Carnegie Mellon

Mining Billion-Node Graphs: Patterns and influence propagation

Christos Faloutsos

CMU

Thank you!

- Sinan Aral
- Foster Provost
- Arun Sundararajan

- Shirley Lau

Our goal:

Open source system for mining huge graphs:

PEGASUS project (PEta GrAph mining System)

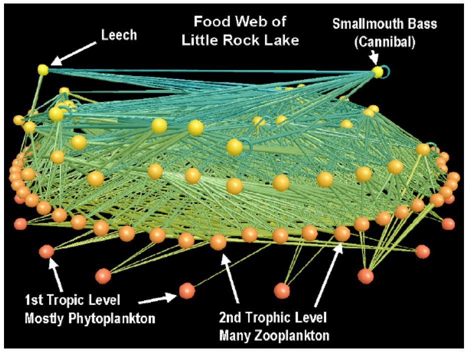
- www.cs.cmu.edu/~pegasus
- code and papers

Carnegie Mellon

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
- Conclusions

Graphs - why should we care?



Food Web
[Martinez '91]

Recommendation

systems

Internet Map
[lumeta.com]

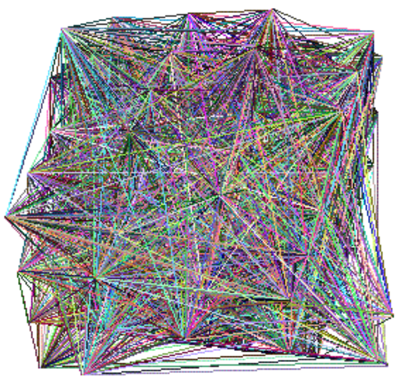
Carnegie Mellon

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - overview
  - Triangles
  - Diameter
  - `Eigenspokes'
  - Phonecall duration
- Problem#2: Tools
- Conclusions

Carnegie Mellon

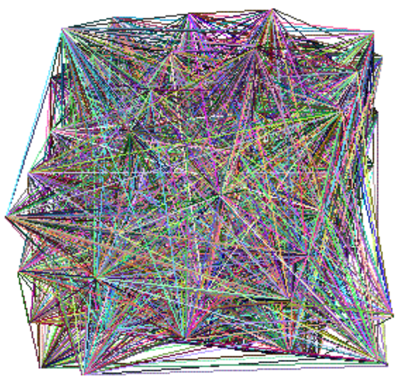
Problem #1 - network and graph mining



- What does the Internet look like?
- What does FaceBook look like?

- What is ‘normal’/‘abnormal’?
- which patterns/laws hold?

Problem #1 - network and graph mining

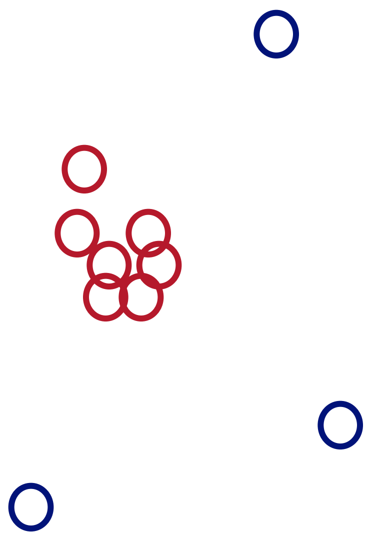


- What does the Internet look like?
- What does FaceBook look like?

- What is 'normal'/'abnormal'?
- which patterns/laws hold?
  - BIG DATA helps: finds patterns that would be 'invisible'

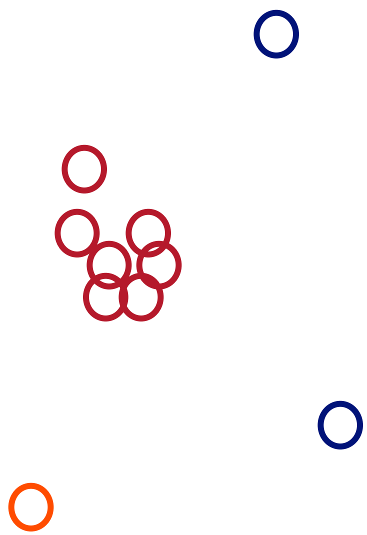
Typical viewpoint

- ‘Signal’ and
- noise

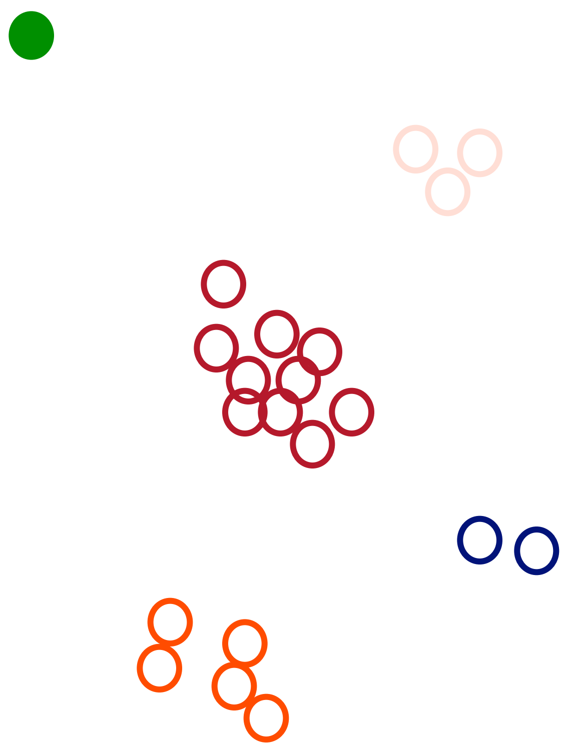


MORE REALISTIC viewpoint

- ‘Signal’ and
- Weaker signal



MORE REALISTIC viewpoint

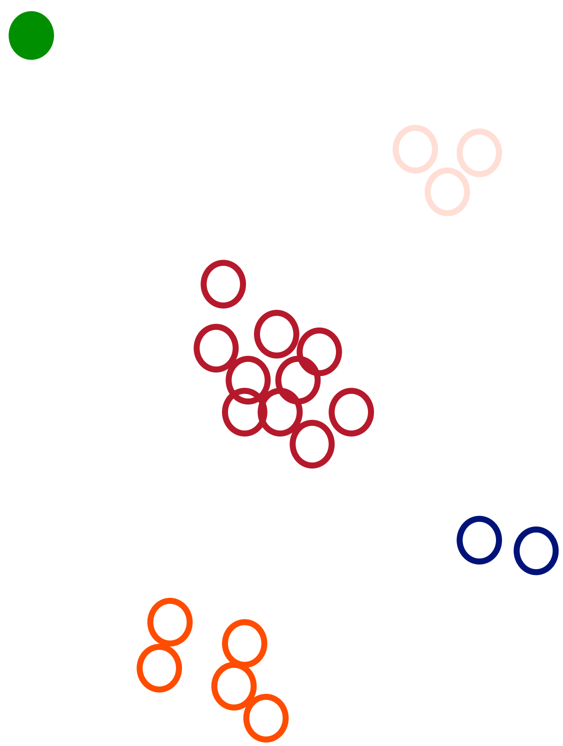


- ‘Signal’ and
- Weaker signal and
- Even weaker signal
- …

BIG DATA helps

Carnegie Mellon

MORE REALISTIC viewpoint



- ‘Signal’ and
- Weaker signal and
- Even weaker signal
- …

BIG DATA helps

(and sampling may hurt)

Graph mining

- Are real graphs random?

Carnegie Mellon

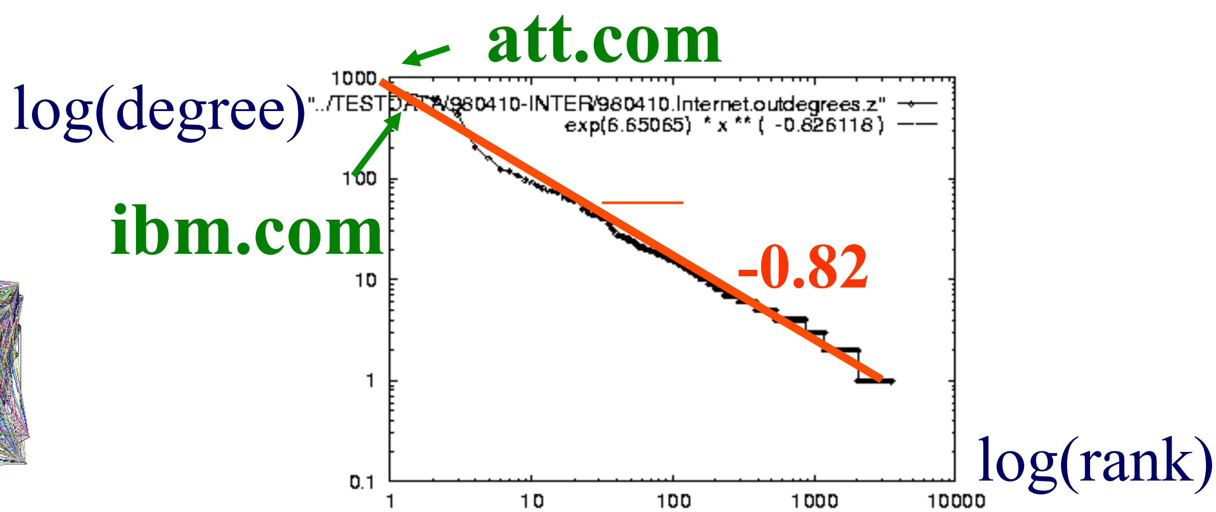
Laws and patterns

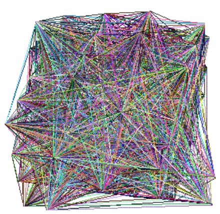
- Are real graphs random?
- A: NO!!
  - Diameter (small, and decreasing!)
  - in- and out- degree distributions (skewed/PL)
  - # triangles (skewed)
  - other (surprising) patterns
- So, let’s look at the data

Solution# S.1

- Power law in the degree distribution [SIGCOMM99]

internet domains

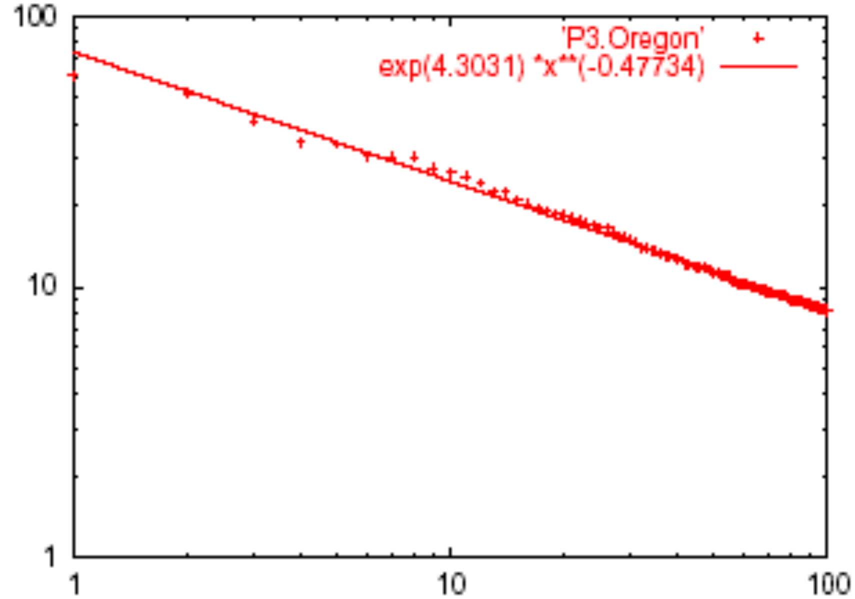




Carnegie Mellon

Solution# S.2: Eigen Exponent *E*

Eigenvalue



Rank of decreasing eigenvalue

Exponent = slope

E = -0.48

May 2001

- A2: power law in the eigenvalues of the adjacency matrix

Carnegie Mellon

Real Graph Patterns

	unweighted	weighted
static	**P01**. Power-law degree distribution [Faloutsos et. al.`99, Kleinberg et. al.`99, Chakrabarti et. al. `04, Newman`04] **P02**. Triangle Power Law [Tsourakakis `08] **P03**. Eigenvalue Power Law [Siganos et. al. `03] **P04**. Community structure [Flake et. al.`02, Girvan and Newman `02] **P05**. Clique Power Laws [Du et. al. ‘09]	**P12**. Snapshot Power Law [McGlohon et. al. `08]
dynamic	**P06**. Densification Power Law [Leskovec et. al.`05] **P07**. Small and shrinking diameter [Albert and Barabási `99, Leskovec et. al. ‘05, McGlohon et. al. ‘08] **P08**. Gelling point [McGlohon et. al. `08] **P09**. Constant size $2^{nd}$ and $3^{rd}$ connected components [McGlohon et. al. `08] **P10**. Principal Eigenvalue Power Law [Akoglu et. al. `08] **P11**. Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et. al. `98, Crovella and Bestavros `99, McGlohon et .al. `08]	**P13**. Weight Power Law [McGlohon et. al. `08] **P14**. Skewed call duration distributions [Vaz de Melo et. al. `10]

RTG: A Recursive Realistic Graph Generator using Random Typing
Leman Akoglu and Christos Faloutsos. *ECML PKDD*’09.

Carnegie Mellon

Real Graph Patterns

	unweighted	weighted
static	**P01**. Power-law degree distribution [Faloutsos et. al.`99, Kleinberg et. al.`99, Chakrabarti et. al. `04, Newman`04] **P02**. Triangle Power Law [Tsourakakis `08] **P03**. Eigenvalue Power Law [Siganos et. al. `03] **P04**. Community structure [Flake et. al.`02, Girvan and Newman `02] **P05**. Clique Power Laws [Du et. al. '09]	**P12**. Snapshot Power Law [McGlohon et. al. `08]
dynamic	**P06**. Densification Power Law [Leskovec et. al.`05] **P07**. Small and shrinking diameter [Albert and Barabási `99, Leskovec et. al. '05, McGlohon et. al. '08] **P08**. Gelling point [McGlohon et. al. `08] **P09**. Constant size 2nd and 3rd connected components [McGlohon et. al. `08] **P10**. Principal Eigenvalue Power Law [Akoglu et. al. `08] **P11**. Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et. al. `98, Crovella and Bestavros `99, McGlohon et .al. `08]	**P13**. Weight Power Law [McGlohon et. al. `08] **P14**. Skewed call duration distributions [Vaz de Melo et. al. `10]

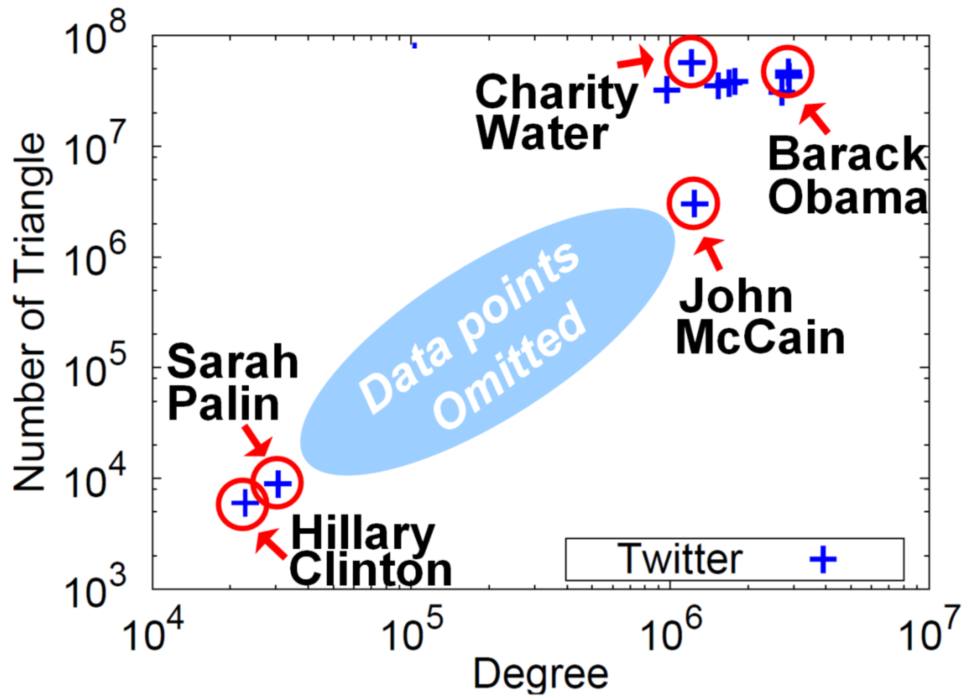
RTG: A Recursive Realistic Graph Generator using Random Typing
Leman Akoglu and Christos Faloutsos. *ECML PKDD*'09.

Triangle counting for large graphs?

Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

Triangle counting for large graphs?

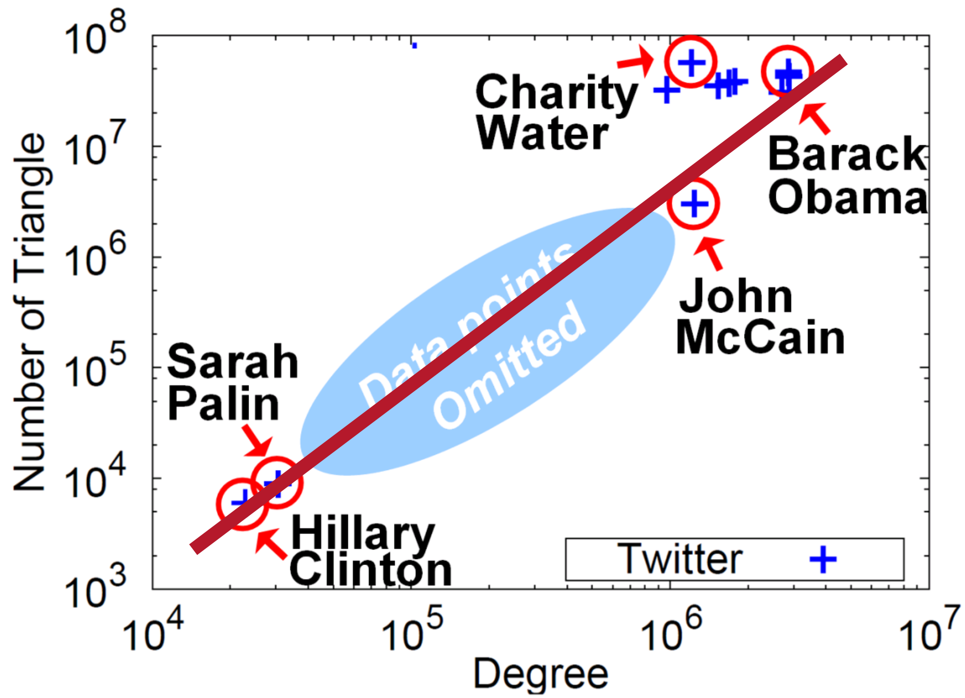


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

Carnegie Mellon

Triangle counting for large graphs?

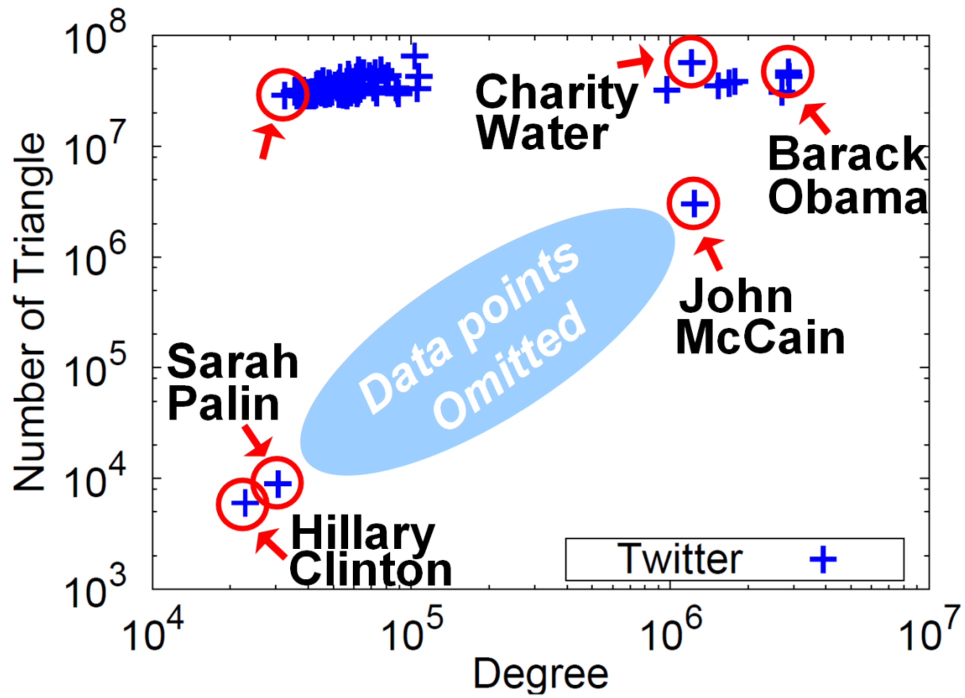


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

Carnegie Mellon

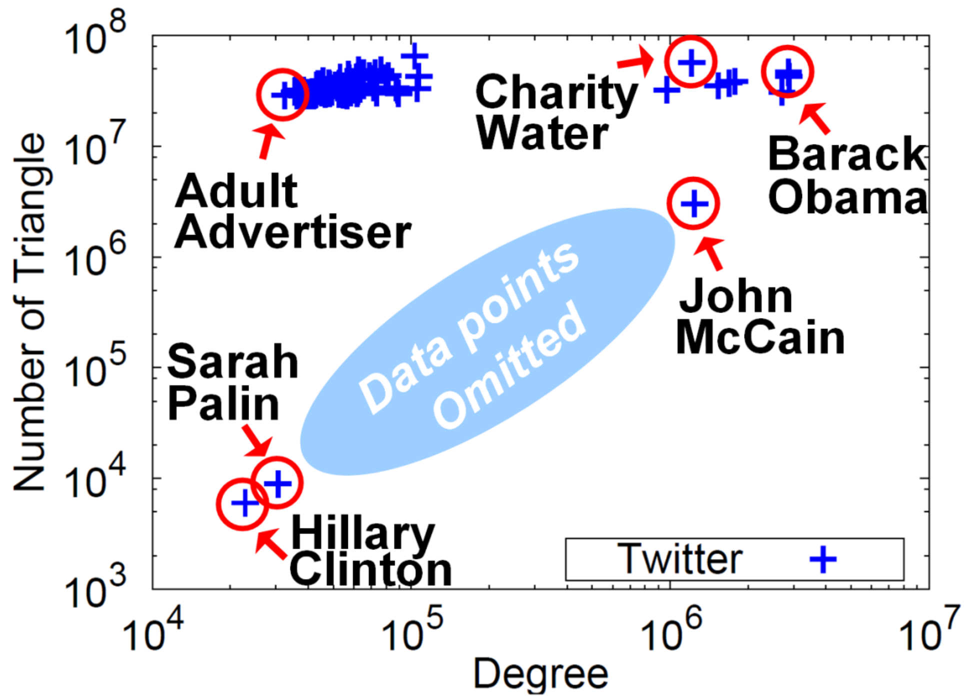
Triangle counting for large graphs?



Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

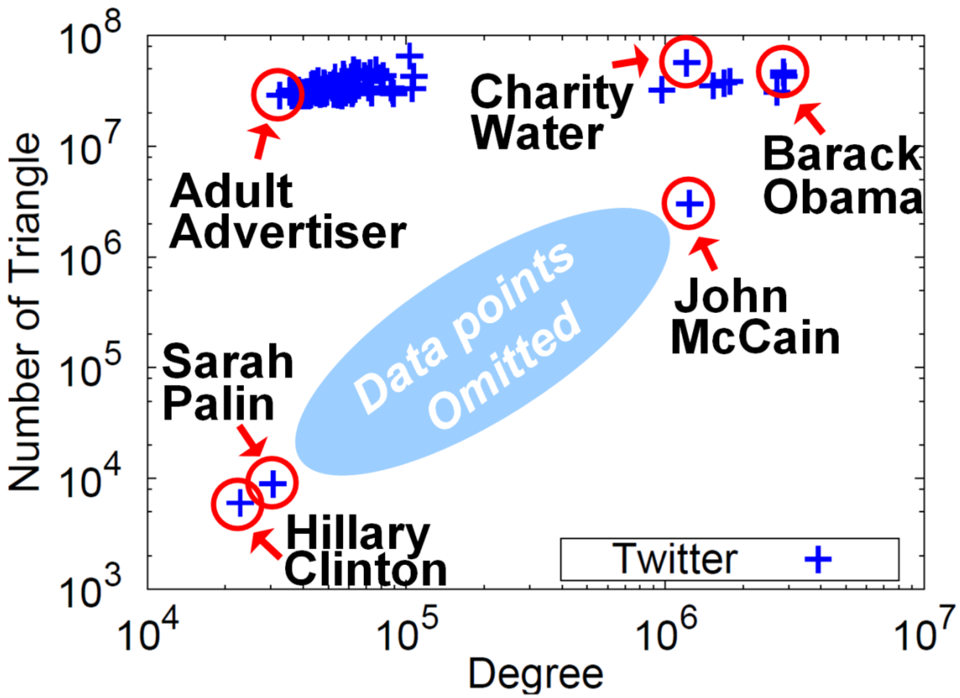
Triangle counting for large graphs?



Anomalous nodes in Twitter(~ 3 billion edges)

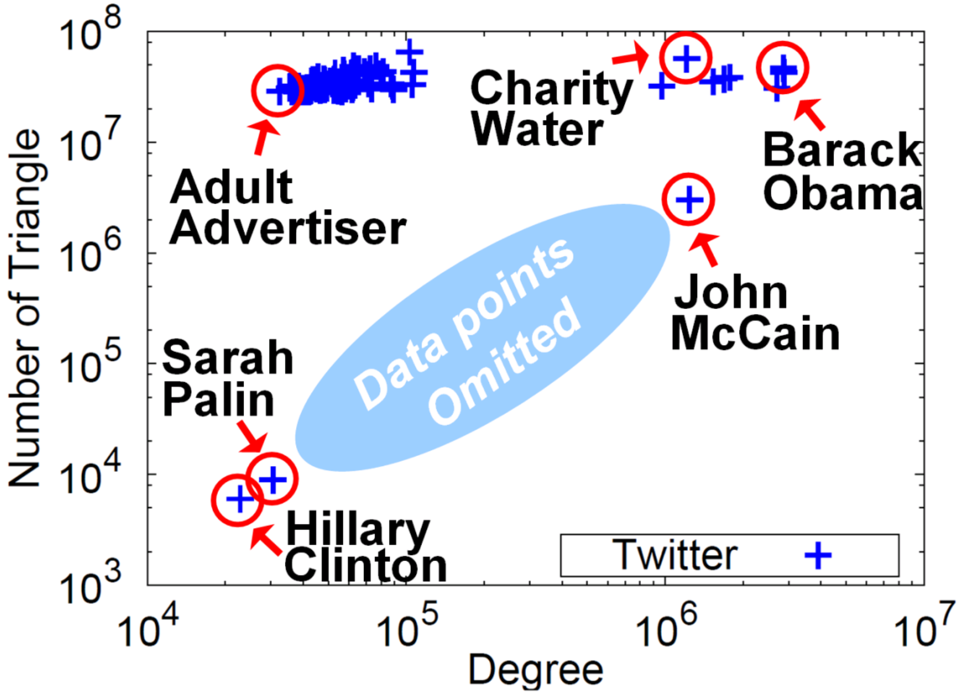
[U Kang, Brendan Meeder, +, PAKDD’11]

Triangle counting for large graphs?



Q: How to compute # triangles in B-node graph? ($O(d_{max}$ ** 2))?

Triangle counting for large graphs?



Q: How to compute # triangles in B-node graph? (O(d_{max} ** 2))? **A: cubes of eigvals**

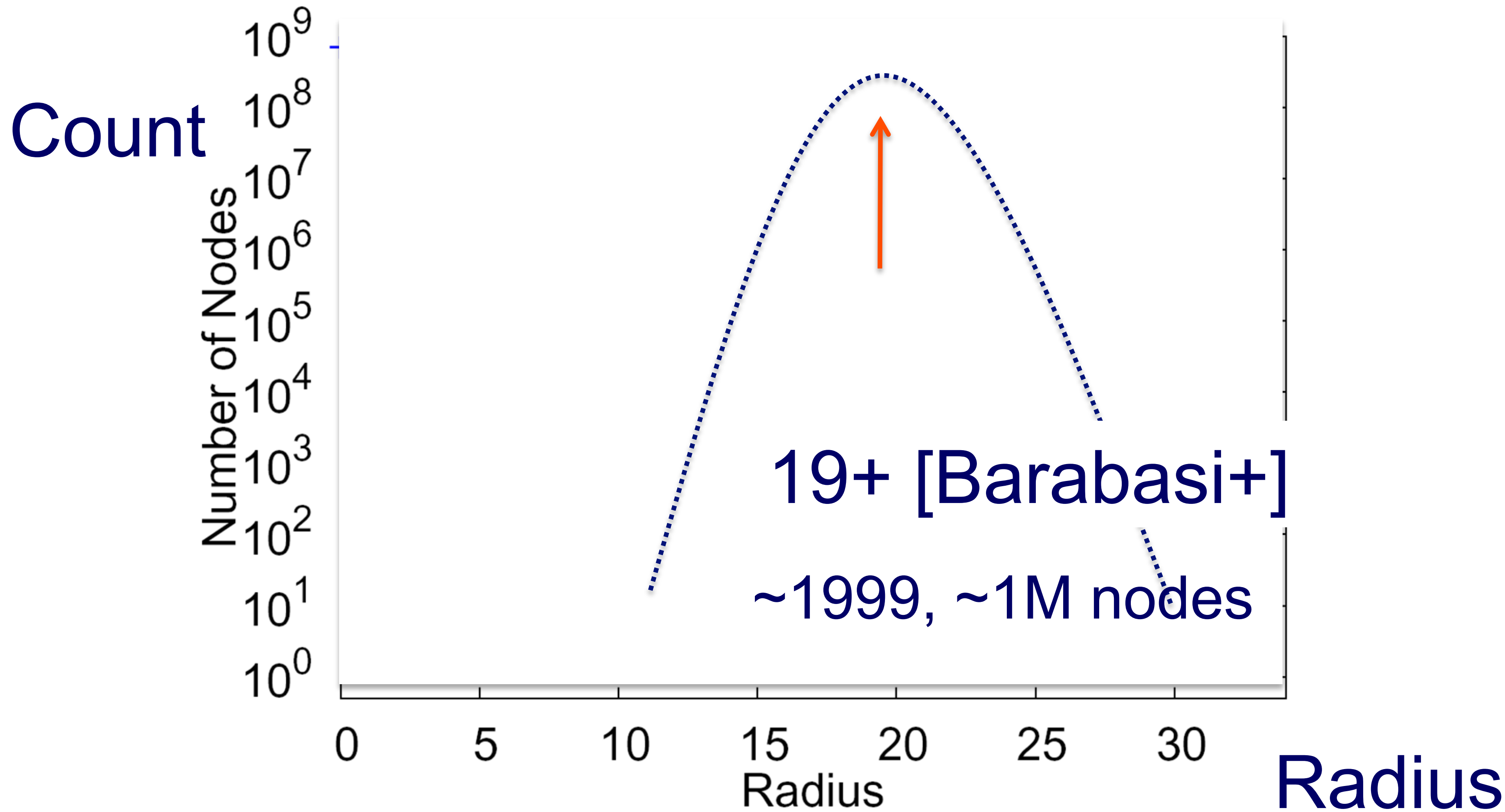
Outline

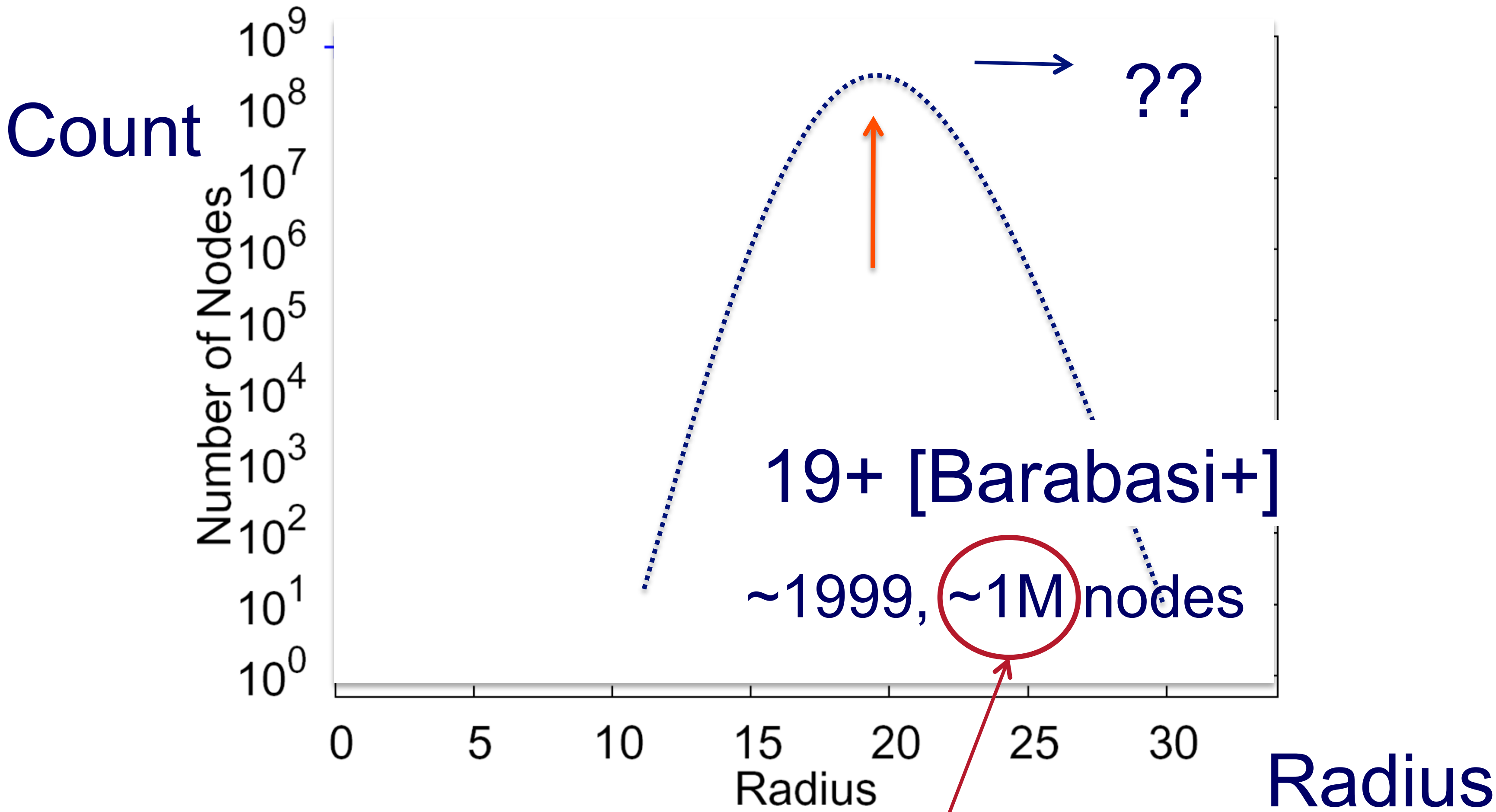
- Introduction – Motivation
- Problem#1: Patterns in graphs
  - overview
  - Triangles
  - Diameter
  - `Eigenspokes'
  - Phonecall duration
- Problem#2: Tools
- Conclusions

HADI for diameter estimation

- *Radius Plots for Mining Tera-byte Scale Graphs* **U Kang**, Charalampos Tsourakakis, Ana Paula Appel, Christos Faloutsos, Jure Leskovec, SDM'10
- Naively: diameter needs **O(N**2)** space and up to O(N**3) time – **prohibitive** (N~1B)
- Our HADI: linear on E (~10B)
  - Near-linear scalability wrt # machines
  - Several optimizations -> 5x faster

Carnegie Mellon

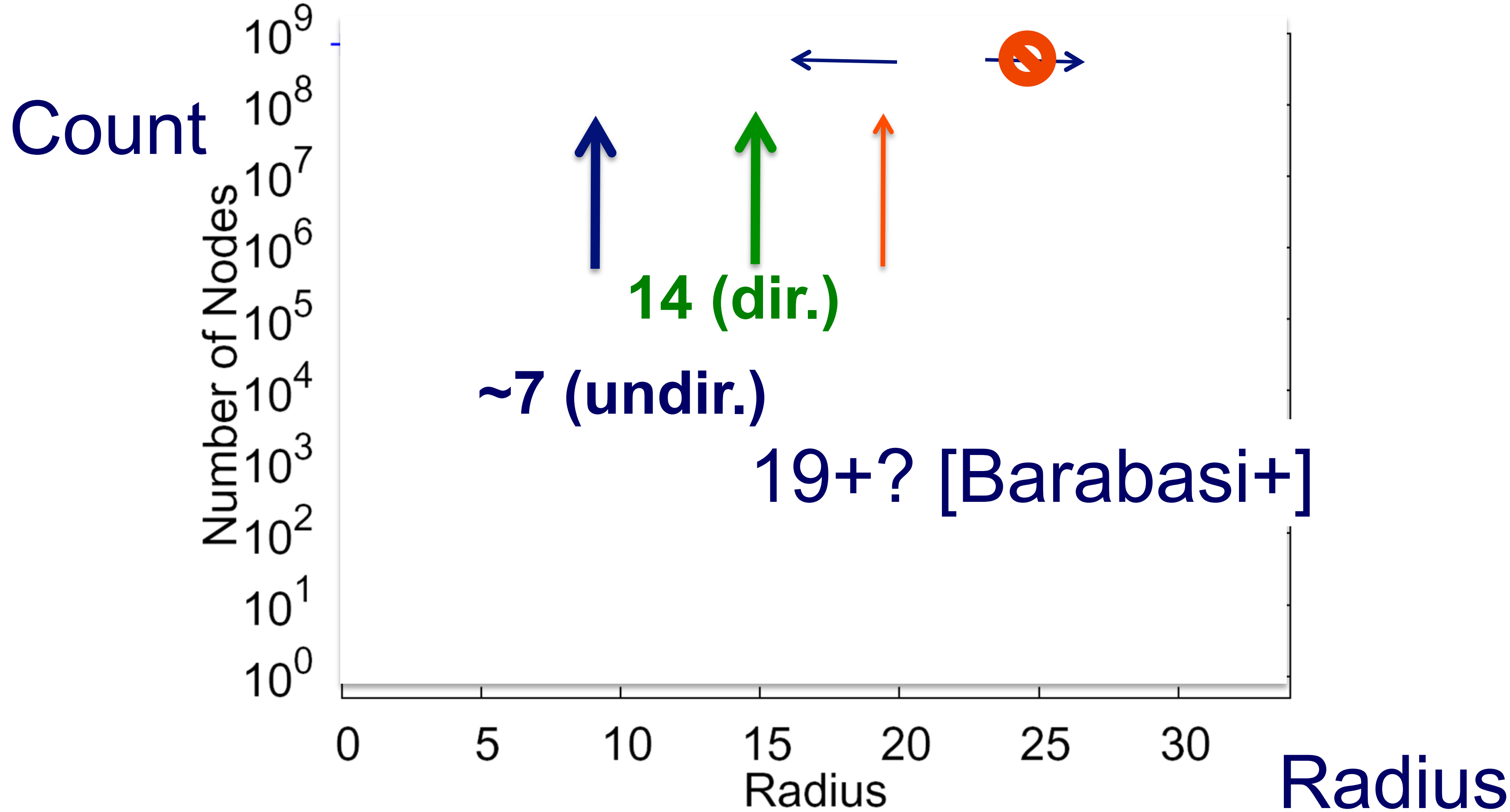




YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

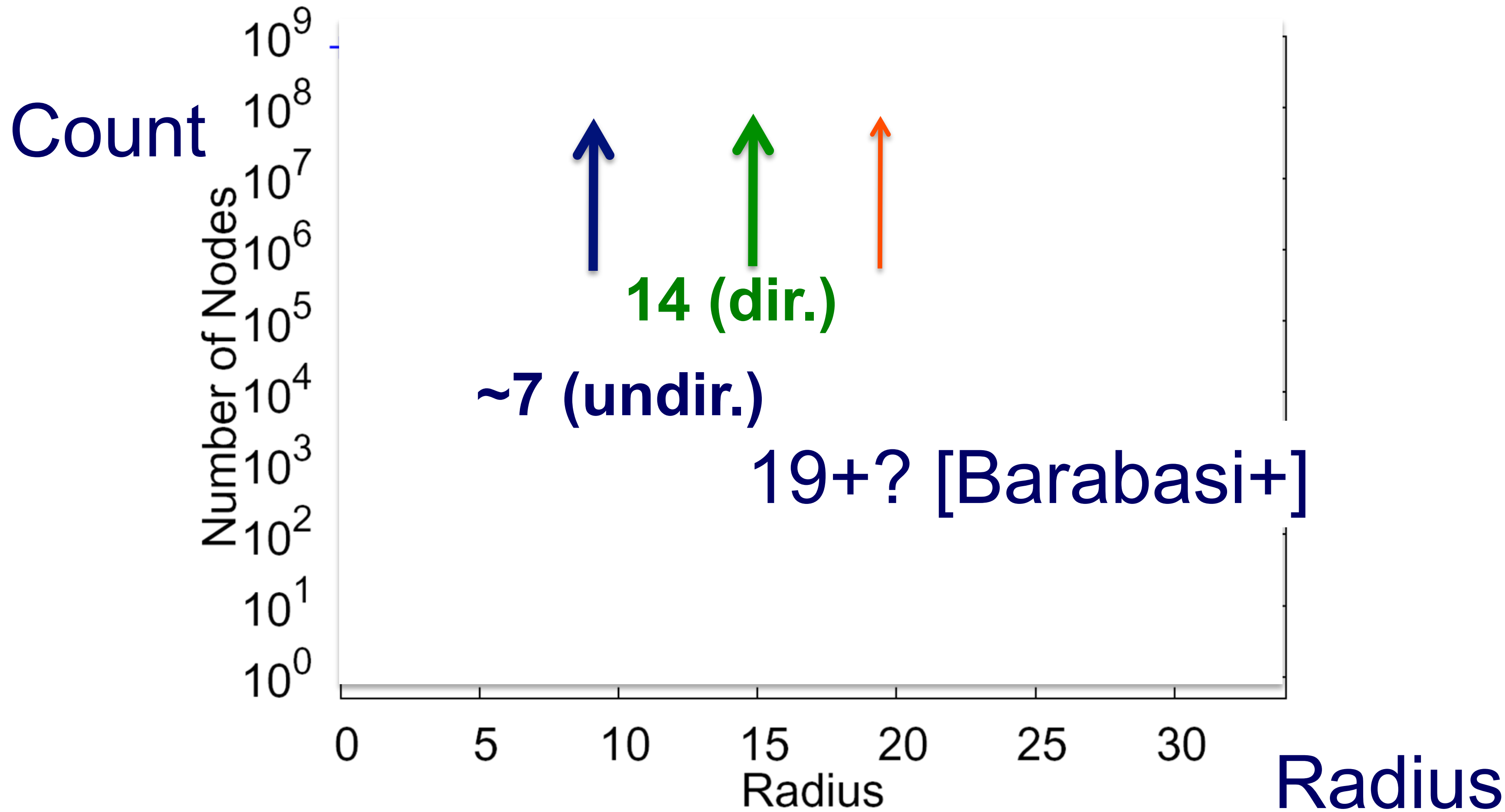
- Largest publicly available graph ever studied.

Carnegie Mellon



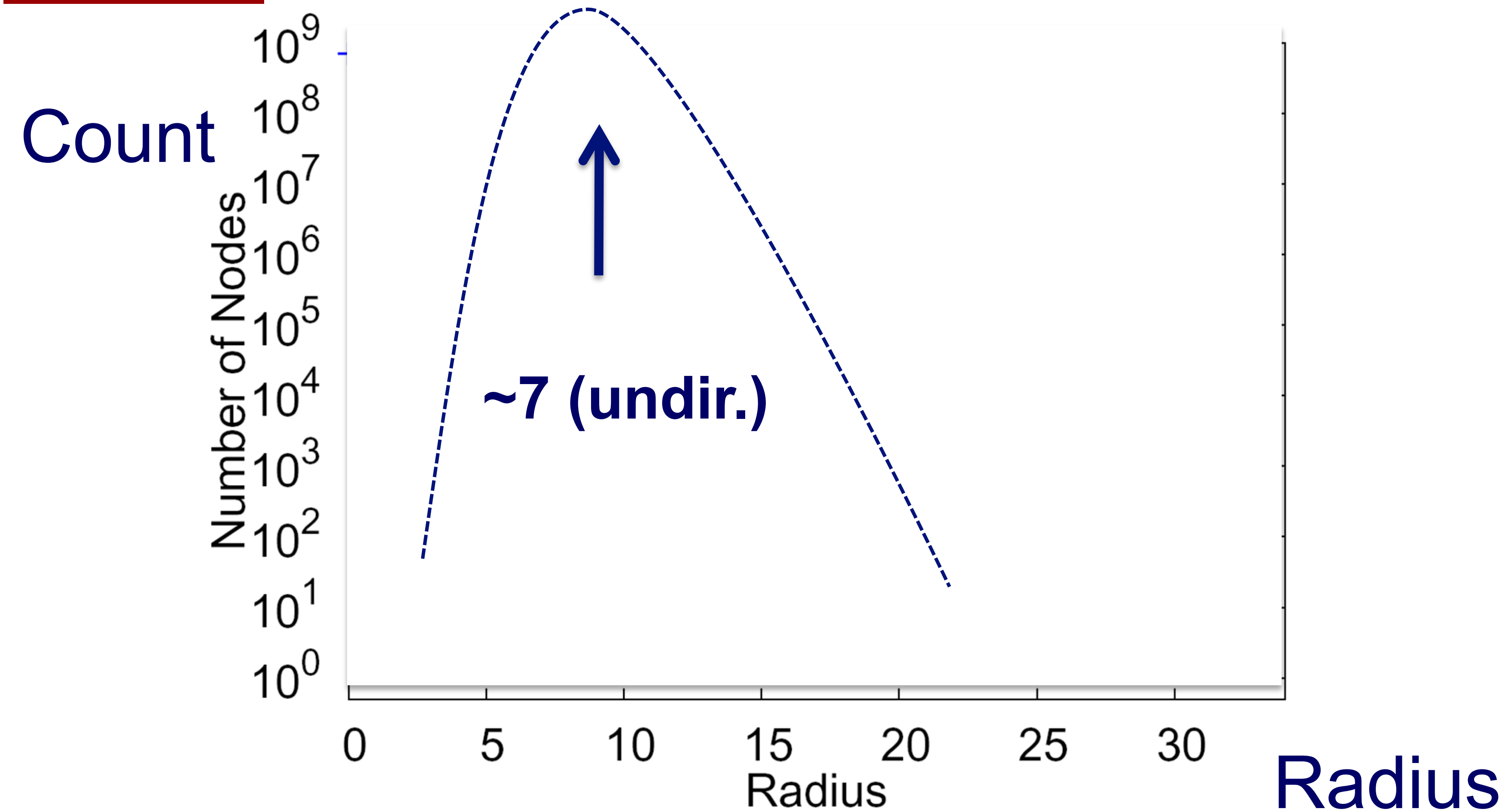
YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

- Largest publicly available graph ever studied.



