Large Graph Mining: Patterns, Cascades, Fraud Detection, and Algorithms

Christos Faloutsos

CMU

Thank you!

- Prof. Chin-Wan Chung

Thank you!

- Prof. Chin-Wan Chung

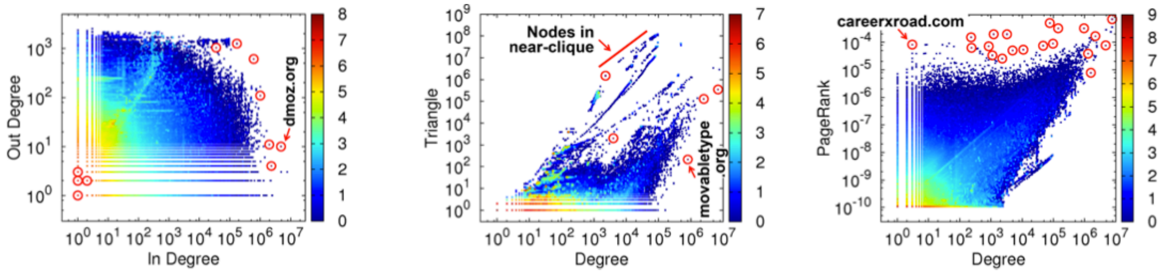
Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Conclusions

Graphs - why should we care?

~1B nodes (web sites)
~6B edges (http links)
'YahooWeb graph'

Graphs - why should we care?



YahooWeb:
(a) In-degree vs. Out-degree (b) Degree vs. Triangles (c) Degree vs. PageRank

~1B nodes (web sites)
~6B edges (http links)
'YahooWeb graph'

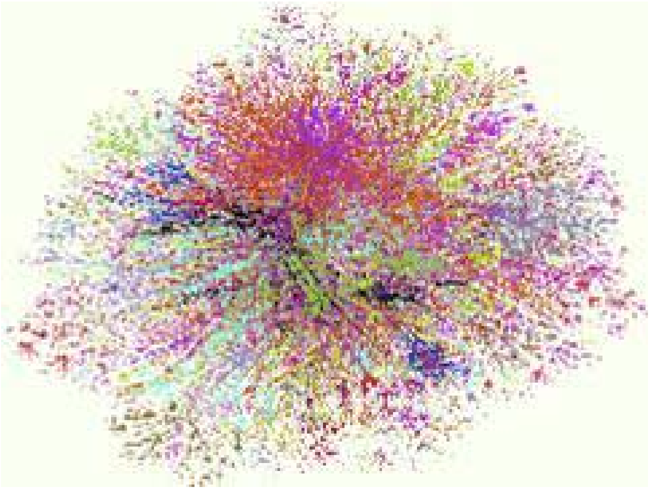


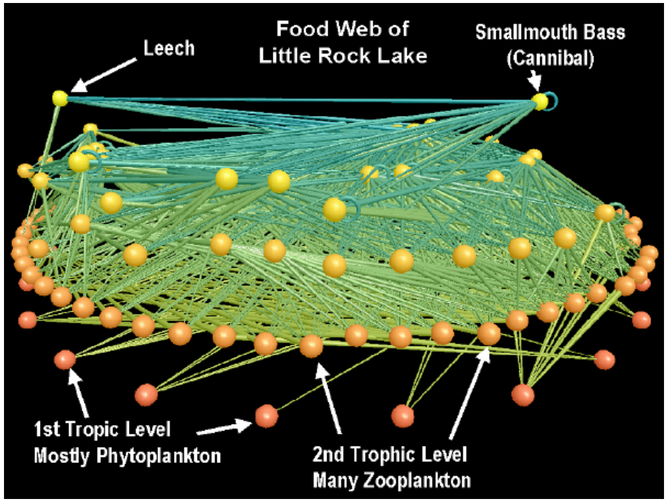
U Kang, Jay-Yoon Lee, DanaiKoutra, and Christos Faloutsos. *Net-Ray: Visualizing and Mining Billion-Scale Graphs* PAKDD 2014, Tainan, Taiwan.

Graphs - why should we care?

>$10B; ~1B users

Graphs - why should we care?



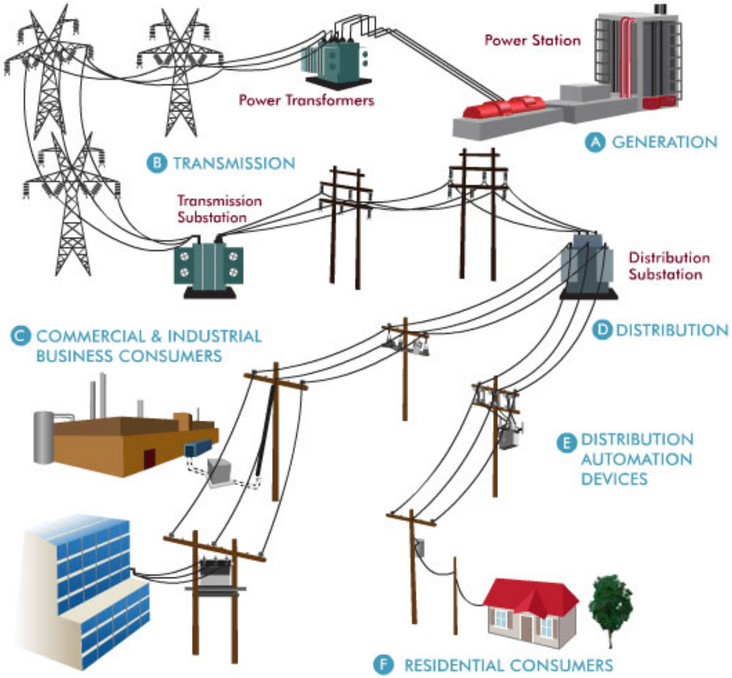


Internet Map
[lumeta.com]

Food Web
[Martinez '91]

Graphs - why should we care?

- Power-grid!
  - Nodes: (plants/consumers)
  - Edges: power lines



Graphs - why should we care?

- web-log ('blog') news propagation
- computer network security: email/IP traffic and anomaly detection
- Recommendation systems
- ....

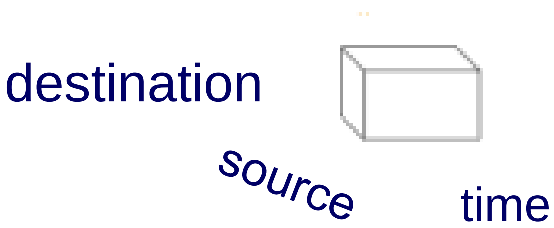
- Many-to-many db relationship -> graph

Carnegie Mellon

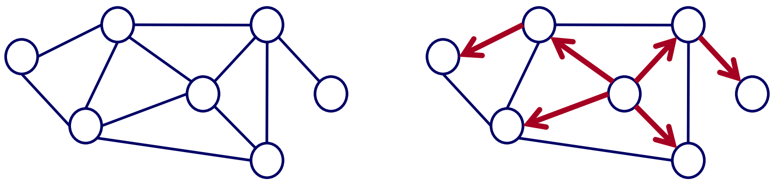
Motivating problems

- P1: patterns? Fraud detection?

