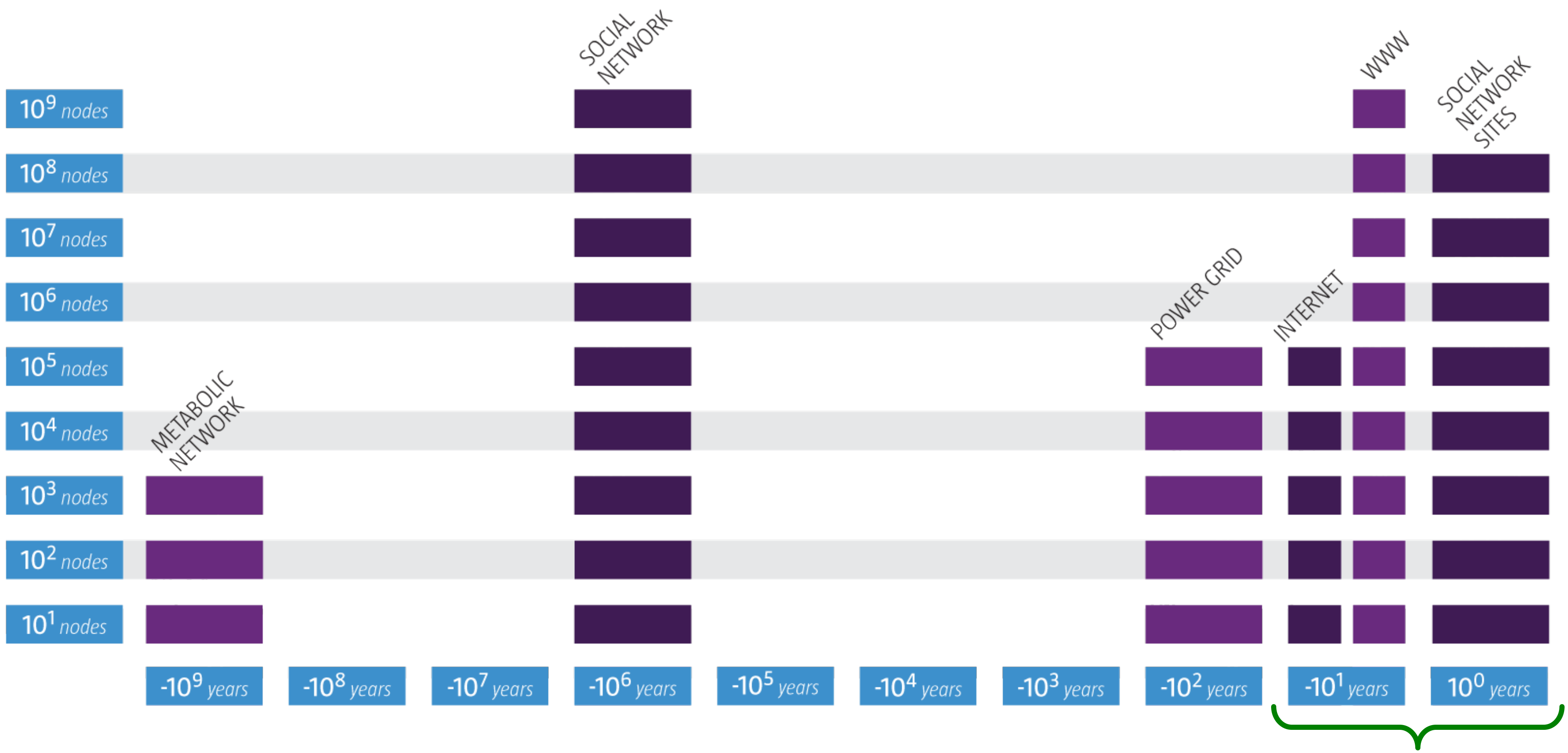


Protein-Protein Interaction Networks:
Nodes: Proteins
Edges: 'physical' interactions

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

But Jure, why should I care about networks?

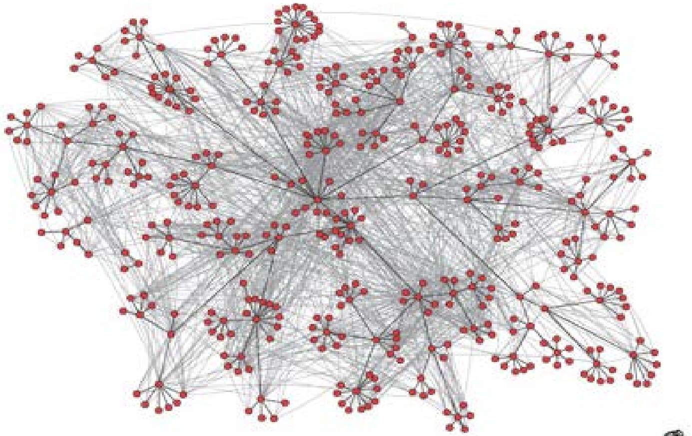
Networks: Why Now?



Age and size of networks

Transformation of Humanity

Online friendships
[Ugander-Karrer-Backstrom-Marlow, '11]



Corporate e-mail communication
[Adamic-Adar, '05]

- **Web: a Social and a Technological network**
- **Profound transformation of humanity:**
  - How knowledge is produced and shared
  - How people interact and communicate

The Internet/Web turned CS into a natural science

The first computational artifact that was never designed, and hence must be approached by the *scientific method:*

- **Measurements**
- **Experiments**
- **Falsifiable theories**
- **Specialized applied mathematics**

… and a social science

The Internet/Web cannot be studied in isolation from the complex social system it enables and serves

Web is an ideal test bed for sociological analysis and experimentation

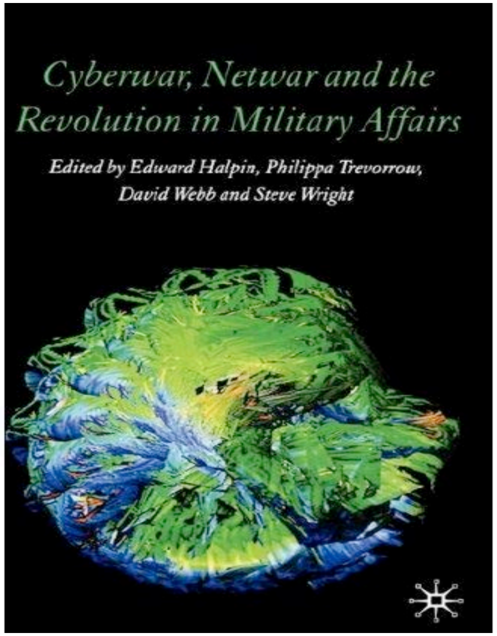
Networks: Impact

licensed under (cc) Attribution-NonCommercial-ShareAlike 2.0 Germany | Ludwig Gatzke | http://flickr.com/photos/stabilo-boss/

- **Google**
  Market cap:
  $366 billion
  (1y ago it was 250b)
- **Cisco**
  Market cap:
  $130 billion
  (1y ago it was 100b)
- **Facebook**
  Market cap:
  $165 billion
  (1y ago it was 50b)

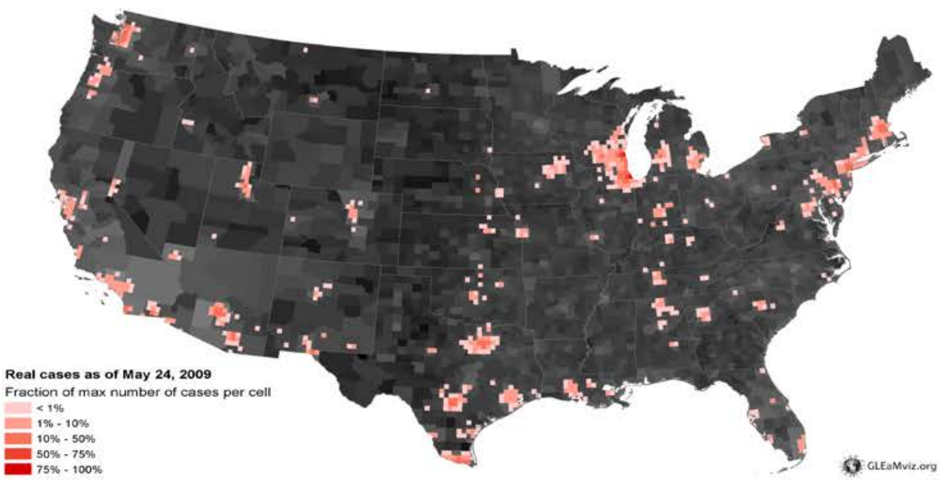
Networks: Impact

- **Intelligence and fighting (cyber) terrorism**

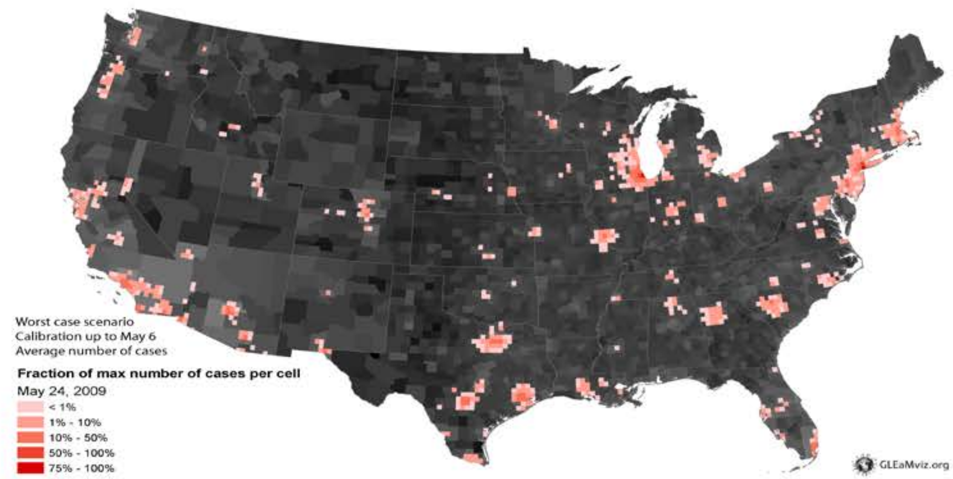


Networks: Impact

- **Predicting epidemics**



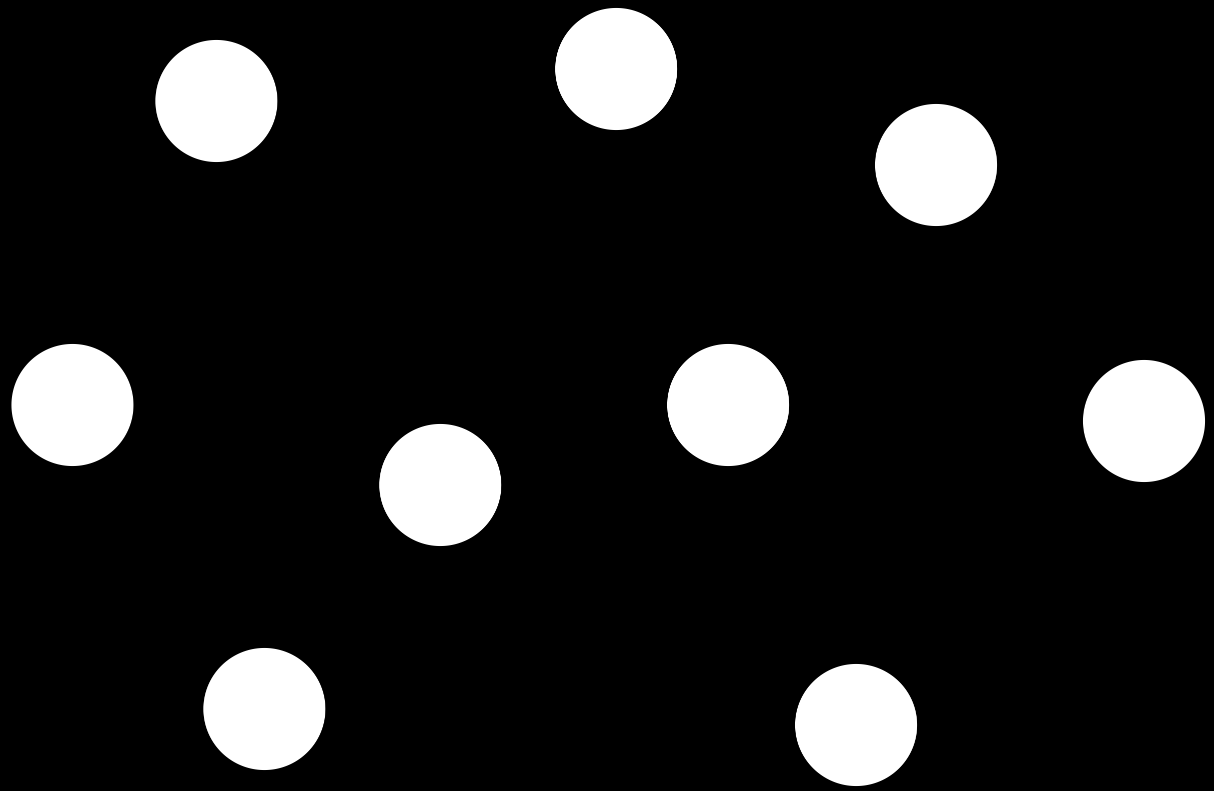
Real



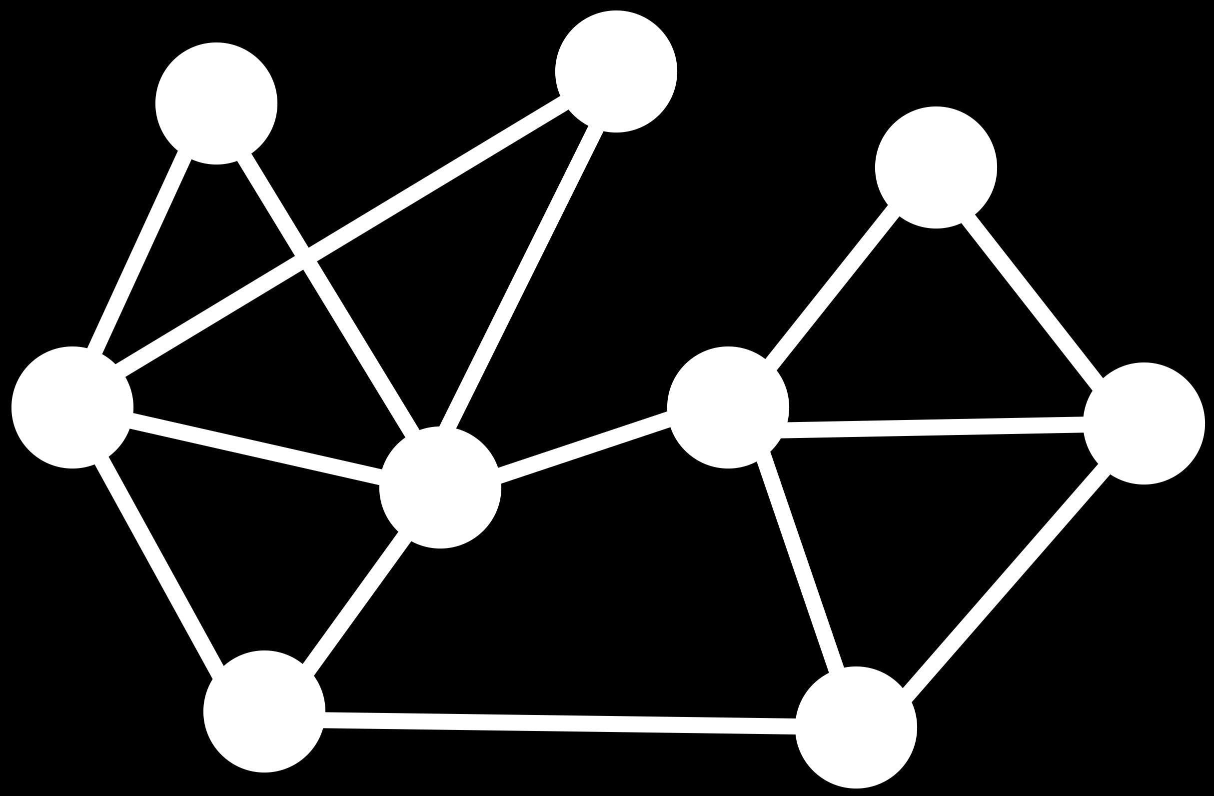
Predicted

Why Networks? Why Now?

- **Universal language for describing 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, Social media, Drug design



Network!



Network!

Working Network Data

- **Network data brings several core machine learning methodologies into play**
  - **Working with network data is messy**
    - Not just "wiring diagrams" but also dynamics and (meta)-data (features, attributes)
  - **Computational challenges**
    - Large scale network data
  - **Algorithmic models as vocabulary for expressing complex scientific questions**
    - Social science, physics, biology

Tools for Networks

- **S**tanford **N**etwork **A**nalysis **P**latform (**SNAP**) is a general purpose, high-performance system for analysis and manipulation of large networks
  - http://snap.stanford.edu
  - Scales to massive networks with hundreds of millions of nodes and billions of edges
- **SNAP software**
  - Snap.py for Python, SNAP C++
  - Tutorial on how to use SNAP: http://snap.stanford.edu/proj/snap-icwsm

Snap.py Resources

- **Prebuilt packages** for Mac OS X, Windows, Linux
  http://snap.stanford.edu/snappy/index.html
- **Snap.py documentation**:
  http://snap.stanford.edu/snappy/doc/index.html
  - Quick Introduction, Tutorial, Reference Manual
- **SNAP user mailing list**
  http://groups.google.com/group/snap-discuss
- **Developer resources**
  - Software available as open source under BSD license
  - GitHub repository
    https://github.com/snap-stanford/snap-python

SNAP C++ Resources

- **Prebuilt packages** for Mac OS X, Windows, Linux
  http://snap.stanford.edu/snap/download.html
- **SNAP documentation**
  http://snap.stanford.edu/snap/doc.html
  - Quick Introduction, User Reference Manual
- **SNAP user mailing list**
  http://groups.google.com/group/snap-discuss
- **Developer resources**
  - Software available as open source under BSD license
  - GitHub repository
    https://github.com/snap-stanford/snap
  - SNAP C++ Programming Guide

Network Data

- **Stanford Large Network Dataset Collection**
  - http://snap.stanford.edu/data
- **Over 70 different networks and communities**
  - **Social networks:** online social networks, edges represent interactions between people
  - **Twitter and Memetracker:** Memetracker phrases, links and 467 million Tweets
  - **Citation networks:** nodes represent papers, edges represent citations
  - **Collaboration networks:** nodes represent scientists, edges represent collaborations
  - **Amazon networks :** nodes represent products and edges link commonly co-purchased products

Books & Courses

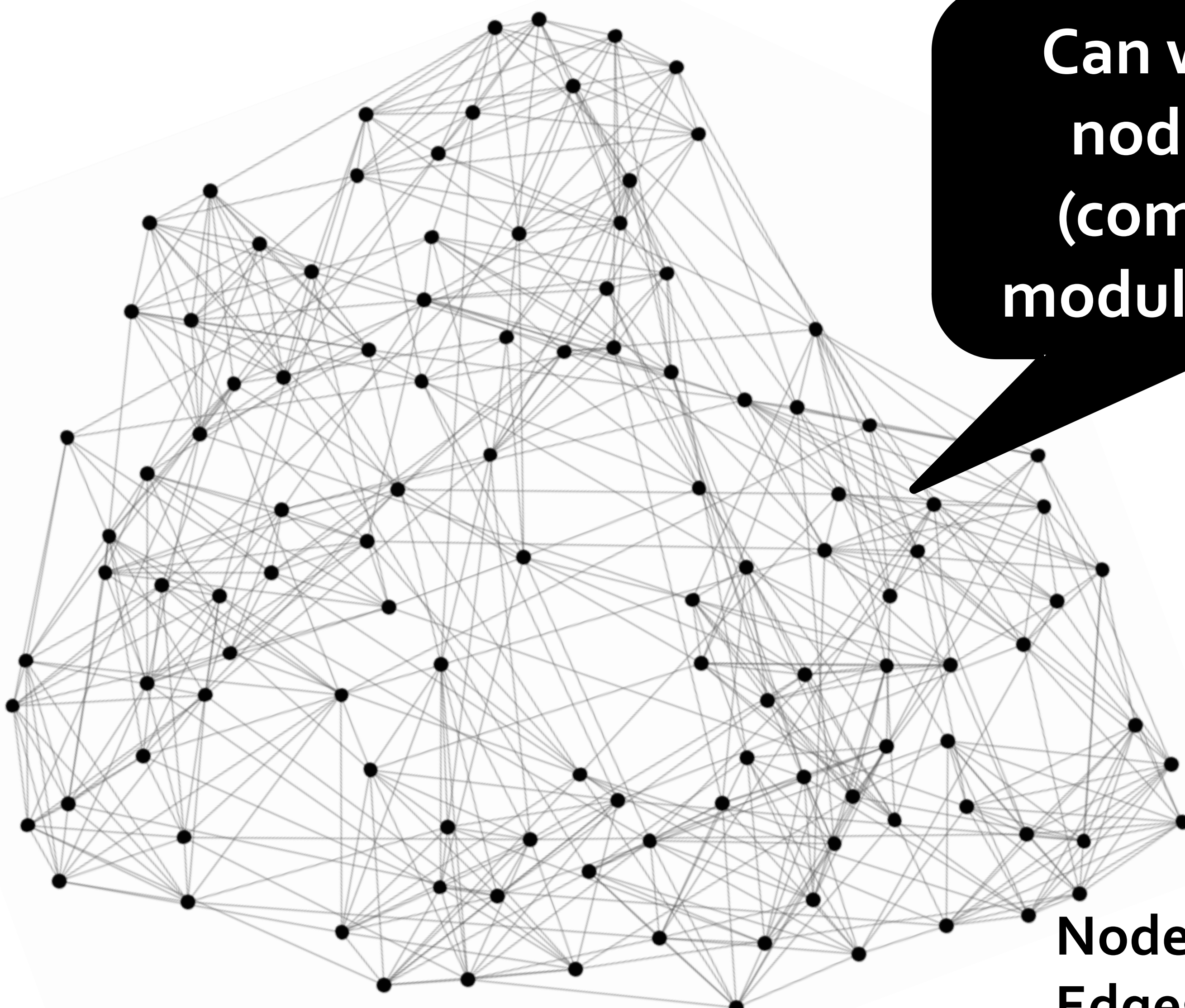
Want to learn more about networks?

- **Social and Information Networks** lectures:
  - http://cs224w.stanford.edu
- **Mining Massive Datasets** lectures:
  - http://cs246.stanford.edu
- **Books (fee PDFs):**
  - **Mining Massive Datasets**
    - http://infolab.stanford.edu/~ullman/mmds.html
  - **Networks, Crowds and Markets**
    - http://www.cs.cornell.edu/home/kleinber/networks-book

Networks: 3 problems

1) Community detection
2) Link & Attribute prediction
3) Social media

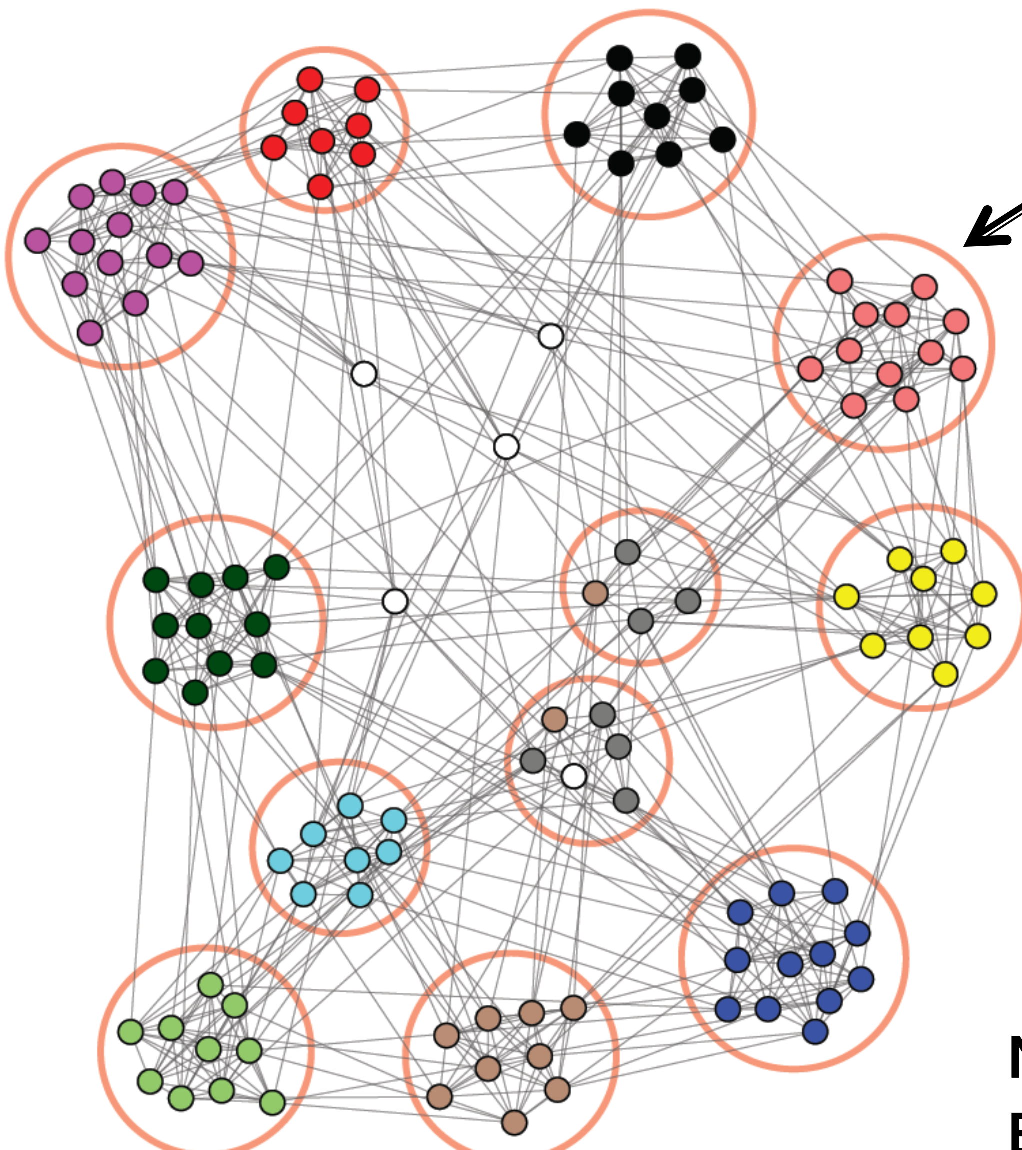
Identifying Structure



Can we identify node groups? (communities, modules, clusters)