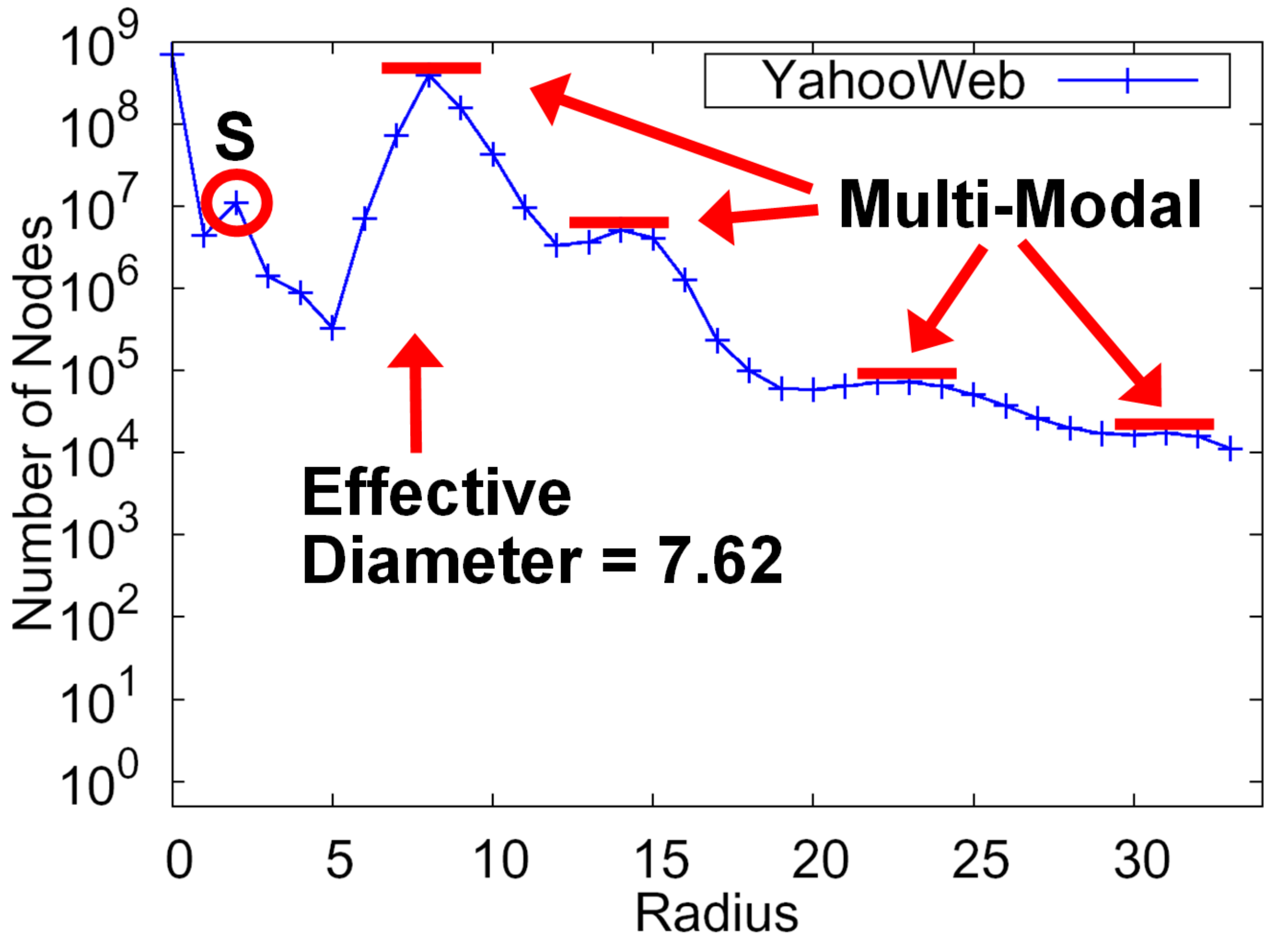
YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

- 7 degrees of separation (!)
- Diameter: shrunk



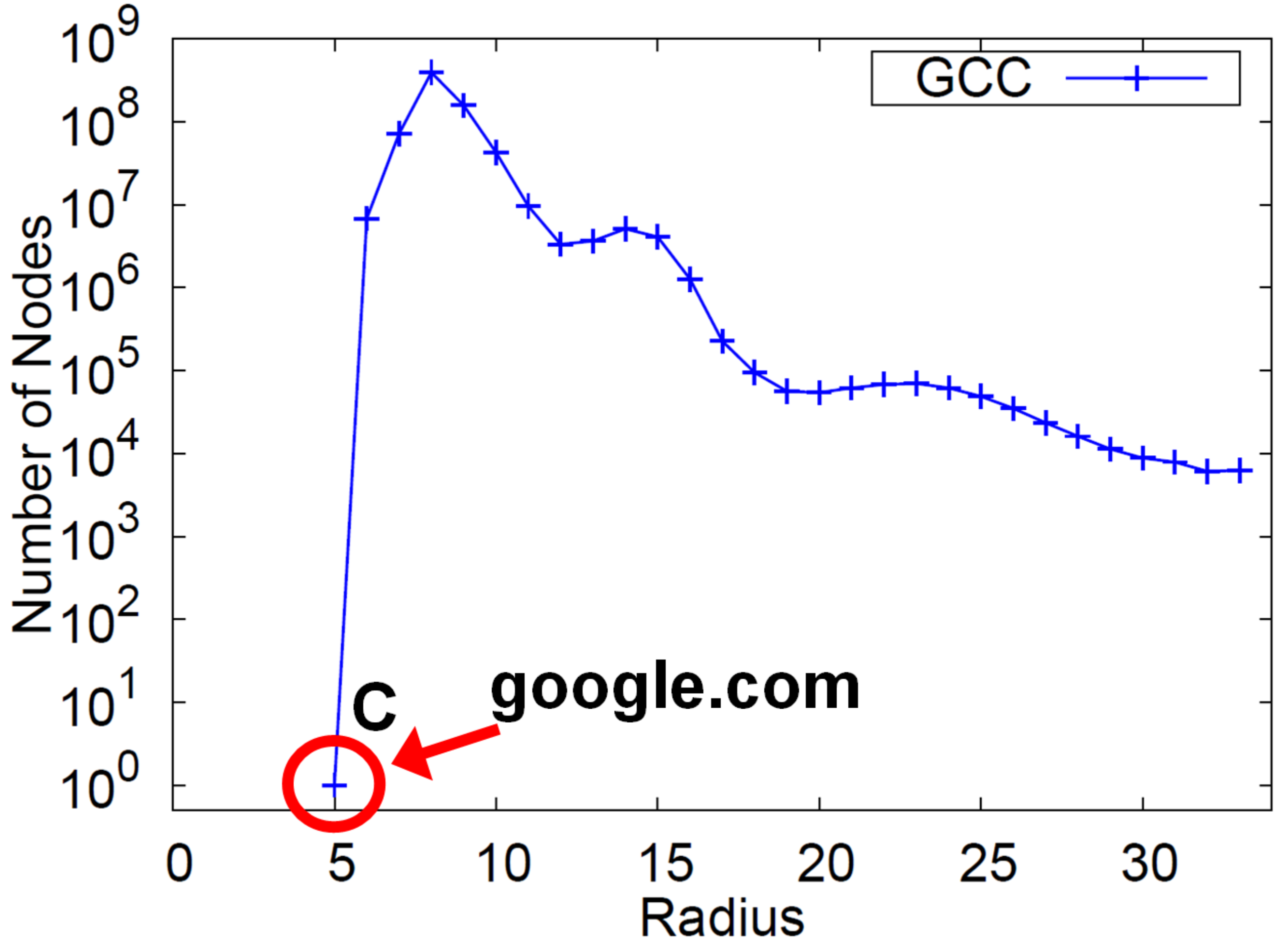
YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

Q: Shape?

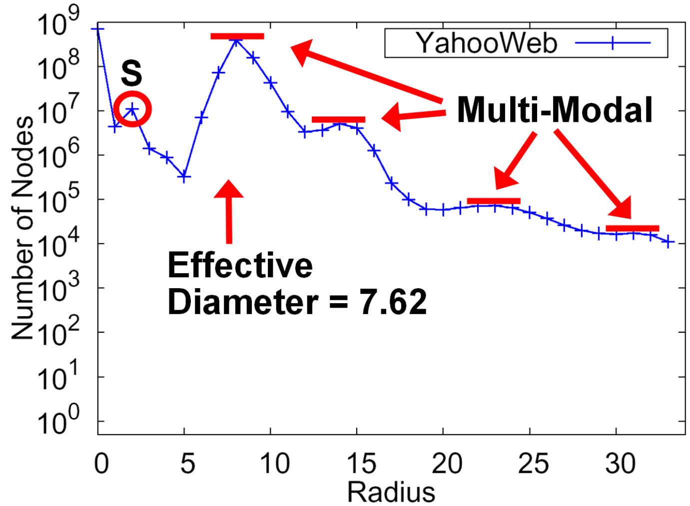


YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

- effective diameter: surprisingly small.
- Multi-modality (?!)



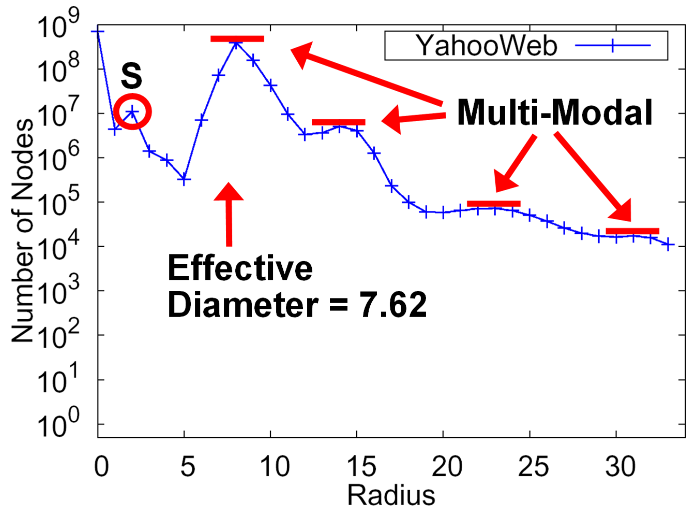
Radius Plot of **GCC** of YahooWeb.

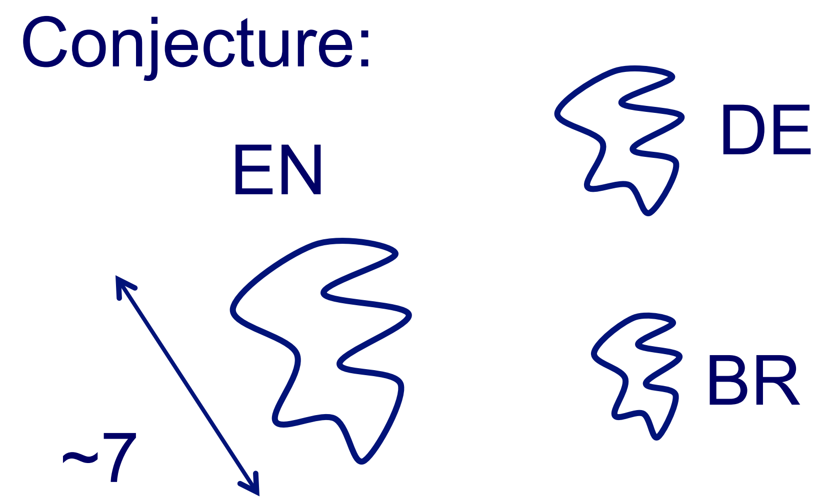


YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

- effective diameter: surprisingly small.
- Multi-modality: probably mixture of cores .

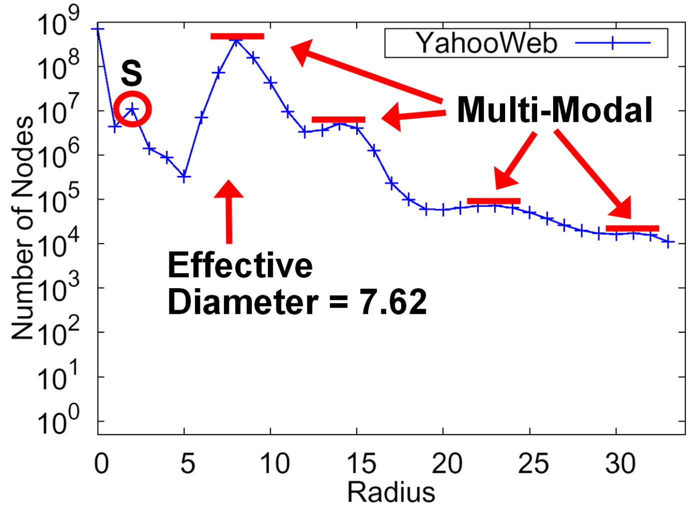
Carnegie Mellon



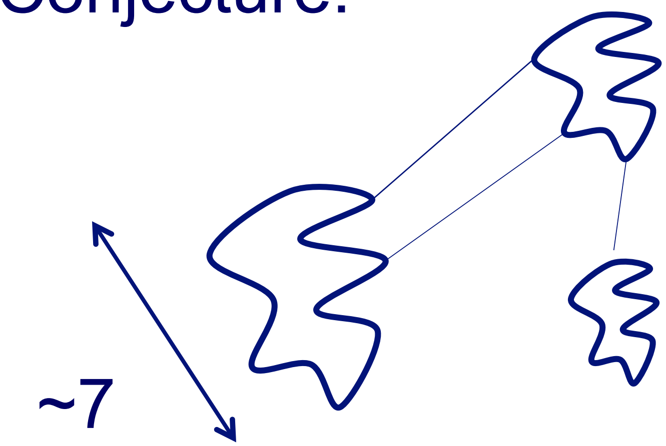


YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

- effective diameter: surprisingly small.
- Multi-modality: probably mixture of cores .



Conjecture:

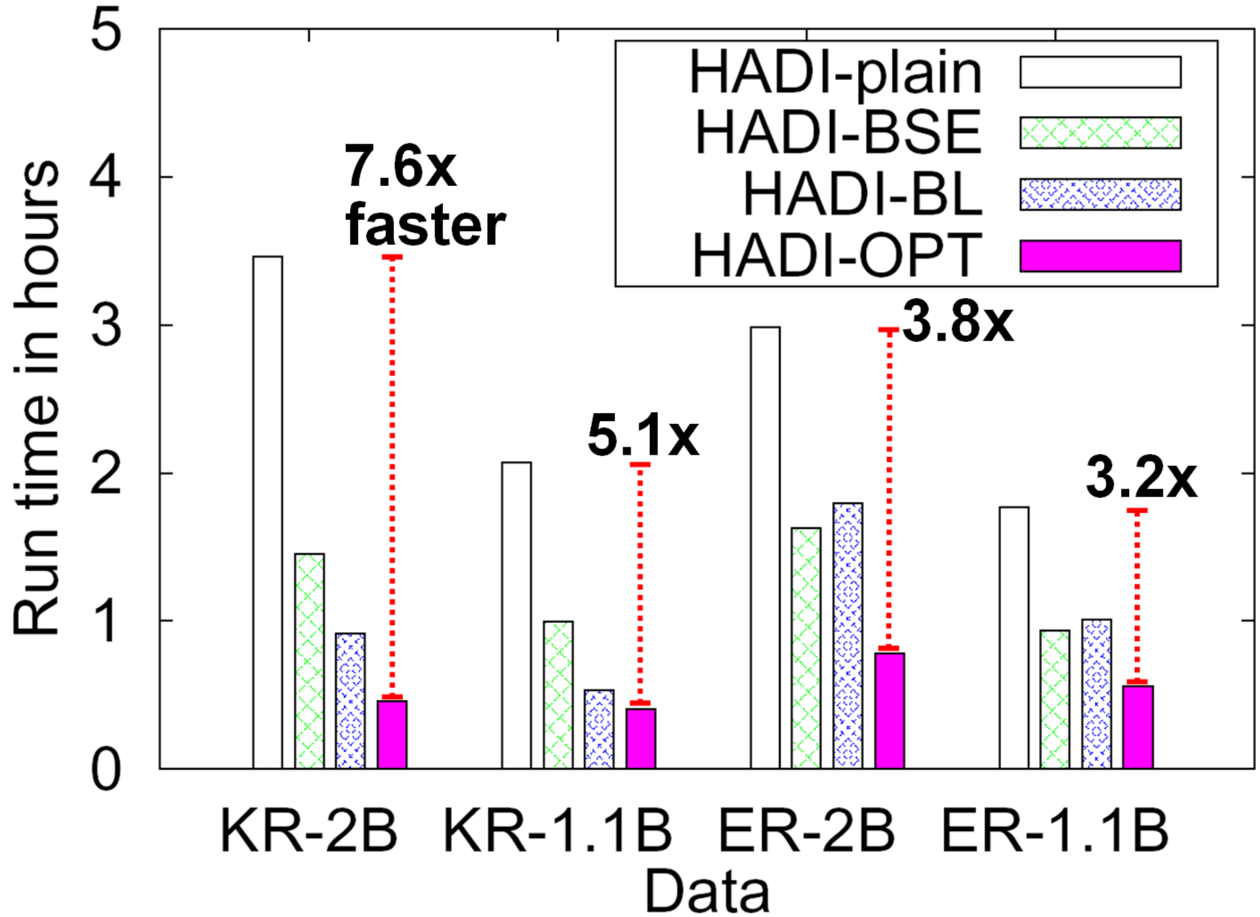


YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

- effective diameter: surprisingly small.
- Multi-modality: probably mixture of cores .

Carnegie Mellon

details



Running time - Kronecker and Erdos-Renyi Graphs with billions edges.

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - overview
  - Triangles
  - Diameter
  - `Eigenspokes'
  - Phonecall duration
- Problem#2: Tools
- Conclusions

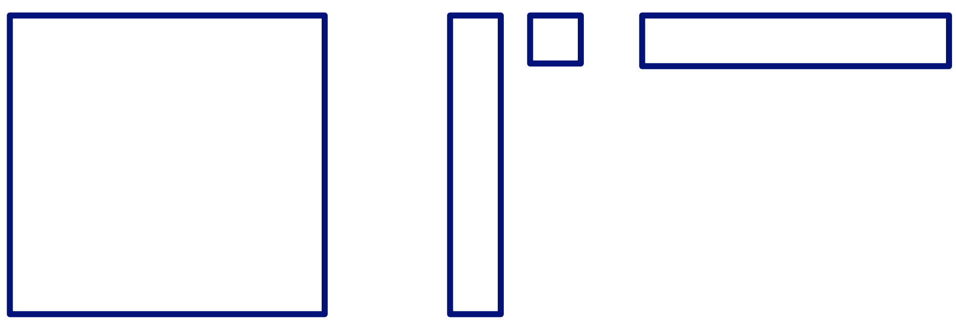
EigenSpokes

B. Aditya Prakash, Mukund Seshadri, Ashwin Sridharan, Sridhar Machiraju and Christos Faloutsos: *EigenSpokes: Surprising Patterns and Scalable Community Chipping in Large Graphs*, PAKDD 2010, Hyderabad, India, 21-24 June 2010.