- P2: patterns in time-evolving graphs / tensors



- P3: cascades – whom to immunize?



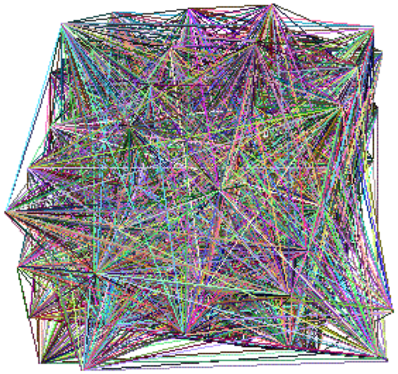
Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Conclusions

Carnegie Mellon

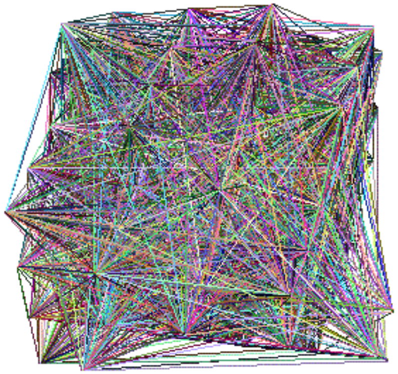
Part 1: Patterns, & fraud detection

Laws and patterns

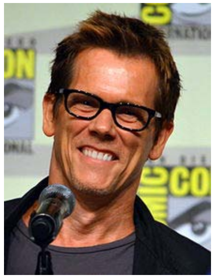


- Q1: Are real graphs random?

Laws and patterns



- Q1: Are real graphs random?
- A1: NO!!
  - Diameter (‘6 degrees’; ‘Kevin Bacon’)
  - in- and out- degree distributions
  - other (surprising) patterns
- So, let’s look at the data



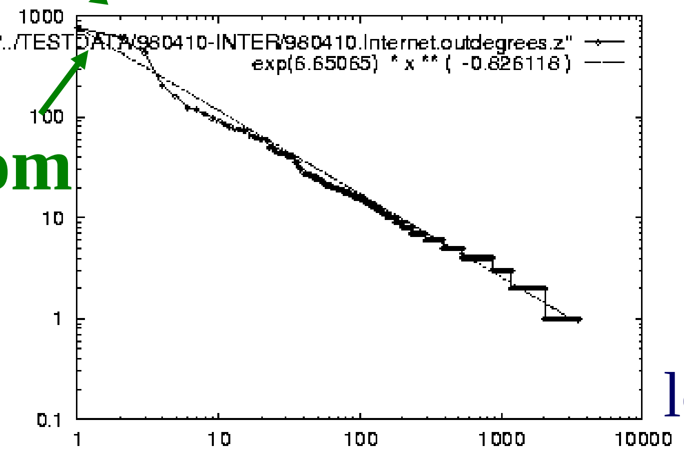
Solution# S.1

- Power law in the degree distribution [SIGCOMM99]

internet domains

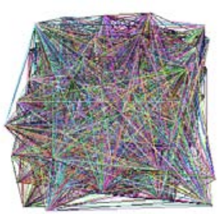
att.com

log(degree)



ibm.com

log(rank)



Solution# S.1

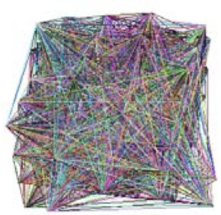
- Power law in the degree distribution [SIGCOMM99]

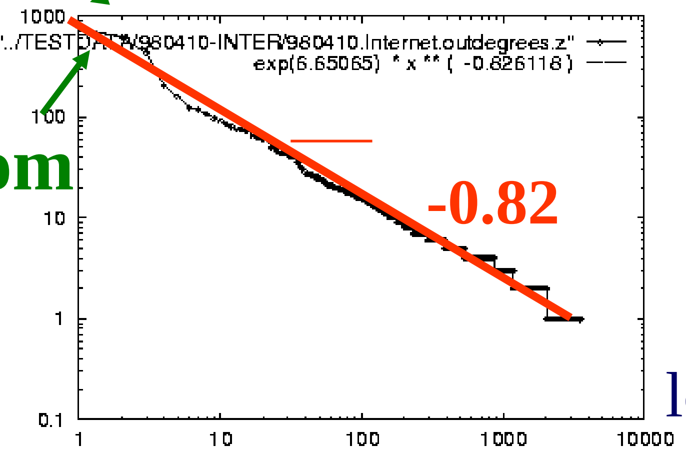
internet domains

att.com

log(degree)

ibm.com



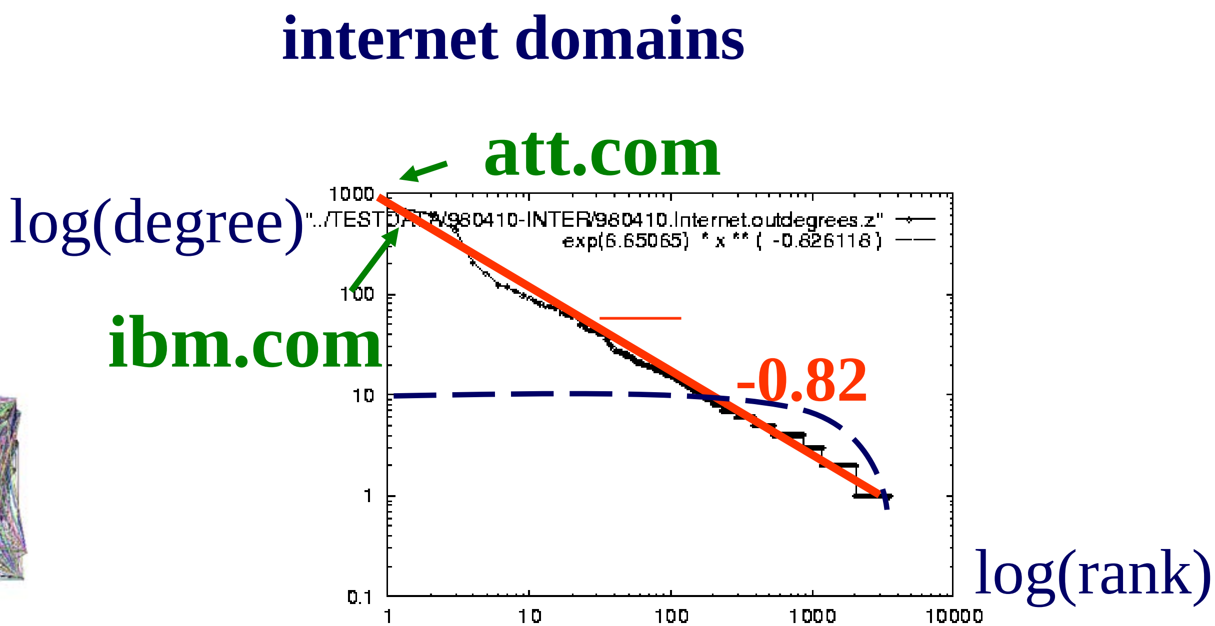


-0.82

log(rank)

Solution# S.1

- Q: So what?