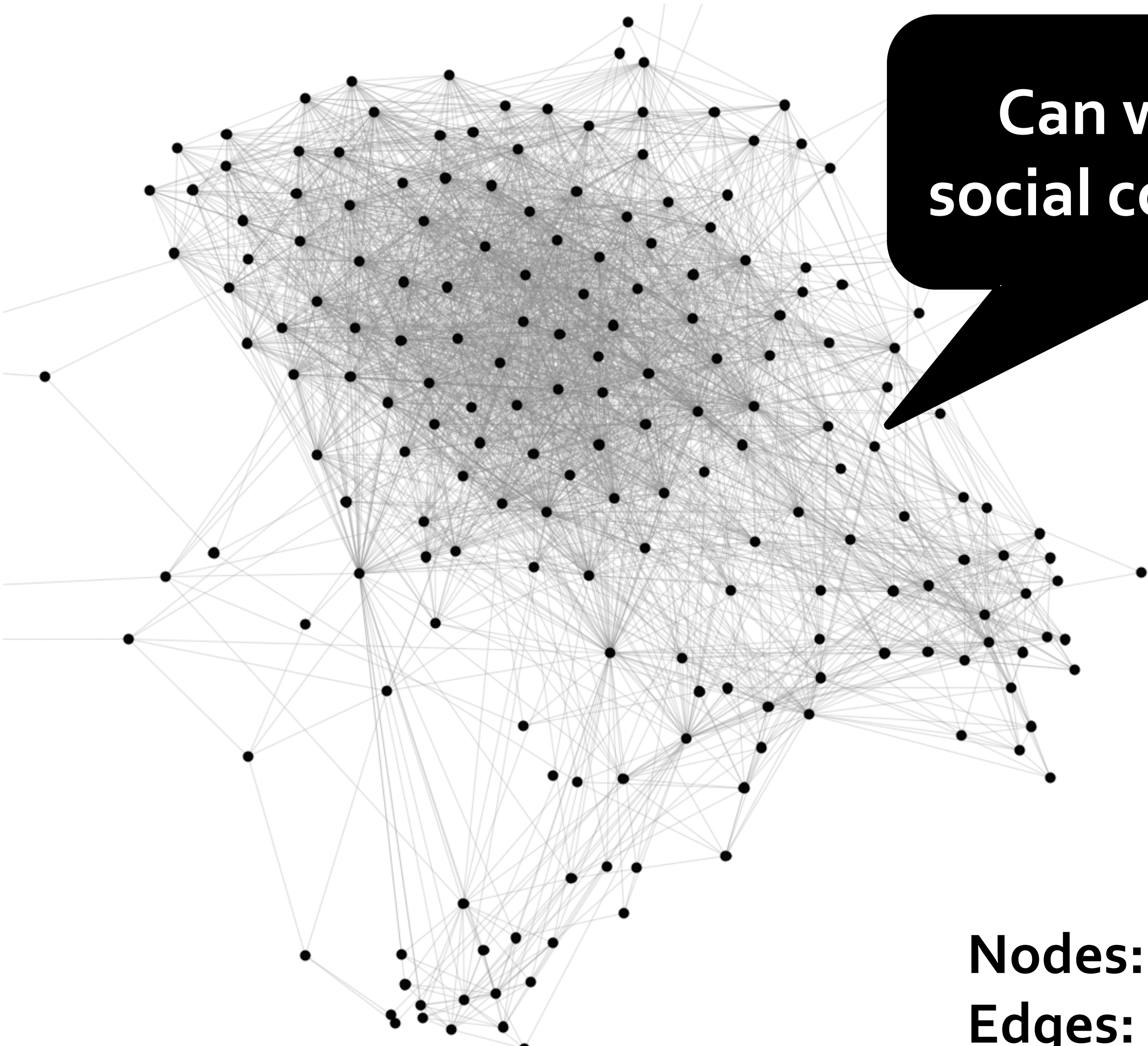
Nodes: Football Teams
Edges: Games played

NCAA Football Network



Nodes: Football Teams
Edges: Games played

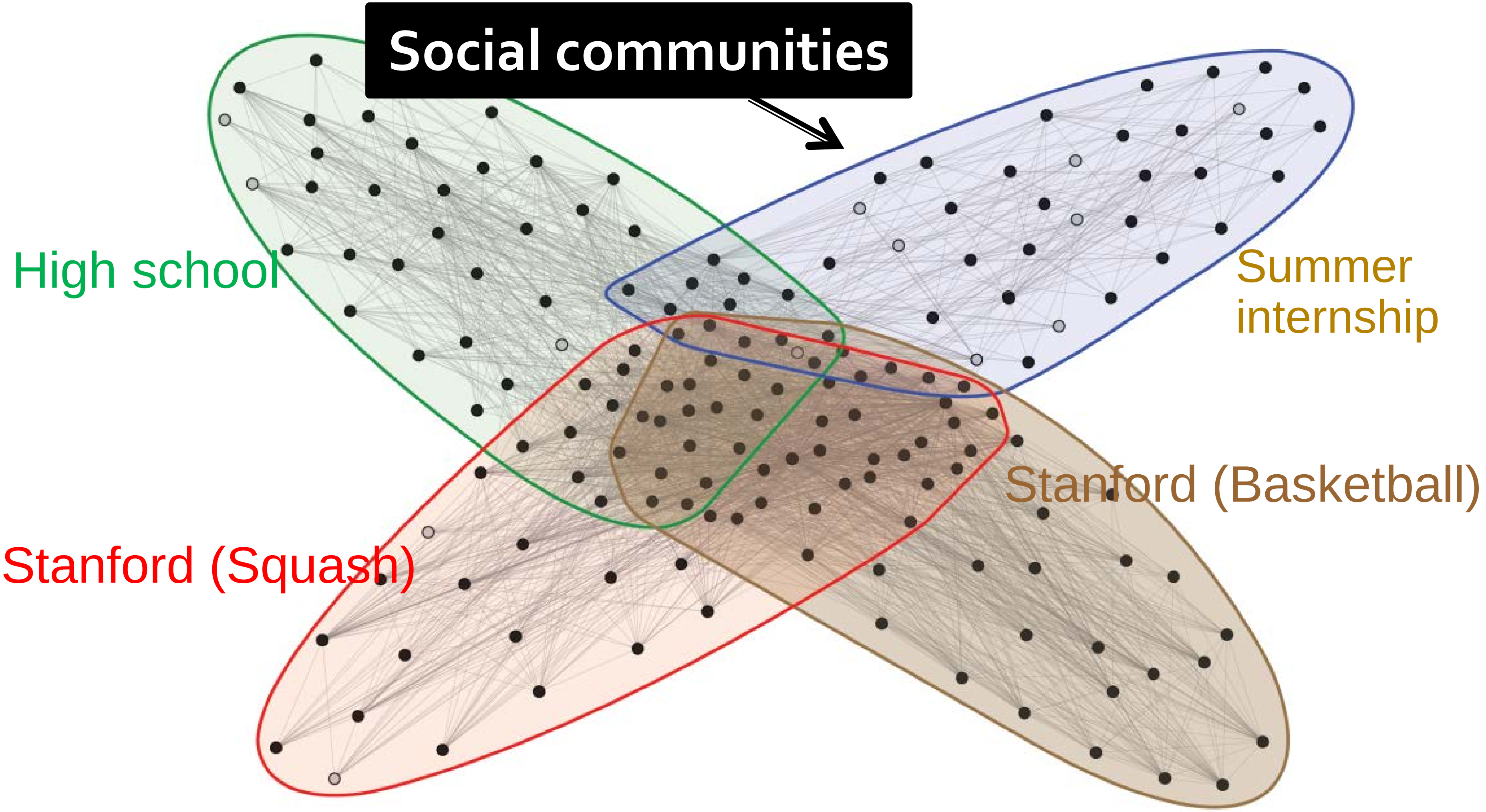
Facebook Network



Can we identify social communities?

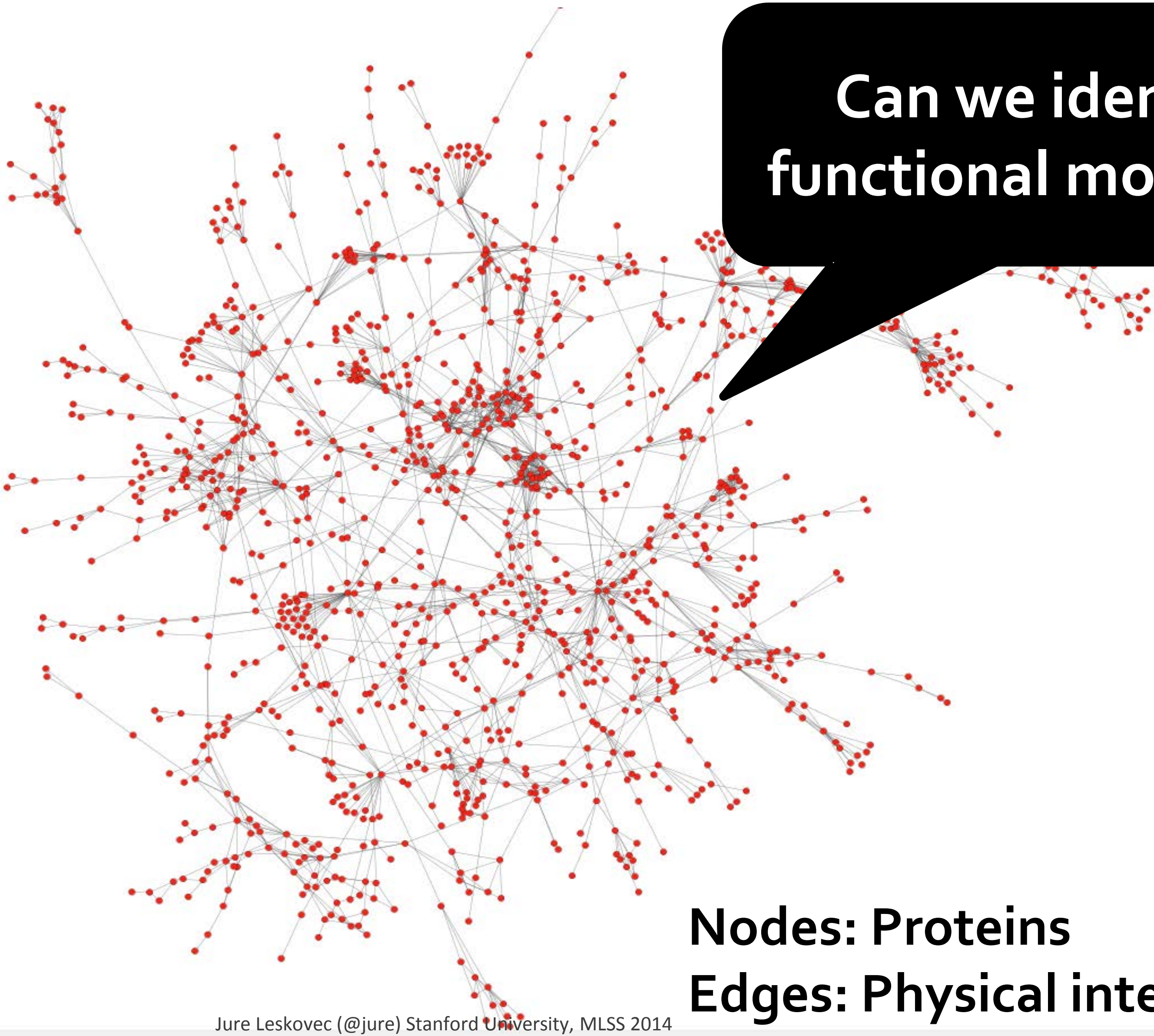
Nodes: Facebook Users
Edges: Friendships

Facebook Network



Nodes: Facebook Users
Edges: Friendships

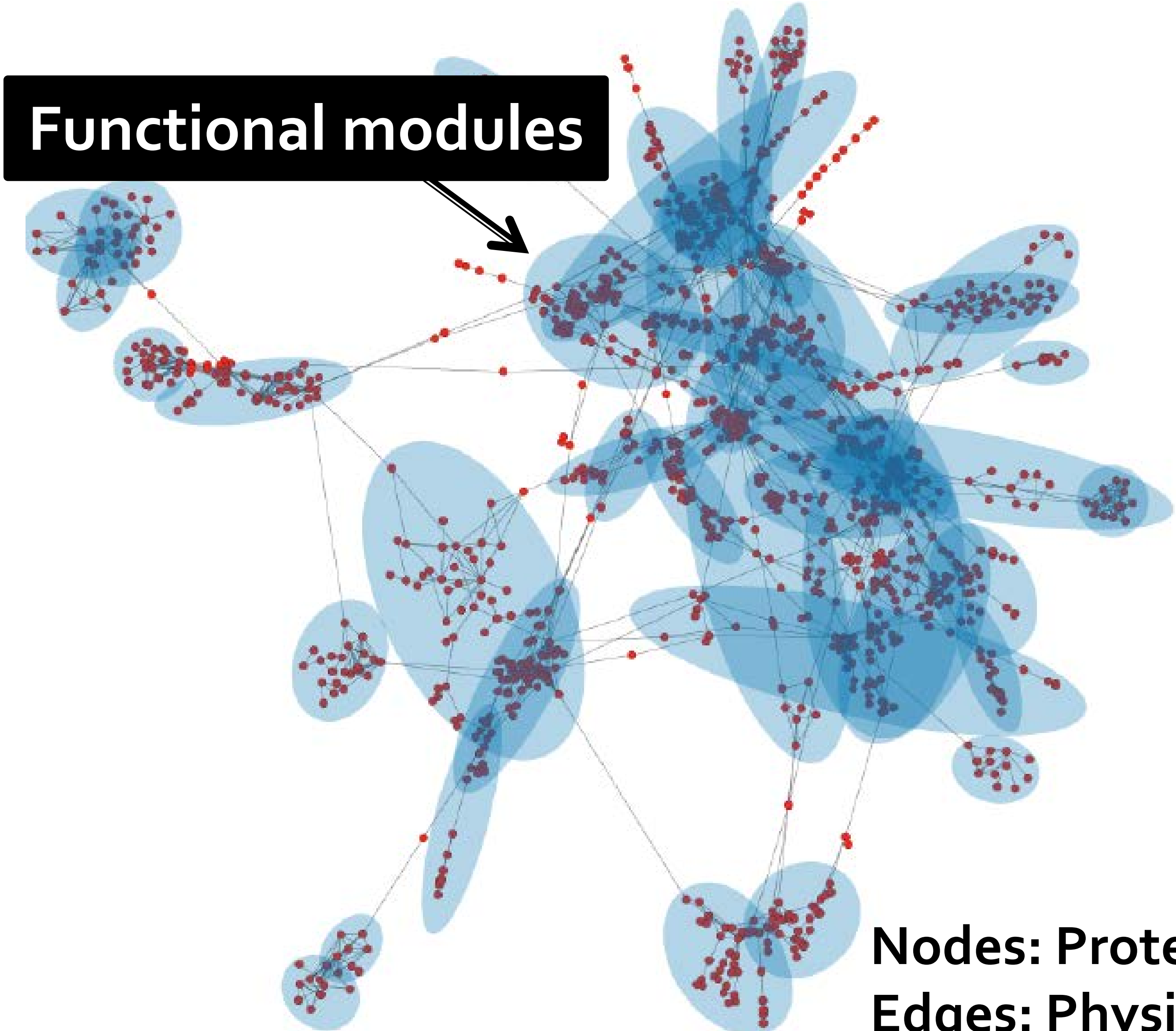
Protein-Protein Interactions



Can we identify functional modules?

Nodes: Proteins
Edges: Physical interactions

Protein-Protein Interactions

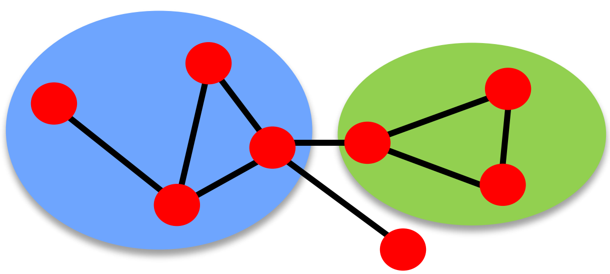


Nodes: Proteins

Edges: Physical interactions

Community Detection

- **Input:**
  A network
- **Output:**
  Community memberships of nodes



Cluster nodes based on network connectivity with the hope to identify sets of objects with common function, role or property.

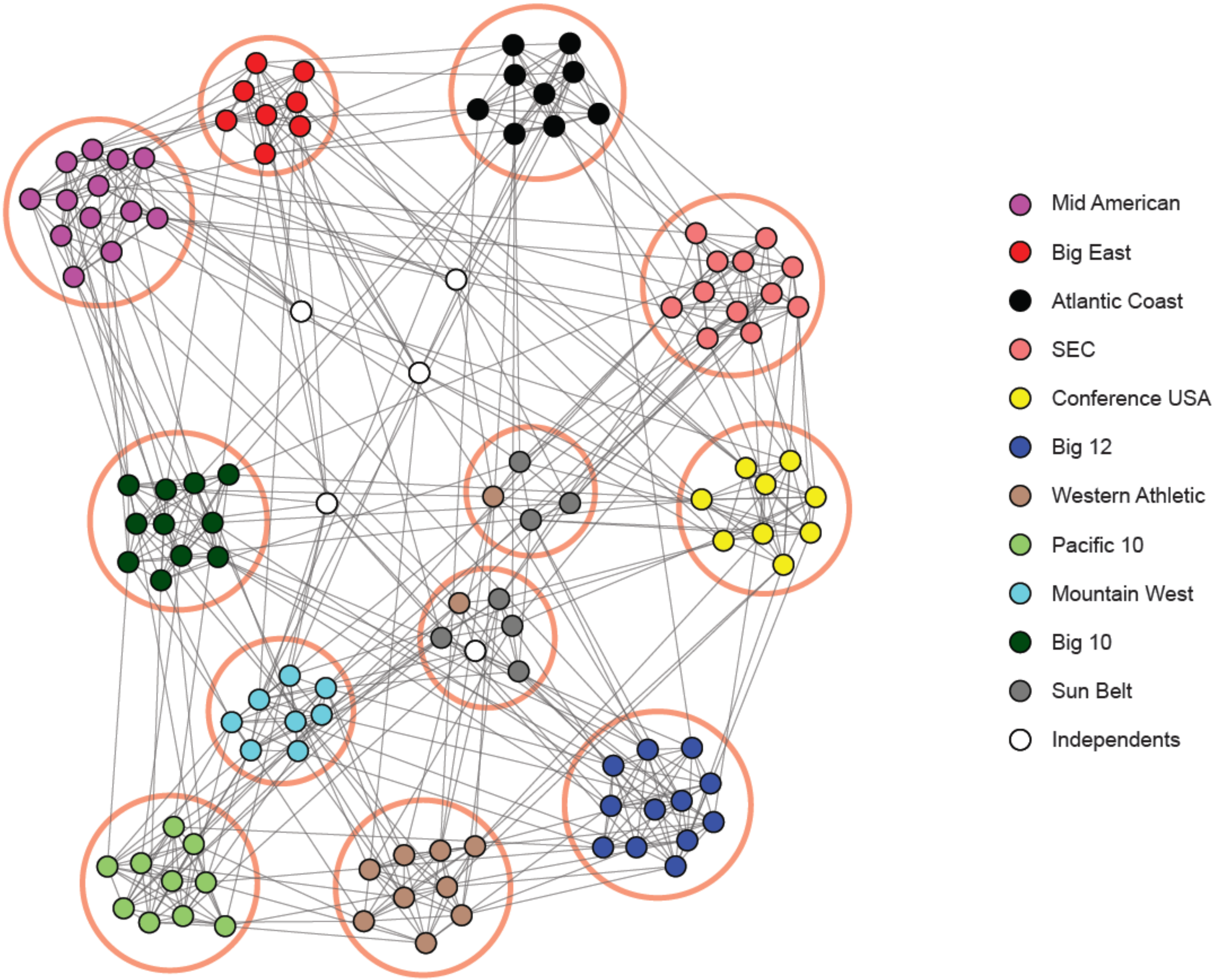
Why is it important?

- **Community detection is a fundamental problem in network analysis allowing for:**
  - **Discovering unknown roles of proteins** [Krogan et al. ‘06]
  - **Identifying module boundaries** [Ahn et al. ‘11]
  - **Detecting missing links** [Kim, L. ‘12]
  - **Observing political factions in the blogosphere** [Adamic, Glance ’05]
  - **Identifying functional modules** [Palla et al. ‘05]

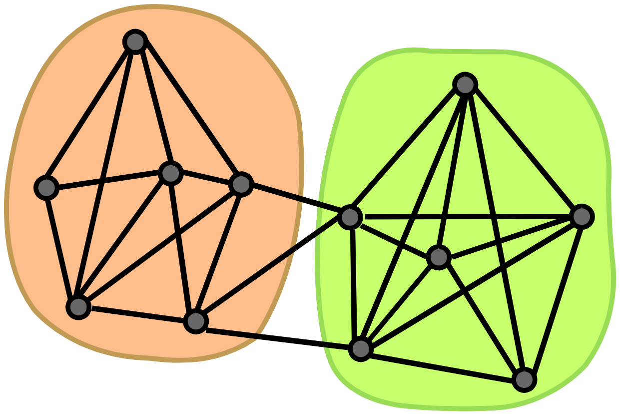
Why is it hard?

- **Modeling:** Communities form **complex structures**: Non-overlapping, overlapping, hierarchically nested
- **Computation:** Many formulations lead to **intractable problems**
  - For 100k node networks many methods take days to run
- **Evaluation:** **Lack of ground-truth**
  - Research relies on anecdotal manual inspection

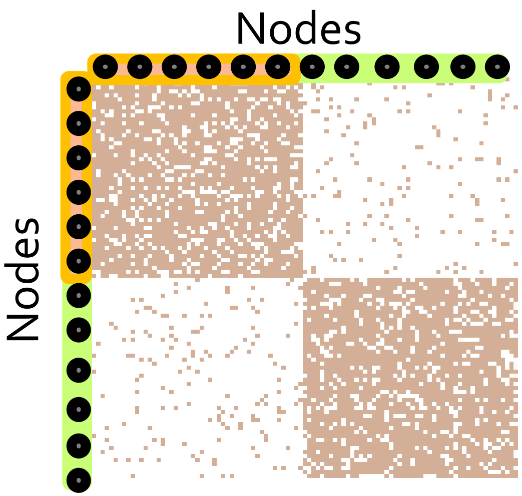
Non-overlapping Communities



Non-overlapping Communities



Network



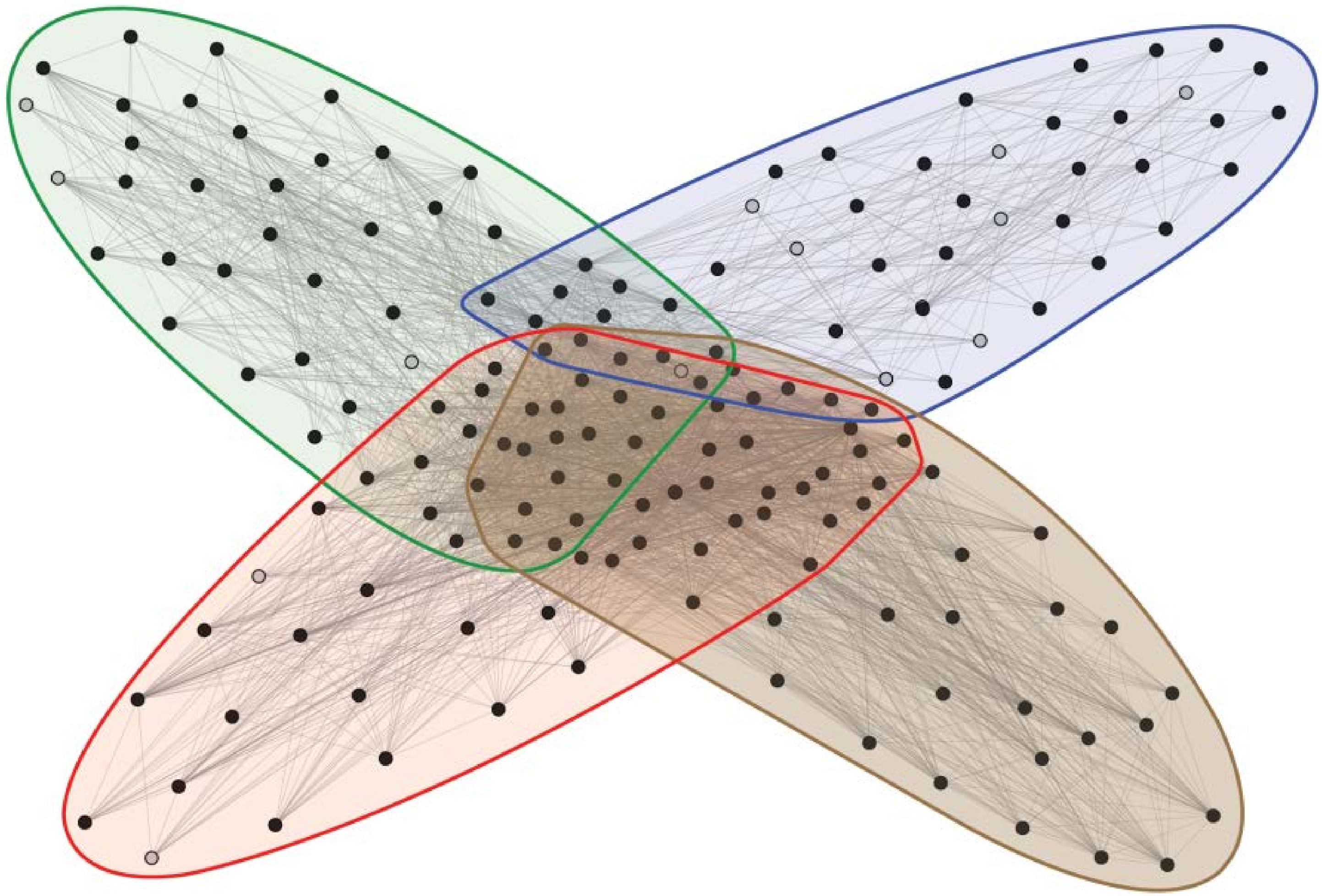
Adjacency matrix

Methods for non-overlapping communities…

- Spectral clustering [Shi&Malik '00], Modularity [Newman '06], Block models [Holland '83], …

…define communities as well-separable clusters

What if communities overlap?



Overlapping Community Detection

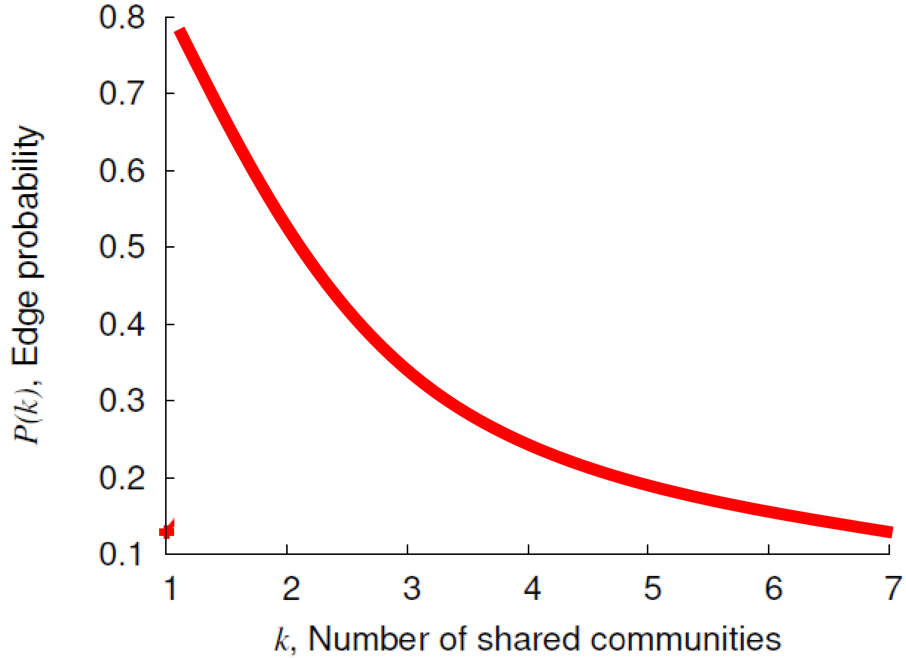
Many methods for overlapping communities:

- **Mixed membership stochastic block models** [Airoldi, Blei, Feinberg, Xing, ’08]
- **Link clustering** [Ahn et al. ‘10] [Evans et al. ‘09]
- **Clique percolation** [Palla et al. ‘05]
- **Clique expansion** [Lee et al. ‘10]
- **Bayesian matrix factorization** [Psorakis et al. ‘11]

What do these methods assume about community overlaps?