EigenSpokes

- Eigenvectors of adjacency matrix
  - equivalent to singular vectors (symmetric, undirected graph)

$$A = U \Sigma U^T$$

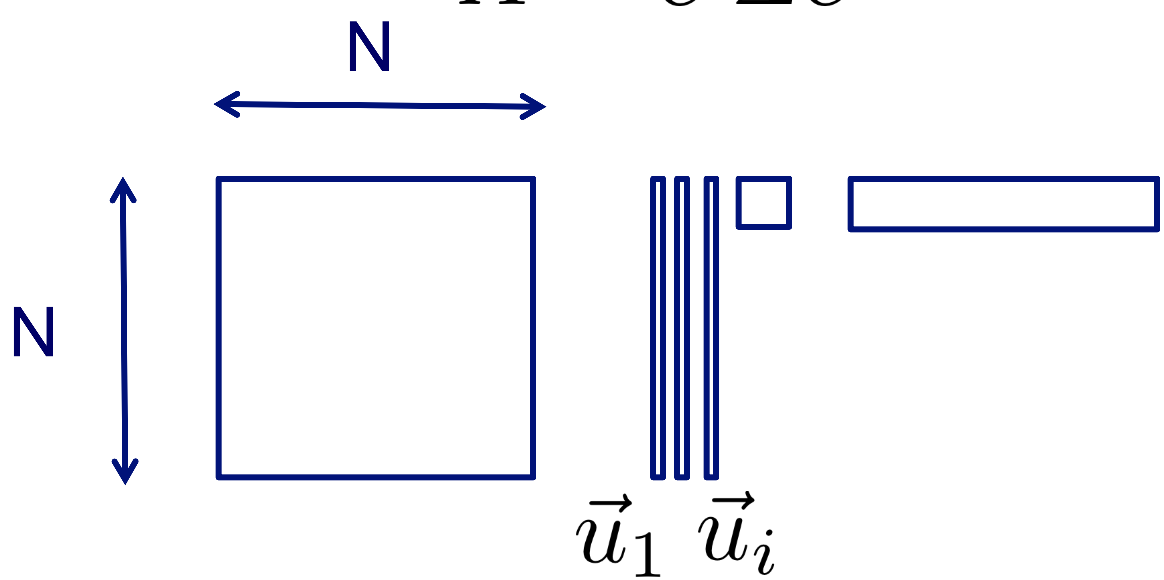


details

EigenSpokes

- Eigenvectors of adjacency matrix
  - equivalent to singular vectors (symmetric, undirected graph)

$$A = U \Sigma U^T$$

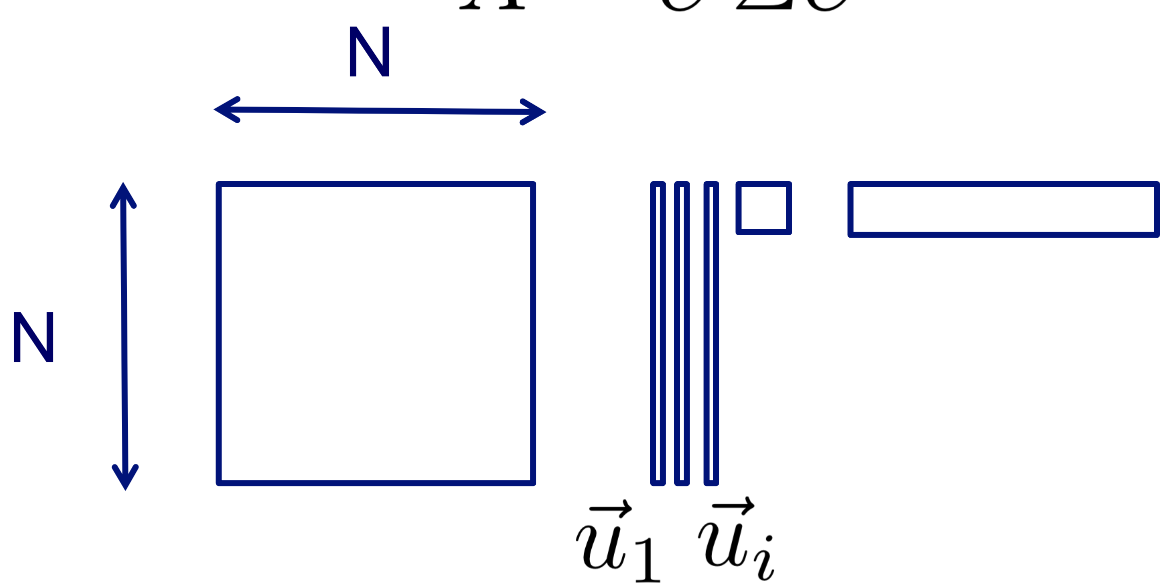


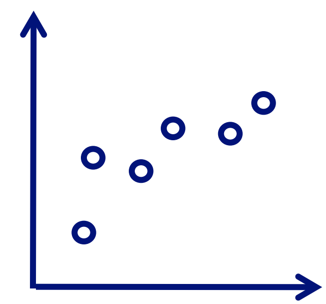
details

EigenSpokes

- Eigenvectors of adjacency matrix
  - equivalent to singular vectors (symmetric, undirected graph)

$$A = U \Sigma U^T$$



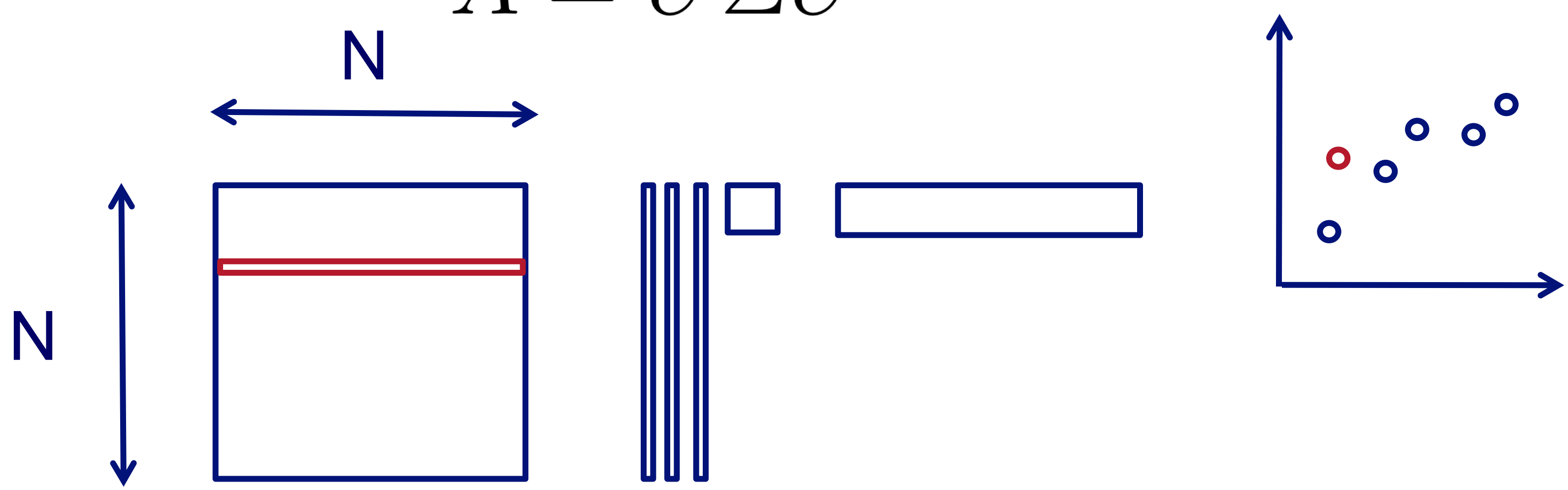


details

EigenSpokes

- Eigenvectors of adjacency matrix
  - equivalent to singular vectors (symmetric, undirected graph)

$$A = U \Sigma U^T$$



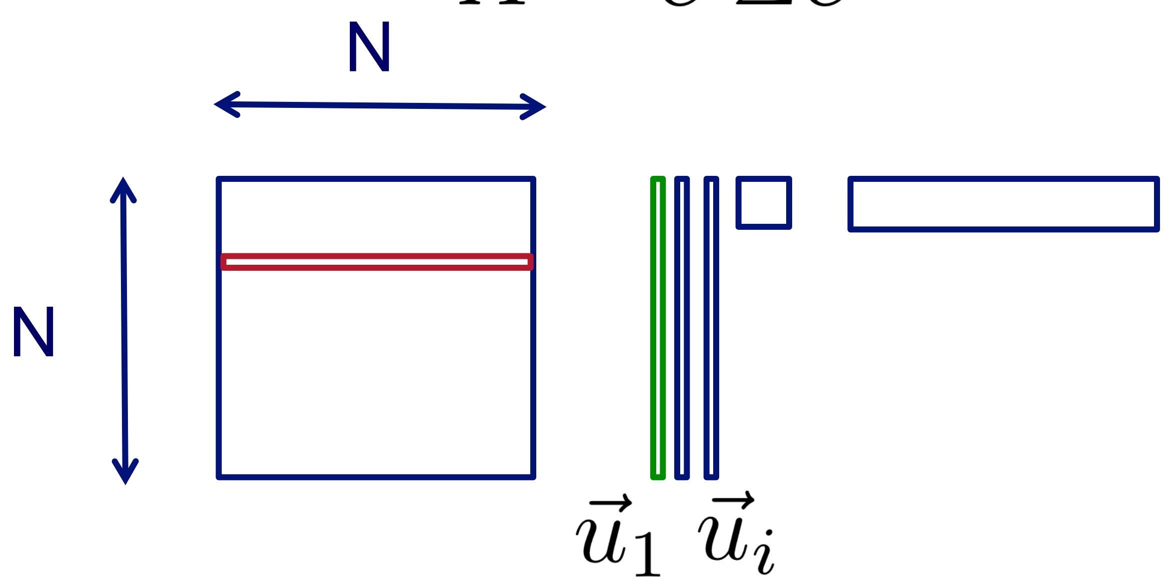
$\vec{u}_1$ $\vec{u}_i$

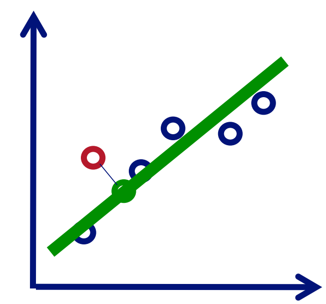
details

EigenSpokes

- Eigenvectors of adjacency matrix
  - equivalent to singular vectors (symmetric, undirected graph)

$$A = U \Sigma U^T$$



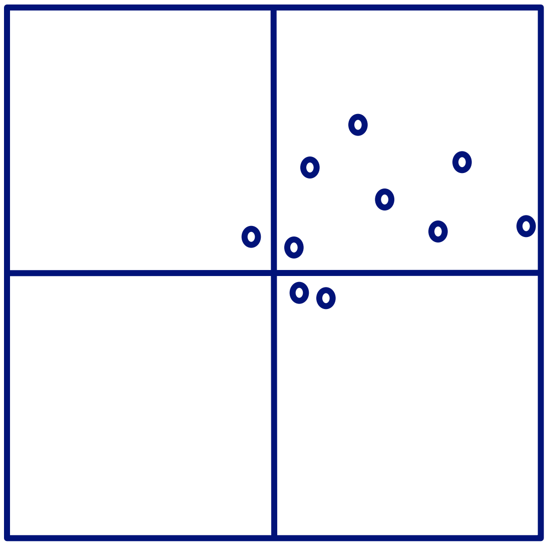


Carnegie Mellon

EigenSpokes

- EE plot:
- Scatter plot of scores of u1 vs u2
- One would expect
  - Many points @ origin
  - A few scattered ~randomly

2nd Principal component
u2

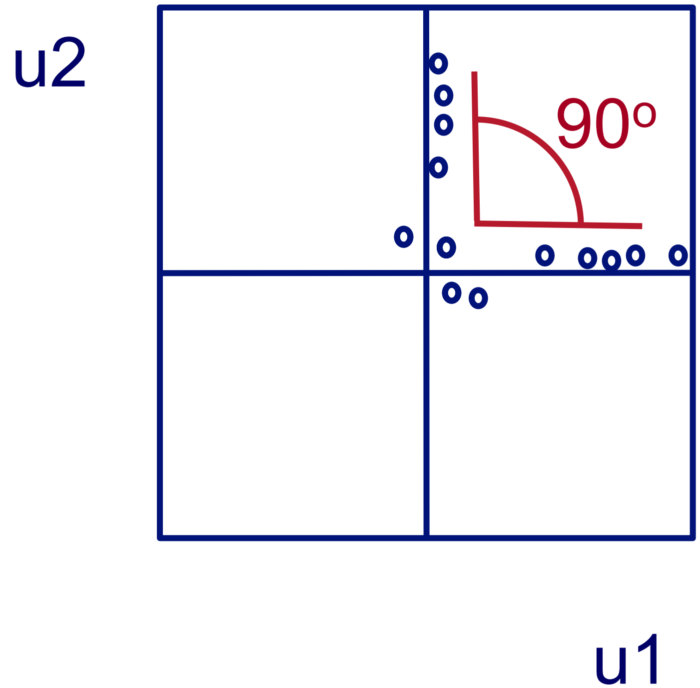


u1

1st Principal component

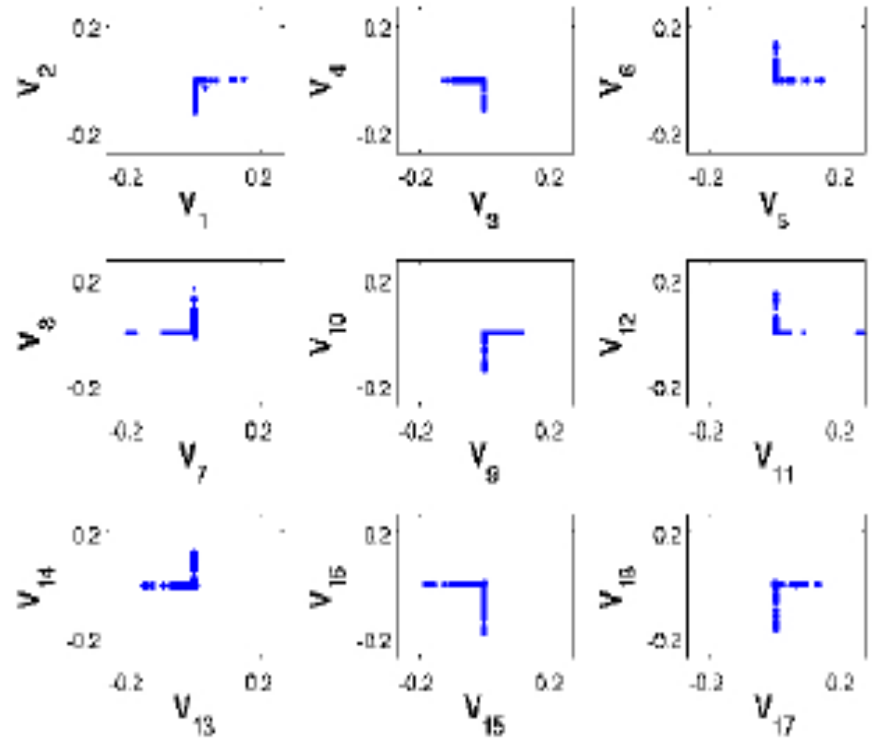
EigenSpokes

- EE plot:
- Scatter plot of scores of u1 vs u2
- One would expect
  - Many points @ origin
  - A few scattered ~randomly

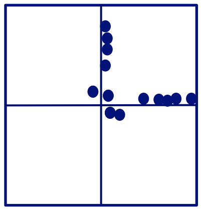


EigenSpokes - pervasiveness

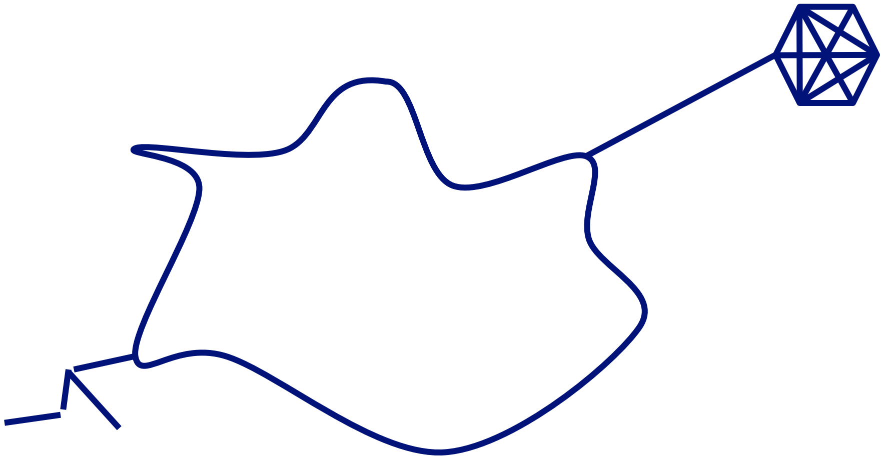
- Present in mobile social graph
  - across time and space
- Patent citation graph



EigenSpokes - explanation

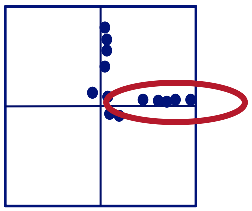


Near-cliques, or near-bipartite-cores, loosely connected

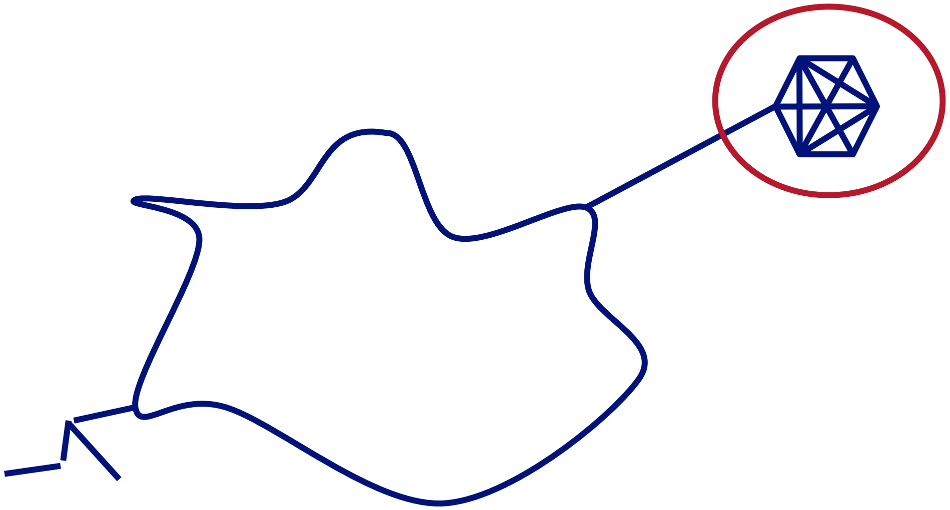


Carnegie Mellon

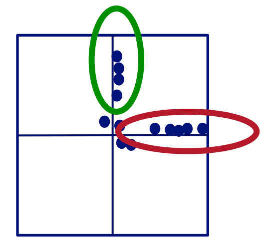
EigenSpokes - explanation



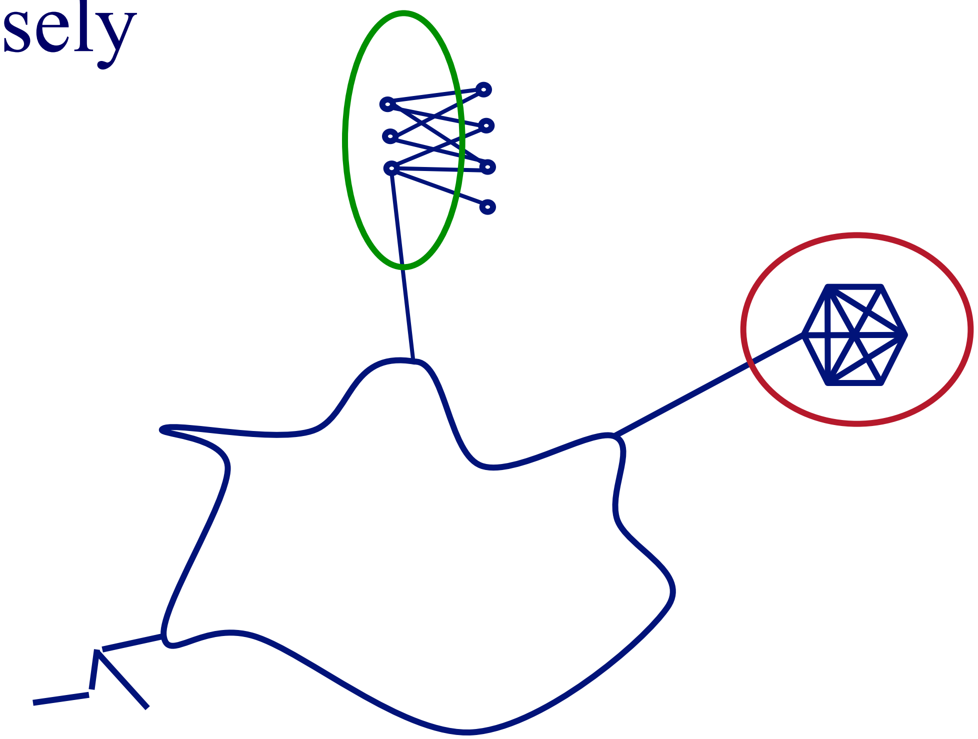
Near-cliques, or near-bipartite-cores, loosely connected



EigenSpokes - explanation

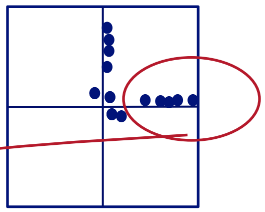


Near-cliques, or near-bipartite-cores, loosely connected

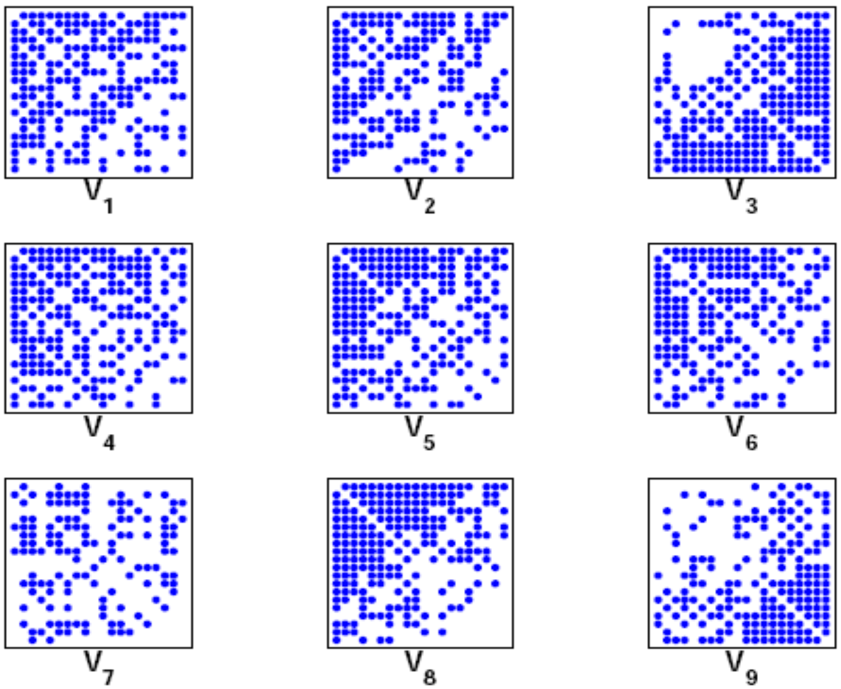


EigenSpokes - explanation

Near-cliques, or near-bipartite-cores, loosely connected



spy plot of top 20 nodes



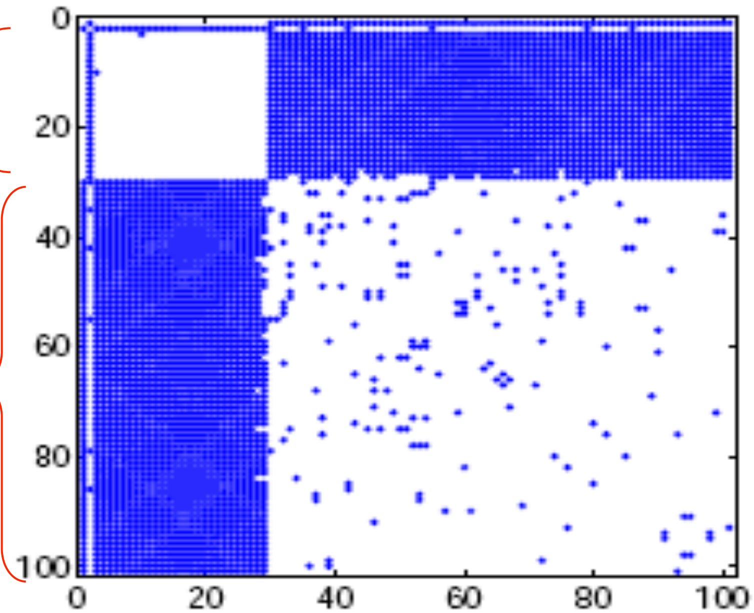
So what?