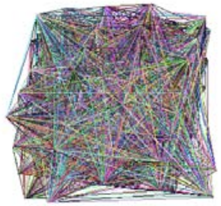


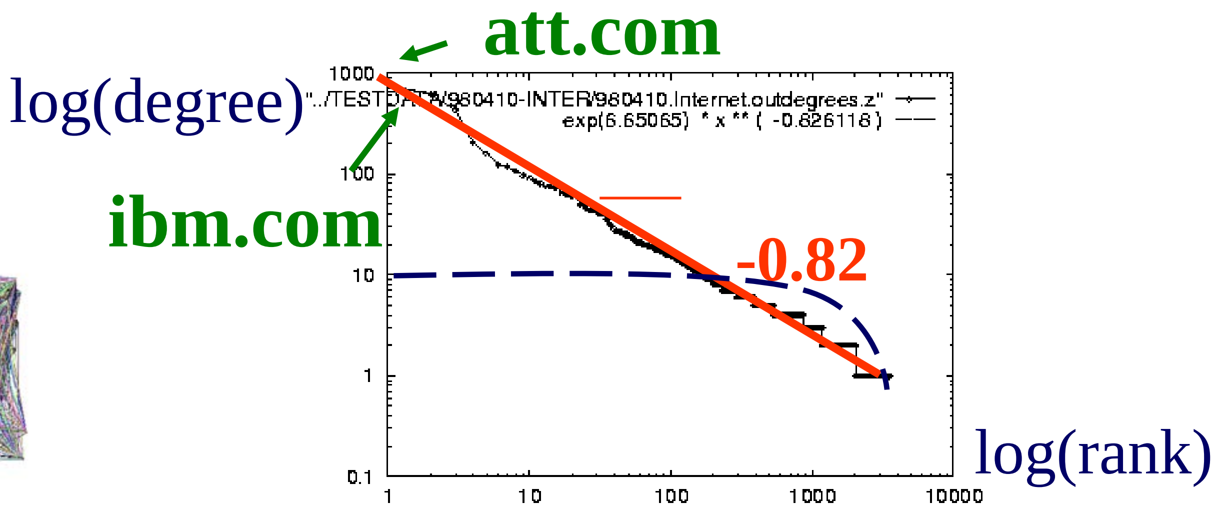
Solution# S.1

- Q: So what?
- A1: # of two-step-away pairs:

= friends of friends (F.O.F.)

internet domains



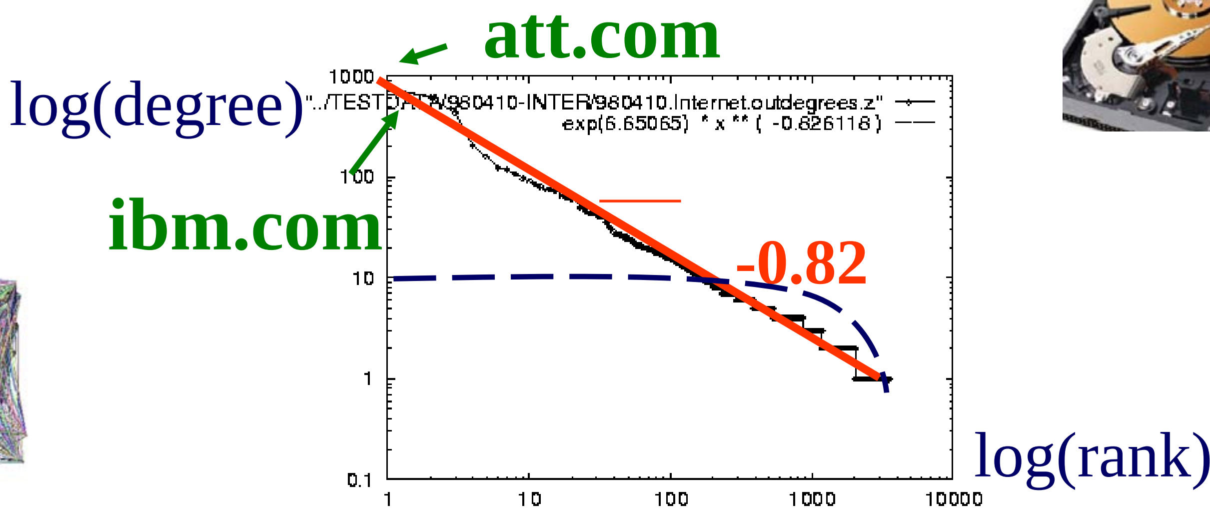


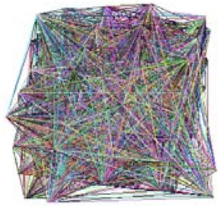
Solution# S.1

- Q: So what?
- A1: # of two-step-away pairs: 100^2 * N= 10 Trillion

= friends of friends (F.O.F.)