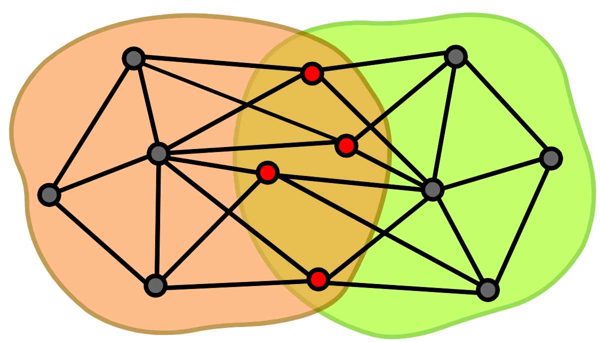
Overlapping Communities

- **Existing methods assume that edge probability <u>decreases</u> with the number of shared communities**

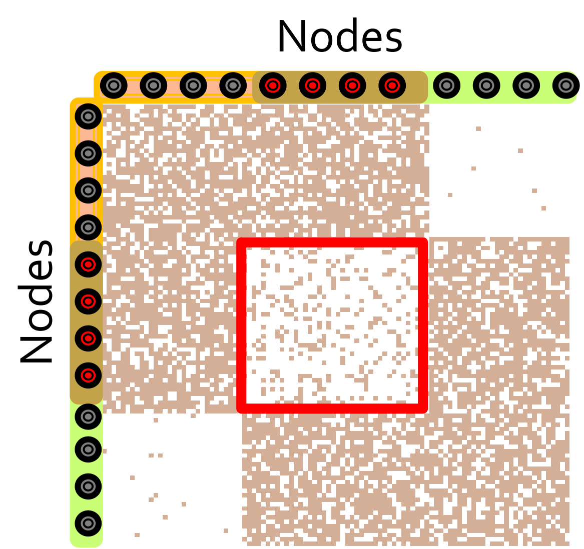


Overlapping Communities

- **Existing methods assume that edge probability <u>decreases</u> with the number of shared communities**



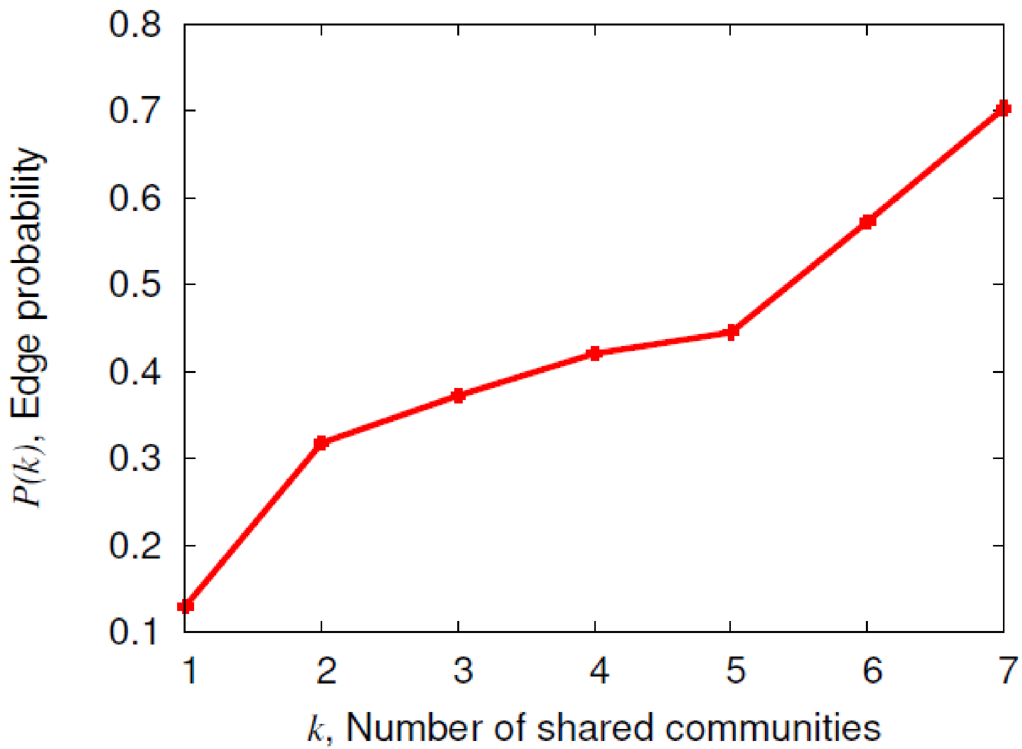
Network



Adjacency matrix

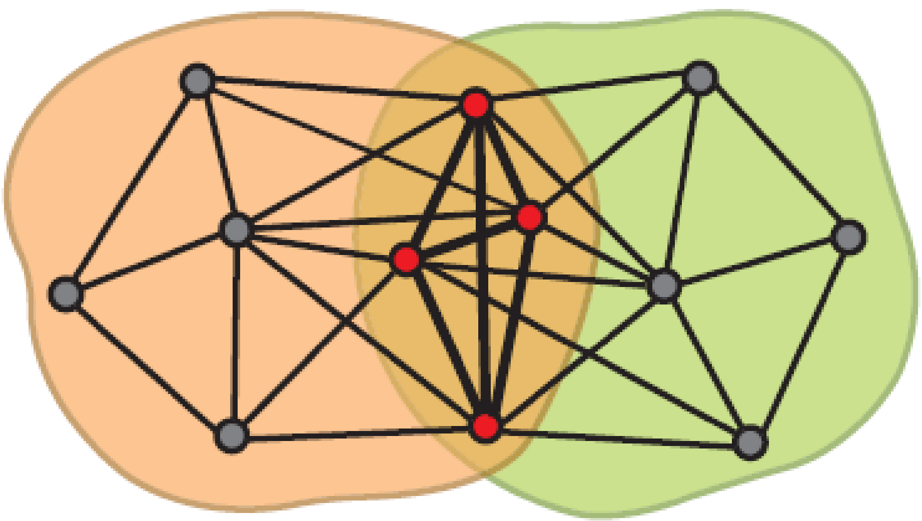
Community Overlaps

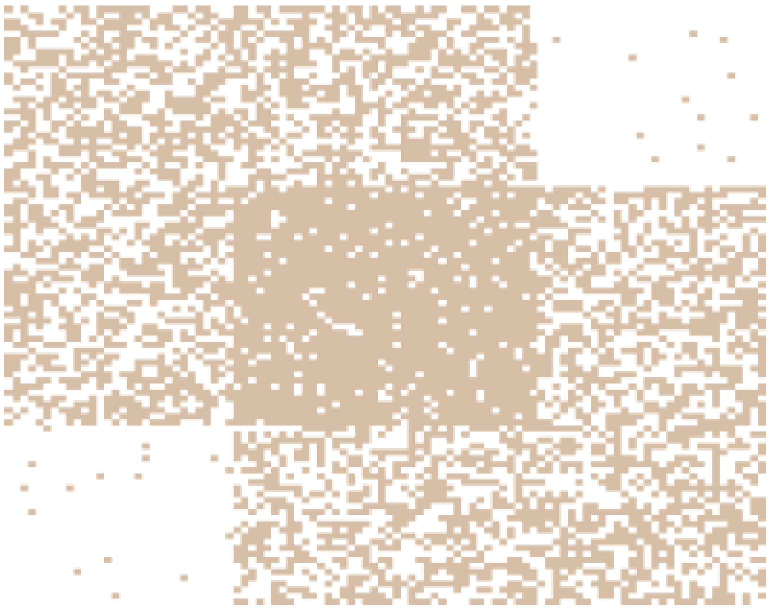
- **More communities** U and V share the **more likely** they are **linked**
  $\Rightarrow$ **Community overlaps are denser**



Community Overlaps

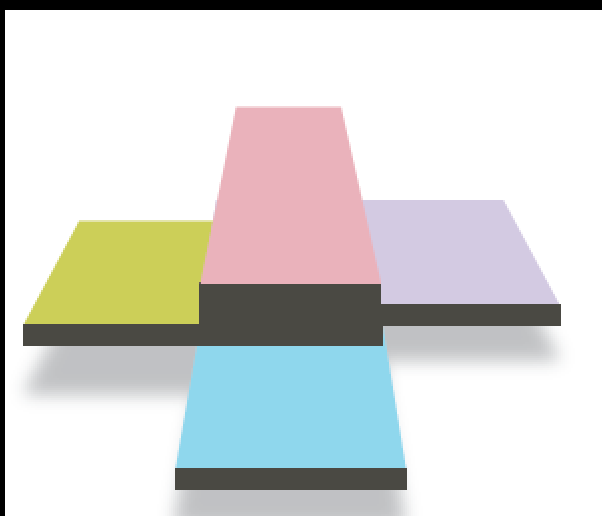
- **More communities** U and V share the **more likely** they are **linked**
  $\Rightarrow$ **Community overlaps are denser**



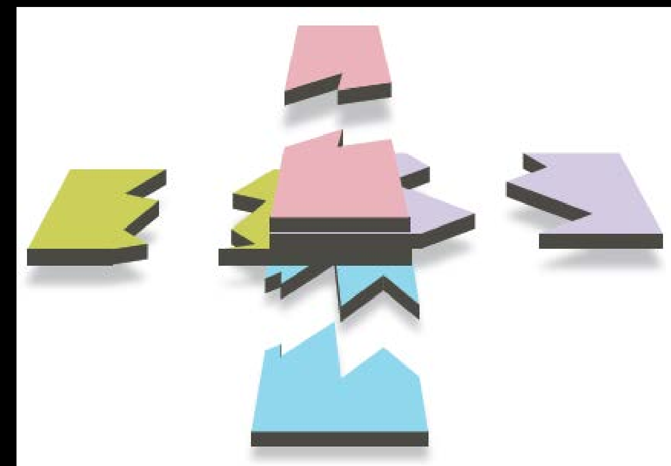


New paradigm: Communities as "tiles"

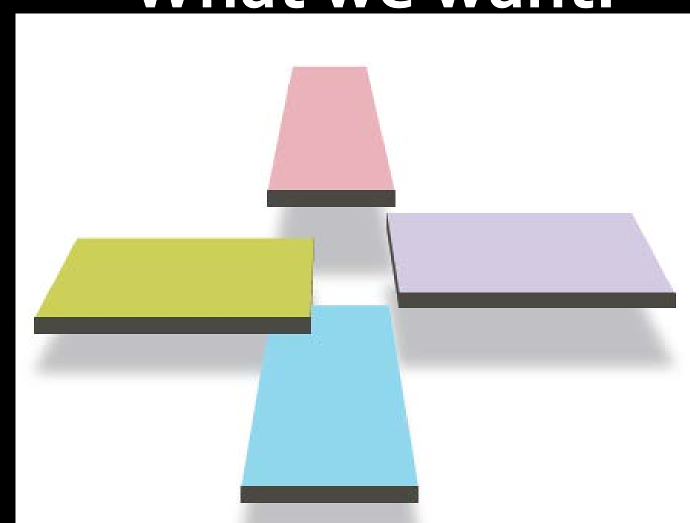
From Networks to Communities



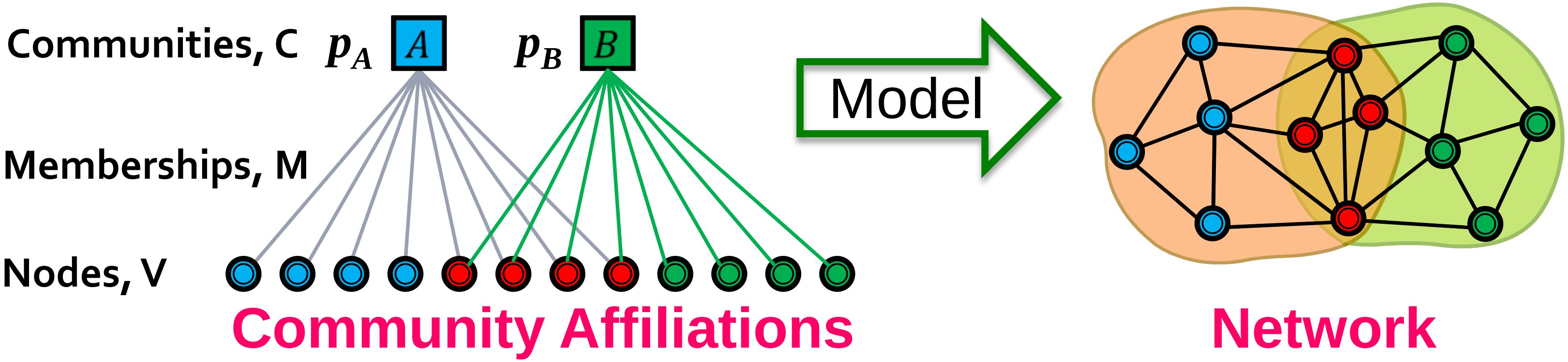
What we have:



What we want:

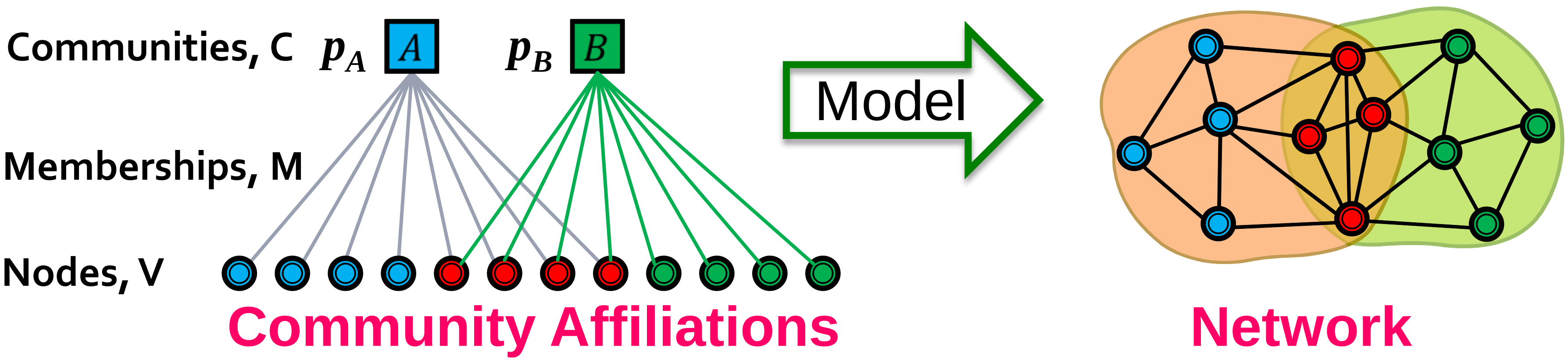


Community-Affiliation Graph



- **Generative model: How is a network generated from community affiliations?**
  - Later, we detect communities by fitting the model
- **Model parameters B**(*V, C, M*, {p_c}) **:**
  - Nodes **V**, Communities **C**, Memberships **M**
  - Each community **c** has a single probability $\mathbf{p_c}$

AGM: Generative Process



- **AGM generates the network:**
  - Nodes in community c connect to each other with probability p_c:

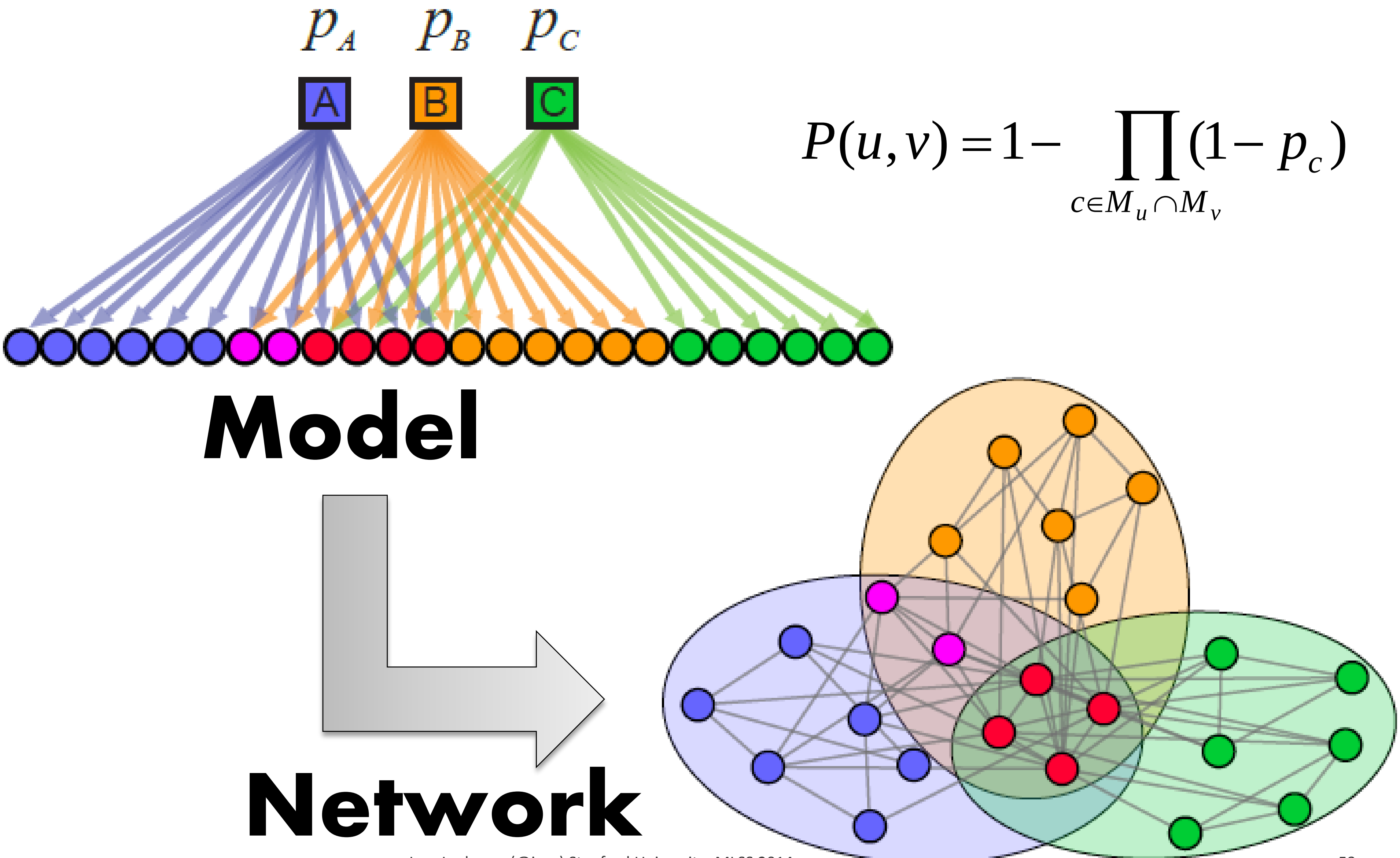
$$P(u,v) = 1 - \prod_{c \in M_u \cap M_v} (1 - p_c)$$

Provably generates power-law degree distributions and other real-world network patterns [Lattanzi, Sivakumar, '09]

[ICDM'12]

AGM Generates Networks

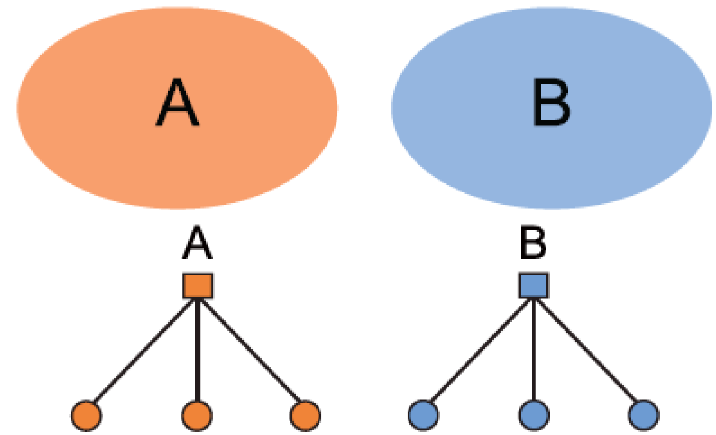
$$P(u,v) = 1 - \prod_{c \in M_u \cap M_v} (1 - p_c)$$

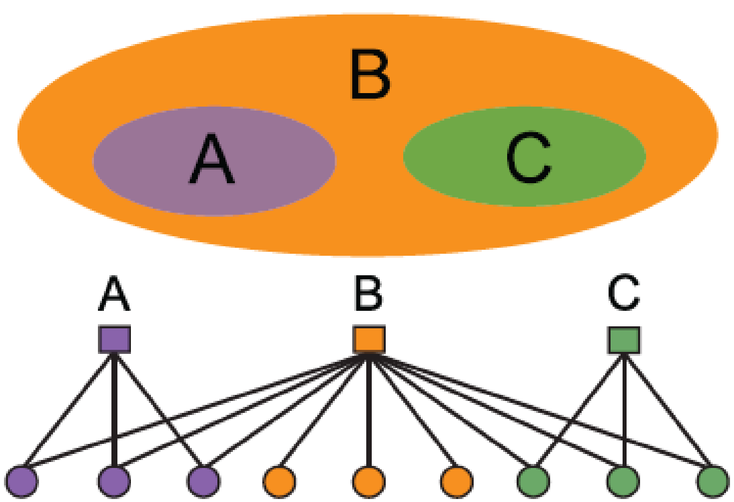


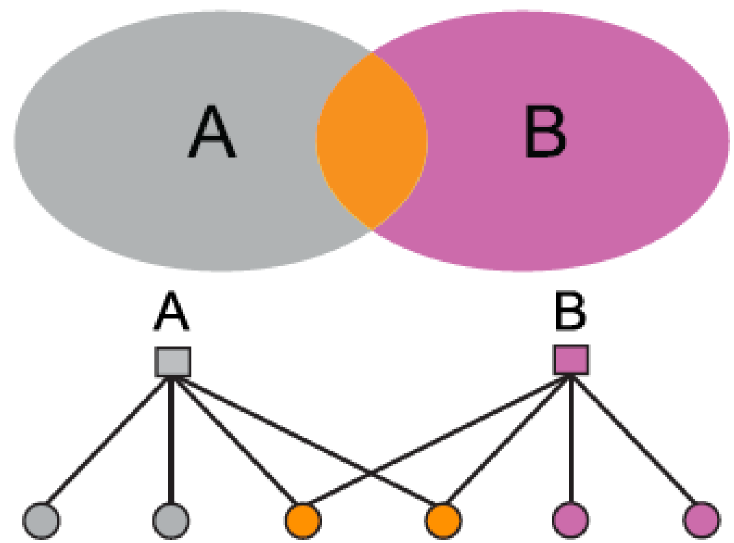
[icdm ‘12]

AGM: Modeling Flexibility

- **AGM can express a variety of community structures:** Non-overlapping, Overlapping, Nested

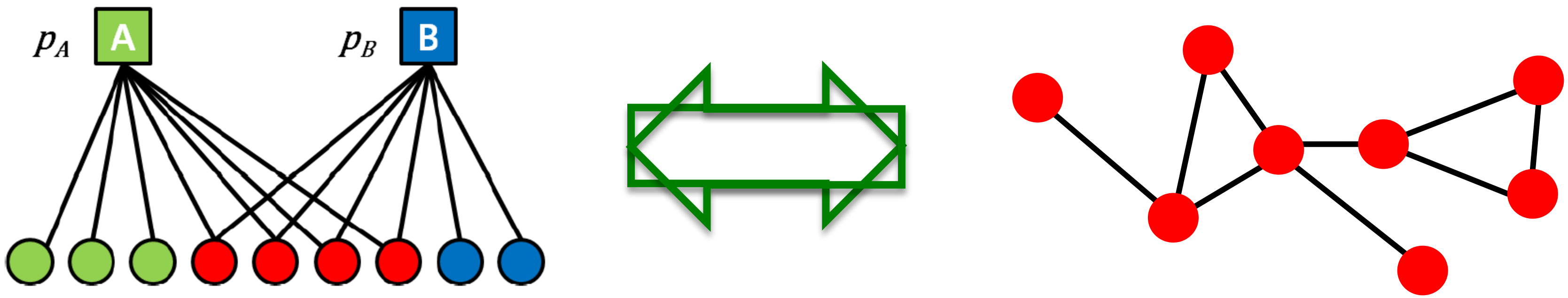






Detecting Communities

- **Detecting communities with AGM:**



- **Given a graph *G*, find the model *B* by maximizing the model likelihood:**

$$\arg\max_B P(G;B) = \prod_{(i,j)\in E} P(i,j) \prod_{(i,j)\notin E} (1 - P(i,j))$$

$$P(i,j) = 1 - \prod_{c\in M_i \cap M_j} (1 - p_c)$$

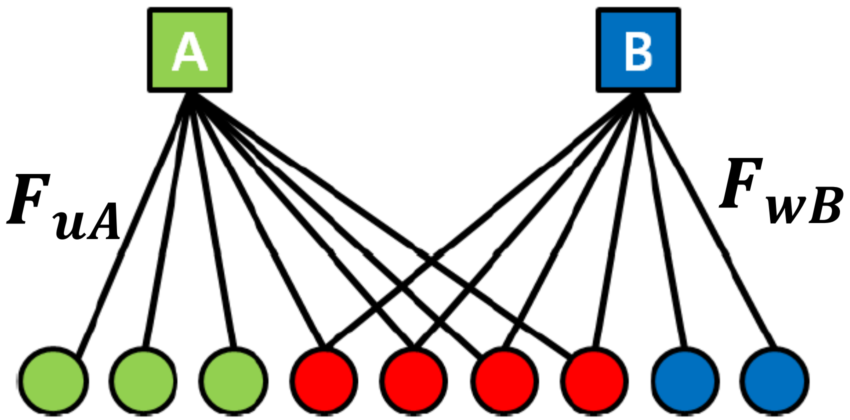
Model B has 3 parts:

1) Affiliation graph ***M***
2) Number of communities **C**
3) Parameters $\boldsymbol{p_c}$

[wsdm '13]

"Relaxing" AGM

- "Relax" the AGM: **Memberships have strengths**



- F_{uA}: The membership strength of node u to community A ($F_{uA} = 0$: no membership)

[wsdm '13]

BigCLAM Model

- **Prob. of nodes linking is proportional to the strengths of shared memberships:**

$$\boldsymbol{P(u,v) = 1 - \exp(-F_u \cdot F_v^T)}$$

- **Now, given a network, we estimate F**

$$l(F) = \sum_{(u,v)\in E} \log(1-\exp(-F_u F_v^T)) - \sum_{(u,v)\notin E} F_u F_v^T$$

- **Non-negative matrix factorization:**
  - Update F_{uC} for node u while fixing the memberships of all other nodes
  - Updating takes linear time in the degree of u

BigCLAM Model

- **Apply block coordinate gradient ascent**

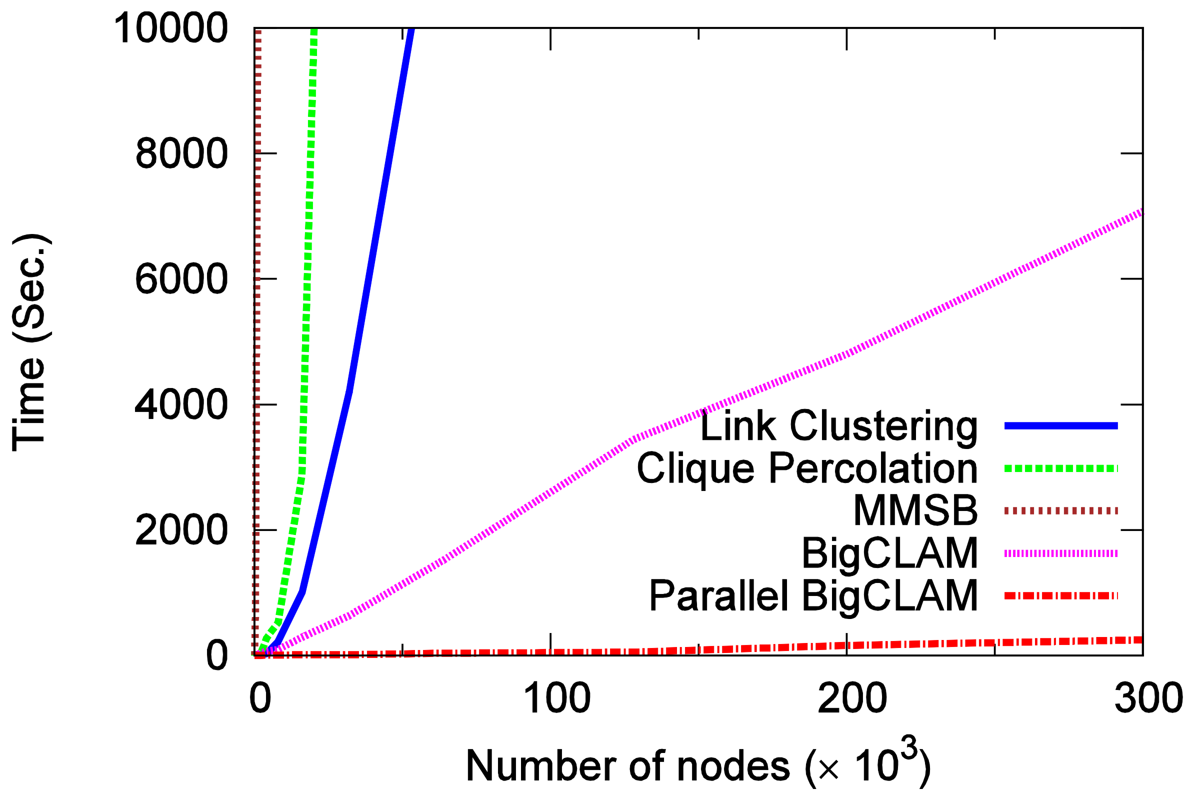
$$\nabla l(F_u) = \sum_{v \in \mathcal{N}(u)} F_v \frac{\exp(-F_u F_v^T)}{1 - \exp(-F_u F_v^T)} - \sum_{v \notin \mathcal{N}(u)} F_v$$