- Extract nodes with high *scores*
- high connectivity
- Good “communities”

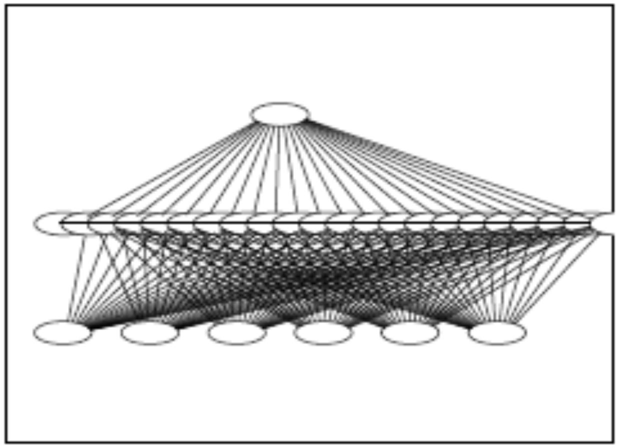
Bipartite Communities!

patents from
same inventor(s)

`cut-and-paste'
bibliography!



magnified bipartite community



Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - overview
  - Triangles
  - Diameter
  - `Eigenspokes’
  - Phonecall duration
- Problem#2: Tools
- Conclusions

Real Graph Patterns

	unweighted	weighted
static	**P01**. Power-law degree distribution [Faloutsos et. al.`99, Kleinberg et. al.`99, Chakrabarti et. al. `04, Newman`04] **P02**. Triangle Power Law [Tsourakakis `08] **P03**. Eigenvalue Power Law [Siganos et. al. `03] **P04**. Community structure [Flake et. al.`02, Girvan and Newman `02] **P05**. Clique Power Laws [Du et. al. '09]	**P12**. Snapshot Power Law [McGlohon et. al. `08]
dynamic	**P06**. Densification Power Law [Leskovec et. al.`05] **P07**. Small and shrinking diameter [Albert and Barabási `99, Leskovec et. al. '05, McGlohon et. al. '08] **P08**. Gelling point [McGlohon et. al. `08] **P09**. Constant size 2nd and 3rd connected components [McGlohon et. al. `08] **P10**. Principal Eigenvalue Power Law [Akoglu et. al. `08] **P11**. Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et. al. `98, Crovella and Bestavros `99, McGlohon et .al. `08]	**P13**. Weight Power Law [McGlohon et. al. `08] **P14**. Skewed call duration distributions [Vaz de Melo et. al. `10]

RTG: A Recursive Realistic Graph Generator using Random Typing
Leman Akoglu and Christos Faloutsos. *ECML PKDD*'09.

Carnegie Mellon

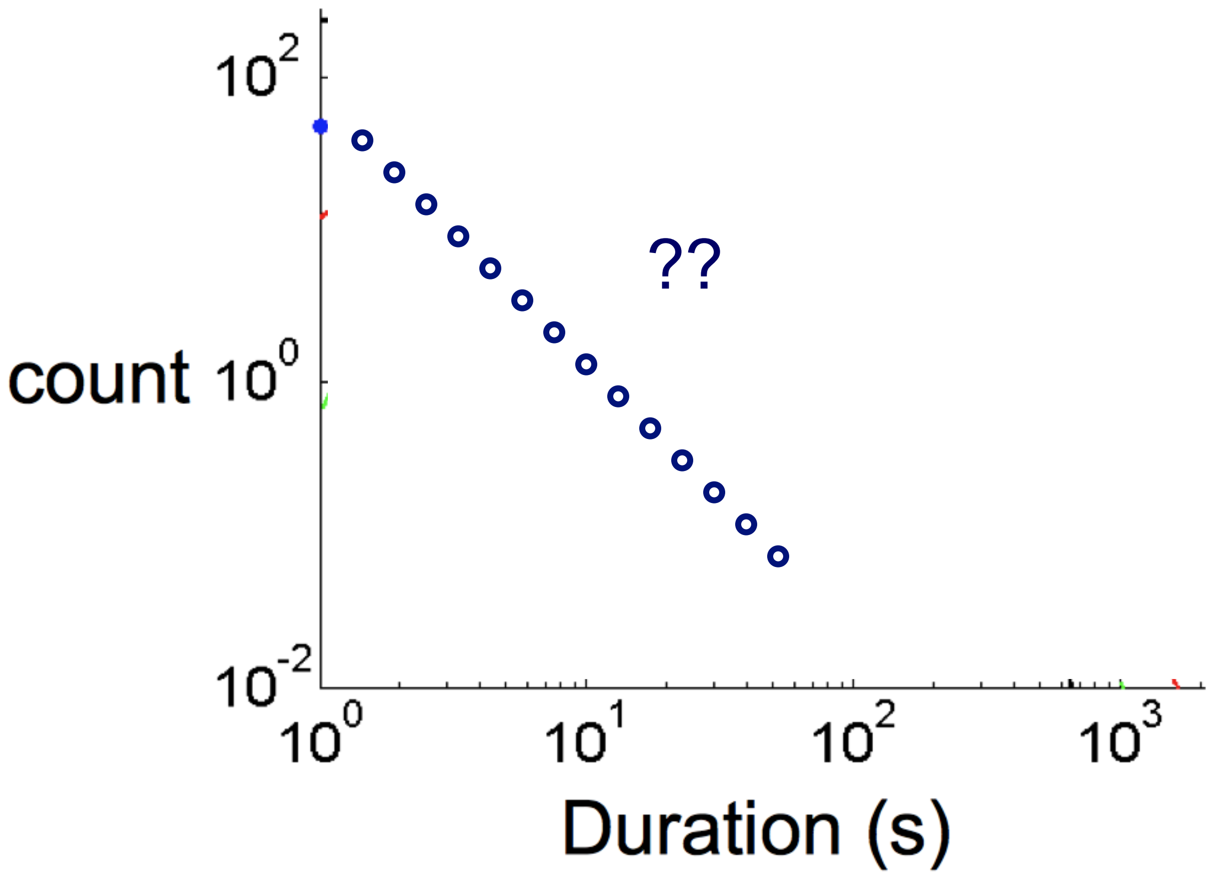
Duration of phonecalls

Surprising Patterns for the Call Duration Distribution of Mobile Phone Users

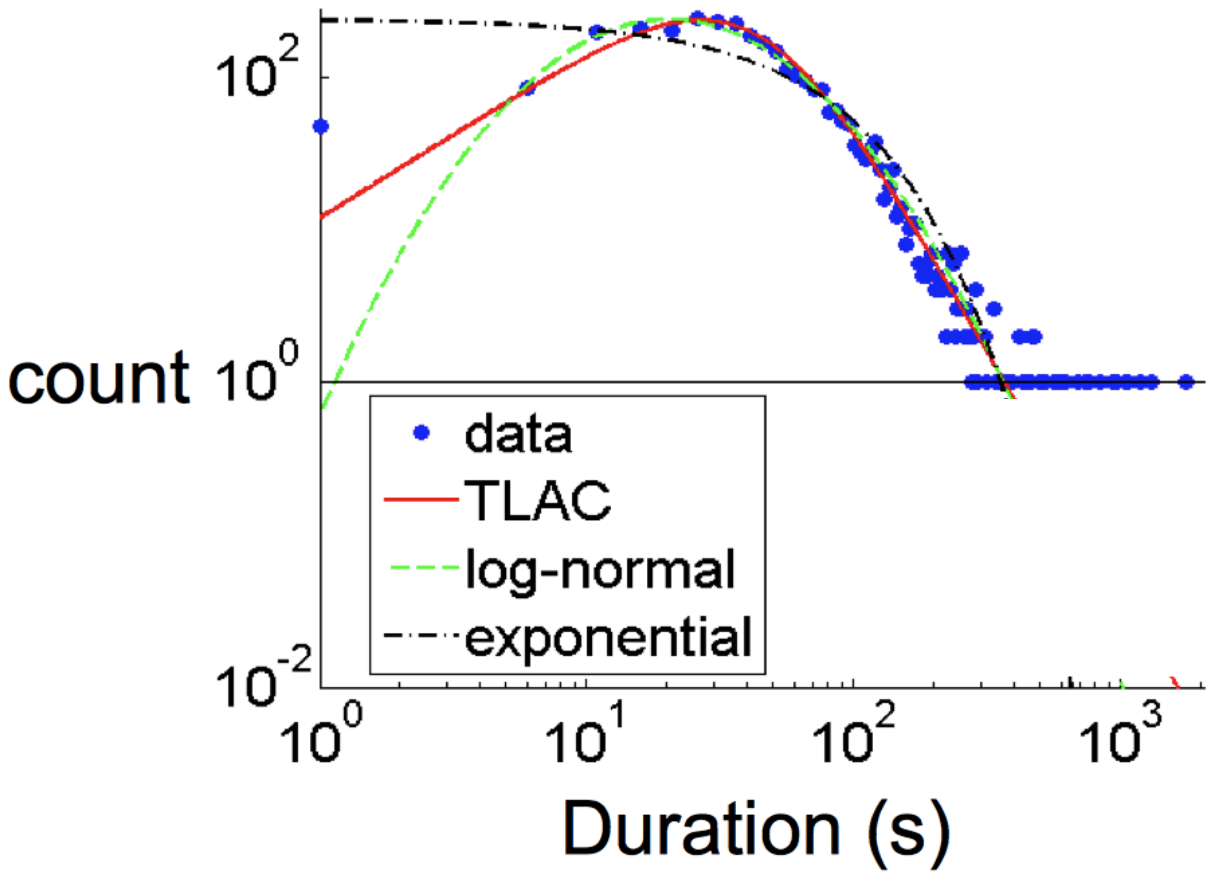
Pedro O. S. Vaz de Melo, Leman Akoglu, Christos Faloutsos, Antonio A. F. Loureiro

PKDD 2010

Probably, power law (?)

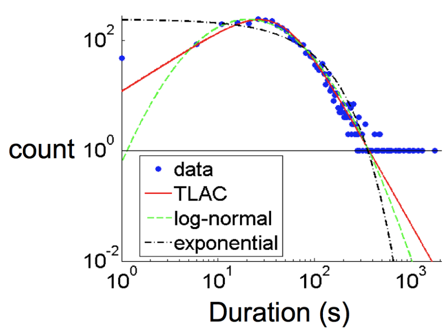


No Power Law!?


count
10^2
10^0
10^{-2}
10^0
10^1
10^2
10^3
Duration (s)
data
TLAC
log-normal
exponential

'TLaC: Lazy Contractor'

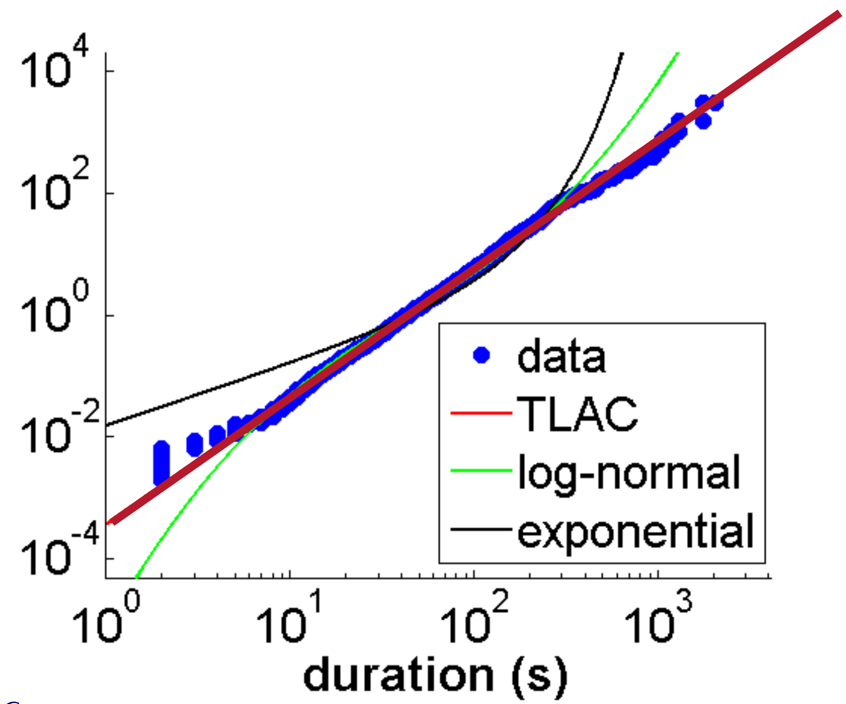
- The longer a task (phonecall) has taken,
- The even longer it will take



Odds ratio=

Casualties(<x):
Survivors(>=x)

== power law

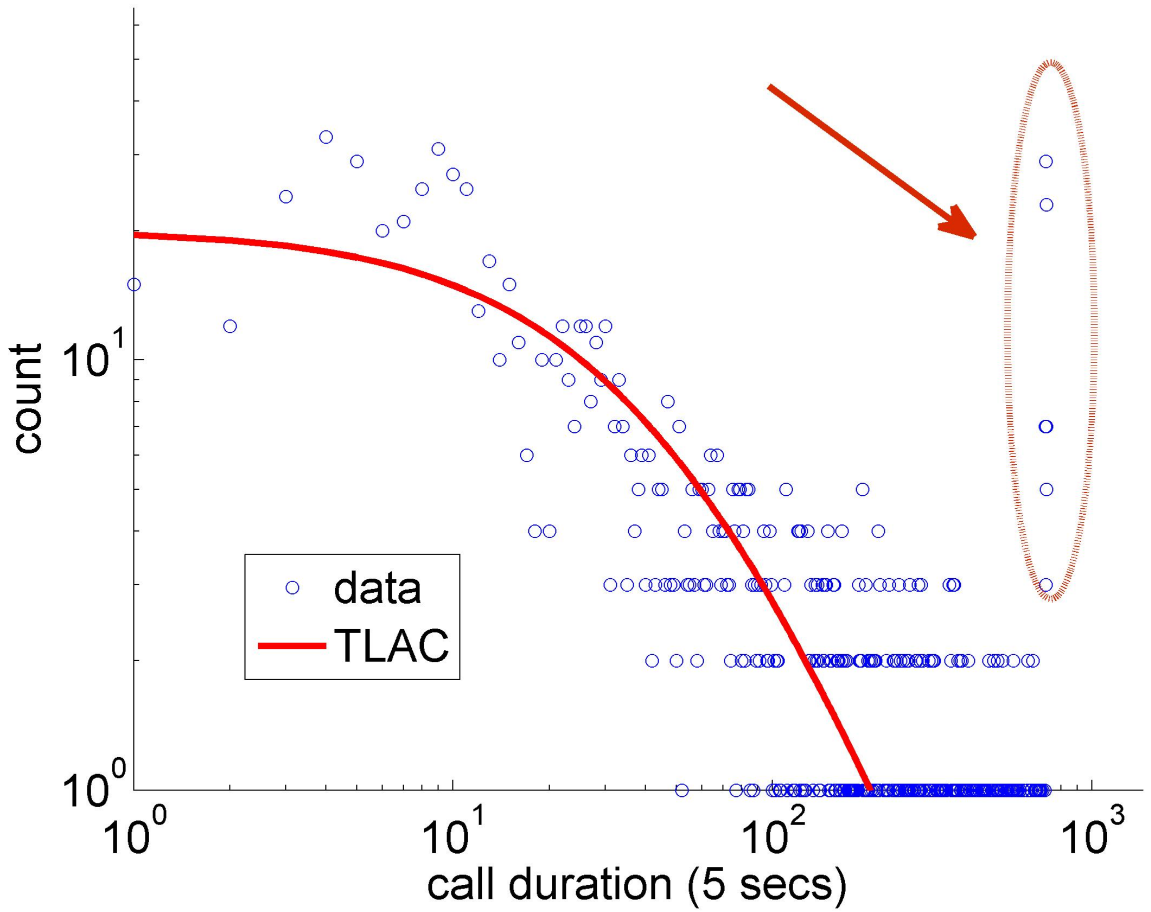


Data Description

- Data from a private mobile operator of a large city
  - 4 months of data
  - 3.1 million users
  - more than 1 billion phone records
- Over 96% of ‘talkative’ users obeyed a TLAC distribution (‘talkative’: >30 calls)

Carnegie Mellon

Outliers:



Carnegie Mellon

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - …
  - `Eigenspokes’
  - Phonecall duration
  - Connected components
- Problem#2: Tools
- Conclusions

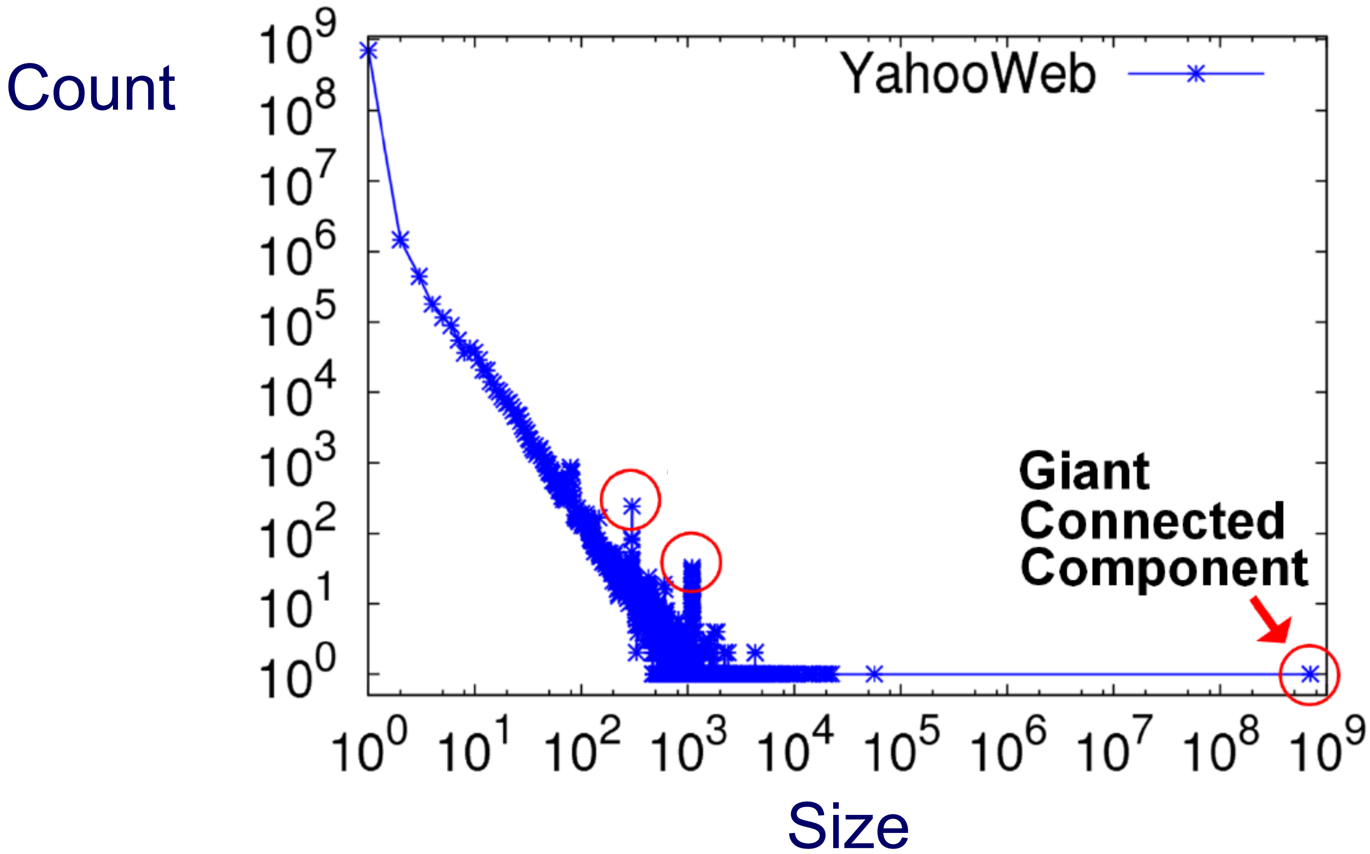
Carnegie Mellon

Generalized Iterated Matrix Vector Multiplication (GIMV)

PEGASUS: A Peta-Scale Graph Mining System - Implementation and Observations. U Kang, Charalampos E. Tsourakakis, and Christos Faloutsos. (ICDM) 2009, Miami, Florida, USA. Best Application Paper (runner-up).