internet domains



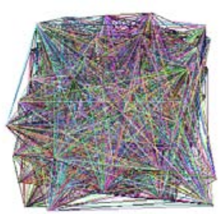


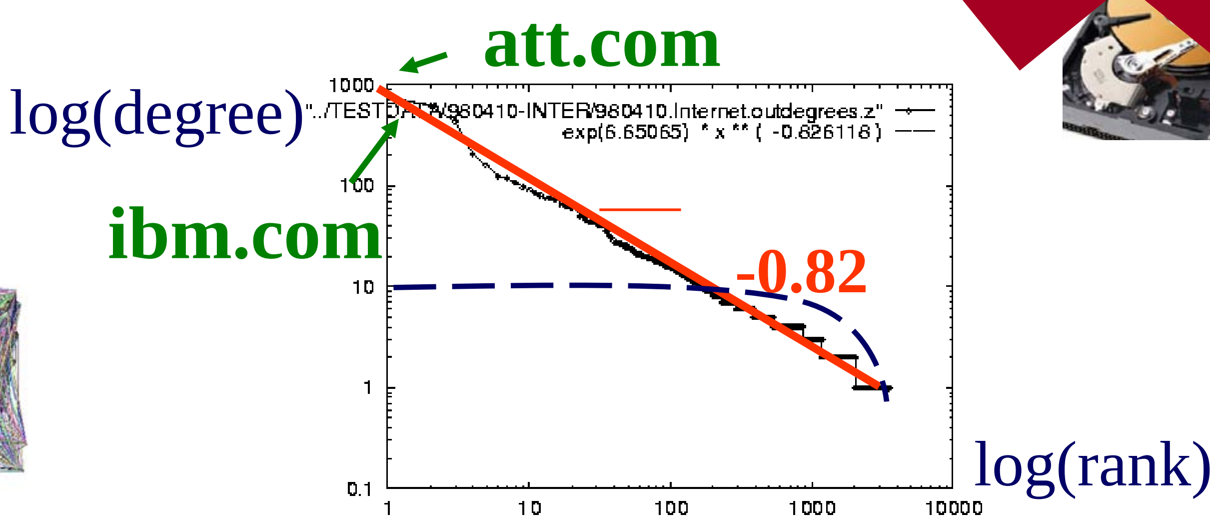
Solution# S.1

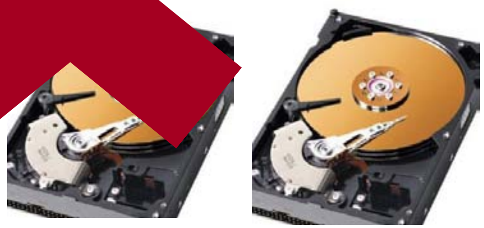
- Q: So what?
- A1: # of two-step-away pairs: 100^ ... N ... Trillion

= friends of friends (F.O.F.)

internet domains







Gaussian trap

Solution# S.1

= friends of friends (F.O.F.)

- Q: So what?
- A1: # of two-step-away pairs: O(d_max ^2) ~ 10M^2

internet domains

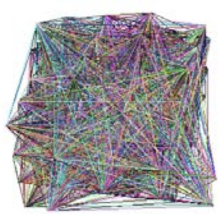
⇩

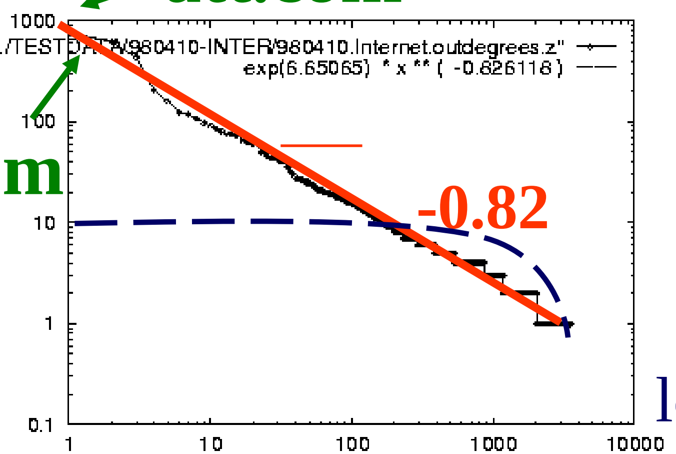
~0.8PB ->
a data center(!)

att.com

log(degree)

ibm.com





DCO @ CMU

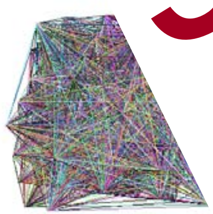
Gaussian trap

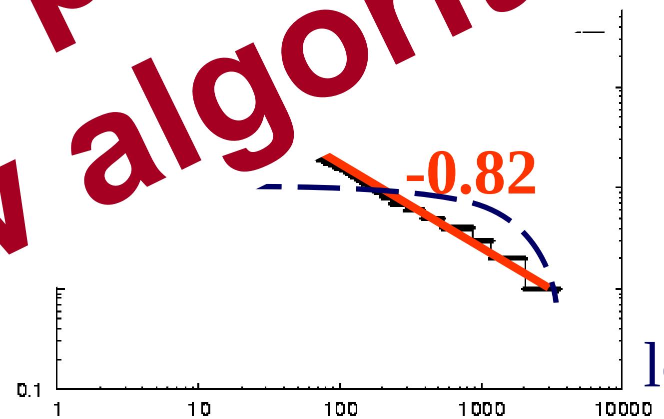
Solution# S.1

- Q: So what?
- A1: # of two-step-aw… ?) ~ 10M^2

inte…

~0.8PB ->
a data center(!)

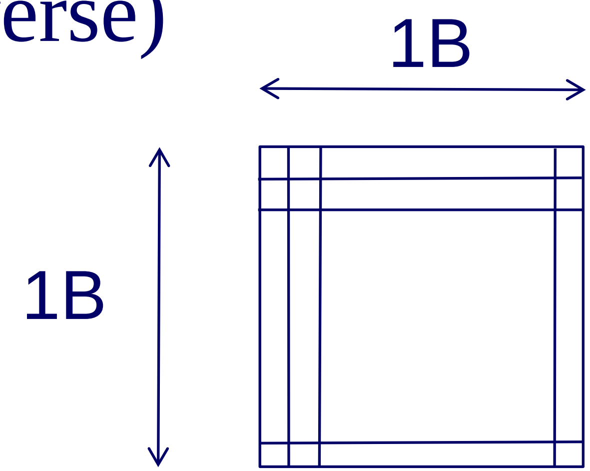




Such patterns -> New algorithms

Observation – big-data:

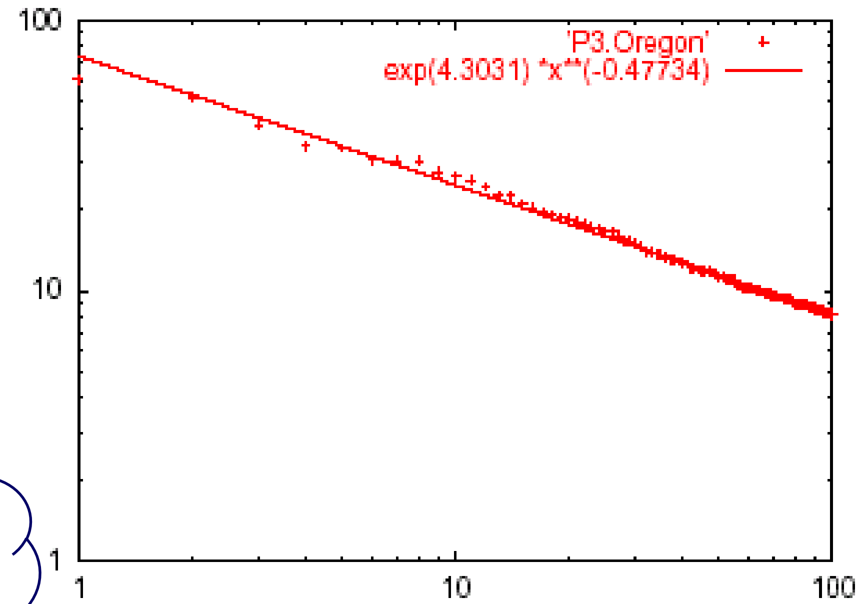
- O(N^2) algorithms are ~intractable - N=1B
- N^2 seconds = 31B years (>2x age of universe)