  - Step size: backtracking line search
  - Project $\boldsymbol{F_u}$ back to a non-negative vector
- Pure gradient ascent is slow! **However:**

$$\sum_{v \notin \mathcal{N}(u)} F_v = \left(\sum_v F_v - F_u - \sum_{v \in \mathcal{N}(u)} F_v\right)$$

- By caching $\boldsymbol{F_v}$ a gradient step takes **linear time** in the degree of $\boldsymbol{u}$

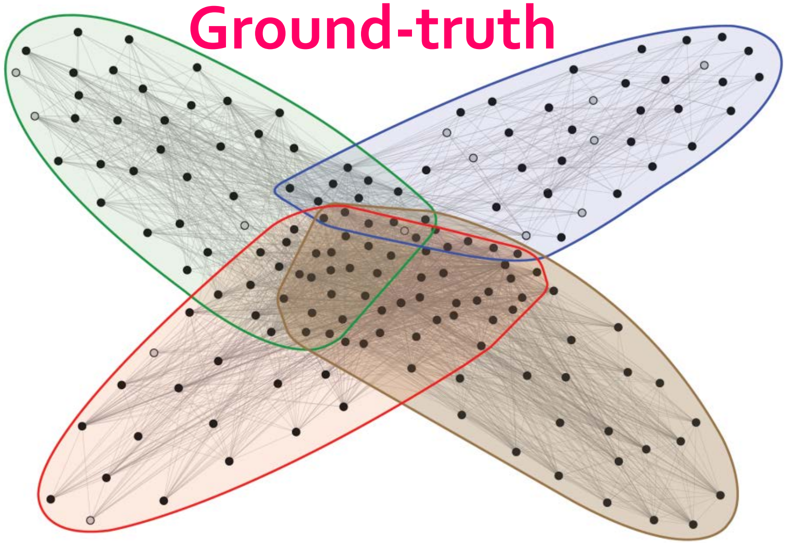
BigClam: Scalability



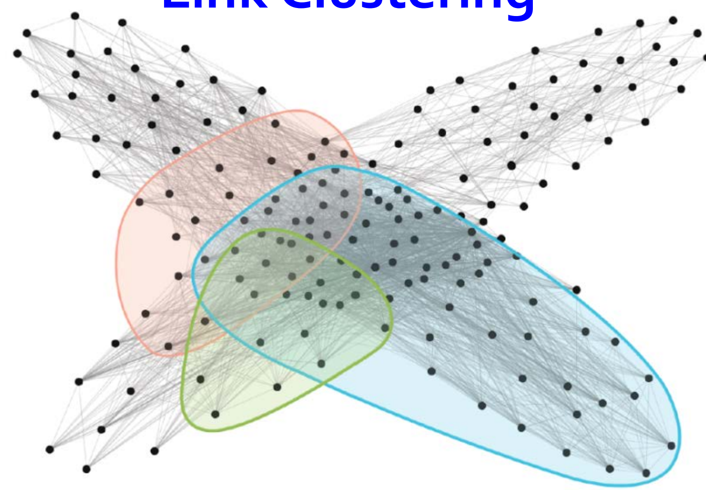
- BigCLAM takes 5min for 300k node networks
  - Other methods take 10 days
- Can process networks with 100M edges!

Results on a Facebook Network

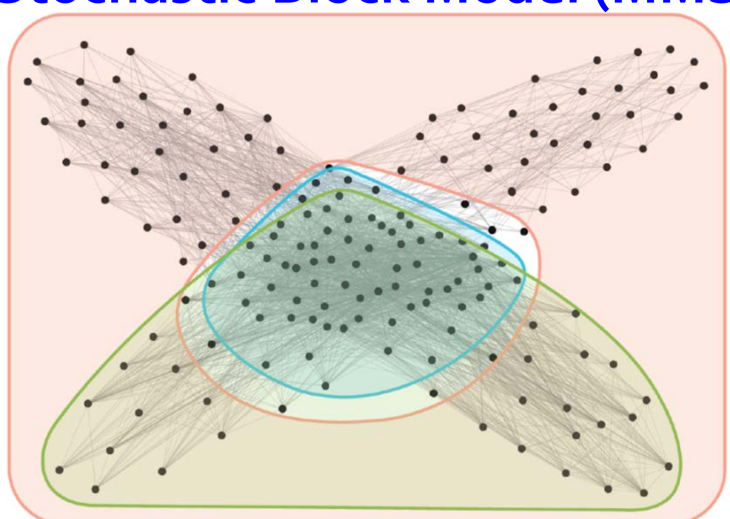
Ground-truth



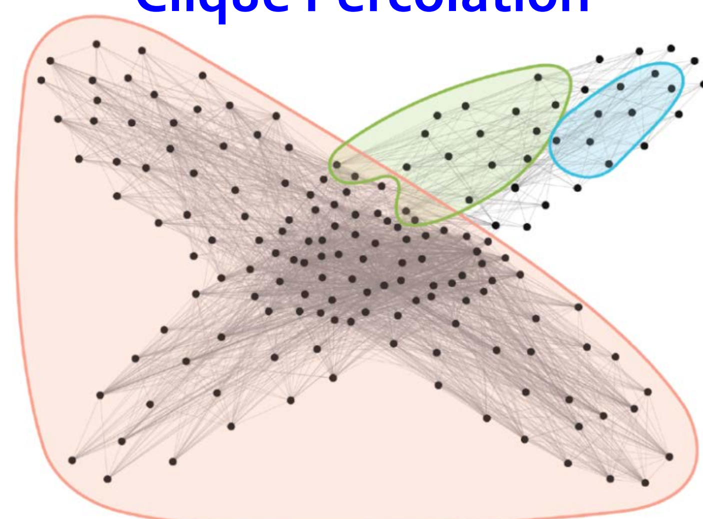
Link Clustering



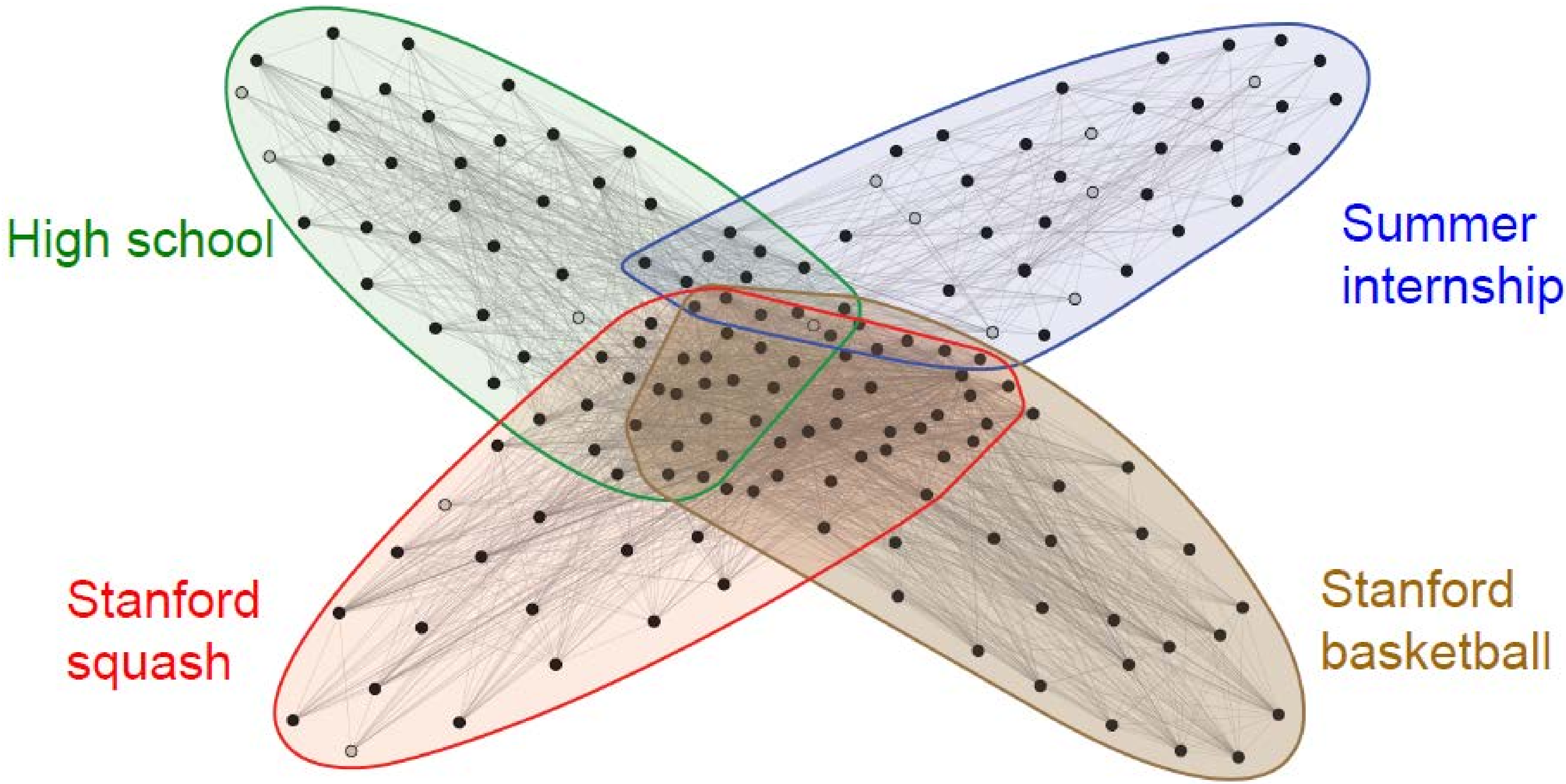
Stochastic Block Model (MMSB)



Clique Percolation



BigClam: Does it work?

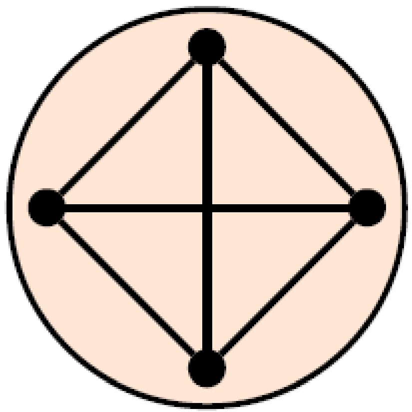


94% accuracy

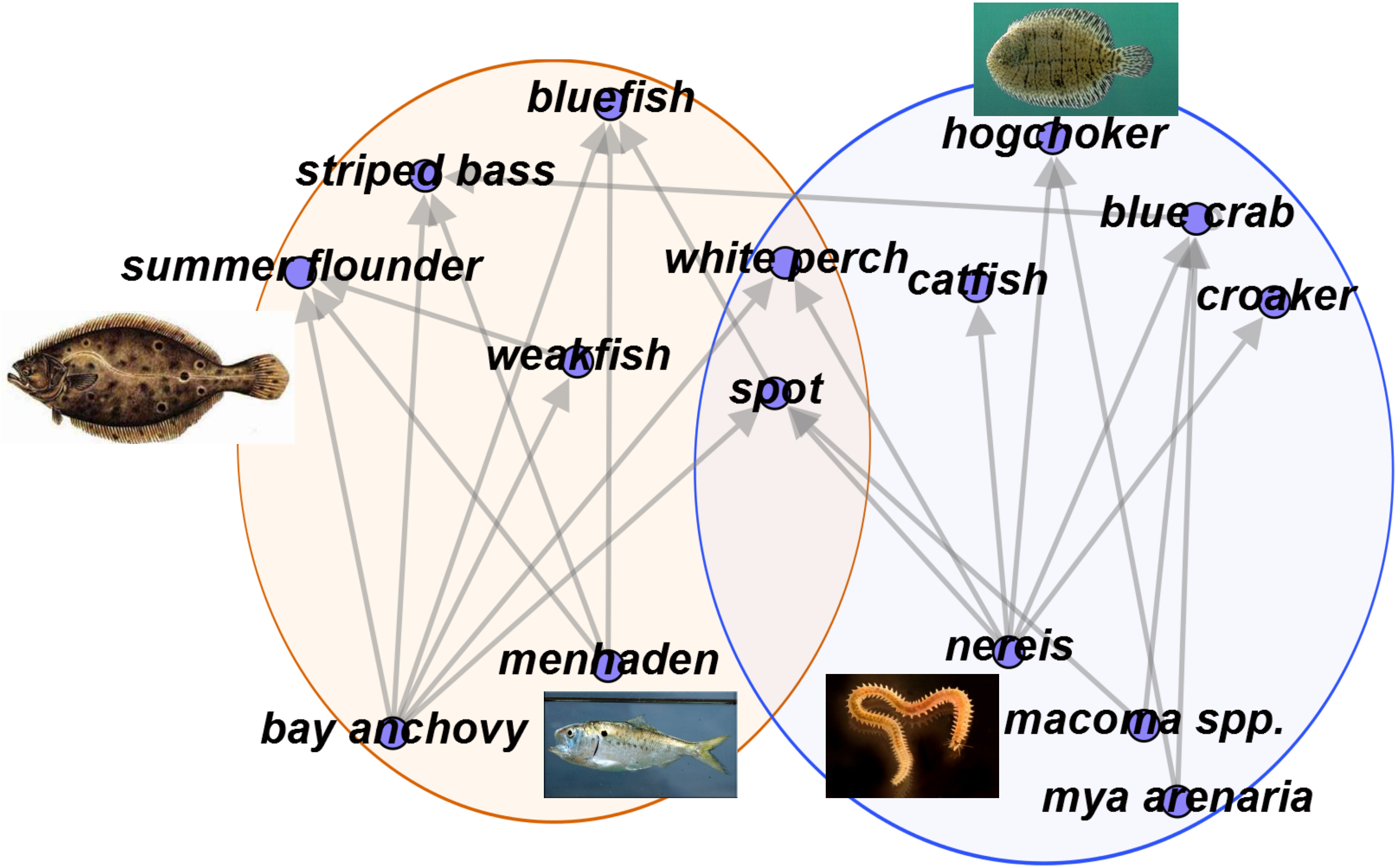
Extensions: Beyond Clusters

Cohesive

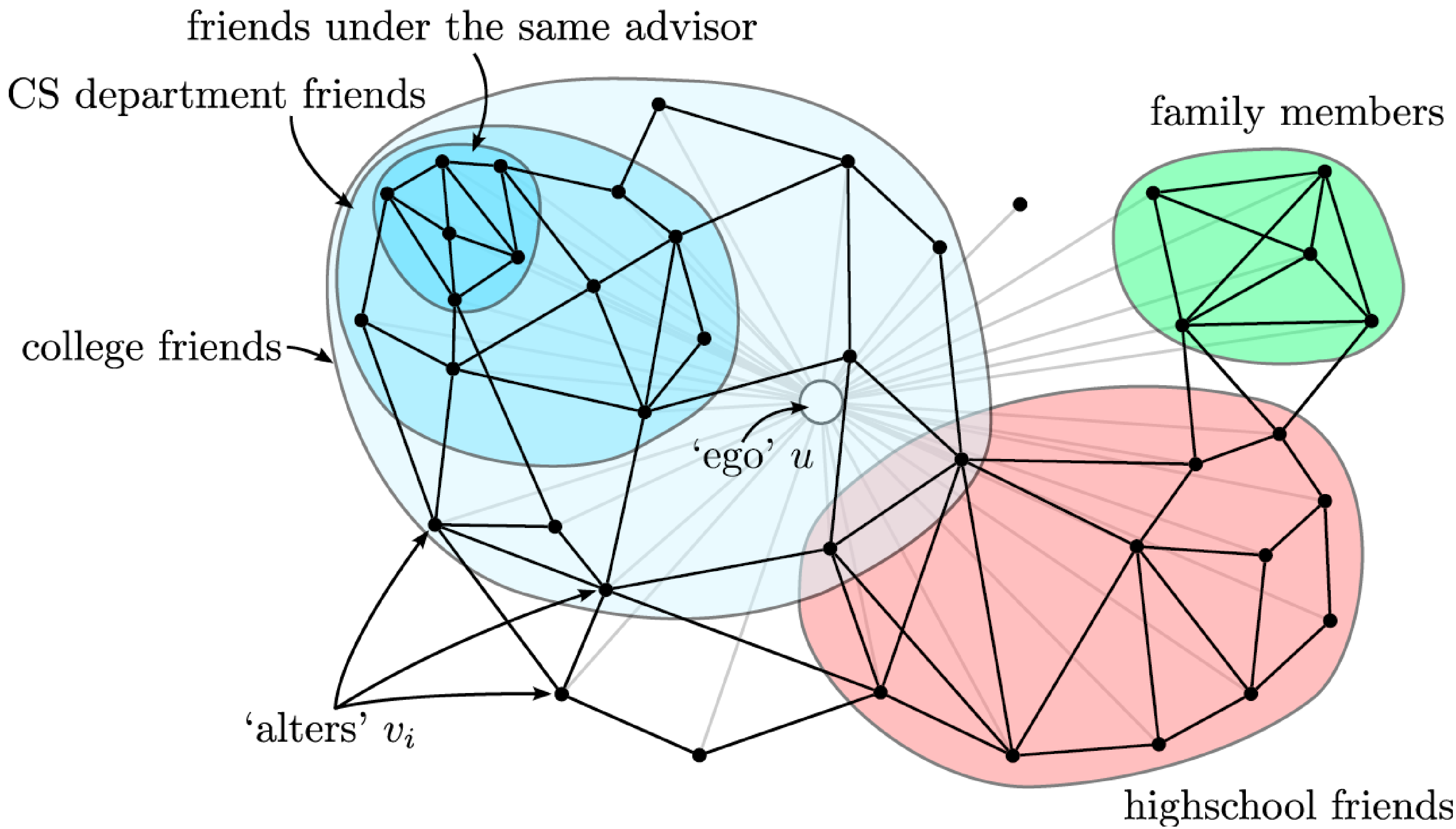
Undirected



Predator-prey Communities

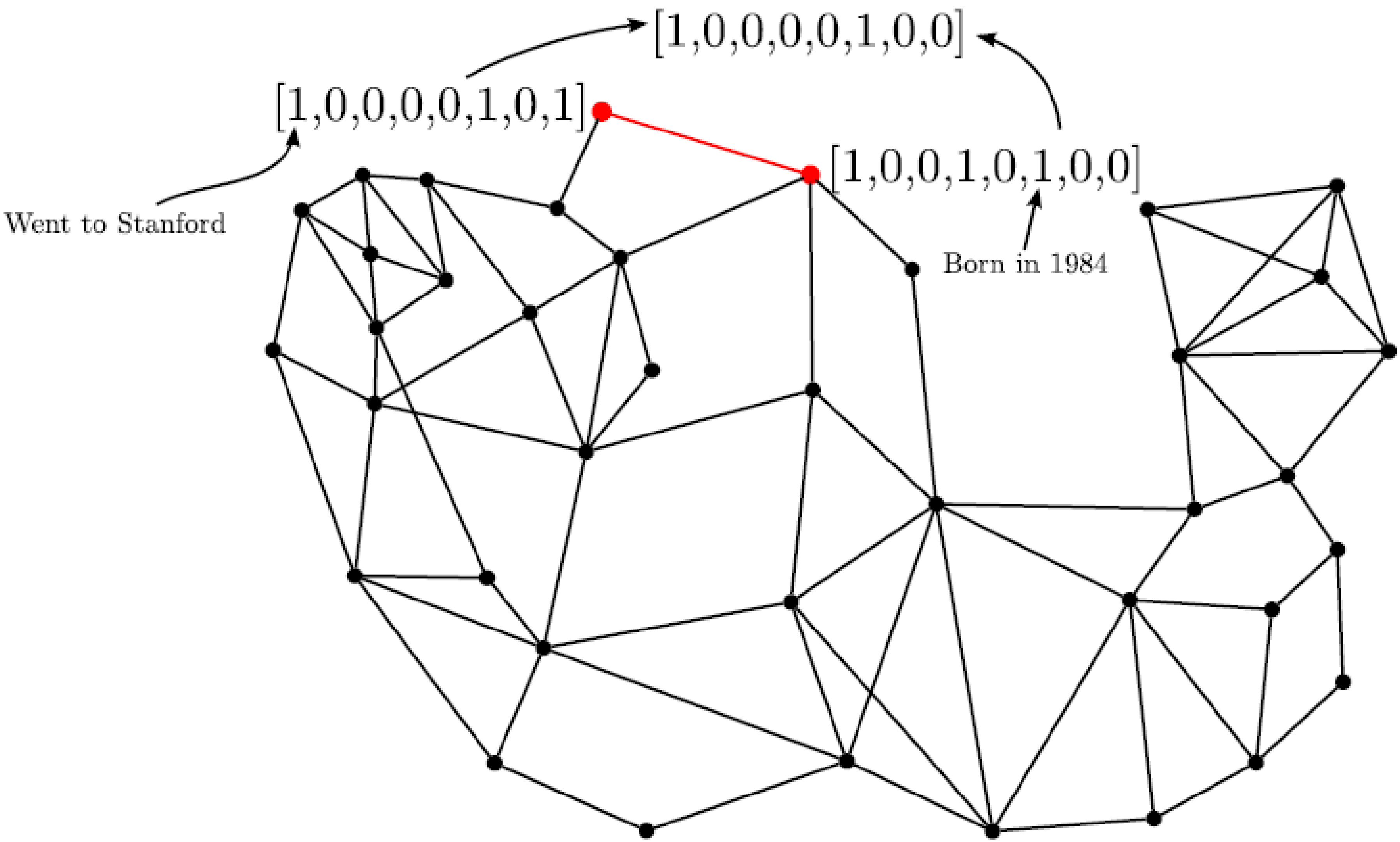


Extension: Organizing Friends



Discover circles and why they exist

Node Features



Model of Social Circles

- Circles arise due to a specific reason
- For a set of circles *c* model edge prob.:

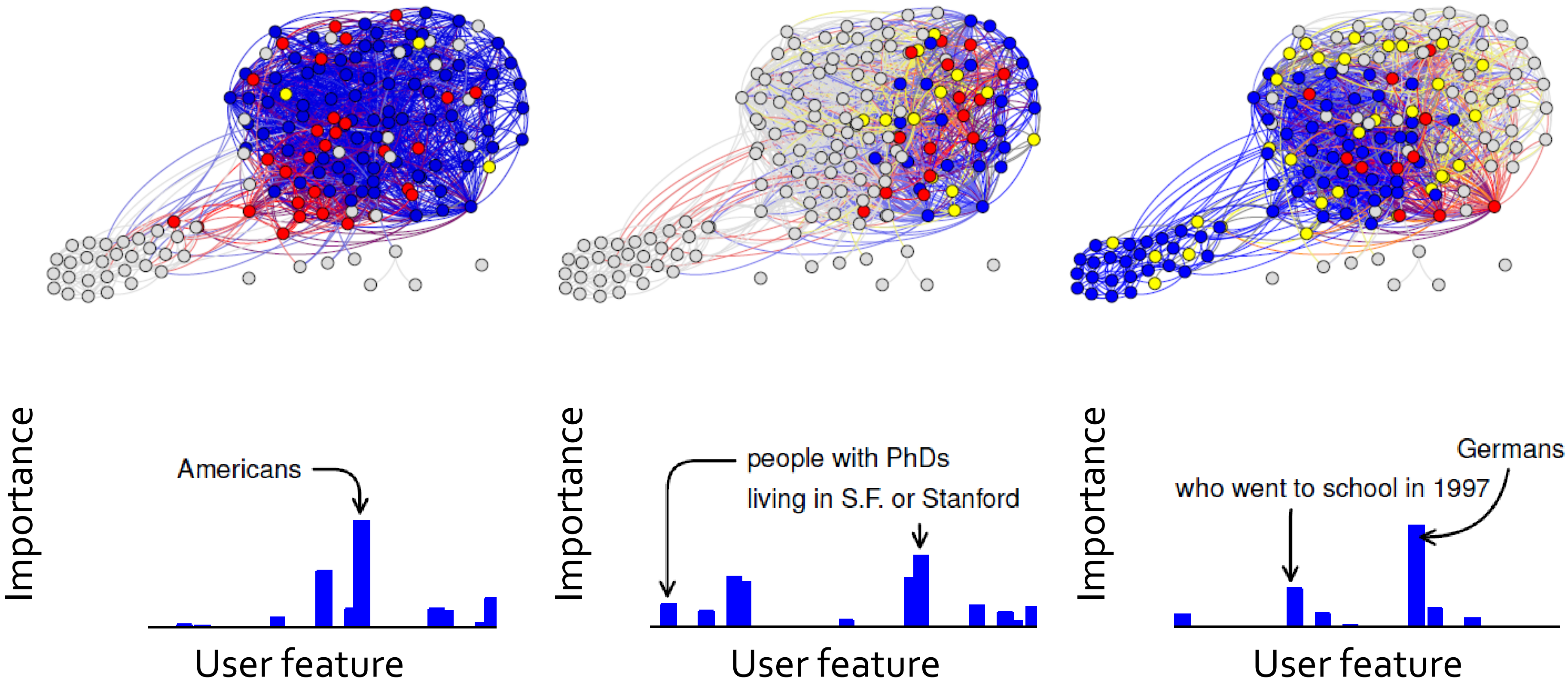
$$p(x, y) \propto \exp(\sum_i \theta_{ci} \cdot \phi_i(x, y))$$

  - $\psi(x, y)$ … edge feature vector describing (x, y)
  - θ_c… circle specific weight vector
- Example:

$$\phi(x,y) = \begin{array}{c|c} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} & \begin{array}{c} \text{Works at MSR} \\ \text{Studied at CMU} \\ \text{From UK} \\ \text{Born in London} \\ \text{Is catholic} \\ \text{Likes SciFi} \\ \text{Studied CS} \end{array} \end{array} \qquad \theta_c = \begin{bmatrix} 1.4 \\ 0.5 \\ 0 \\ 0 \\ 0 \\ 0.3 \\ 1.1 \end{bmatrix}$$

Extensions: Social Circles

- How well do we recover human circles?
- Social circles of a particular person:



Further Questions

Interesting research directions:

- Community detection in **dynamic networks**
  - Communities merge, split, are born, and die
- **Detecting communities of different structural types**
  - Cohesive vs. bipartite communities
- **Robustness/significance of communities**
  - Which communities in a network are "significant"?
- **Scaling to massive networks**

Networks: 3 problems

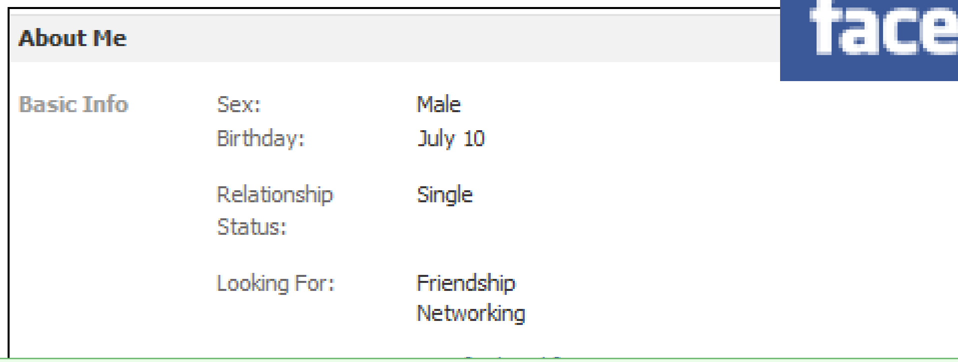
1) Community detection
2) Link & Attribute prediction
3) Social media

Finding Friends

- **What links will occur next?** [LibenNowell, Kleinberg ‘03]
  - **Networks + many other features:** Location, School, Job, Hobbies, Interests, etc.