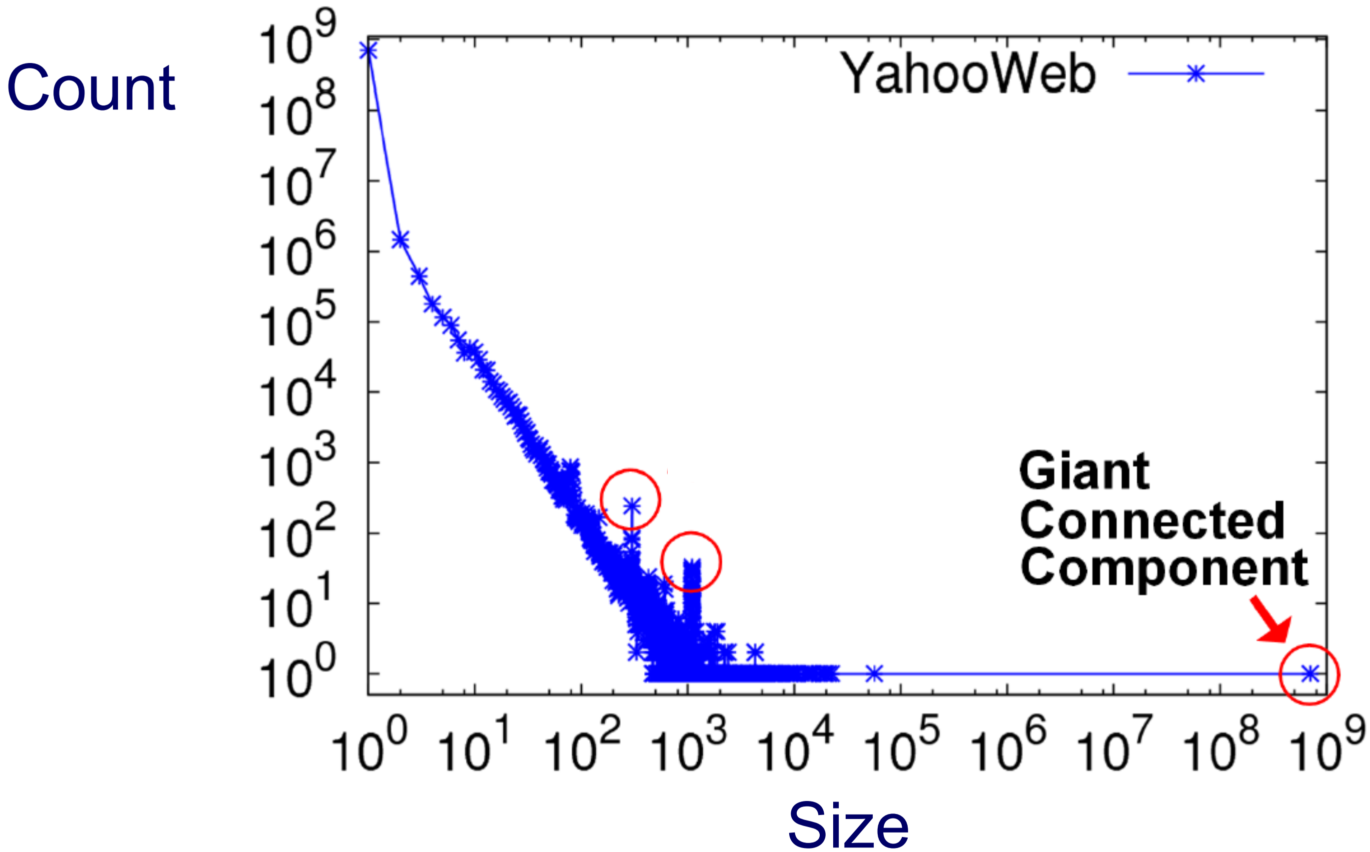
Example: GIM-V At Work

- Connected Components – 4 observations:



Example: GIM-V At Work

- Connected Components



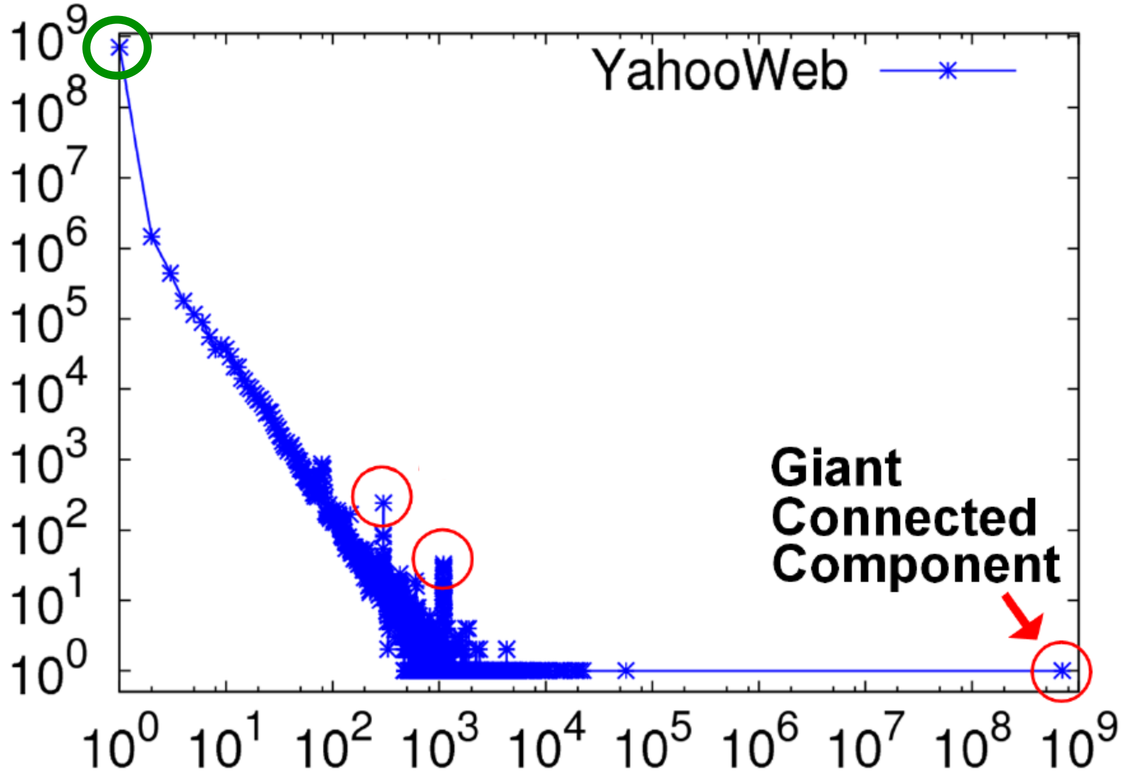
1) 10K x larger than next

Example: GIM-V At Work

- Connected Components

Count

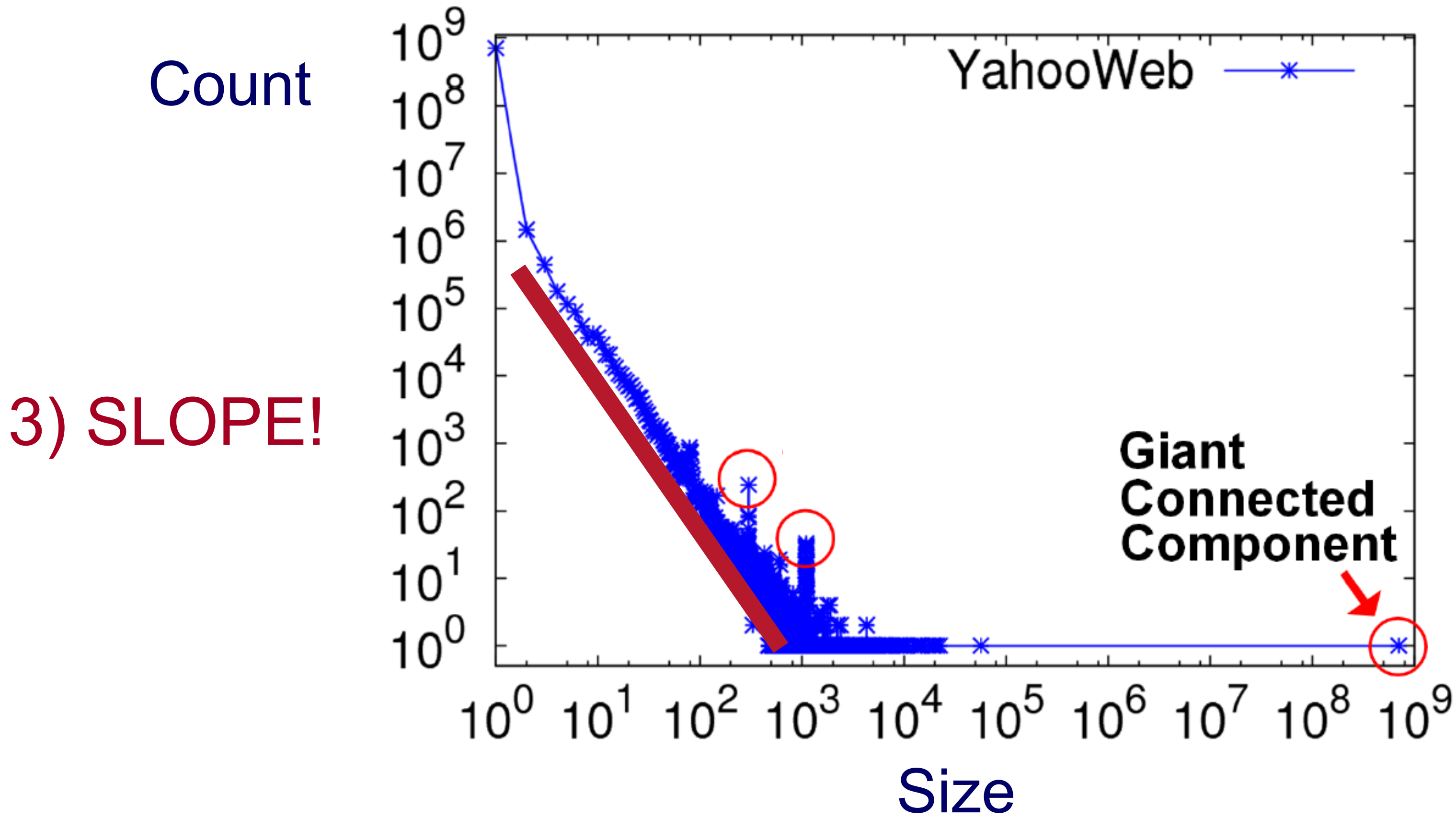
2) ~0.7B singleton nodes



Size

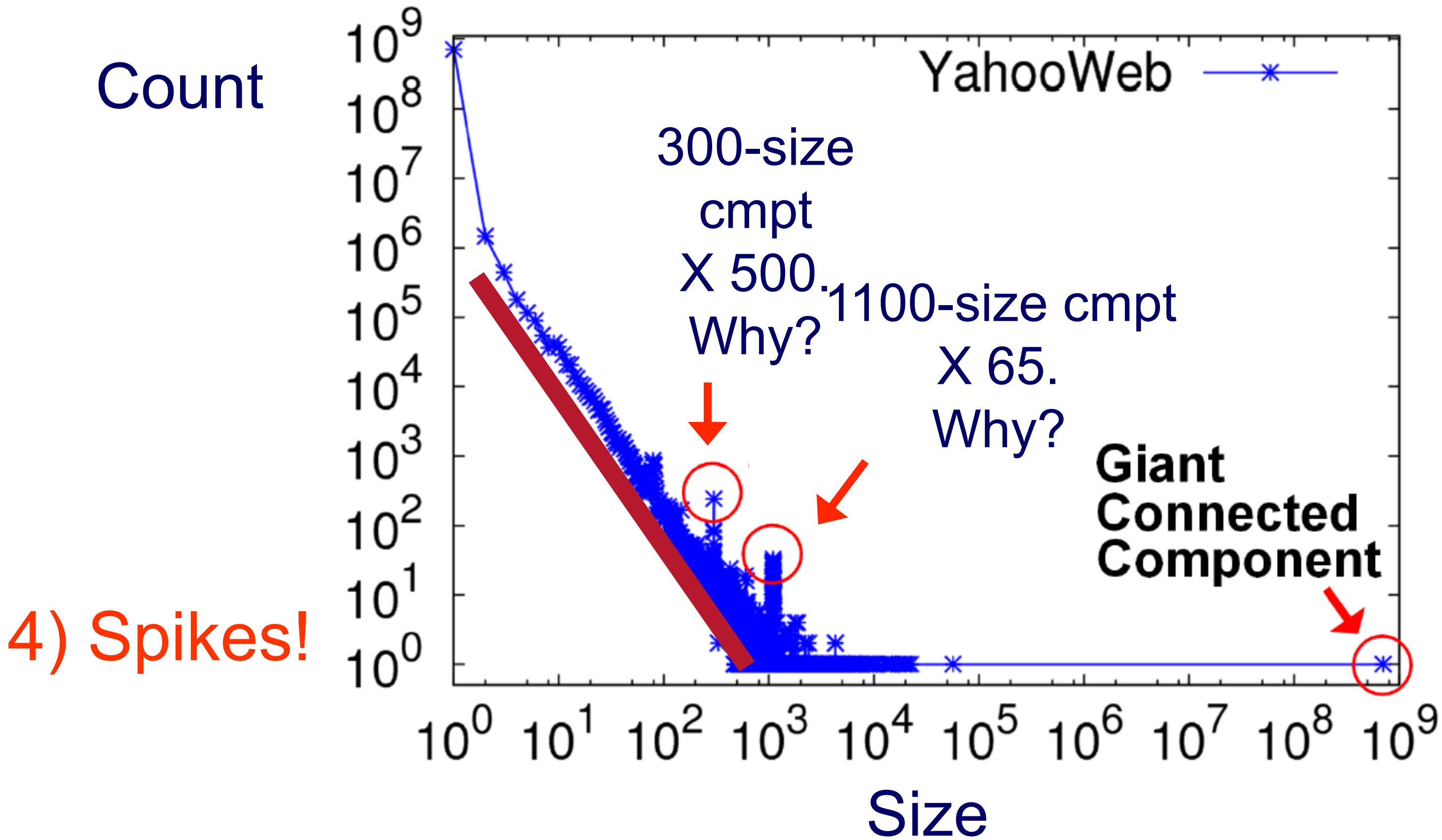
Example: GIM-V At Work

- Connected Components



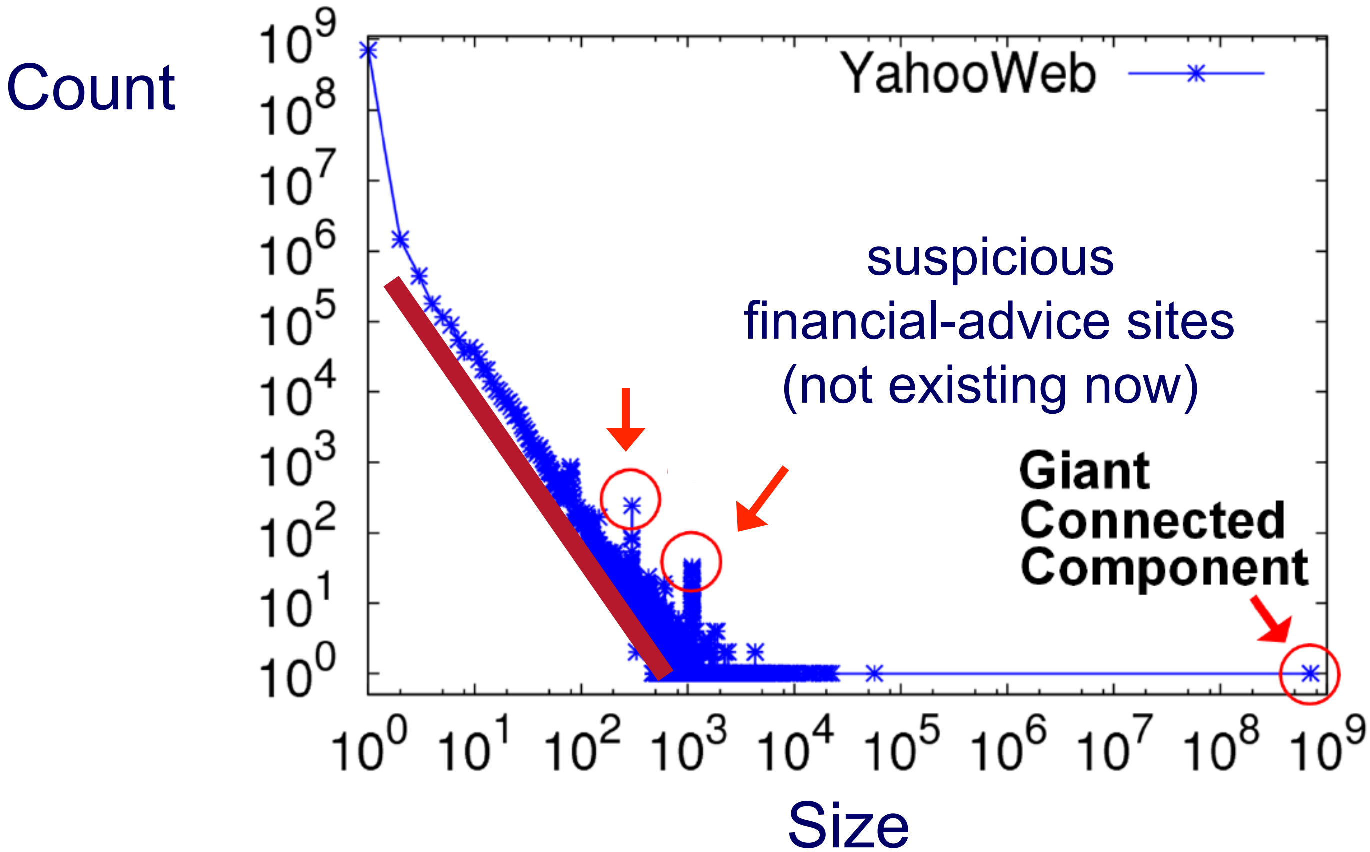
Example: GIM-V At Work

- Connected Components



Example: GIM-V At Work

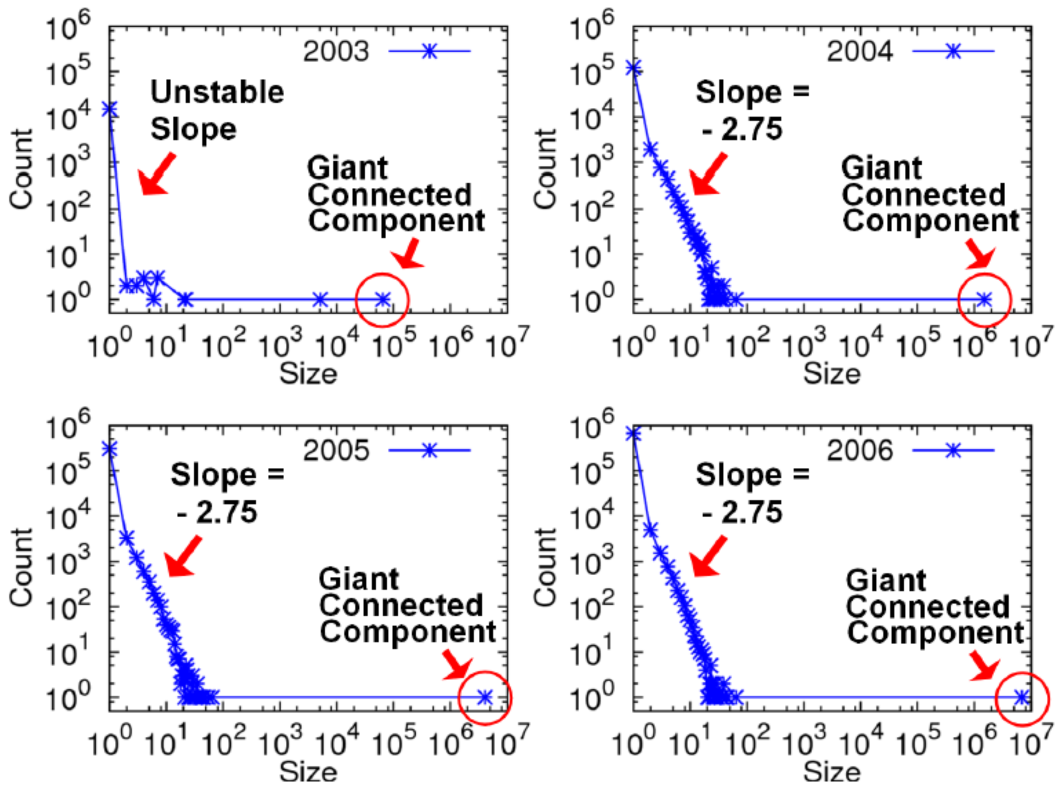
- Connected Components



Carnegie Mellon

GIM-V At Work

- Connected Components over Time
- **LinkedIn: 7.5M nodes and 58M edges**



Stable tail slope after the gelling point

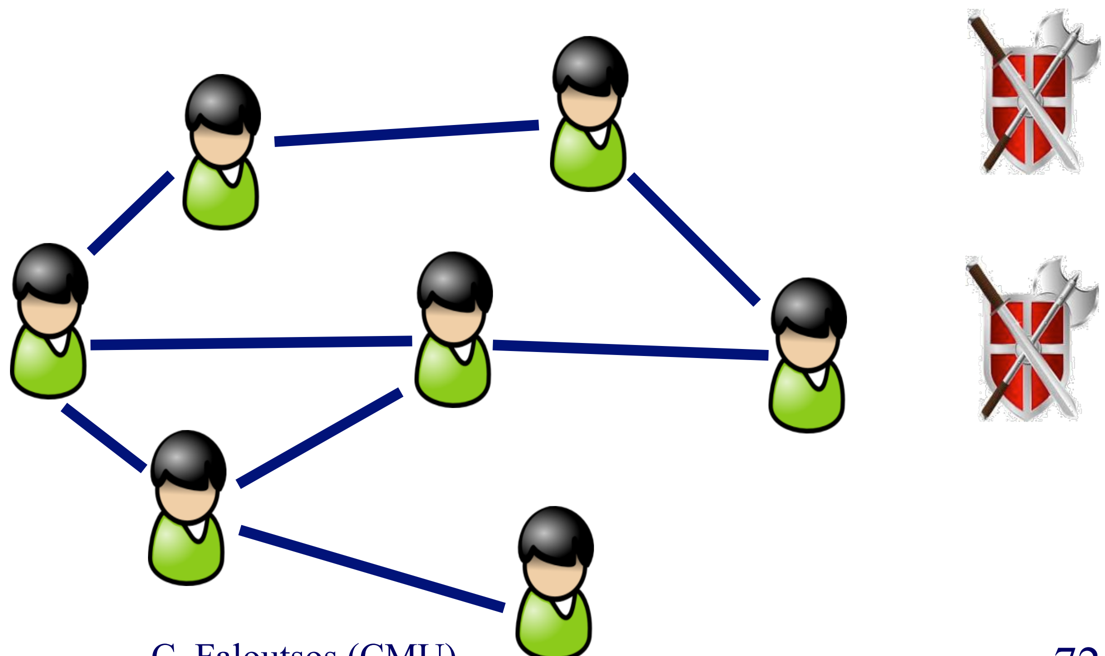
Carnegie Mellon

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - Immunization
  - BP
  - visualization
- Conclusions

Immunization and epidemic thresholds

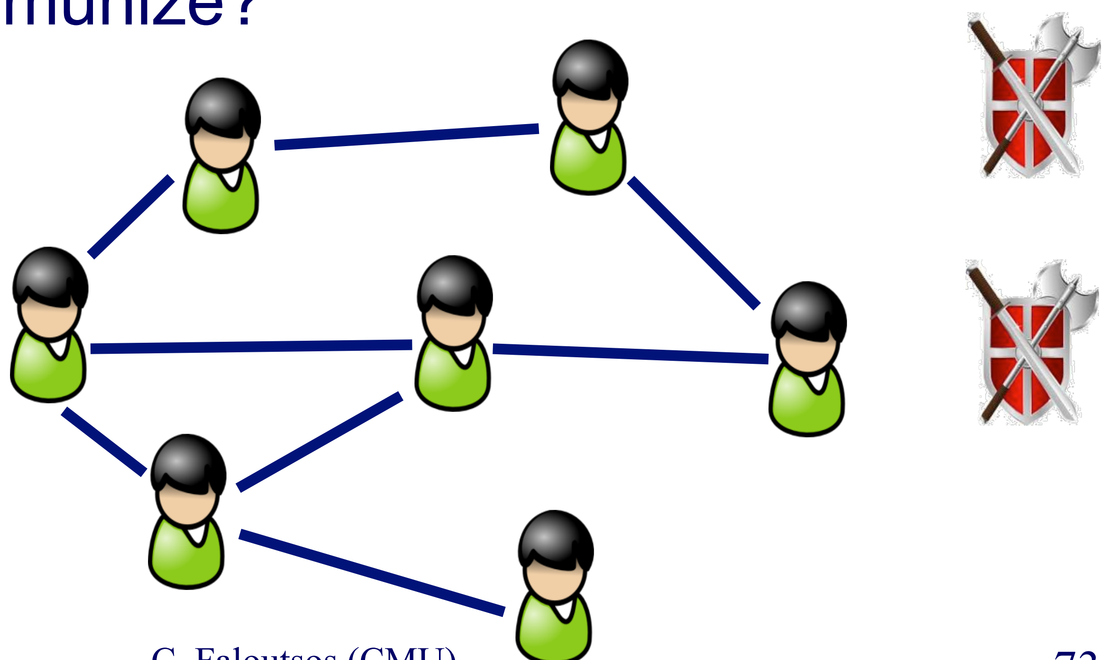
- Q1: which nodes to immunize?
- Q2: will a virus vanish, or will it create an epidemic?