Carnegie Mellon

Solution# S.2: Eigen Exponent *E*

Eigenvalue



Rank of decreasing eigenvalue

Exponent = slope

E = -0.48

May 2001

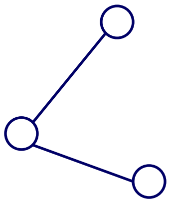
A x = λx

- A2: power law in the eigenvalues of the adjacency matrix (‘eig()’)

Roadmap

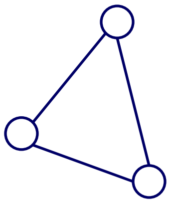
- Introduction – Motivation
- Part#1: Patterns in graphs
  - Patterns: Degree; Triangles
  - Anomaly/fraud detection
  - Graph understanding
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Conclusions

Solution# S.3: Triangle ‘Laws’



- Real social networks have a lot of triangles

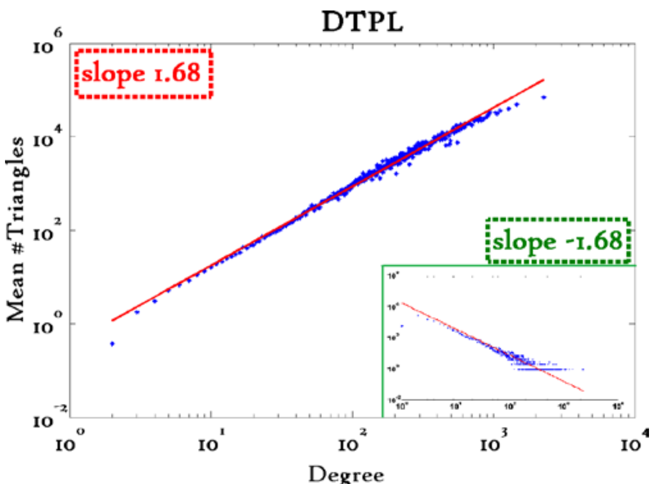
Solution# S.3: Triangle 'Laws'



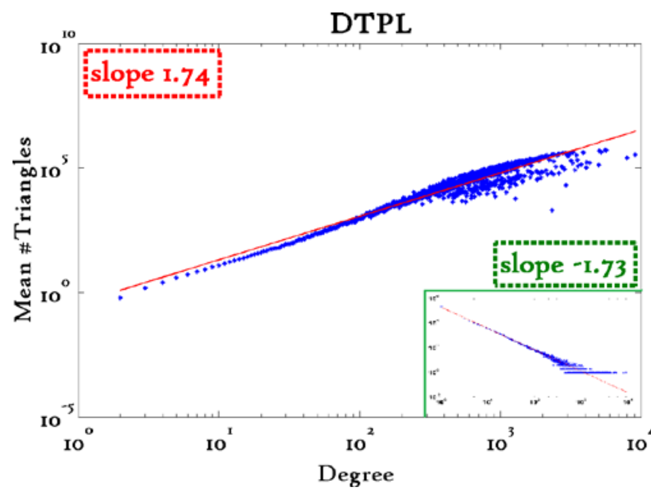
- Real social networks have a lot of triangles
  - Friends of friends are friends
- Any patterns?
  - 2x the friends, 2x the triangles ?

Triangle Law: #S.3
[Tsourakakis ICDM 2008]

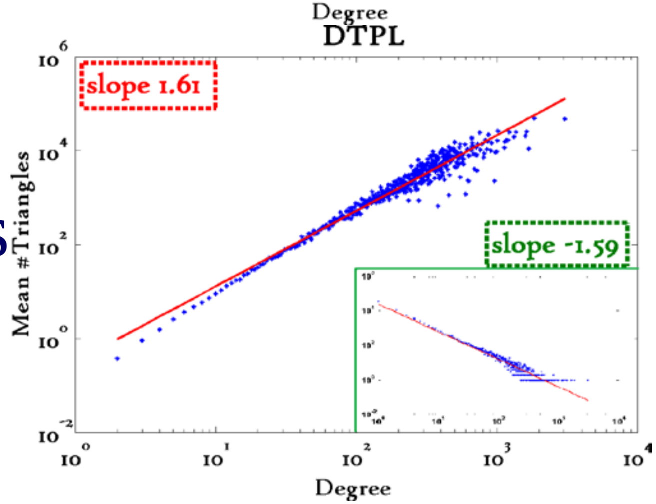
Reuters

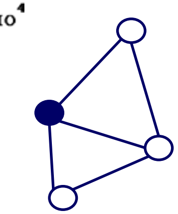


SN



Epinions





X-axis: degree

Y-axis: mean # triangles

n friends -> ~$n^{1.6}$ triangles

details

Triangle Law: Computations
[Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos) – $O(d_{max}^2)$
Q: Can we do that quickly?
A:

details

Triangle Law: Computations

[Tsourakakis ICDM 2008]

But: triangles are expensive to compute

(3-way join; several approx. algos) – $O(d_{max}^2)$

Q: Can we do that quickly?

A: Yes!