Modeling Links in Networks

- Nodes in networks have rich ***attributes:***



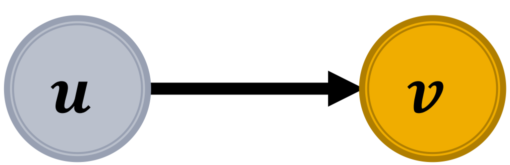
GOAL: Develop a ***model*** of links in a network that considers ***node attributes***

How do the node attributes form a network?

[Internet Math. '12]

Approach: Node attributes

- **Each node has a set of categorical attributes**
  - Gender: Male, Female
  - Home country: US, Canada, Russia, etc.
- **How do node attributes influence link formation?**
  - **Example:** MSN Instant Messenger



Chatting network

v's gender

u's gender

u \ v	FEMALE	MALE
FEMALE	0.3	0.7
MALE	0.7	0.3

Link probability

Link-Affinity Matrix

- Let the values of the ***i-th attribute*** for node u and v be $\boldsymbol{a_i(u)}$ and $\boldsymbol{a_i(v)}$
  - $a_i(u)$ and $a_i(v)$ can take values $\{0, \cdots, d_i - 1\}$
- **Question: How can we capture the influence of the attributes on link formation?**
  - Insight: ***Attribute link-affinity matrix*** $\boldsymbol{\Theta}$

	$a_i(v) = 0$	$a_i(v) = 1$
$a_i(u) = 0$	$\boldsymbol{\Theta[0,0]}$	$\boldsymbol{\Theta[0,1]}$
$a_i(u) = 1$	$\boldsymbol{\Theta[1,0]}$	$\boldsymbol{\Theta[1,1]}$

$$\boxed{\boldsymbol{P(u,v) = \Theta[a_i(u), a_i(v)]}}$$

  - Each entry captures the ***affinity of a link*** between two nodes associated with the attributes of them

Attribute Interactions

- ***MAG modeling flexibility:***
  - ***Homophily*** : love of the ***same***
    - e.g., political views, hobbies

0.9	0.1
0.1	0.8

  - ***Heterophily*** : love of the ***opposite***
    - e.g., genders

0.2	0.9
0.9	0.1

  - ***Core-periphery*** : love of the ***core***
    - e.g. extrovert personalities

0.9	0.5
0.5	0.2

[Internet Math. '12]

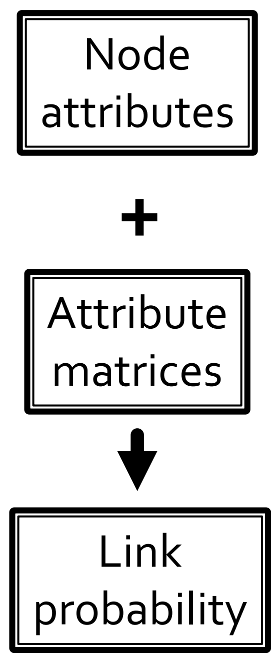
From Attributes to Links

- ***How do we combine the effects of multiple attributes?***
  - We ***multiply the probabilities*** from all attributes

$$a(u) = [\ 0 \quad 0 \quad 1 \quad 0\ ]$$
$$a(v) = [\ 0 \quad 1 \quad 1 \quad 0\ ]$$

$$\Theta_i = \begin{bmatrix} \alpha_1 & \beta_1 \\ \beta_1 & \gamma_1 \end{bmatrix} \begin{bmatrix} \alpha_2 & \beta_2 \\ \beta_2 & \gamma_2 \end{bmatrix} \begin{bmatrix} \alpha_3 & \beta_3 \\ \beta_3 & \gamma_3 \end{bmatrix} \begin{bmatrix} \alpha_4 & \beta_4 \\ \beta_4 & \gamma_4 \end{bmatrix}$$

$$P(u,v) = \alpha_1 \times \beta_2 \times \gamma_3 \times \alpha_4$$



Multiplicative Attribute Graph

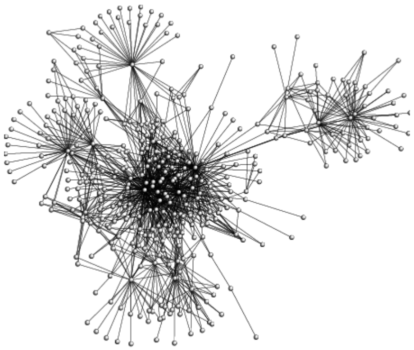
- **The MAG model** $M(n, l, A, \vec{\Theta})$
  A network contains n **nodes**
  - Each node has l **categorical attributes**
  - $A = [a_i(u)]$ represents the ***i*-th attribute of node** u
  - Each attribute can take d_i **different values**
  - Each attribute has a $d_i \times d_i$ **link-affinity matrix** Θ_i
  - Edge probability between nodes u and v

$$\boxed{P(u, v) = \prod_{i=1}^{l} \Theta_i[a_i(u), a_i(v)]}$$

Fitting the MAG model

- **Find model parameters from the data**
  - **Given:**

    Links of the network
  - **Estimate:**
    - **Latent** node attributes
    - Link-affinity matrices
- Formulate as a maximum likelihood problem
- **Solve it using variational EM**

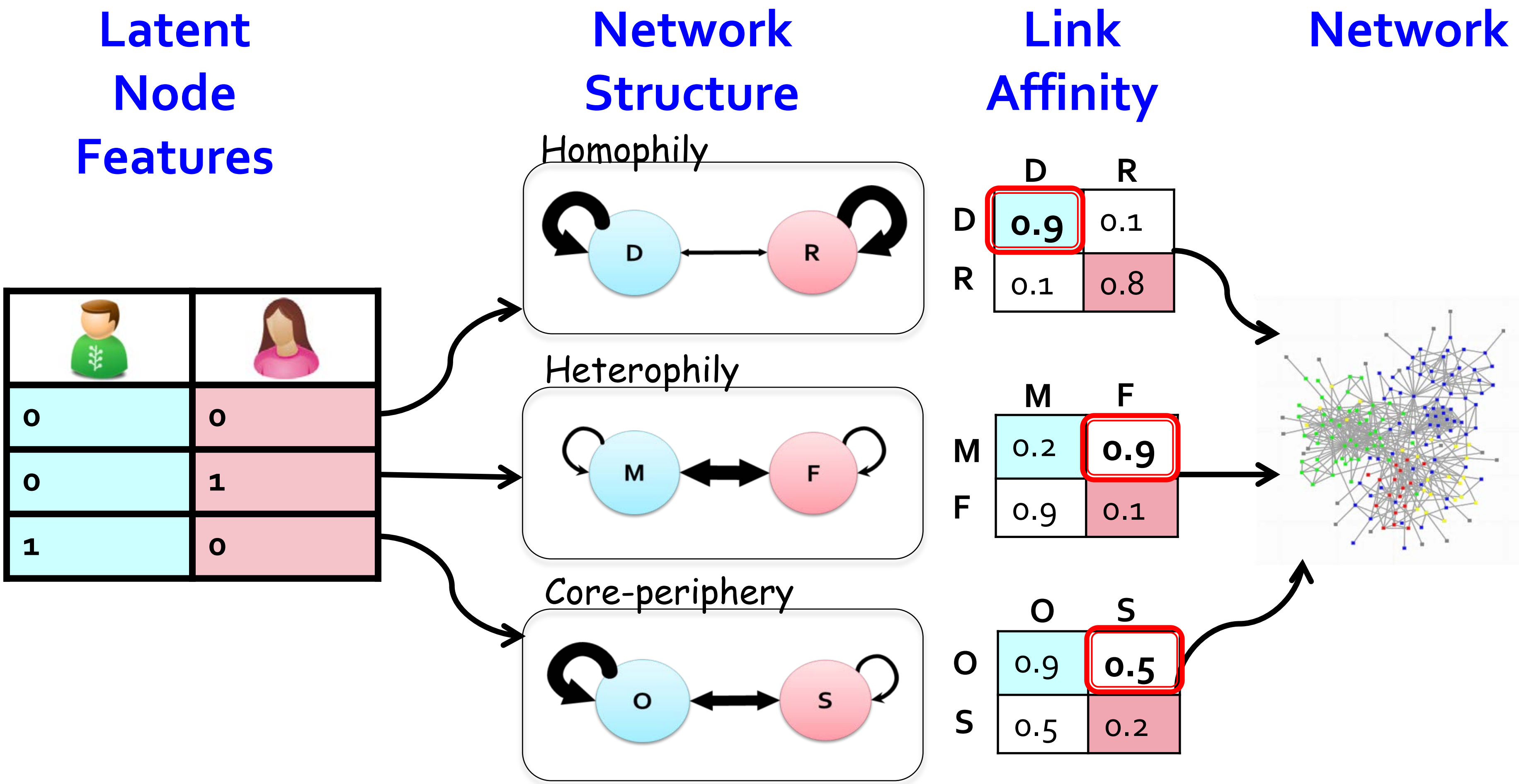


$$\boldsymbol{a(u)} = [\ldots]$$

$$\boldsymbol{\Theta_i} = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.8 \end{bmatrix}$$

Fitting MAG to Data

[UAI. '11]



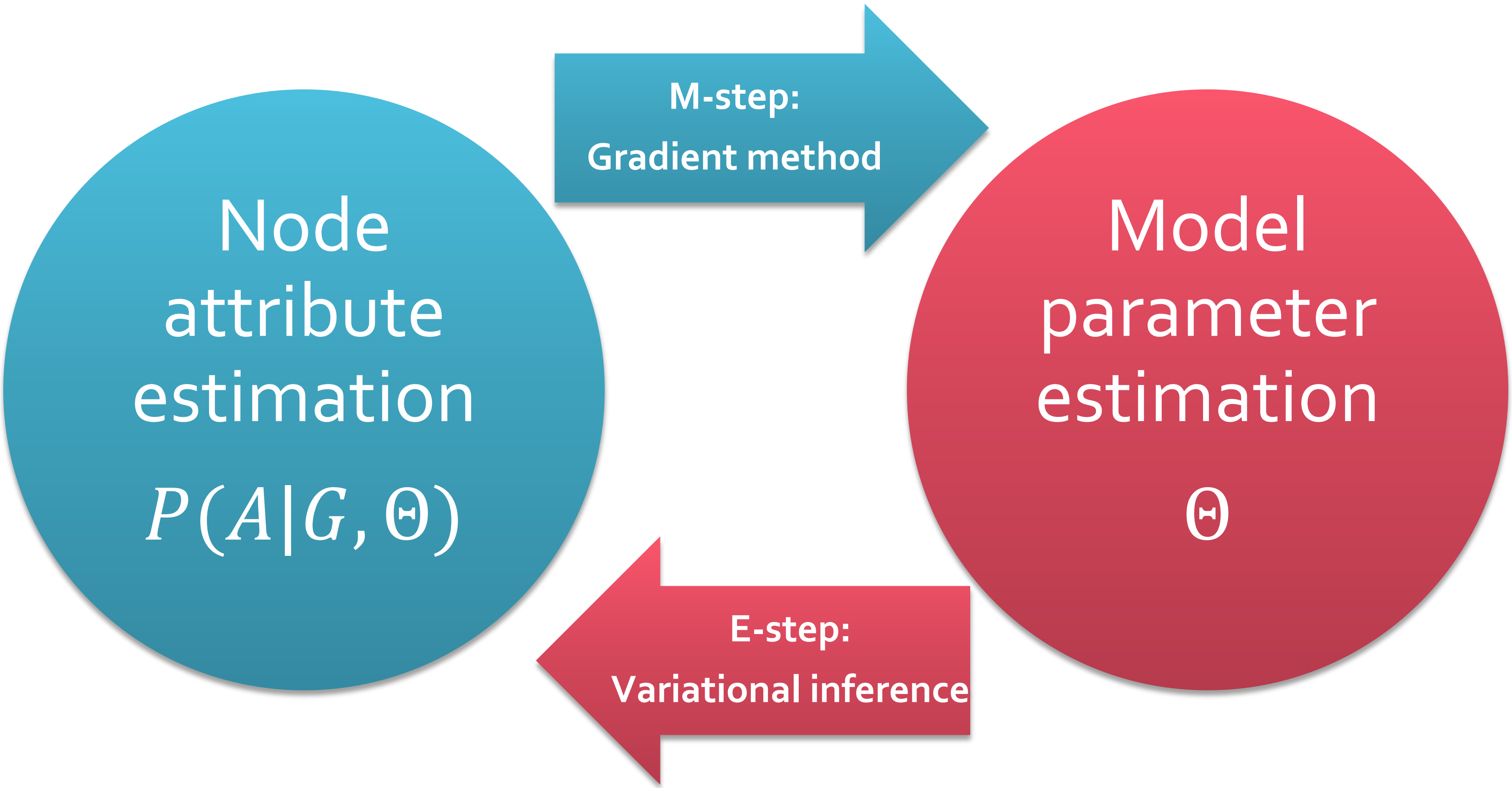
$$P(\;\to\;) = 0.9 \times 0.9 \times 0.5$$

Fitting the MAG model

- **Edge probability:**
  - $\boldsymbol{P(u,v) = \prod_{i=1}^{l} \Theta_i[a_i(u), a_i(v)]}$
- **Network likelihood:**
  - $P(G|A,\Theta) = \prod_{G_{uv}=1} P(u,v) \cdot \prod_{G_{uv}=0} 1 - P(u,v)$
    - G … graph adjacency matrix
    - A … matrix of node attributes
    - Θ… link-affinity matrices
- **Want to solve:**
  - $\arg\max_{A,\Theta} P(G|A,\Theta)$

Variational EM

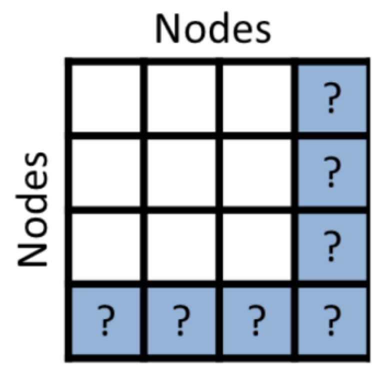
[UAI. '11]



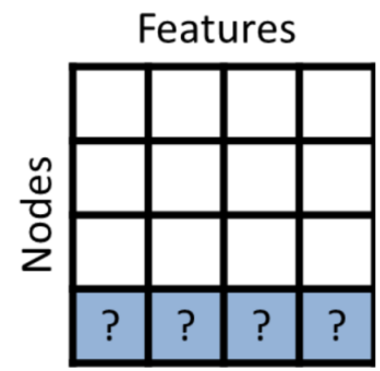
[ICML. '12]

Predictive Tasks in Networks

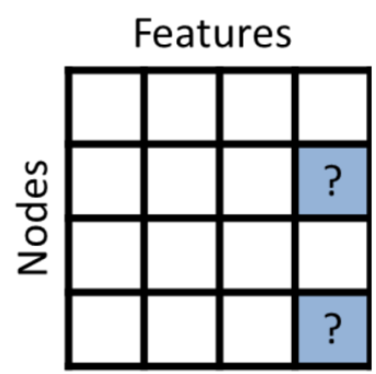
- **Predictive tasks:**
  - **Predict missing links**
    - Predict future friends



  - **Predicting node feature values**
    - Infer user profile features



  - **Node classification**
    - Predict users from China



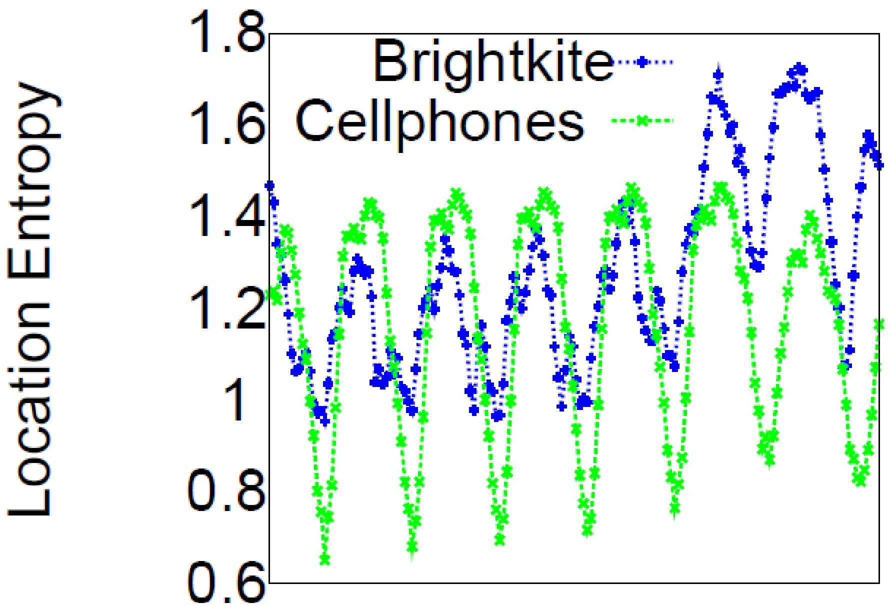
Beyond Static Attributes

- **Dynamic network attributes:**
  - **Location and social networks**

- **Examples:**
  - **Location-based online social networks**
    - Foursquare, Yelp, Brightkite, Gowalla
  - **Cell phones**

Modeling Mobility

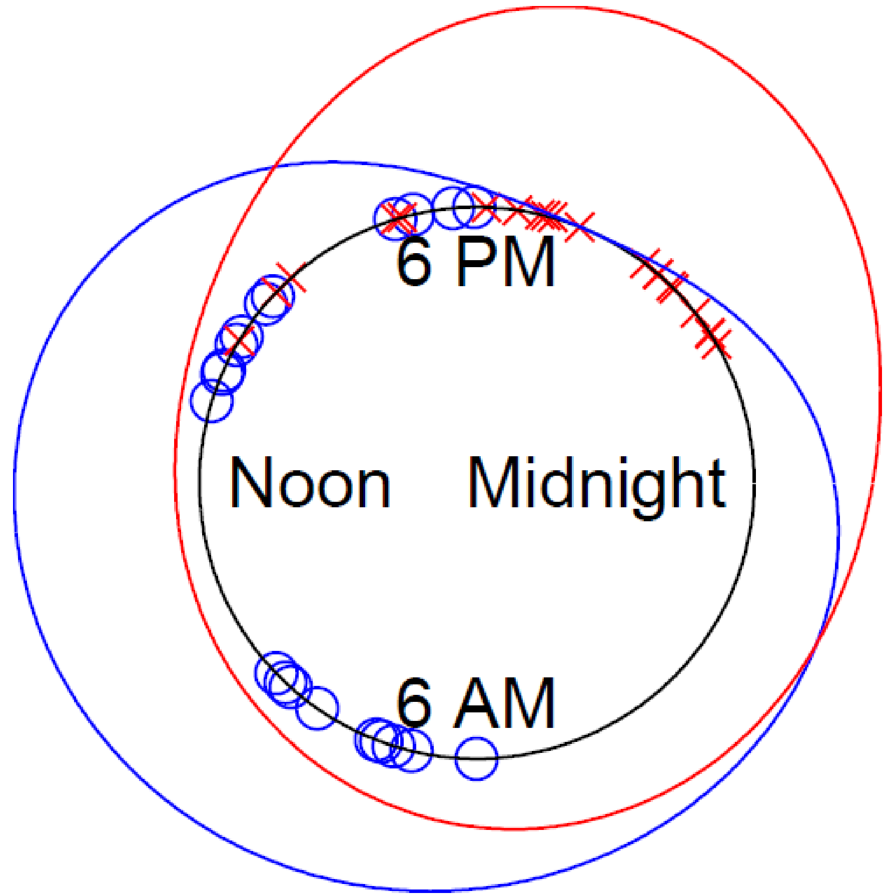
- **Goal:** Model and predict human mobility patterns



- **Observation:**
  - Low location entropy at night/morning
  - Higher entropy over the weekend
- **3 ingredients of the model:**
  - **Spatial, Temporal, Social**