Carnegie Mellon

Q1: Immunization:

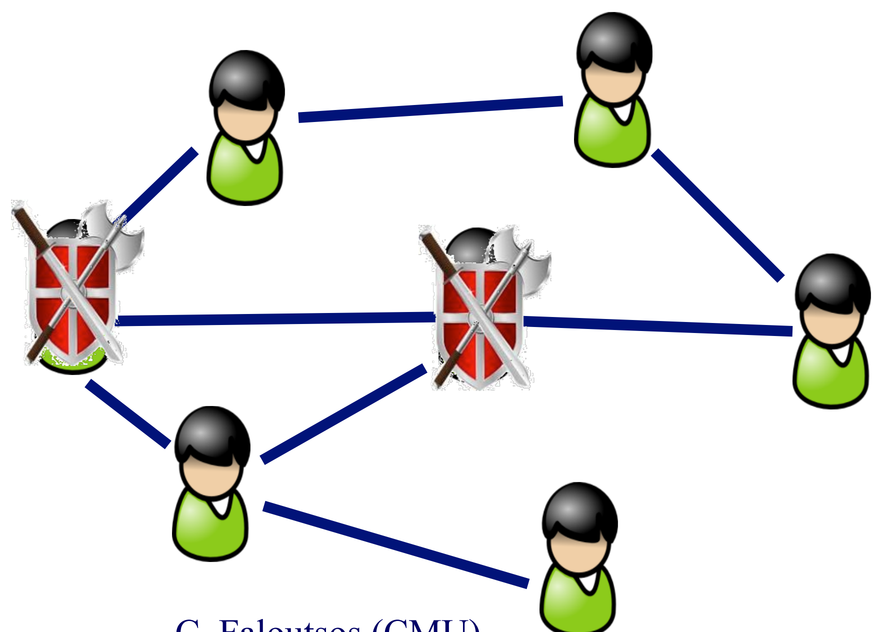
- Given
  - a network,
  - k vaccines, and
  - the virus details
- Which nodes to immunize?



Carnegie Mellon

Q1: Immunization:

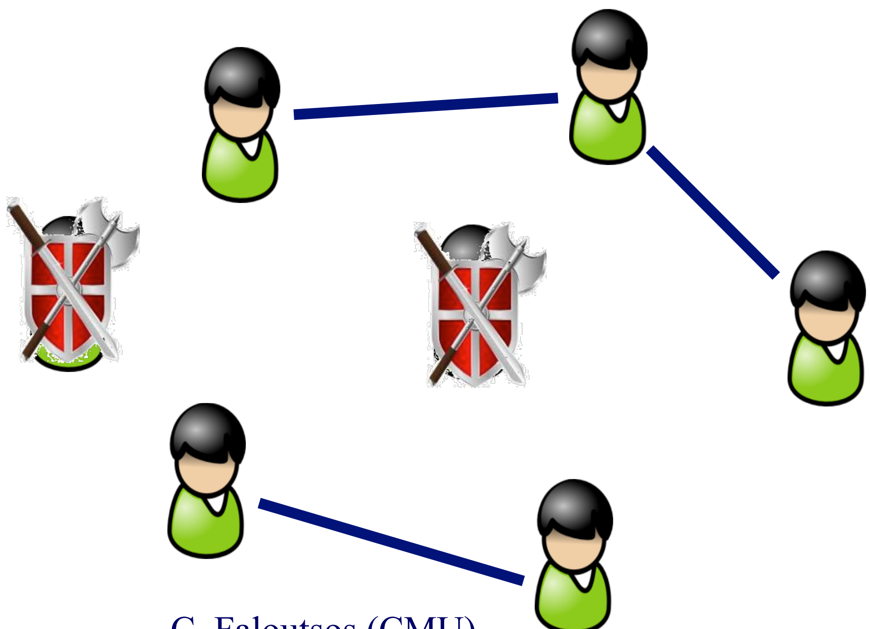
- Given
  - a network,
  - k vaccines, and
  - the virus details
- Which nodes to immunize?



Carnegie Mellon

Q1: Immunization:

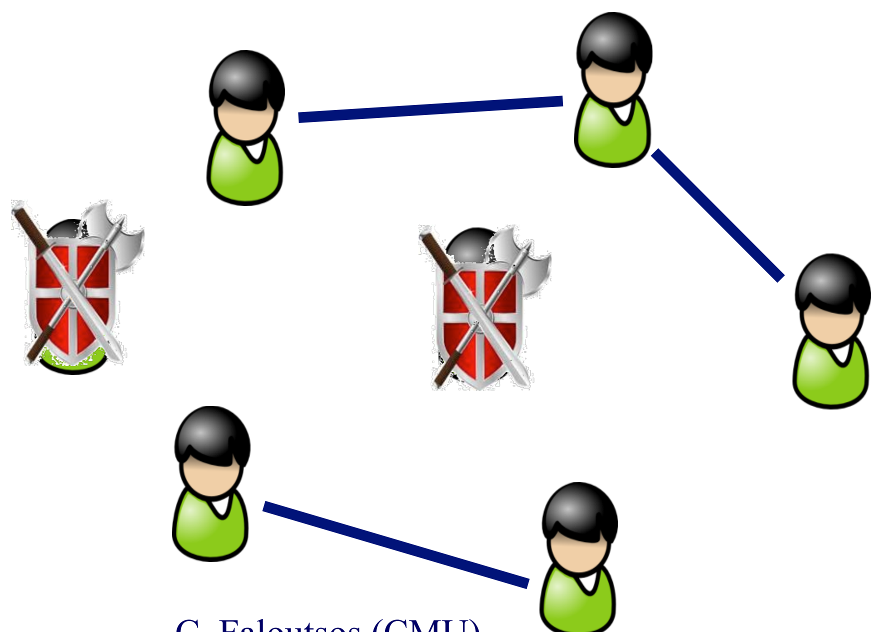
- Given
  - a network,
  - k vaccines, and
  - the virus details
- Which nodes to immunize?



Q1: Immunization:

- Given
  - a network,
  - k vaccines, and
  - the virus details
- Which nodes to immunize?

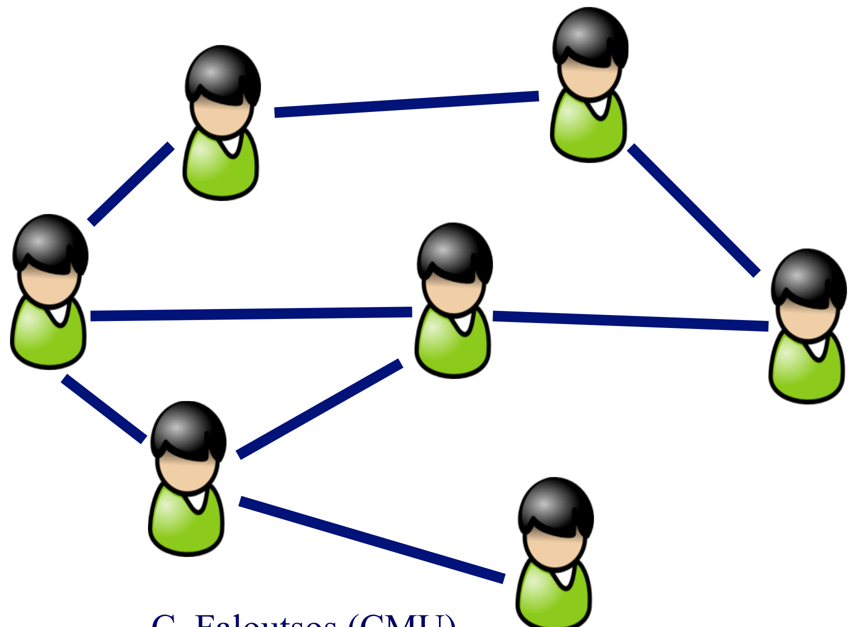
A: immunize the ones that maximally raise the `epidemic threshold' [Tong+, ICDM'10]



Q2: will a virus take over?

- Flu-like virus (no immunity, 'SIS')
- Mumps (life-time immunity, 'SIR')
- Pertussis (finite-length immunity, 'SIRS')

β: attack prob
δ: heal prob

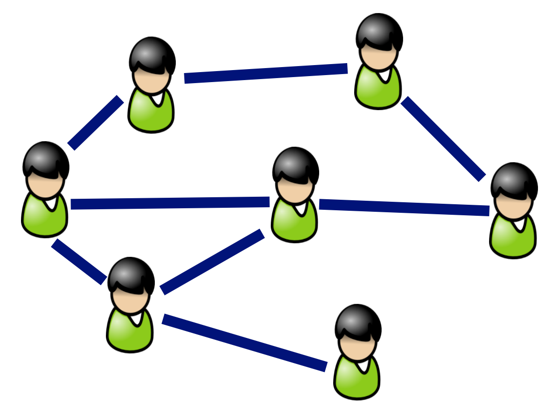


Q2: will a virus take over?

- Flu-like virus (no immunity, ‘SIS’)
- Mumps (life-time immunity, ‘SIR’)
- Pertussis (finite-length immunity, ‘SIRS’)

β: attack prob
δ: heal prob

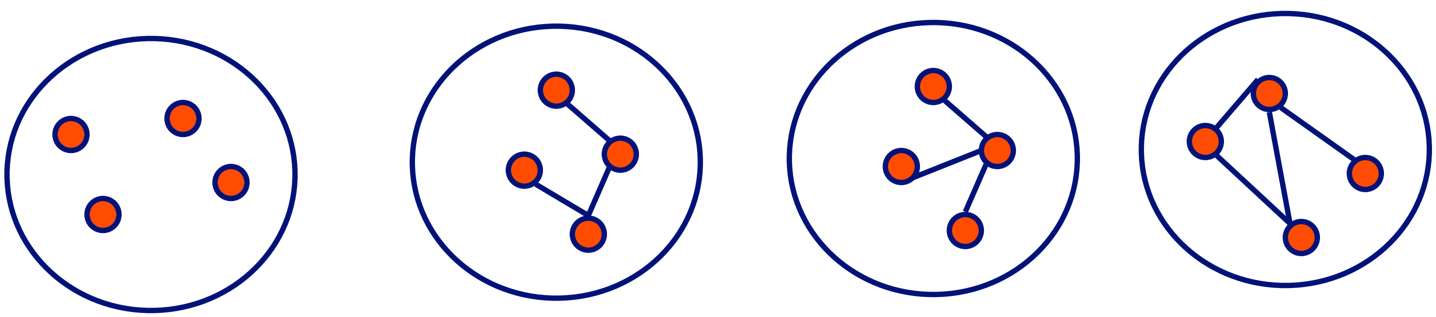
A: depends on connectivity
(avg degree? Something else?)



Epidemic threshold τ

What should τ depend on?

- avg. degree? and/or highest degree?
- and/or variance of degree?
- and/or third moment of degree?
- and/or diameter?



Epidemic threshold - SIS

- [Theorem] We have no epidemic, if

$$\boxed{\beta/\delta < \tau = 1/\lambda_{1,A}}$$

Carnegie Mellon

Epidemic threshold - SIS

- [Theorem] We have no epidemic, if

epidemic threshold

recovery prob.

$$\boxed{\beta/\delta < \tau = 1/\lambda_{1,A}}$$

attack prob.

largest eigenvalue of adj. matrix ***A***

Proof: [Wang+03] (for **SIS**=flu **only**)

Carnegie Mellon

Epidemic threshold - SIS

- [Theorem] We have no epidemic, if

$$\beta/\delta < \tau = 1/\ \lambda_{1,A}$$

What about other V.P.M.? (SIR, SIRS, etc?)

Proof: [Wang+03] (for SIS=flu only)

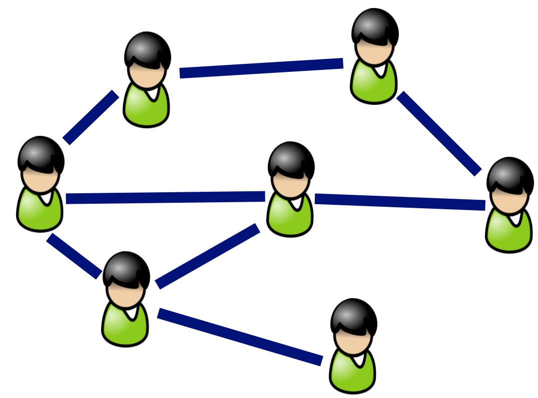
Theorem:

- For **all** typical virus propagation models (flu, mumps, pertussis, HIV, etc)
- The **only** connectivity measure that matters, is

  $1/\lambda_1$

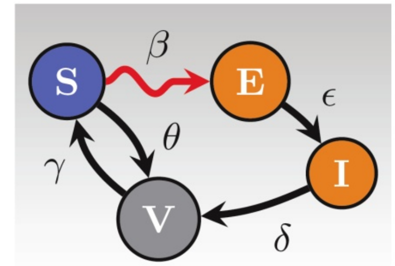
  the first eigenvalue of the adj. matrix

  [Prakash+, '10, arxiv]



Carnegie Mellon

Thresholds for some models



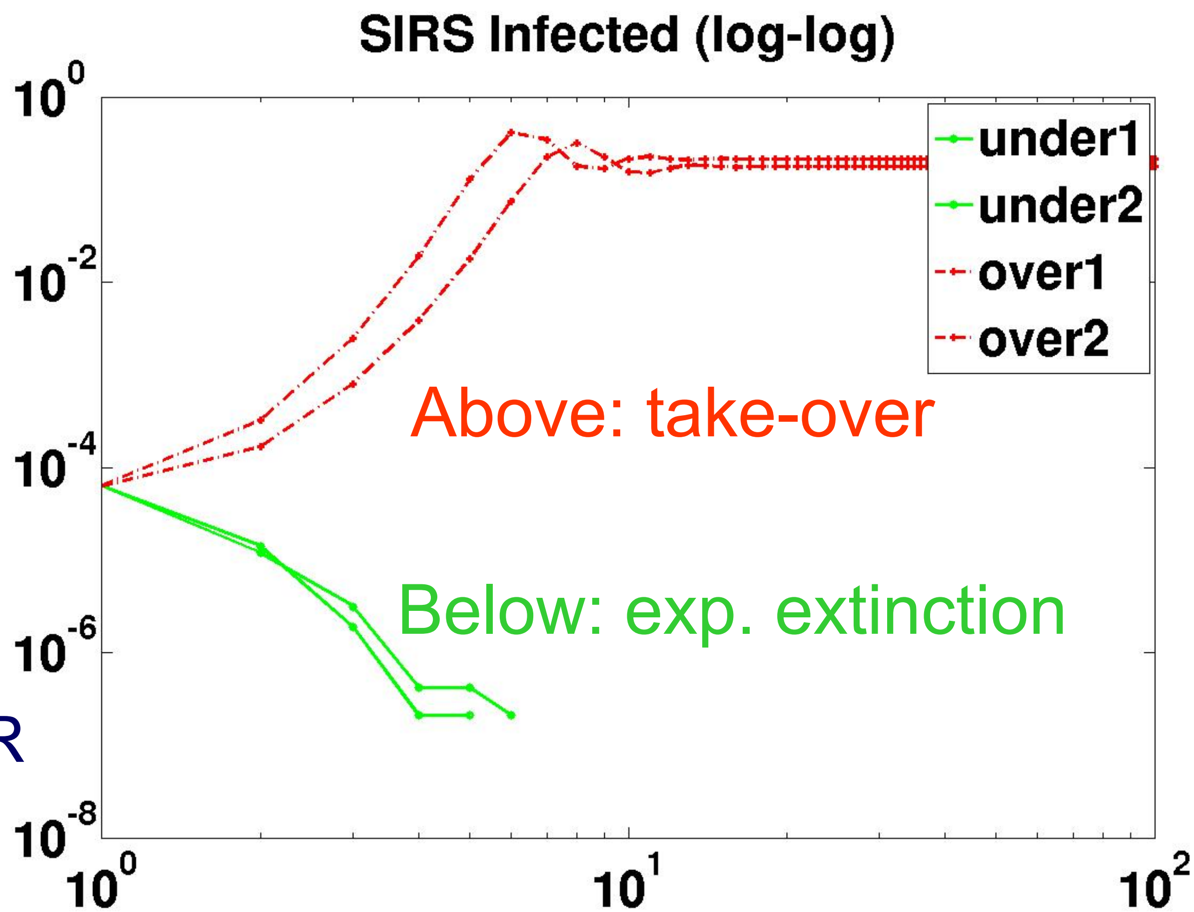
- s = *effective strength*
- $s < 1$: *below threshold*

Models	Effective Strength (s)	Threshold (tipping point)
SIS, SIR, SIRS, SEIR	$s = \lambda \,.\, \left(\frac{\beta}{\delta}\right)$	
SIV, SEIV	$s = \lambda \,.\, \left(\frac{\beta\gamma}{\delta(\gamma+\theta)}\right)$	$s = 1$
$\mathrm{SI_1I_2V_1V_2}$ (**H.I.V.**)	$s = \lambda \,.\, \left(\frac{\beta_1 v_2 + \beta_2 \varepsilon}{v_2(\varepsilon + v_1)}\right)$	

Carnegie Mellon

A2: will a virus take over?

Fraction of infected



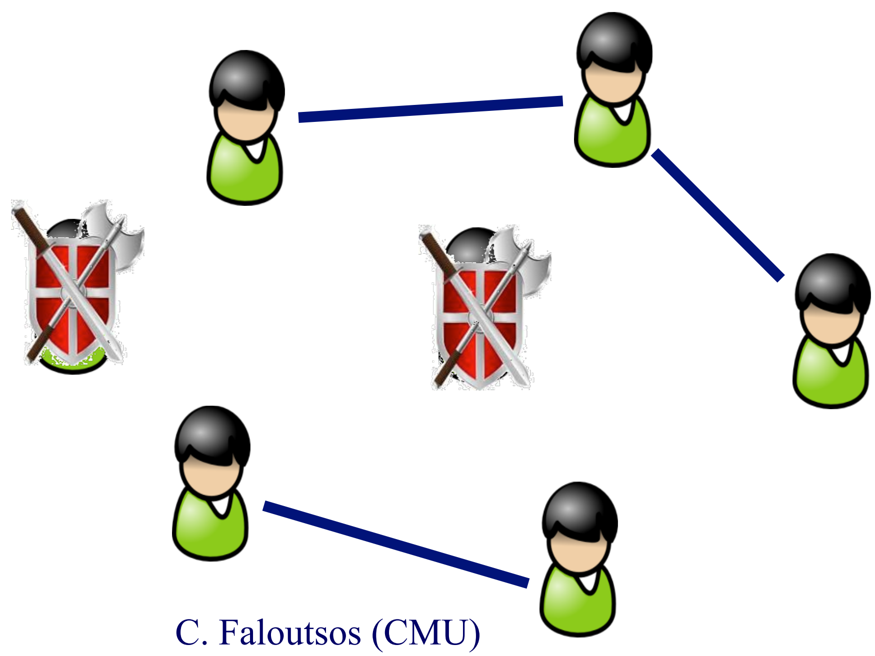
Time ticks

Graph:
Portland, OR
31M links
1.5M nodes

Q1: Immunization:

- Given
  - a network,
  - k vaccines, and
  - the virus details
- Which nodes to immunize?

A: immunize the ones that maximally raise the `epidemic threshold' [Tong+, ICDM'10]

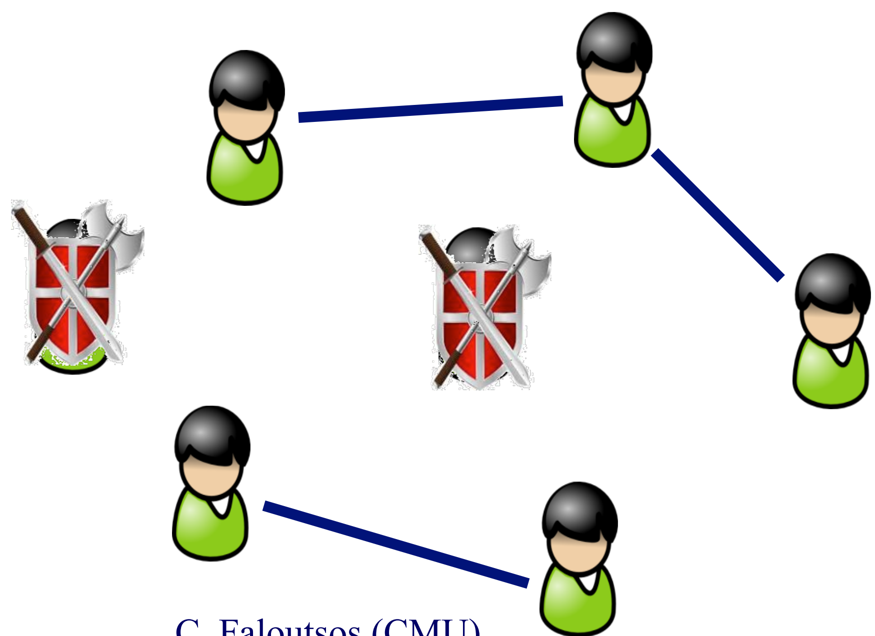


Q1: Immunization:

- Given
    - a network,
    - k vaccines, and
    - the virus details
- Which nodes to immunize?

A: immunize the ones that

Max eigen-drop $\Delta\lambda$
For any virus!