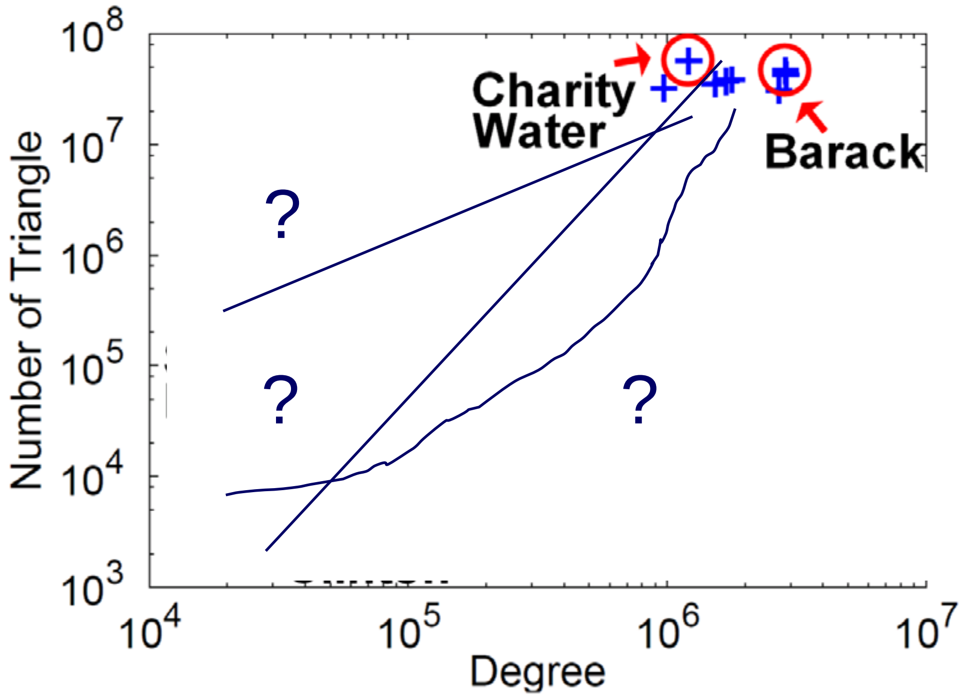
$A\,x = \lambda x$

#triangles = 1/6 Sum (λ_i^3)

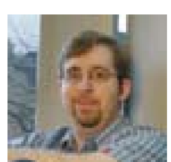
(and, because of skewness (S2) ,

we only need the top few eigenvalues! - O(E)

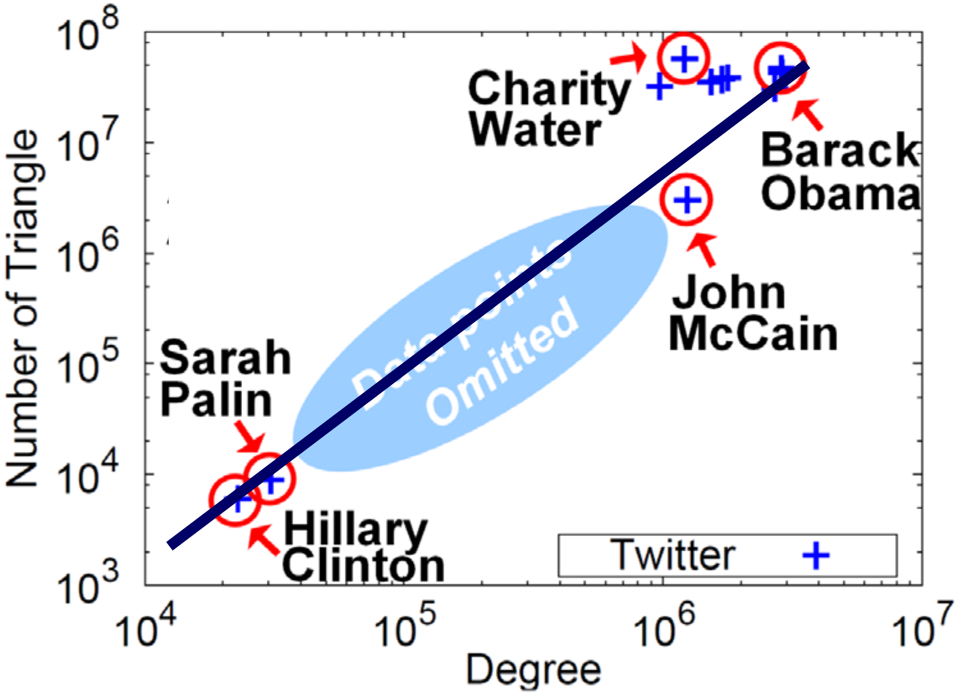
Triangle counting for large graphs?



Anomalous nodes in Twitter(~ 3 billion edges)
[U Kang, Brendan Meeder, +, PAKDD'11]



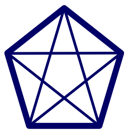
Triangle counting for large graphs?

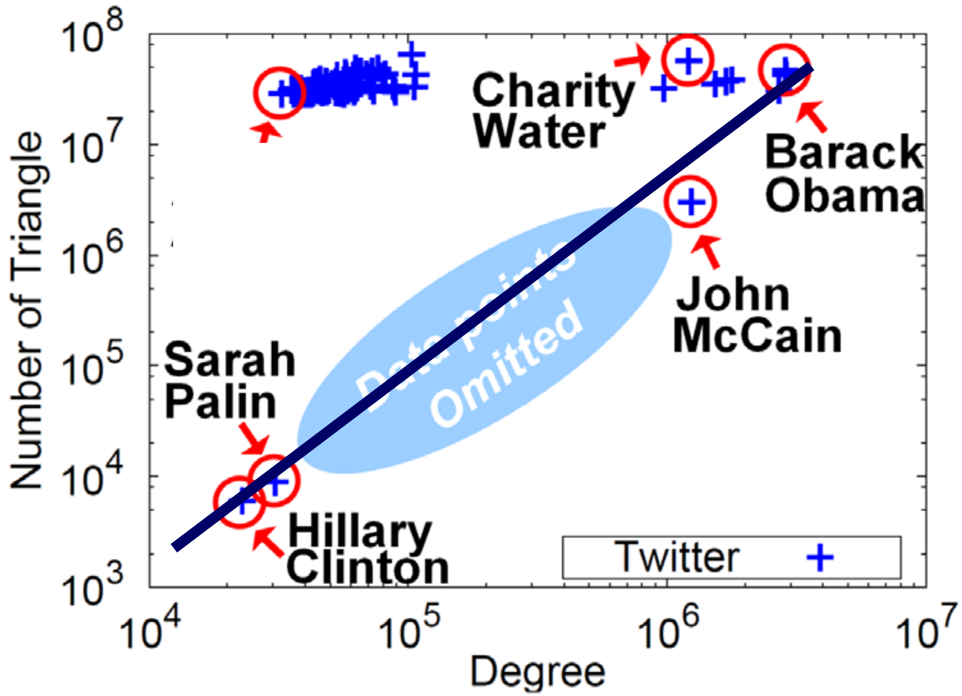


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD’11]

Triangle counting for large graphs?