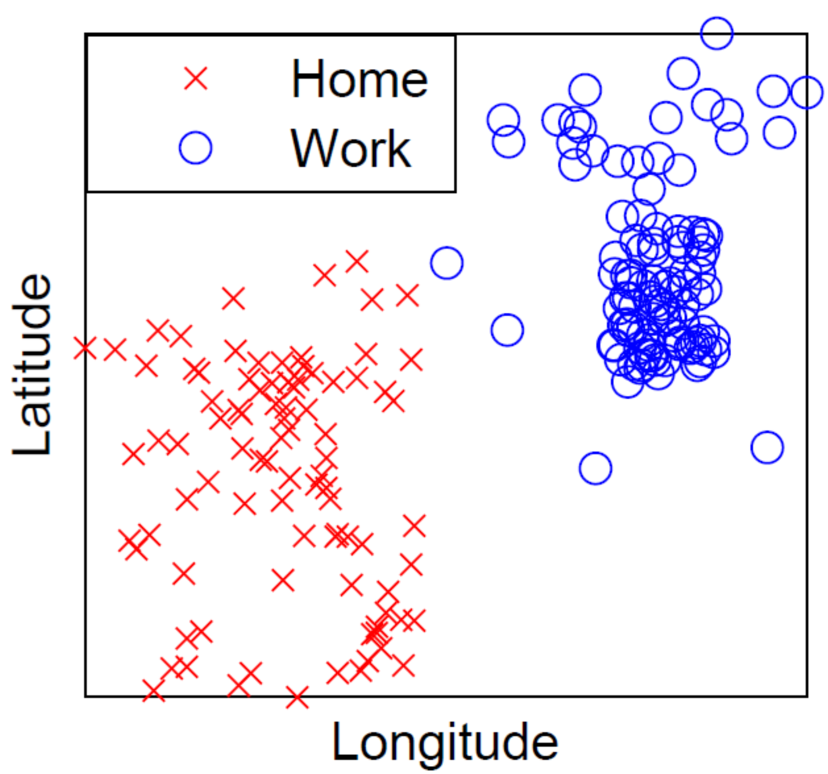
Modeling Mobility

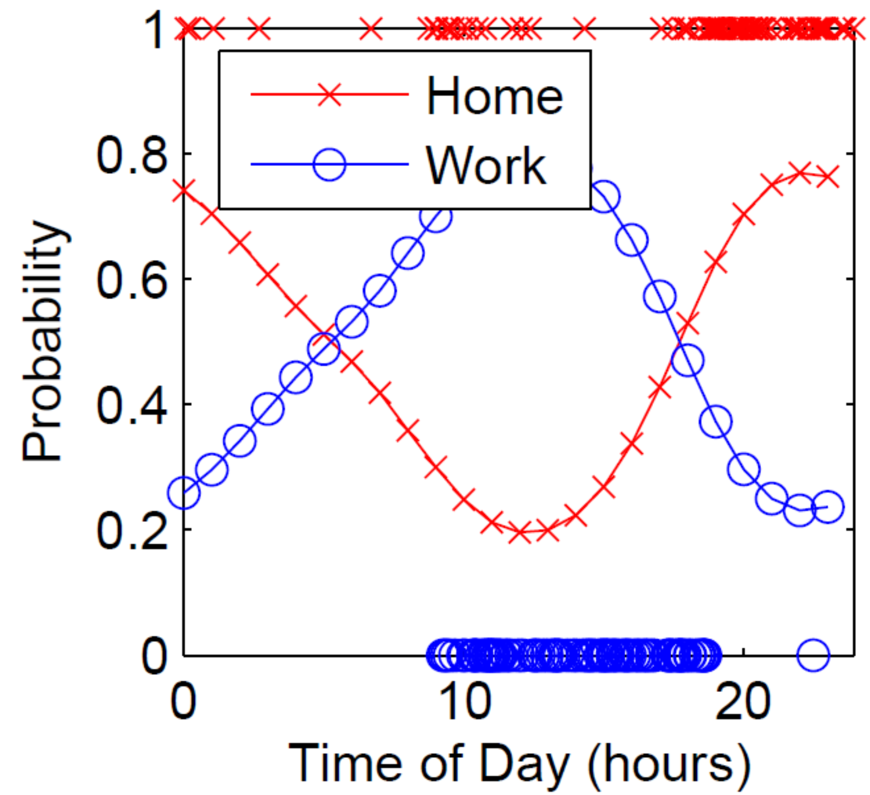
Spatial model:
Home vs. Work Location

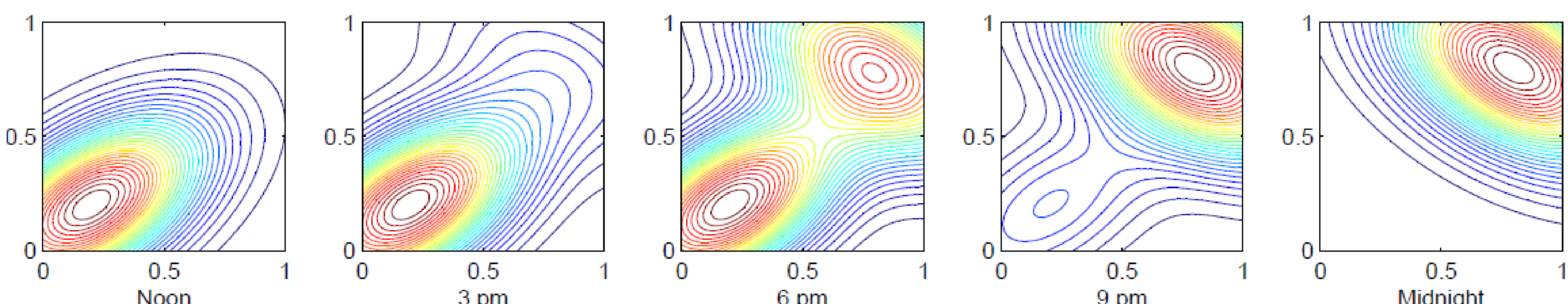


Temporal model:
Mobility Home vs. Work

Example User

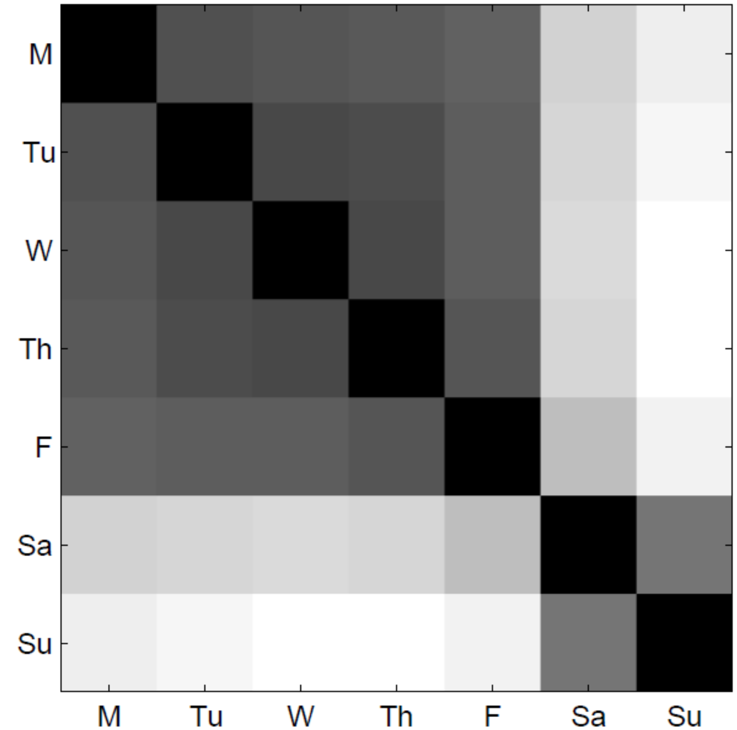






Weekend Mobility

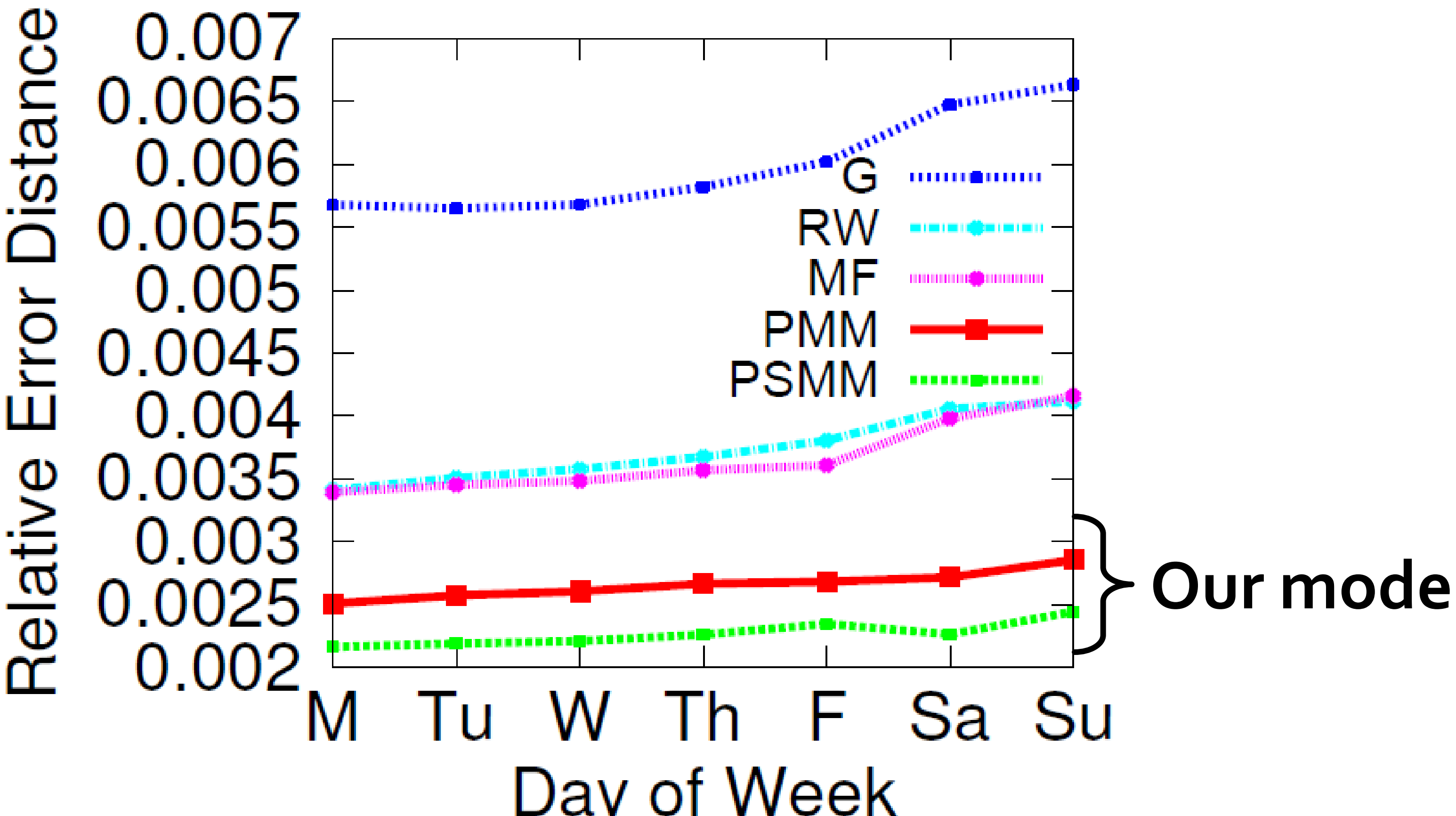
- **Social network plays particularly important role on weekends**
- **Include social network into the model**
  - Prob. that user visits location ***X*** depends on:
    - Distance(***X***, ***F***)
    - Time since a friend was at location ***F***
      - ***F*** = Friend's last known location



Mobility similarity

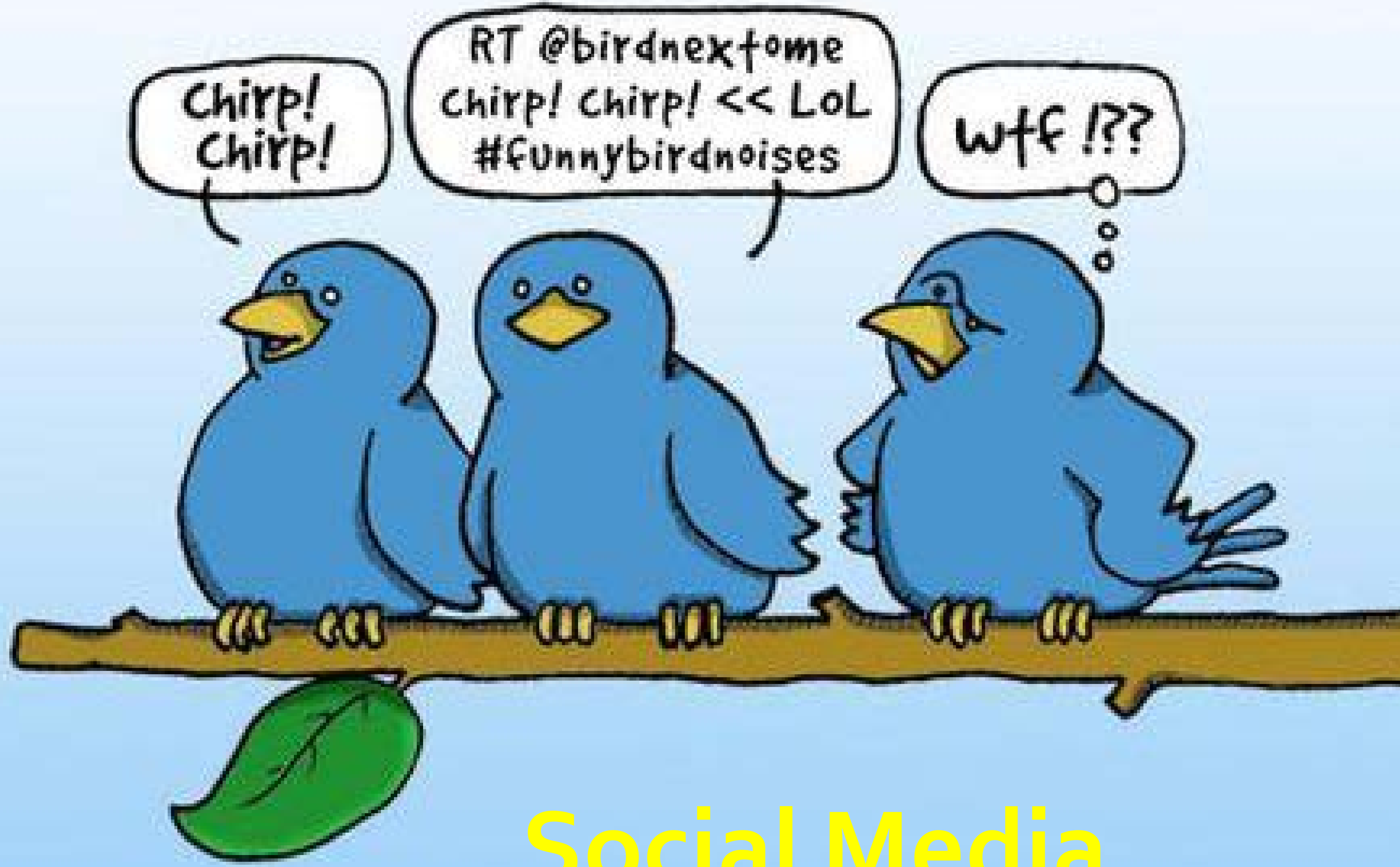
Mobility: Results

- **Cellphones:** Whenever user receives or makes a call predict her location



Networks: 3 problems

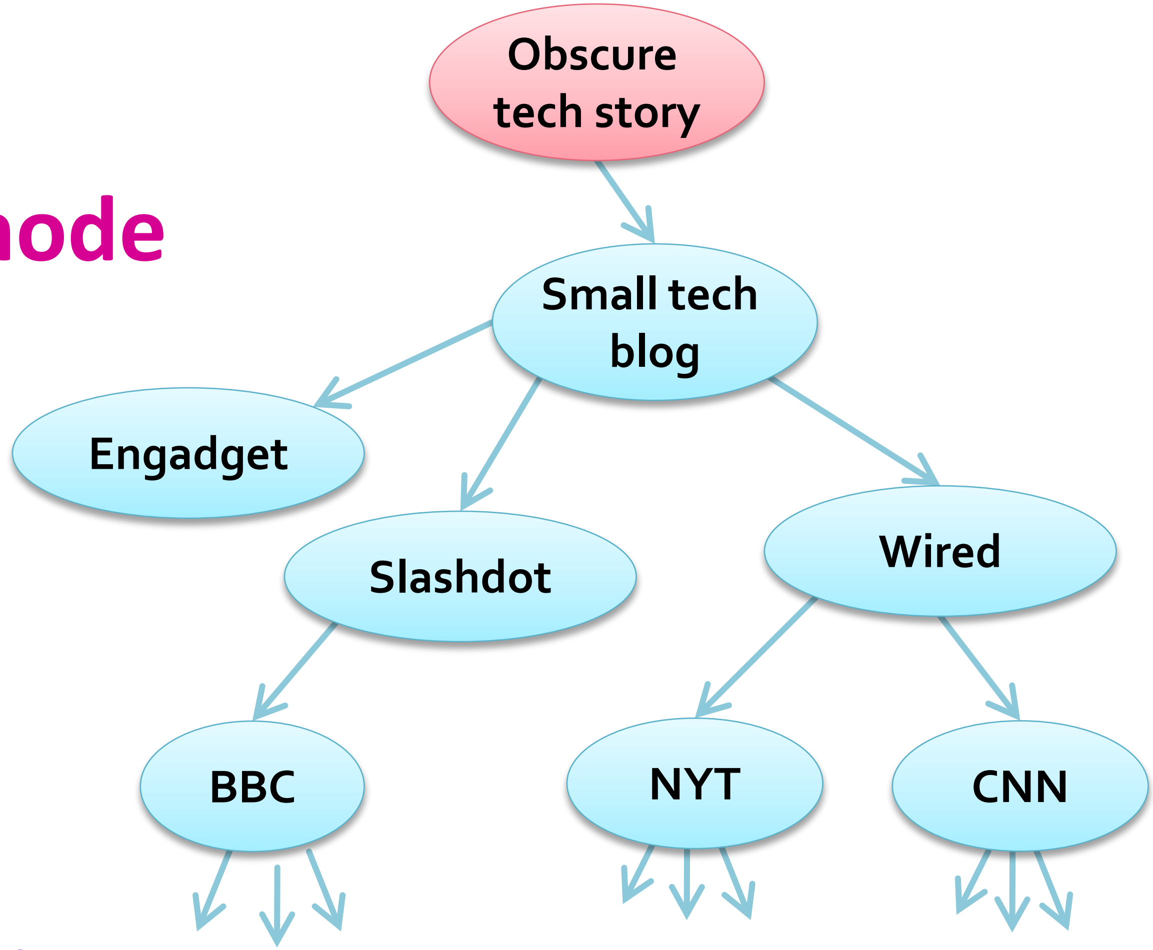
1) Community detection
2) Link & Attribute prediction
3) Social media



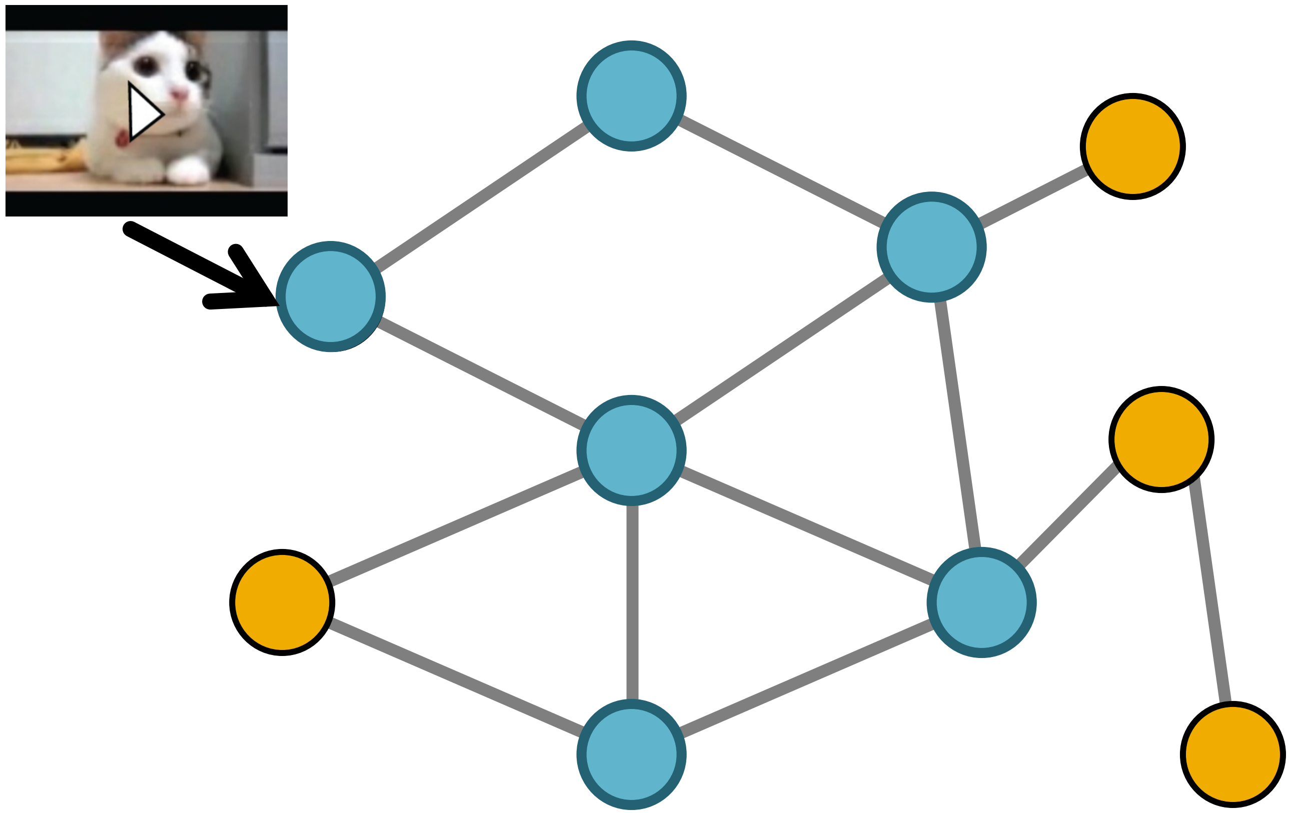
Social Media

Diffusion in Networks

- Information flows from a node to node like an epidemic
- How does information transmitted by mainstream media interact with social networks?



Information Flows through Links



Information spreads over the links of the network

Diffusion in Online Media

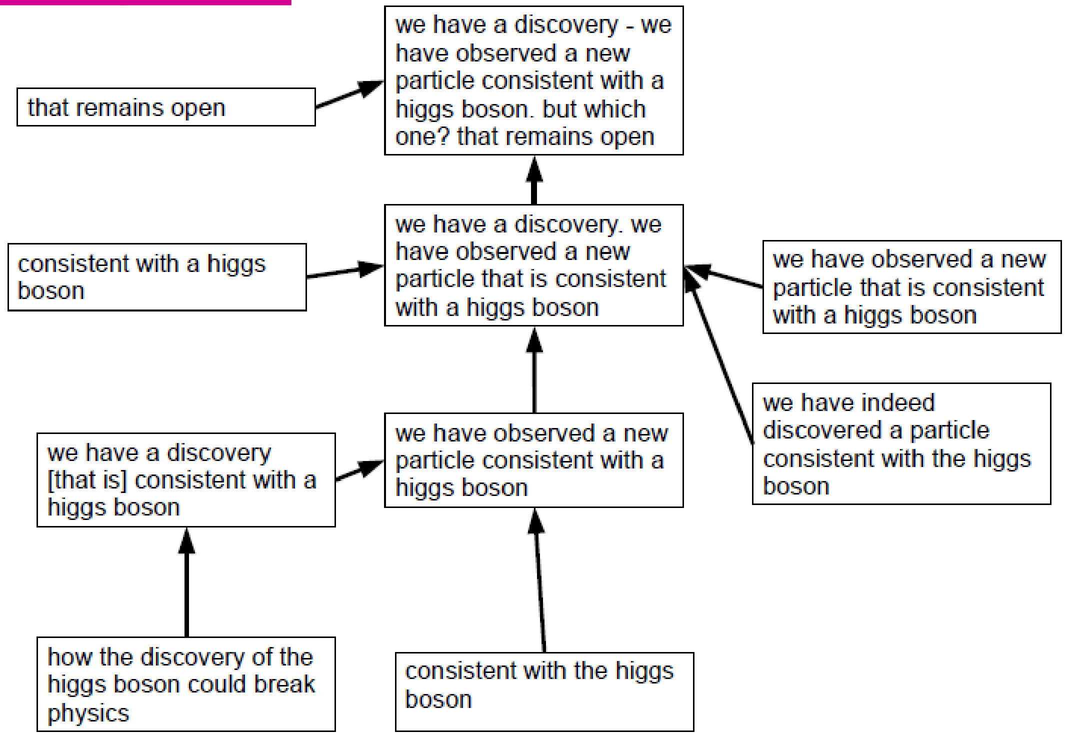
- **Since August 2008 we have been collecting 30M articles/day:** 6B articles, 20TB of data
- **Challenge:**

  How to track information as it spreads?

Meme-tracking

[WWW '13]

- **Goal:** **Trace textual phrases that spread through many news articles**
- **Challenge 1: Phrases mutate!**

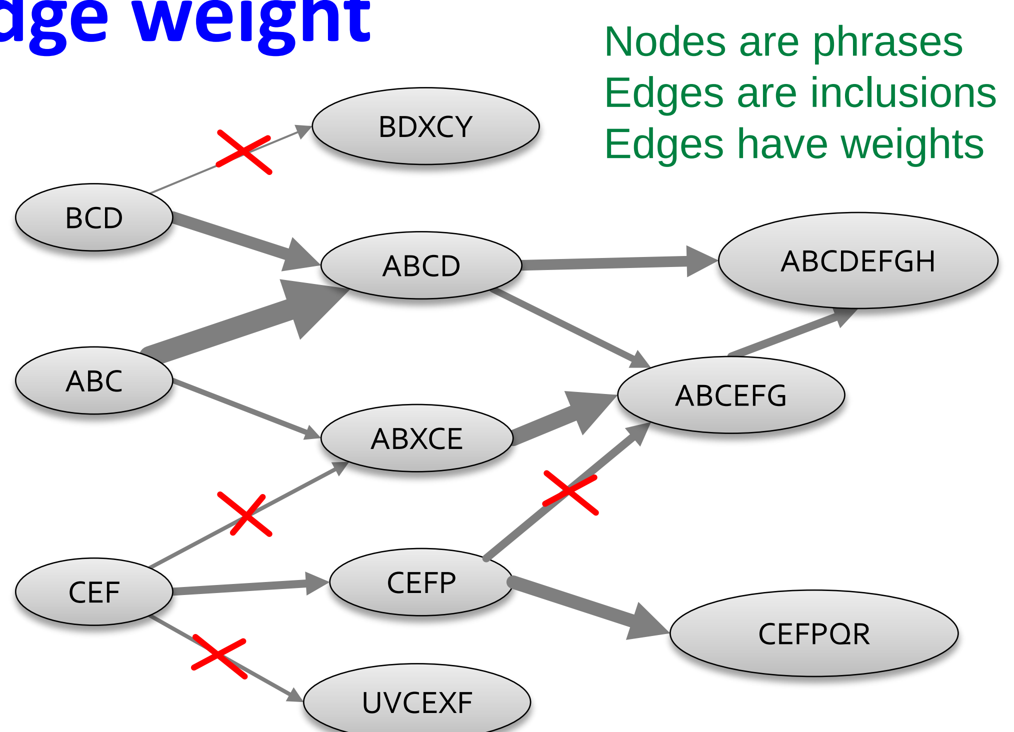


Mutations of a meme about the **Higgs boson particle**.

Finding Mutational Variants

[KDD '09]

- **Goal:** Find mutational variants of a phrase
- **Objective:**
  - In a DAG of approx. phrase inclusion, **delete min total edge weight** such that **each component has a single "sink"**



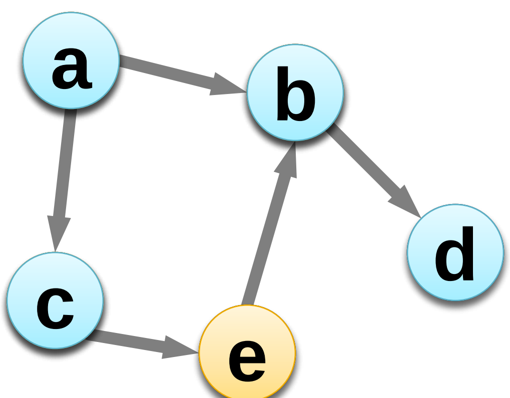
Memes over Time

Visualization of 1 month of data from October 2012

- **Browse all 4 years of data at** http://snap.stanford.edu/nifty

Inferring Diffusion Networks

- Challenge 3: Information network is hidden
- **Goal: Infer the information diffusion network**
  - There is a **hidden** network, and
  - We only see **times** when nodes get "infected"



- **Yellow** info: (a,1), (c,2), (b,3), (e,4)
- **Blue** info: (c,1), (a,4), (b,5), (d,6)

[KDD '10]

Inferring Networks

	Virus propagation	Word of mouth & Viral marketing
Process	Viruses propagate through the network	Recommendations and influence propagate
We observe	We only observe when people get sick	We only observe when people buy products
It's hidden	But NOT who **infected** them	But NOT who **influenced** them

Can we infer the underlying network?

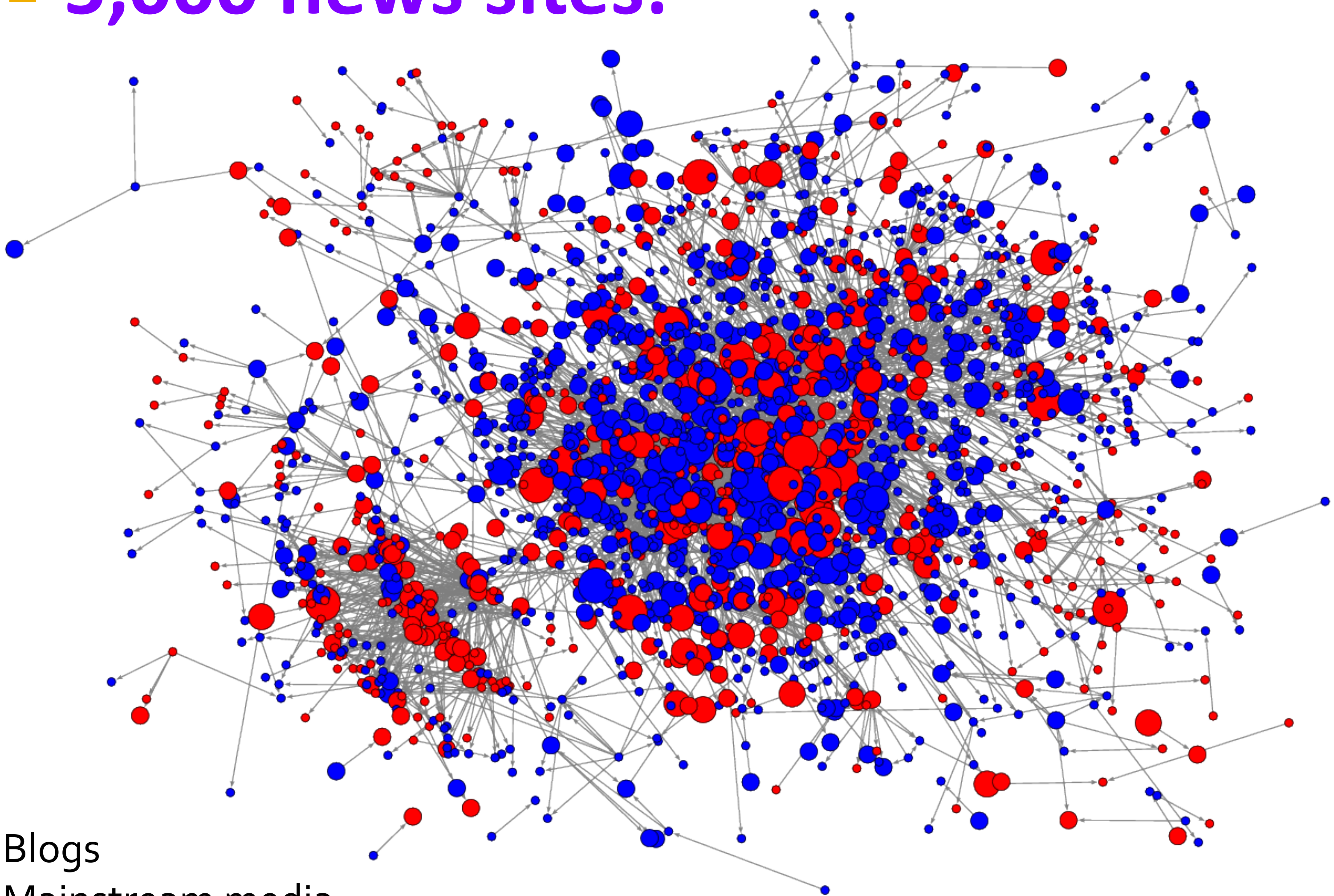
Yes, convex optimization problem!