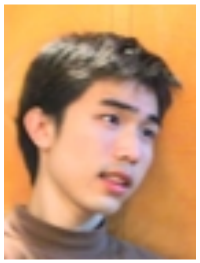


Outline

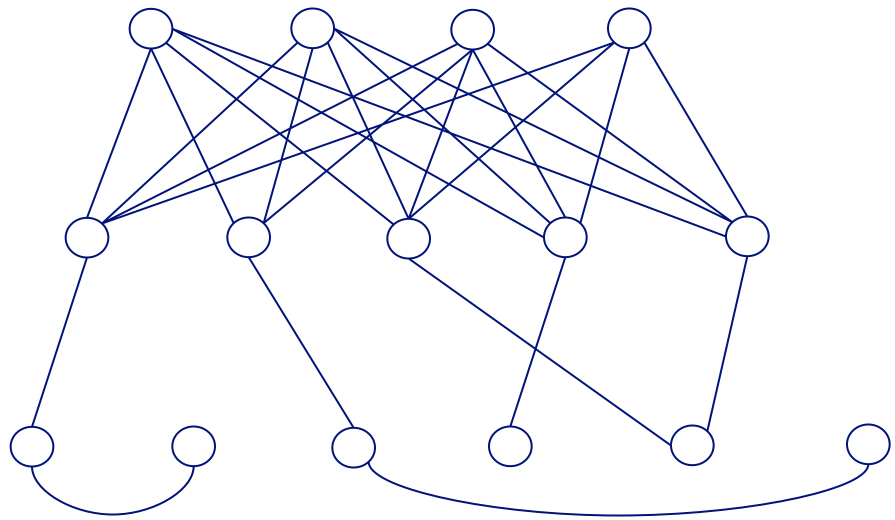
- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - Immunization
  - BP
  - visualization
- Conclusions

Carnegie Mellon

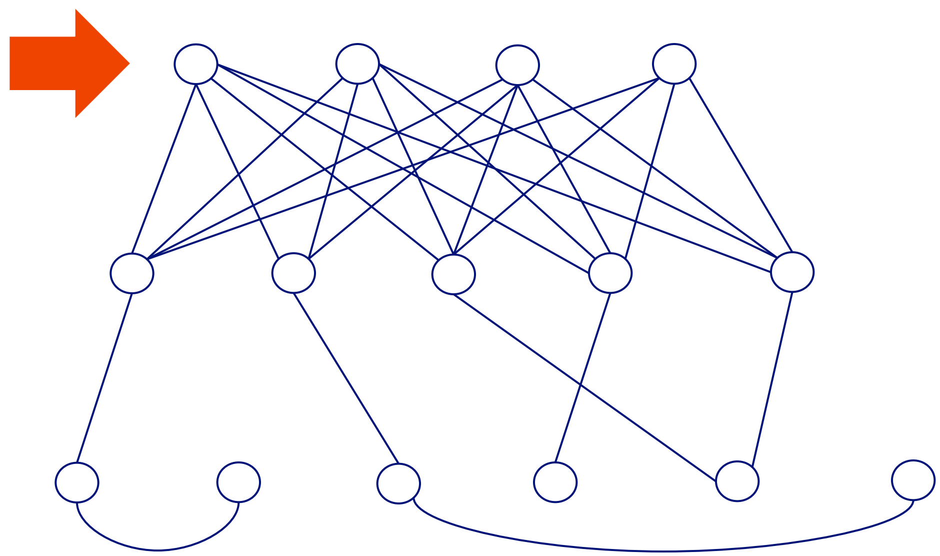
E-bay Fraud detection




w/ Polo Chau &
Shashank Pandit, CMU
[www'07]

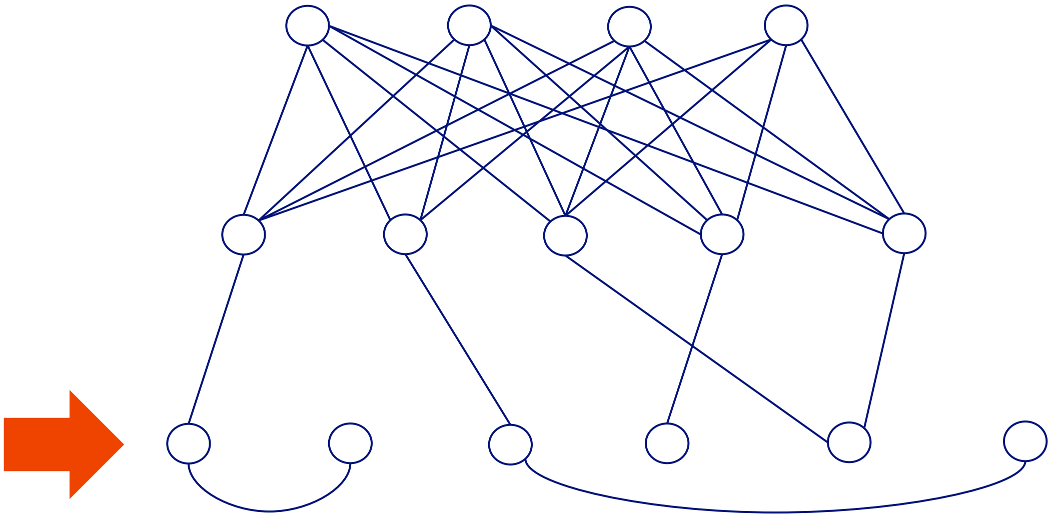


E-bay Fraud detection

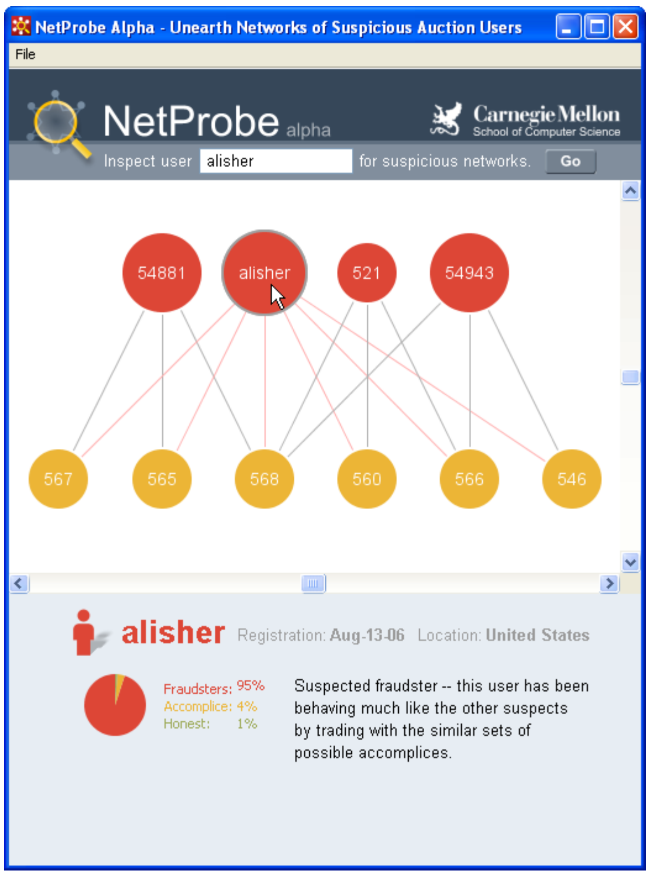


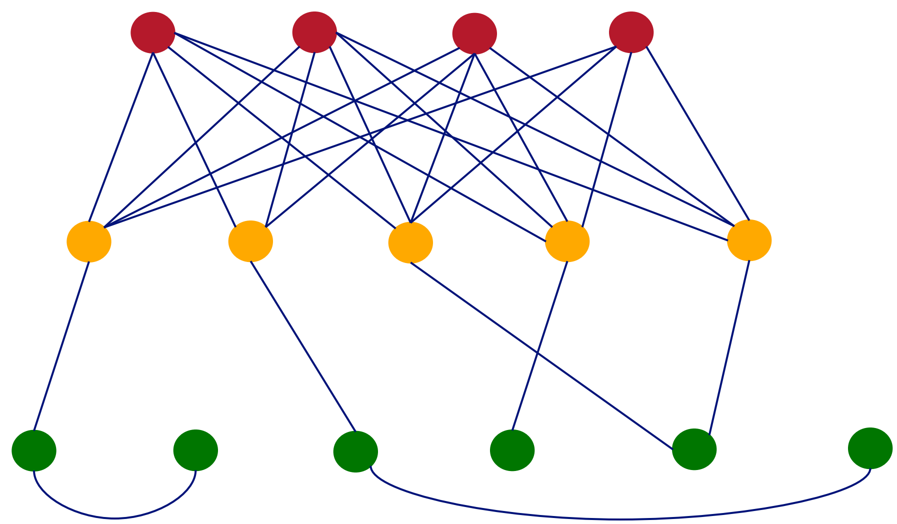
Carnegie Mellon

E-bay Fraud detection



E-bay Fraud detection - NetProbe





Popular press

USA TODAY

The Washington Post

Los Angeles Times

And less desirable attention:

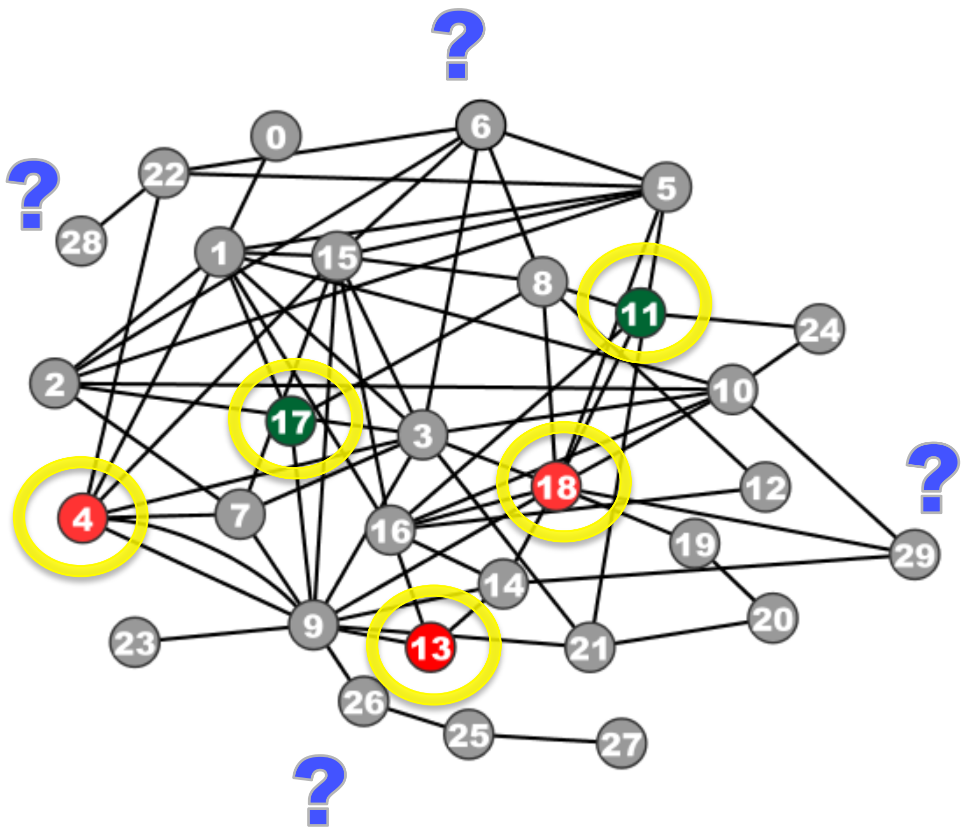
- E-mail from ‘Belgium police’ (‘copy of your code?’)

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - Immunization
  - BP - theory
  - visualization
- Conclusions

Carnegie Mellon

Guilt-by-Association Techniques



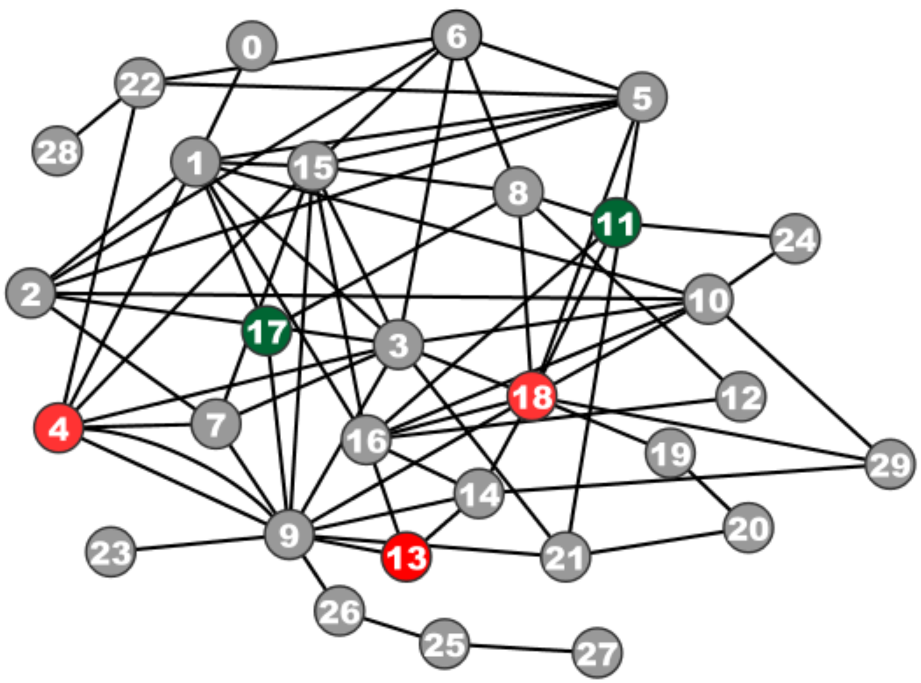
Given:
- graph and
- few labeled nodes

Find: class (red/green) for rest nodes

Assuming: network effects (homophily/ heterophily)

Correspondence of Methods

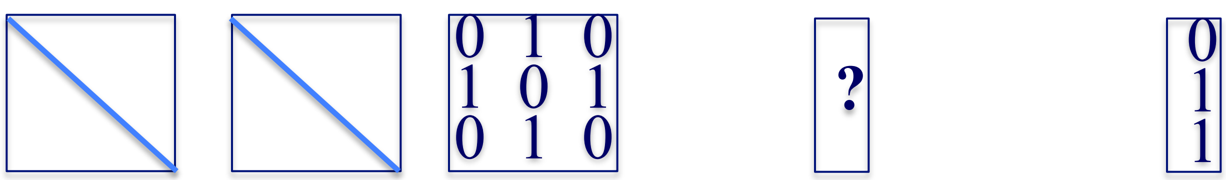
Random Walk with Restarts (**RWR**) Google
Semi-supervised Learning (**SSL**)
Belief Propagation (**BP**) Bayesian



Correspondence of Methods

Random Walk with Restarts (**RWR**) ≈
Semi-supervised Learning (**SSL**) ≈
Belief Propagation (**BP**)

Method	Matrix		unknown		known
RWR	$[\mathbf{I} - c\ \mathbf{A}\mathbf{D}^{-1}]$	$\times$	$\mathbf{x}$	$=$	$(1-c)\mathbf{y}$
SSL	$[\mathbf{I} + \alpha(\mathbf{D} - \mathbf{A})]$	$\times$	$\mathbf{x}$	$=$	$\mathbf{y}$
FABP	$[\mathbf{I} + a\,\mathbf{D} - c'\mathbf{A}]$	$\times$	$\mathbf{b_h}$	$=$	$\boldsymbol{\phi}_\mathbf{h}$



Unifying Guilt-by-Association Approaches: Theorems and Fast Algorithms. Danai Koutra, et al *PKDD'11*

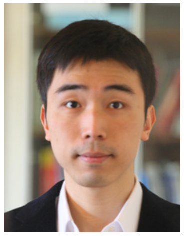
Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - Immunization
  - BP
  - visualization
- Conclusions

Apolo

Making Sense of Large Network Data: Combining Rich User Interaction & Machine Learning

CHI 2011, Vancouver, Canada

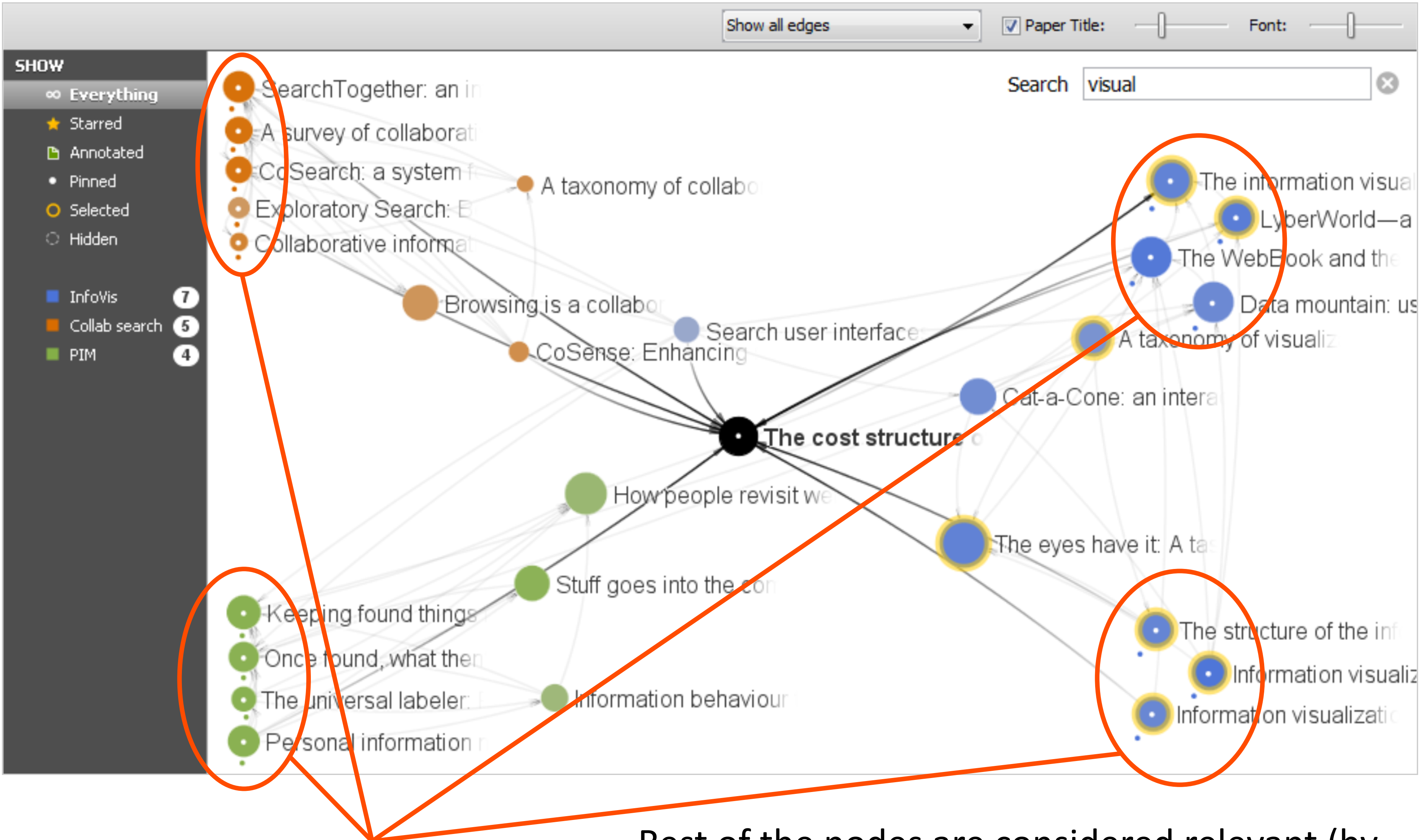


Polo Chau

Prof. Niki Kittur

Prof. Jason Hong

Prof. Christos Faloutsos



Exemplars

Rest of the nodes are considered relevant (by BP); relevance indicated by color saturation.

Note that BP supports multiple groups

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - Immunization
  - BP
  - visualization
- Conclusions

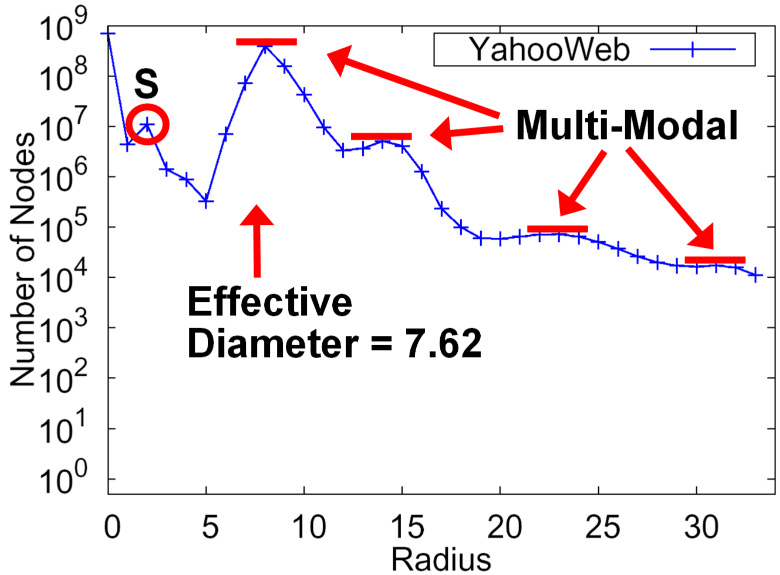
OVERALL CONCLUSIONS – low level:

- Several new **patterns** (eigenspokes, radius plot etc)
- New **tools** and theoretical results
  - belief propagation (~ RWR ~ SSL)
  - Immunization: **$\Delta\lambda$,** for ‘any’ V.P.M.

Carnegie Mellon

OVERALL CONCLUSIONS – high level

- **BIG DATA: ->** patterns/outliers that are invisible otherwise



References

- Leman Akoglu, Christos Faloutsos: *RTG: A Recursive Realistic Graph Generator Using Random Typing*. ECML/PKDD (1) 2009: 13-28

- Deepayan Chakrabarti, Christos Faloutsos: *Graph mining: Laws, generators, and algorithms*. ACM Comput. Surv. 38(1): (2006)

Carnegie Mellon

References

- Deepayan Chakrabarti, Yang Wang, Chenxi Wang, Jure Leskovec, Christos Faloutsos: *Epidemic thresholds in real networks.* ACM Trans. Inf. Syst. Secur. 10(4): (2008)

- Deepayan Chakrabarti, Jure Leskovec, Christos Faloutsos, Samuel Madden, Carlos Guestrin, Michalis Faloutsos: *Information Survival Threshold in Sensor and P2P Networks*. INFOCOM 2007: 1316-1324

References

- Christos Faloutsos, Tamara G. Kolda, Jimeng Sun: *Mining large graphs and streams using matrix and tensor tools*. Tutorial, SIGMOD Conference 2007: 1174

References

- T. G. Kolda and J. Sun. *Scalable Tensor Decompositions for Multi-aspect Data Mining*. In: ICDM 2008, pp. 363-372, December 2008.

References

- Jure Leskovec, Jon Kleinberg and Christos Faloutsos *Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations*, KDD 2005 (Best Research paper award).
- Jure Leskovec, Deepayan Chakrabarti, Jon M. Kleinberg, Christos Faloutsos: *Realistic, Mathematically Tractable Graph Generation and Evolution, Using Kronecker Multiplication*. PKDD 2005: 133-145

References

- Jimeng Sun, Yinglian Xie, Hui Zhang, Christos Faloutsos. *Less is More: Compact Matrix Decomposition for Large Sparse Graphs*, SDM, Minneapolis, Minnesota, Apr 2007.
- Jimeng Sun, Spiros Papadimitriou, Philip S. Yu, and Christos Faloutsos, *GraphScope: Parameter-free Mining of Large Time-evolving Graphs* ACM SIGKDD Conference, San Jose, CA, August 2007

Carnegie Mellon

References

- Jimeng Sun, Dacheng Tao, Christos Faloutsos: *Beyond streams and graphs: dynamic tensor analysis*. KDD 2006: 374-383

References

- Hanghang Tong, Christos Faloutsos, and Jia-Yu Pan, *Fast Random Walk with Restart and Its Applications*, ICDM 2006, Hong Kong.
- Hanghang Tong, Christos Faloutsos, *Center-Piece Subgraphs: Problem Definition and Fast Solutions*, KDD 2006, Philadelphia, PA

References

- Hanghang Tong, Christos Faloutsos, Brian Gallagher, Tina Eliassi-Rad: Fast best-effort pattern matching in large attributed graphs. KDD 2007: 737-746

Carnegie Mellon

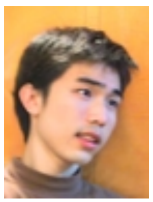
Project info

www.cs.cmu.edu/~pegasus

★ 

Akoglu, Leman

★ Chau, Polo

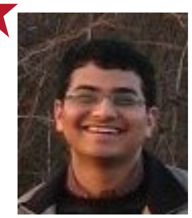


Kang, U

Koutra, Danae

McGlohon, Mary

Prakash, Aditya

★ 

Tong, Hanghang

★ Out, in '12

Thanks to: NSF IIS-1017415, 0705359, 0534205, CTA-INARC; Yahoo (M45), LLNL, IBM, SPRINT, Google, INTEL, HP, iLab

OVERALL CONCLUSIONS – high level

- **BIG DATA: ->** patterns/outliers that are invisible otherwise