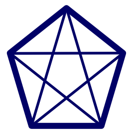


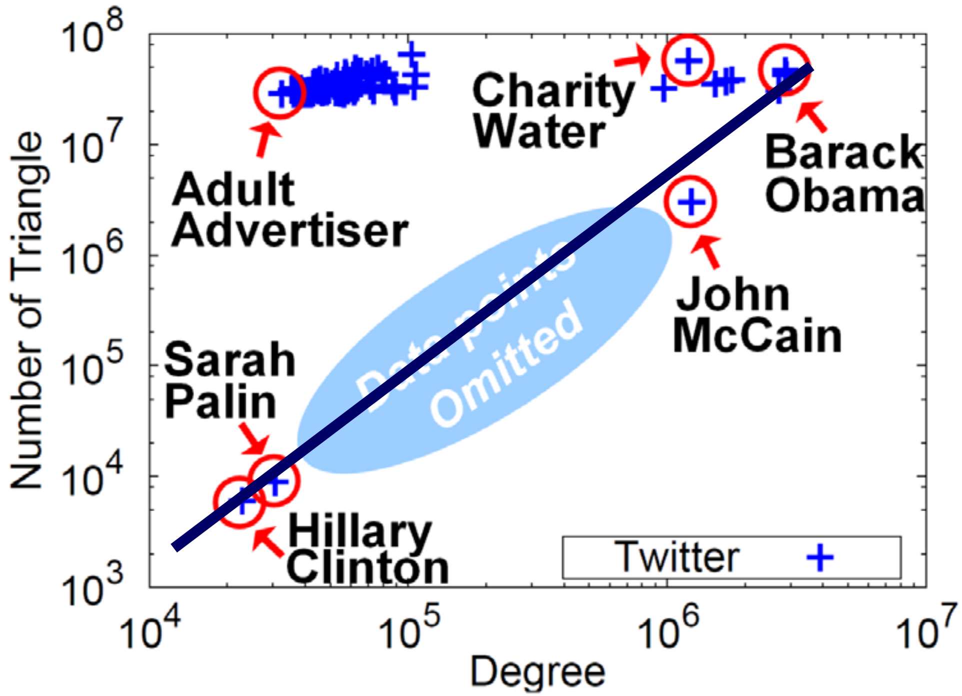


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD’11]

Triangle counting for large graphs?





Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD’11]

MORE Graph Patterns

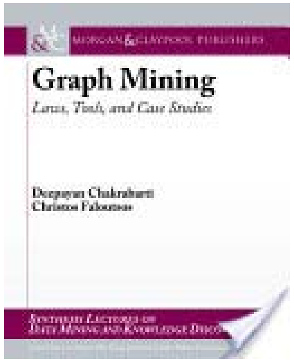
	Unweighted	Weighted
Static	**L01.** Power-law degree distribution [Faloutsos et al. `99, Kleinberg et al. `99, Chakrabarti et al. `04, Newman `04] **L02.** Triangle Power Law (TPL) [Tsourakakis `08] **L03.** Eigenvalue Power Law (EPL) [Siganos et al. `03] **L04.** Community structure [Flake et al. `02, Girvan and Newman `02]	**L10.** Snapshot Power Law (SPL) [McGlohon et al. `08]
Dynamic	**L05.** Densification Power Law (DPL) [Leskovec et al. `05] **L06.** Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05] **L07.** Constant size $2^{nd}$ and $3^{rd}$ connected components [McGlohon et al. `08] **L08.** Principal Eigenvalue Power Law (λ_1PL) [Akoglu et al. `08] **L09.** Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et al. `98, Crovella and	**L11.** Weight Power Law (WPL) [McGlohon et al. `08]

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

MORE Graph Patterns

	Unweighted	Weighted
Static	L01. Power-law degree distribution [Faloutsos et al. `99, Kleinberg et al. `99, Chakrabarti et al. `04, Newman `04] L02. Triangle Power Law (TPL) [Tsourakakis `08] L03. Eigenvalue Power Law (EPL) [Siganos et al. `03] L04. Community structure [Flake et al. `02, Girvan and Newman `02]	L10. Snapshot Power Law (SPL) [McGlohon et al. `08]
Dynamic	L05. Densification Power Law (DPL) [Leskovec et al. `05] L06. Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05] L07. Constant size 2nd and 3rd connected components [McGlohon et al. `08] L08. Principal Eigenvalue Power Law (λ_1PL) [Akoglu et al. `08] L09. Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et al. `98, Crovella and Bestavros `99, McGlohon et al. `08]	L11. Weight Power Law (WPL) [McGlohon et al. `08]


- Mary McGlohon, Leman Akoglu, Christos Faloutsos. *Statistical Properties of Social Networks.* in "Social Network Data Analytics" (Ed.: CharuAggarwal)

- Deepayan Chakrabarti and Christos Faloutsos, *Graph Mining: Laws, Tools, and Case Studies* Oct. 2012, Morgan Claypool.

Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
  - Patterns
  - Anomaly / fraud detection
  - Graph understanding
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Conclusions

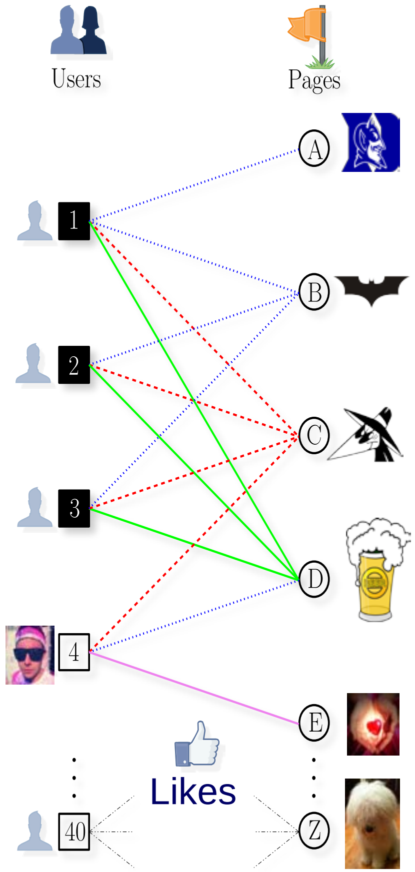
Fraud

- Given
  - Who ‘likes’ what page, and when
- Find
  - Suspicious users and suspicious products

CopyCatch: Stopping Group Attacks by Spotting Lockstep Behavior in Social Networks, Alex Beutel, WanhongXu, VenkatesanGuruswami, Christopher Palow, Christos Faloutsos *WWW, 2013.*