[Gomez-Rodriguez, L., Krause, '10, Myers, L., '10]

News Diffusion Network

[KDD '10]

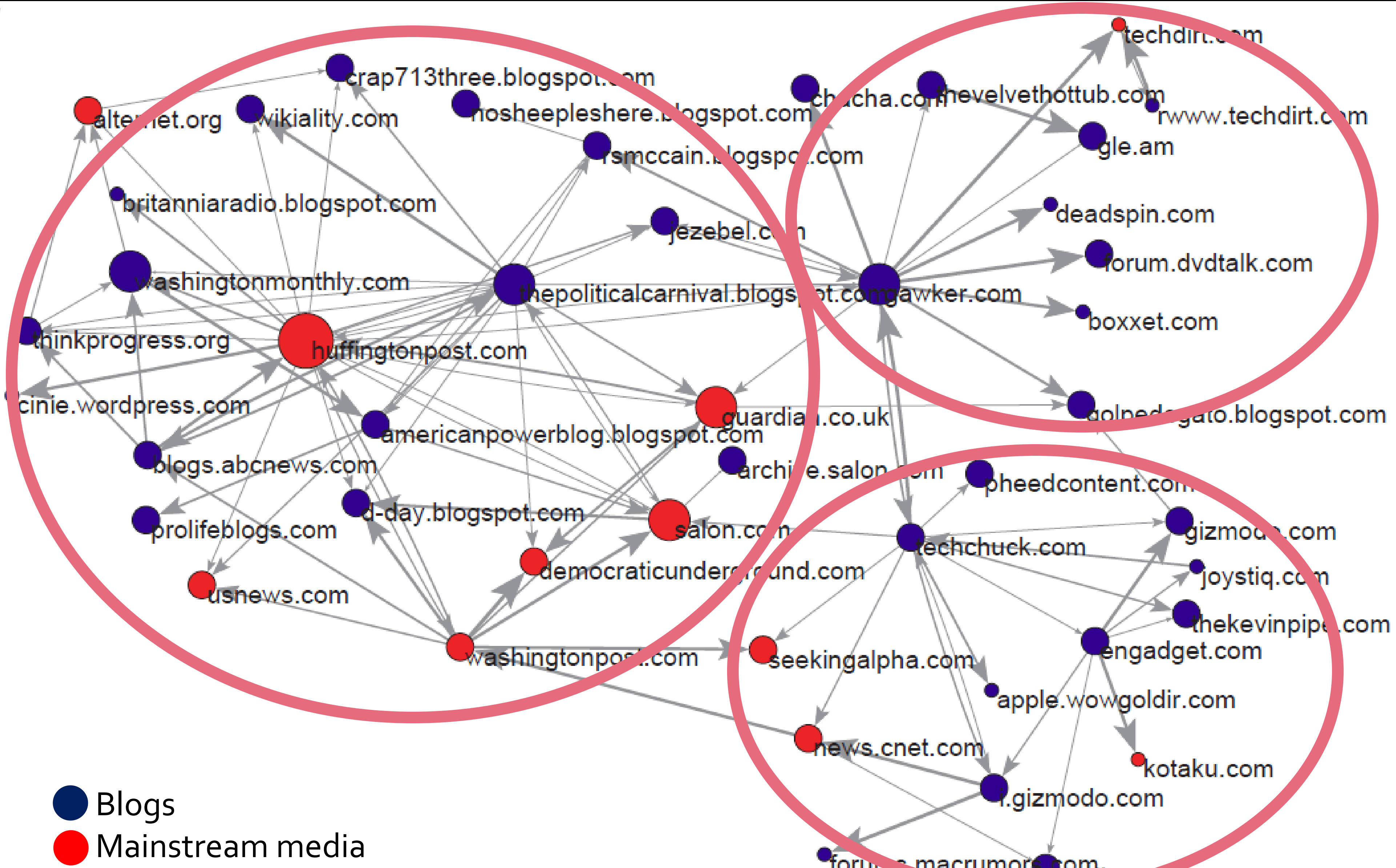
- **5,000 news sites:**



Blogs
Mainstream media

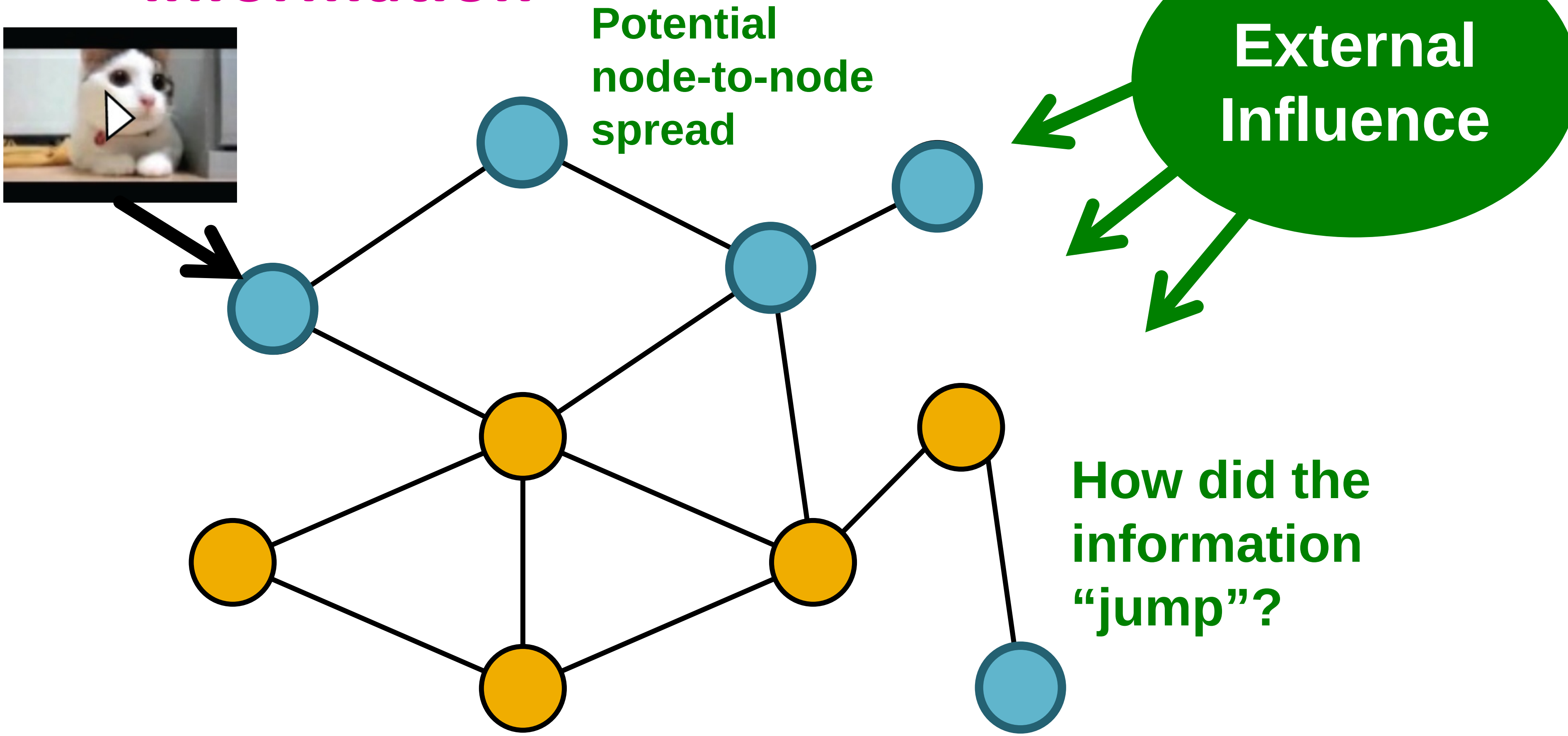
News Diffusion Network

[KDD '10]



Information in Networks

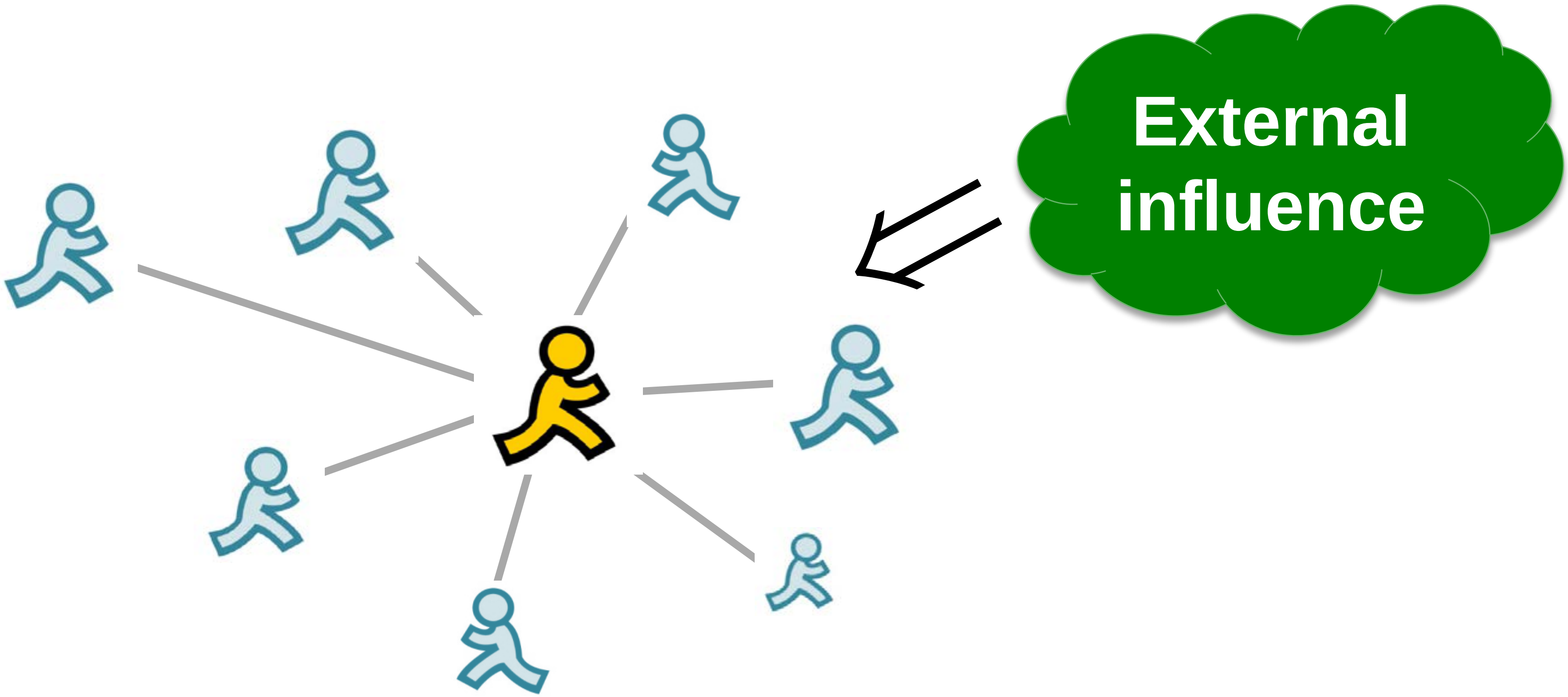
- **Observe times when nodes adopt the information**



Exposures and Adoptions

- **Exposure**: When a node sees a contagion, whether from a neighbor's adoption or elsewhere
- **Adoption**: The node posts the contagion for her neighbors to see

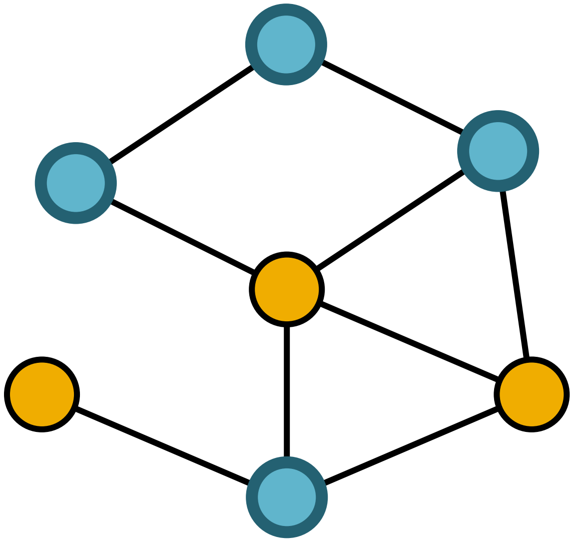
Network & External Exposures



- **Two sources of exposures:**
  - Exposures from the network
  - External exposures

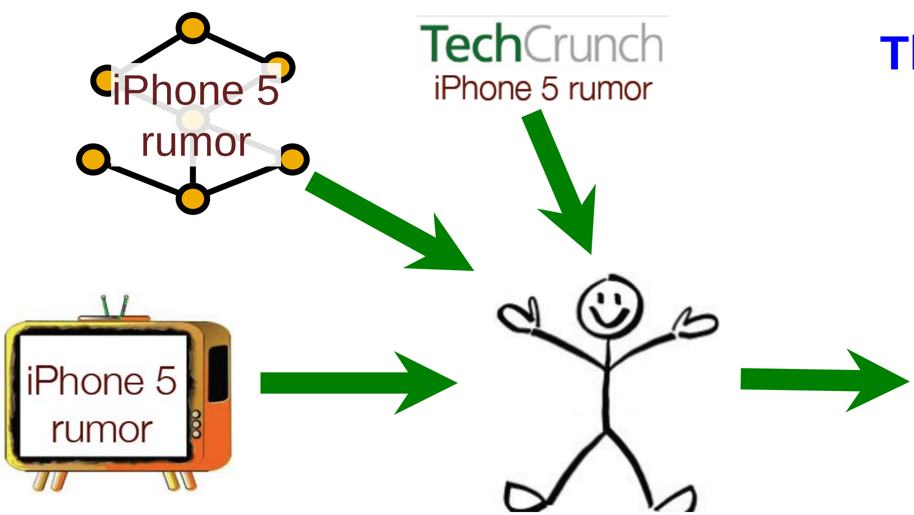
Why is it important?

- **Why separating network effects from the external influence?**
  - Detecting external events
  - Estimating information virality
  - Building better models of diffusion
  - Better targeting and influence maximization



Why is it hard?

- **Why is modeling external influence hard?**
  - External sources are unobservable
  - Amount of external influence varies over time
  - External influence can be confused with network influence



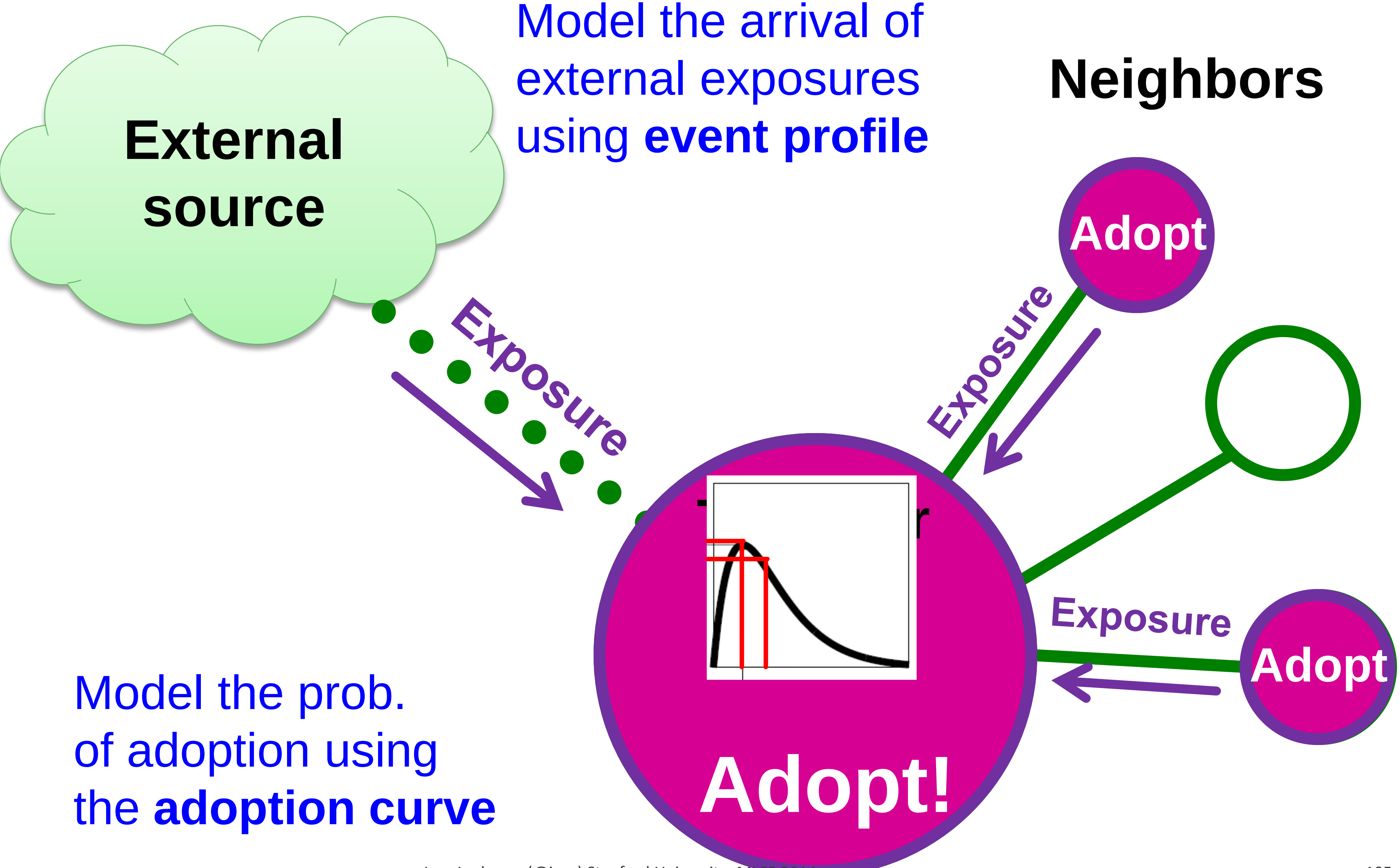
The post occurred both due to external and network effects!

Posts the rumor

Towards the Model

[KDD '12]

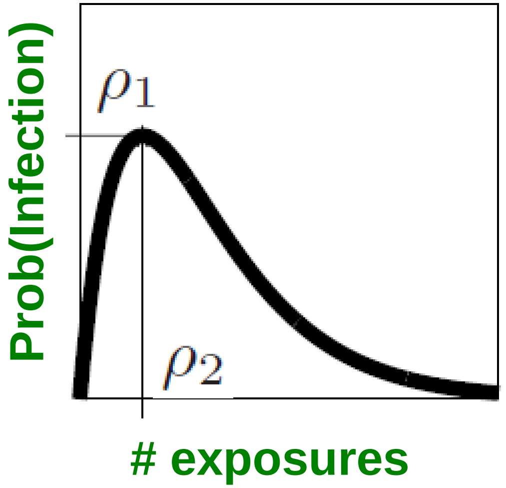
Model the arrival of external exposures using **event profile**



Model the prob. of adoption using the **adoption curve**

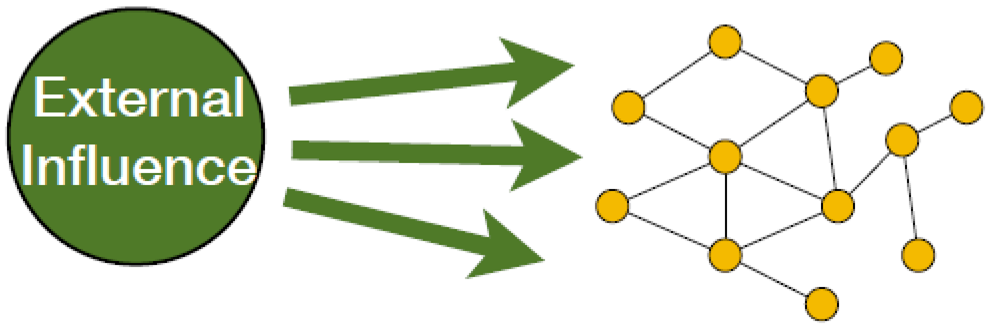
Adoption Curves

- **From exposures** to **adoptions**
  - **Exposure**: Node is exposed to information
  - **Adoption**: The node acts on the information
- **Adoption curve:** $\eta(x) = \frac{\rho_1}{\rho_2} \cdot x \cdot \exp\left(1 - \frac{x}{\rho_2}\right)$



Modeling External Influence

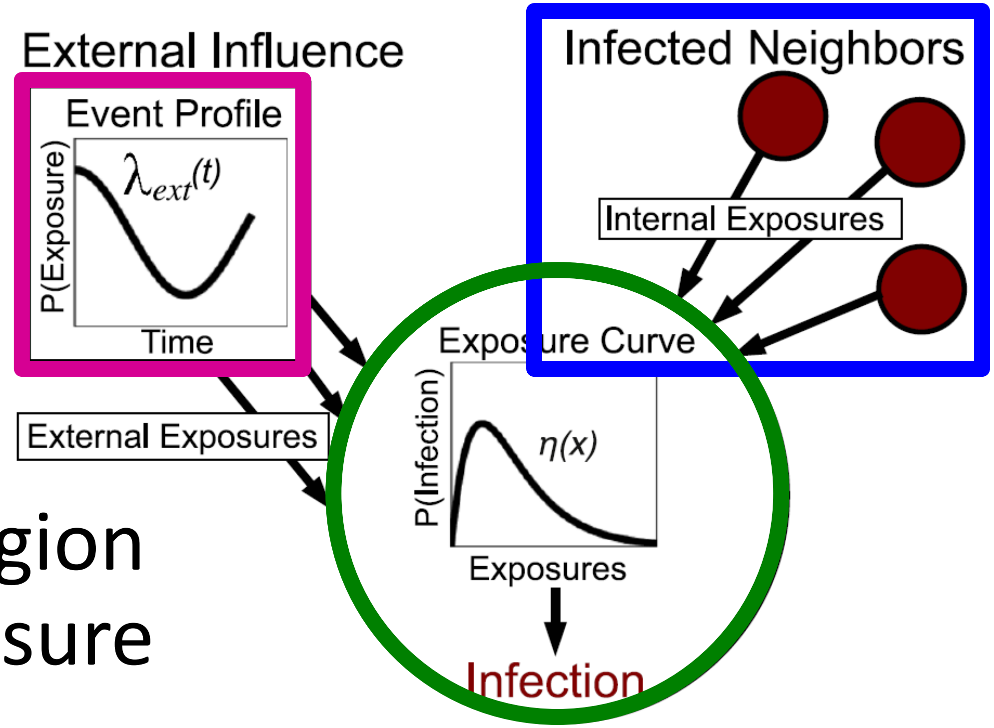
- **Assume an external source generating exposures uniformly across the network**



- **Event profile**
  - $\lambda_{ext}(t) = \mathrm{P}\left[\begin{array}{c}\text{any user receiving an} \\ \text{external exposure at time } t\end{array}\right]$
    - For each t_i we have a separate parameter $\lambda_{ext}(t_i)$

Putting it all together

- User receives **external exposures** by the event profile
- Each **neighbor** that posts the contagion also creates an exposure
- With each exposure, the **adoption curve** is sampled: Does the user adopt the contagion?



Objective Function

- **Prob. that user *i* adopted contagion**

$$F^{(i)}(t) = P(\ i \text{ has adopted contagion by } t)$$

$$= \sum_{n=1}^{\infty} P(\ i \text{ has } n \text{ exposures at } t) \times \left[1 - \prod_{k=1}^{n} [1 - \eta(k)]\right]$$

At least one exposure lead to adoption

- **Where:**

Total internal exposures

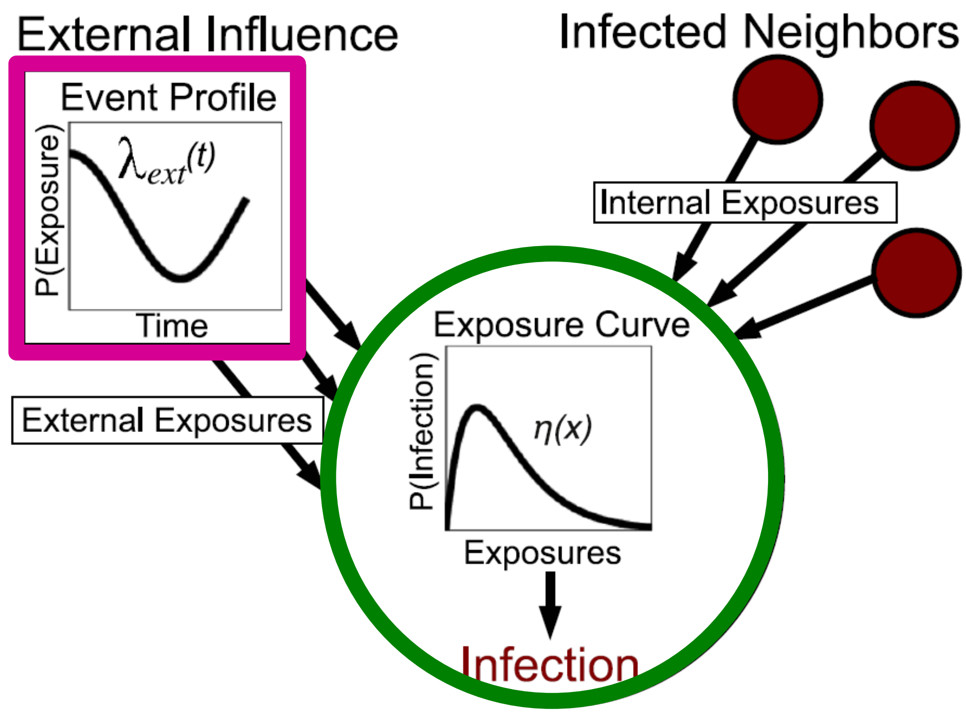
Total external exposures

$$P(\ i \text{ has } n \text{ exposures at } t\ ) \approx \binom{t/dt}{n} \left(\frac{\Lambda_{int}^{(i)}(t) + \Lambda_{ext}(t)}{t} \cdot dt \right)^{n}$$

$$\times \left(1 - \frac{\Lambda_{int}^{(i)}(t) + \Lambda_{ext}(t)}{t} \cdot dt \right)^{t/dt - n}$$

Model Inference Task

- **Given:**
  - Network G
  - Node adoption times (i, t) of a contagion
- **Goal: Infer**
  - **(1) External event profile**
  - **(2) Adoption curve**



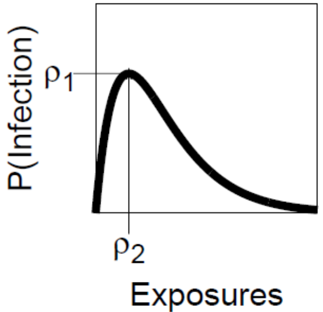
such that observed adoption times fit best

[KDD '12]

Results: Different Topics

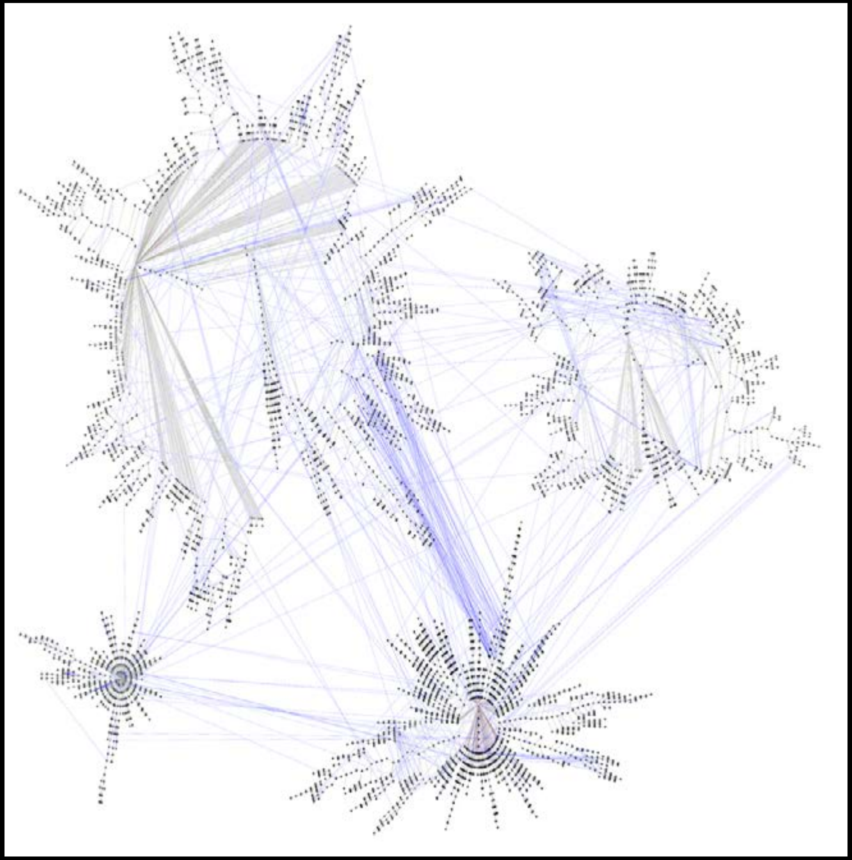
- **Complete data from Jan 2011: 3 billion tweets**

	max P(k)	k at max P(k)	Duration (hours)	% Ext. Exposures
Politics (25)	0.0007 +/- 0.0001	4.59 +/- 0.76	51.24 +/- 16.66	47.38 +/- 6.12
World (824)	0.0013 +/- 0.0000	2.97 +/- 0.10	43.54 +/- 2.94	26.07 +/- 1.19
Entertain. (117)	0.0015 +/- 0.0002	3.52 +/- 0.28	89.89 +/- 16.13	17.87 +/- 2.51
Sports (24)	0.0010 +/- 0.0003	4.76 +/- 0.83	87.85 +/- 38.03	43.88 +/- 6.97
Health (81)	0.0016 +/- 0.0002	3.25 +/- 0.30	100.09 +/- 17.57	18.81 +/- 3.33
Tech. (226)	0.0013 +/- 0.0001	3.00 +/- 0.16	83.05 +/- 8.73	18.36 +/- 1.80
Business (298)	0.0015 +/- 0.0001	3.18 +/- 0.16	49.61 +/- 5.14	22.27 +/- 1.79
Science (106)	0.0012 +/- 0.0002	4.06 +/- 0.30	135.28 +/- 16.19	20.53 +/- 2.78
Travel (16)	0.0005 +/- 0.0001	2.33 +/- 0.29	151.73 +/- 39.70	39.99 +/- 6.60
Art (32)	0.0006 +/- 0.0001	5.26 +/- 0.66	188.55 +/- 48.17	27.54 +/- 5.30
Edu. (31)	0.0009 +/- 0.0001	3.77 +/- 0.51	130.53 +/- 38.63	21.45 +/- 6.40



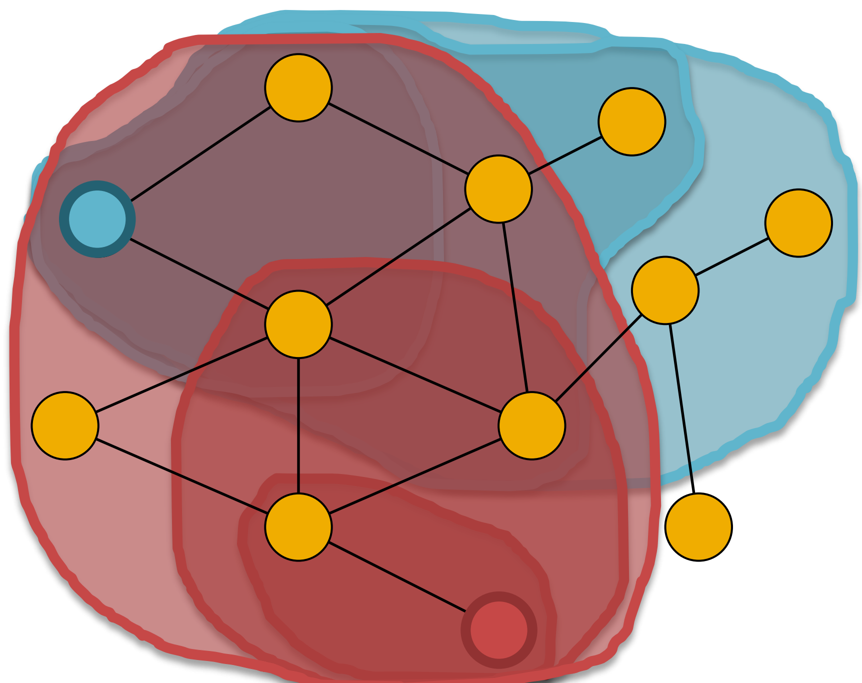
More details: S. Myers, C. Zhu, J. Leskovec: Information diffusion and external influence in networks, *KDD* 2012.

How about Interactions between cascades?



Contagion Interactions

- So far we considered contagions as **independently** propagating
- **How do contagions interact?**
  - Does being exposed to **blue** change the probability of talking about **red** contagion?

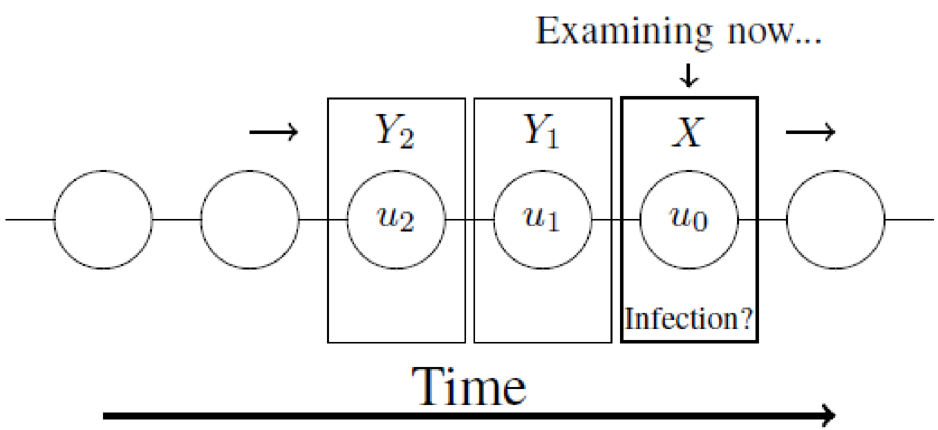


Modeling Interactions

- **Goal: Model interaction between many contagions spreading over the network simultaneously**
  - **Some contagions may help each other in adoption**
  - **Others may compete for attention**

Modeling Interactions

- **User is reading posts on Twitter:**
  - User examines posts one by one
  - Currently she is examining post X
  - How does the probability of reposting X depend on what she has seen in the past?



P(post **X** | exposed to **X**, **Y$_1$**, **Y$_2$**, **Y$_3$**) = ?

What's the goal?

- **Given:**
  - **For a single user:**
    Exposure and infection events

- **Goal:** Infer tweet **topic memberships** and **topic interactions**
  - reinforces
  - But suppresses

The Model

- **Goal:** Model P(post X | exp. X, Y_1, Y_2, Y_3)
- **Assume exposures are independent:**

$$P\left(X|\{Y_k\}_{k=1}^{K}\right) = \frac{P(X) \cdot P\left(\{Y_k\}_{k=1}^{K}|X\right)}{P\left(\{Y_k\}_{k=1}^{K}\right)}$$

$$= \frac{1}{P(X)^{K-1}} \prod_{k=1}^{K} P(X|Y_k)$$

- **How many parameters?** $K \cdot w^2$ **!!!**
  - K … history size
  - w … number of posts

The Model

- **Goal:** Model P(post X | exp. X, Y_1, Y_2, Y_3)
- **First, assume:**

$$P(X = u_j | Y_k = u_i) \approx \underbrace{P(X = u_j)}_{\substack{\textbf{Prior infection} \\ \textbf{prob.}}} + \underbrace{\Delta^{(k)}_{cont.}(u_i, u_j)}_{\substack{\textbf{Interaction term} \\ \text{(still has } w^2 \text{ entries!)}}}$$

- **Next, assume "topics":**

$$\Delta^{(k)}_{cont.}(u_i, u_j) = \sum_t \sum_s \mathbf{M}_{j,t} \cdot \Delta^{(k)}_{clust}(c_t, c_s) \cdot \mathbf{M}_{i,s}$$

  - Each contagion u_i has a vector M_i
    - Entry M_{is} models how much u_i belongs to topic s
  - $\Delta^{(k)}_{clust}(s,t)$ … change in infection prob. given that u_i is on topic s and exposure k-steps ago was on topic t

The Model

- **Goal:** Model P(post X | exp. X, Y_1, Y_2, Y_3)
- **First, assume:**

$$P(X = u_j | Y_k = u_i) \approx \underbrace{P(X = u_j)}_{\text{Prior infection prob.}} + \underbrace{\Delta^{(k)}_{cont.}(u_i, u_j)}_{\text{Interaction term (still has } w^2 \text{ entries!)}}$$

- **Next, assume "topics":**

$$\Delta^{(k)}_{cont.}(u_i, u_j) = \sum_t \sum_s \mathbf{M}_{j,t} \cdot \Delta^{(k)}_{clust}(c_t, c_s) \cdot \mathbf{M}_{i,s}$$

$$\left[\quad \Delta^{(k)}_{cont.} \quad \right] = \left[\mathbf{M} \right] \times \left[\Delta^{(k)}_{clust} \right] \times \left[\quad \mathbf{M}^T \quad \right]$$

The Model

- **So we arrive to the full model:**

$$P(X = u_j | Y_k = u_i) = P(X = u_j) + \sum_t \sum_s \mathbf{M}_{i,t} \cdot \Delta_{t,s}^{(k)} \cdot \mathbf{M}_{j,s}$$

- **And then the adoption probability is:**

$$P\left(X | \{Y_k\}_{k=1}^{K}\right) = \frac{1}{P(X)^{K-1}} \prod_{k=1}^{K} P(X|Y_k)$$

Inferring the Model

[ICDM '12]

- **Model parameters:**
  - Δ^k ... topic interaction matrix
  - $M_{i,t}$ ... topic membership vector
  - $P(X)$ ... Prior infection prob.
- **Maximize data likelihood:**

$$\arg\max_{P(x),M,\Delta} \prod_{X\in R} P(X|X,Y_1 \dots Y_K) \prod_{X\notin R} 1 - P(X|X,Y_1 \dots Y_K)$$

  - R ... posts X that resulted in retweets
  - **Solve using stochastic coordinate ascent:**
    - Alternate between optimizing Δ and M

Dataset: Twitter

- **Data from Twitter**
  - *Complete* data from Jan 2011: 3 billion tweets
  - All URLs tweeted by at least 50 users: 191k
- **Task:**
  Predict whether a user will post URL $\boldsymbol{X}$
  - Train on 90% of the data, test on 10%
- **Baselines:** $P(X = u_i | Y_k = u_j) =$
  - **Infection Probability** **(IP)**: $= P(X = u_i)$
  - **IP + Node bias** **(NB)**: $= P(X = u_i) + \gamma_n$
  - **Exposure curve** **(EC)**: $= P(X \mid \#\ times\ exposed\ to\ X)$

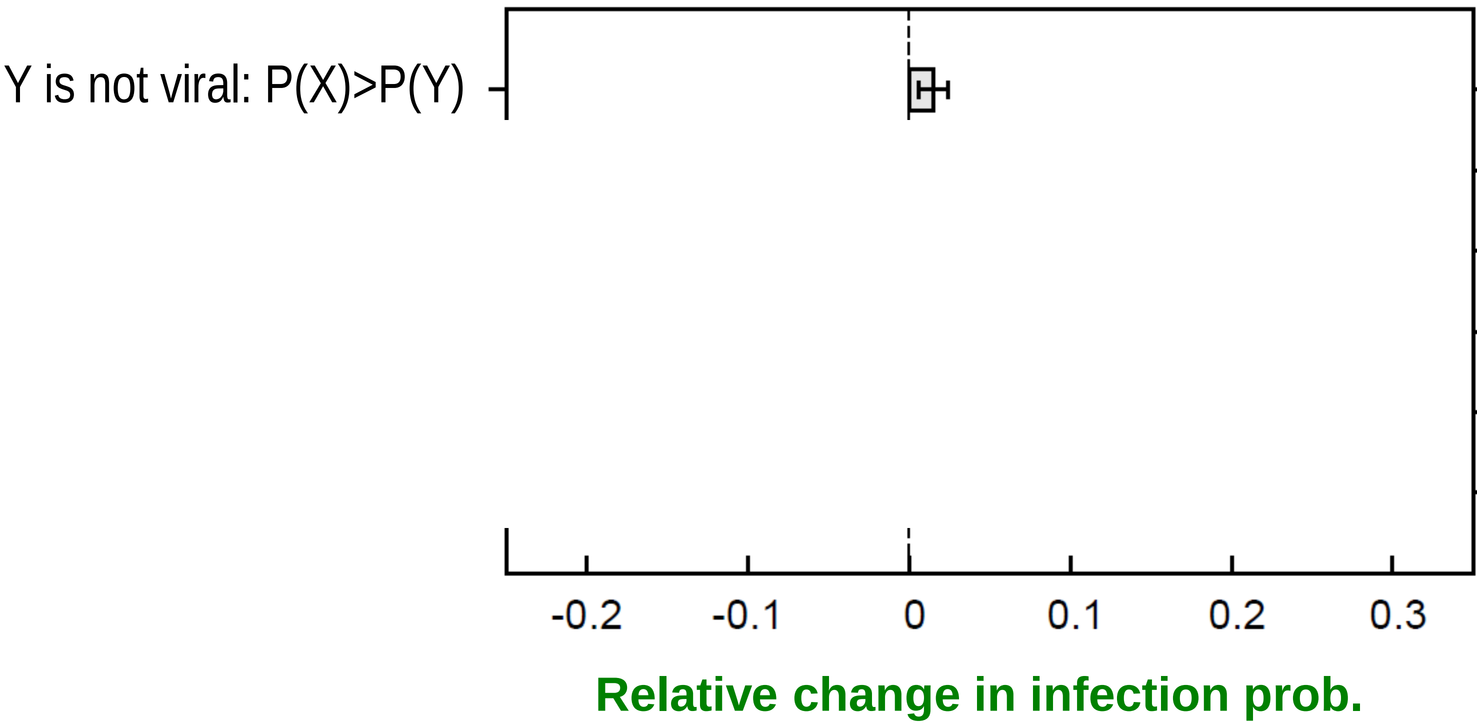
Predicting Retweets

- Task: Predict a retweet given the context

Model Name	Log-Like.	max F_1	Area under PR
IP	-335,550.39	0.0150	0.0157
UB	-338,821.54	0.0112	0.0123
EC	-338,367.86	0.0181	0.0250
Our Model - With Prior			
IMM K=1	-313,843.93	0.0412	0.0515
IMM K=2	-299,884.86	**0.0465**	**0.1238**
IMM K=3	**-299,352.32**	0.0380	0.0926
IMM K=4	-315,319.54	0.0321	0.0804
IMM K=5	-352,687.54	0.0386	0.0924

How do Tweets Interact?

- **How *P(post X| exposed Y)* changes if …**
  - **X and Y are similar/different in content?**
  - **Y is highly viral (Prob. reshare is high)?**

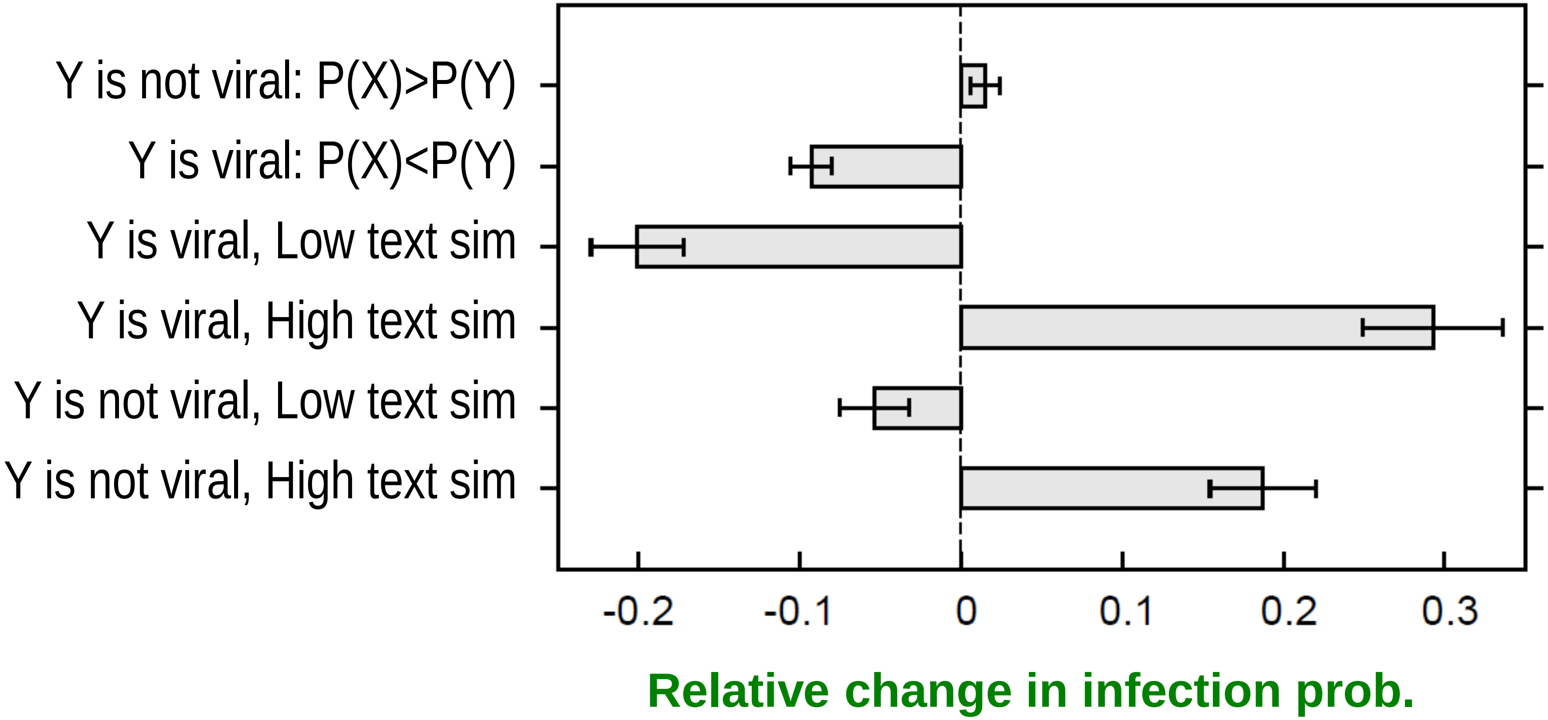


Observations:
- If **Y** is not viral, this boost **X**
- If **Y** is highly viral, this kills **X**

BUT:
Only if **Y** and **X** are of high content similarity **Y** helps **X**

How do Tweets Interact?

- How *P(post X| exposed Y)* changes if …
  - X and Y are similar/different in content?
  - Y is highly viral (Prob. reshare is high)?



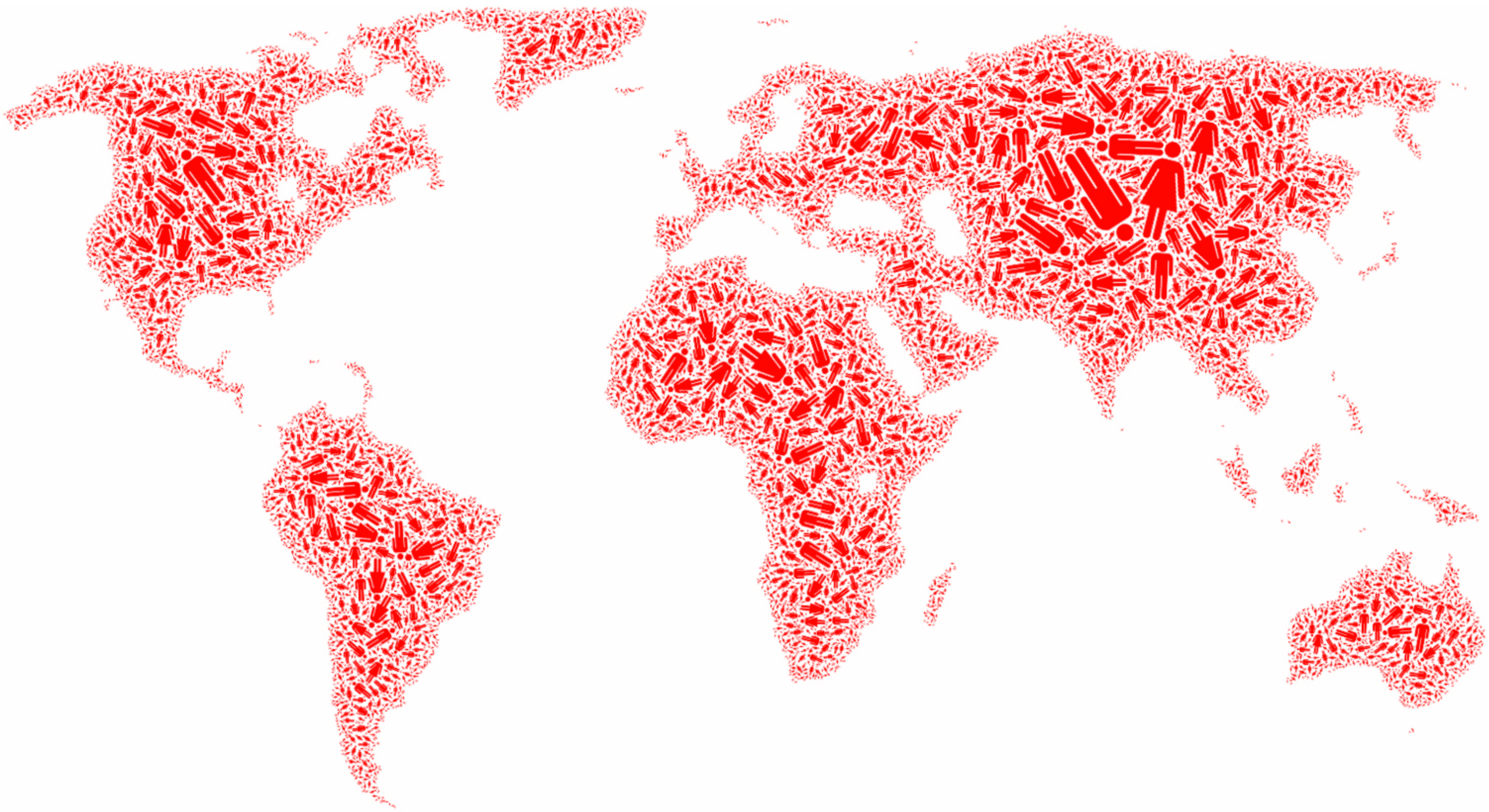
Observations:
- If **Y** is not viral, this boost **X**
- If **Y** is highly viral, this kills **X**

BUT:
Only if **Y** and **X** are of high content similarity **Y** helps **X**

Further Questions

- **Today: Messages arriving through networks from real-time sources requires new ways of thinking about information dynamics and consumption**
- **Predictive models of information diffusion**
  - Where is the information going to spread?
  - What will go viral?
- **User personalization**
  - New models of how users consume information
- **Connections to mutation of information:**
  - How does **attitude** and **sentiment change** in different parts of the network?
  - How does **information change** in different parts of the network?

What’s beyond?



Networks are a natural language for reasoning about problems spanning society, technology and information

Conclusion & Reflections

- **Only recently has large scale network data become available**
  - Opportunity for large scale analyses
  - **Benefits of working with massive data**
    - Observe “invisible” patterns
- **Lots of interesting networks questions both in CS as well as in general science**
  - Need scalable algorithms & models

Network Data & Code

- **Research on networks is both algorithmic and empirical**
- Need to network data:
  - **Stanford Large Network Dataset Collection**
    - Over 60 large online networks with metadata
    - http://snap.stanford.edu/data
  - **SNAP: Stanford Network Analysis Platform**
    - A general purpose, high performance system for dynamic network manipulation and analysis
    - Can process 1B nodes, 10B edges
    - http://snap.stanford.edu

Networks

Networks — implicit for millenia — are finally becoming visible

Models based on algorithmic ideas will be crucial in understanding these developments

THANKS!

Data + Code:

http://snap.stanford.edu

Twitter: @jure

Tools for Networks

- **S**tanford **N**etwork **A**nalysis **P**latform (SNAP) is a general purpose, high-performance system for analysis and manipulation of large networks
  - http://snap.stanford.edu
  - Scales to massive networks with hundreds of millions of nodes and billions of edges
- **SNAP software**
  - Snap.py for Python, SNAP C++
  - Tutorial on how to use SNAP: http://snap.stanford.edu/proj/snap-icwsm

Snap.py Resources

- **Prebuilt packages** for Mac OS X, Windows, Linux
  http://snap.stanford.edu/snappy/index.html
- **Snap.py documentation**:
  http://snap.stanford.edu/snappy/doc/index.html
  - Quick Introduction, Tutorial, Reference Manual
- **SNAP user mailing list**
  http://groups.google.com/group/snap-discuss
- **Developer resources**
  - Software available as open source under BSD license
  - GitHub repository
    https://github.com/snap-stanford/snap-python

SNAP C++ Resources

- **Prebuilt packages** for Mac OS X, Windows, Linux
  - http://snap.stanford.edu/snap/download.html
- **SNAP documentation**
  - http://snap.stanford.edu/snap/doc.html
  - Quick Introduction, User Reference Manual
- **SNAP user mailing list**
  - http://groups.google.com/group/snap-discuss
- **Developer resources**
  - Software available as open source under BSD license
  - GitHub repository
    - https://github.com/snap-stanford/snap
  - SNAP C++ Programming Guide

Network Data

- **Stanford Large Network Dataset Collection**
  - http://snap.stanford.edu/data
- **Over 70 different networks and communities**
  - **Social networks:** online social networks, edges represent interactions between people
  - **Twitter and Memetracker:** Memetracker phrases, links and 467 million Tweets
  - **Citation networks:** nodes represent papers, edges represent citations
  - **Collaboration networks:** nodes represent scientists, edges represent collaborations
  - **Amazon networks :** nodes represent products and edges link commonly co-purchased products

Books & Courses

Want to learn more about networks?

- **Social and Information Networks** lectures:
  - http://cs224w.stanford.edu
- **Mining Massive Datasets** lectures:
  - http://cs246.stanford.edu
- **Books (fee PDFs):**
  - **Mining Massive Datasets**
    - http://infolab.stanford.edu/~ullman/mmds.html
  - **Networks, Crowds and Markets**
    - http://www.cs.cornell.edu/home/kleinber/networks-book

References

Community detection

- Community detection in graphs by S. Fortunato. *Physics Reports* 2010.
- Community-Affiliation Graph Model for Overlapping Community Detection by J. Yang, J. Leskovec.*IEEE Intl. Conference On Data Mining (ICDM)*, 2012.
- Defining and Evaluating Network Communities based on Ground-truth by J. Yang, J. Leskovec. *IEEE Intl. Conference On Data Mining (ICDM)*, 2012.
- Overlapping Community Detection at Scale: A Nonnegative Matrix Factorization Approach by J. Yang, J. Leskovec. *ACM International Conference on Web Search and Data Mining (WSDM)*, 2013.
- Discovering Social Circles in Ego Networks by J. McAuley, J. Leskovec. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2014.
- Community Detection in Networks with Node Attributes by J. Yang, J. McAuley, J. Leskovec. *IEEE Intl. Conference On Data Mining (ICDM)*, 2013.
- Detecting Cohesive and 2-mode Communities in Directed and Undirected Networks by J. Yang, J. McAuley, J. Leskovec. *ACM Web Search and Data Mining (WSDM)*, 2014.

References

Link prediction

- Link Prediction in Complex Networks: A Survey by L. Lu, T. Zhou. Arxiv 2010.
- Multiplicative Attribute Graph Model of Real-World Networks by M. Kim, J. Leskovec. *Internet Mathematics 8(1-2) 113--160* , 2012.
- Latent Multi-group Membership Graph Model by M. Kim, J. Leskovec. *International Conference on Machine Learning (ICML)*, 2012.
- Nonparametric Multi-group Membership Model for Dynamic Networks by M. Kim, J. Leskovec. *Neural Information Processing Systems (NIPS)*, 2013.
- Supervised Random Walks: Predicting and Recommending Links in Social Networks by L. Backstrom, J. Leskovec. *ACM Web Search and Data Mining (WSDM)*, 2011.
- Friendship and Mobility: User Movement In Location-Based Social Networks by E. Cho, S. A. Myers, J. Leskovec. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2011.
- Predicting Positive and Negative Links in Online Social Networks by J. Leskovec, D. Huttenlocher, J. Kleinberg. *ACM World Wide Web (WWW)*, 2010.

References

Social Media

- Meme-tracking and the Dynamics of the News Cycle by J. Leskovec, L. Backstrom, J. Kleinberg. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2009.
- Inferring Networks of Diffusion and Influence by M. Gomez-Rodriguez, J. Leskovec, A. Krause. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2010.
- On the Convexity of Latent Social Network Inference by S. A. Myers, J. Leskovec. *Neural Information Processing Systems (NIPS)*, 2010.
- Structure and Dynamics of Information Pathways in Online Media by M. Gomez-Rodriguez, J. Leskovec, B. Schoelkopf. *ACM International Conference on Web Search and Data Mining (WSDM)*, 2013.
- Modeling Information Diffusion in Implicit Networks by J. Yang, J. Leskovec. *IEEE International Conference On Data Mining (ICDM)*, 2010.
- Information Diffusion and External Influence in Networks by S. Myers, C. Zhu, J. Leskovec. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2012.
- Clash of the Contagions: Cooperation and Competition in Information Diffusion by S. Myers, J. Leskovec. *IEEE International Conference On Data Mining (ICDM)*, 2012.