Fraud

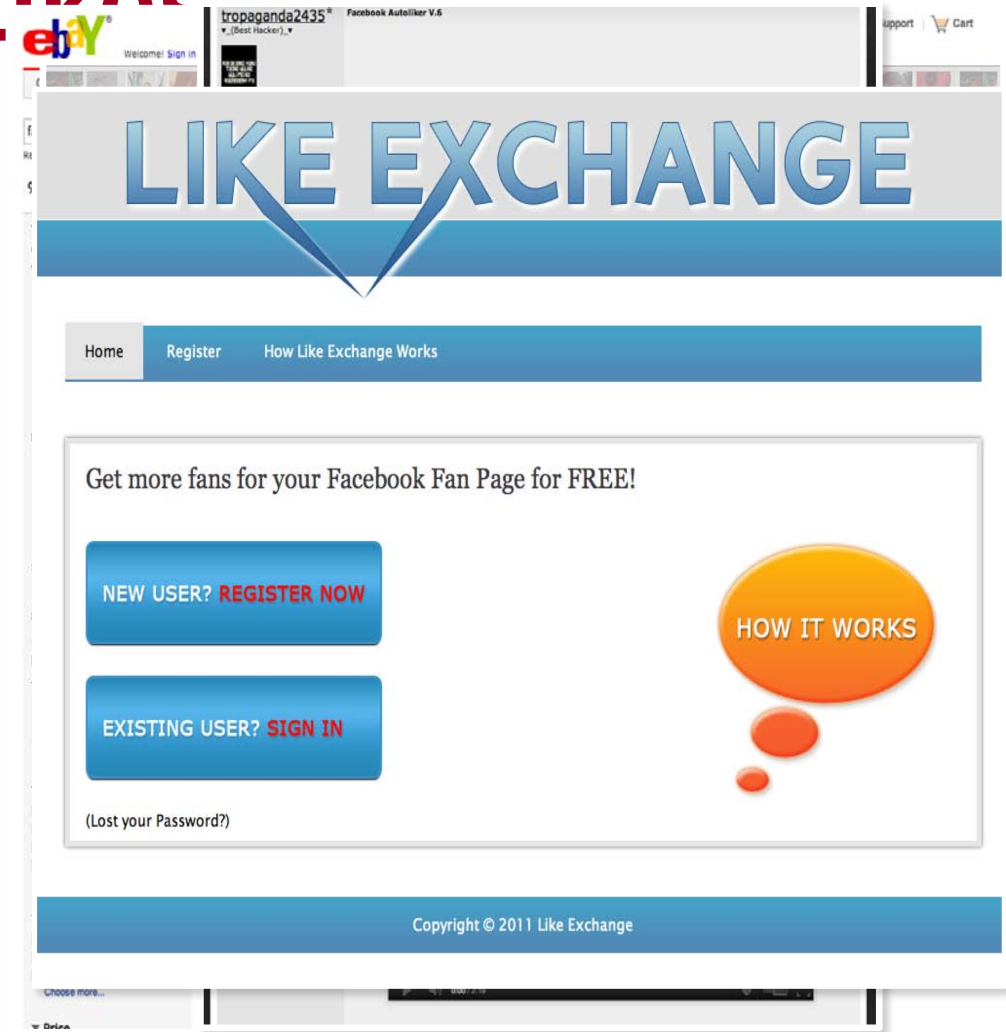
- Given
  - Who ‘likes’ what page, and when
- Find
  - Suspicious users and suspicious products



CopyCatch: Stopping Group Attacks by Spotting Lockstep Behavior in Social Networks, Alex Beutel, WanhongXu, VenkatesanGuruswami, Christopher Palow, Christos Faloutsos *WWW, 2013.*

Ill-gotten Facebook Pages Likes

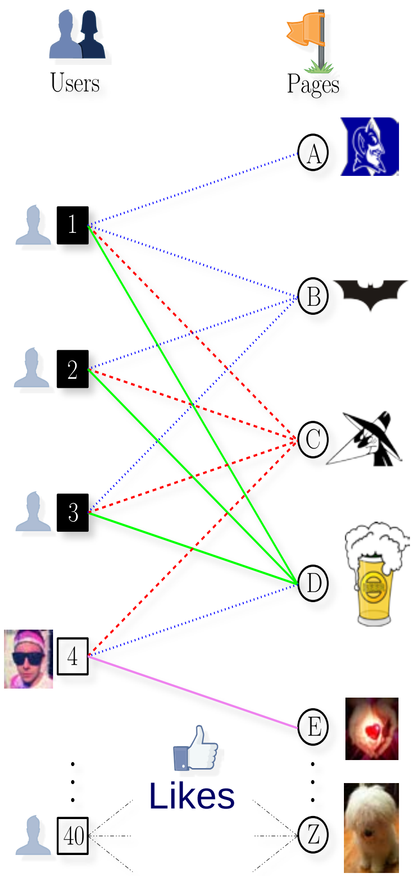
- Popular Page = $
- Fake 'likes' through unethical means:
  - Fake accounts
  - Malware
  - Credential stealing
  - Social Engineering

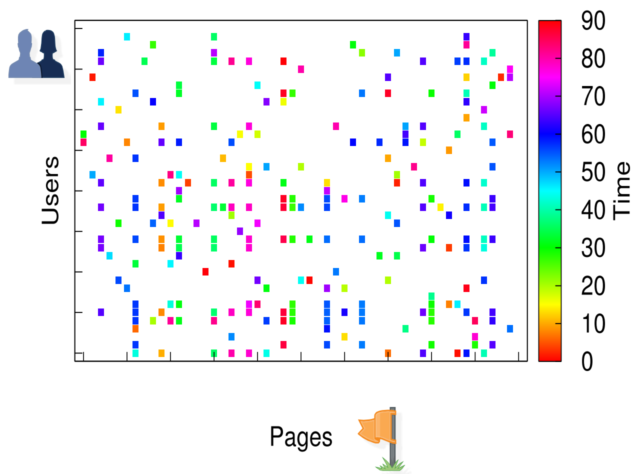


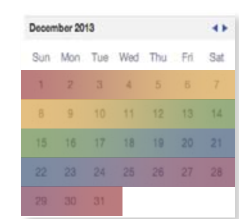
Graph Patterns and Lockstep Behavior

Our intuition

- Lockstep behavior: Same Likes, same time



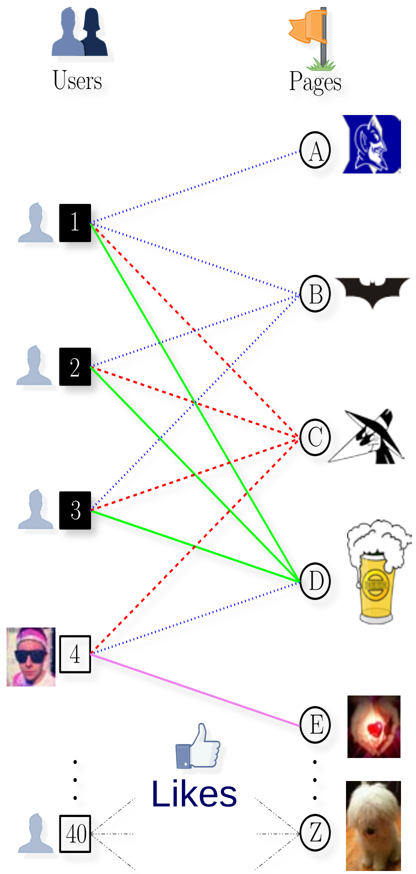


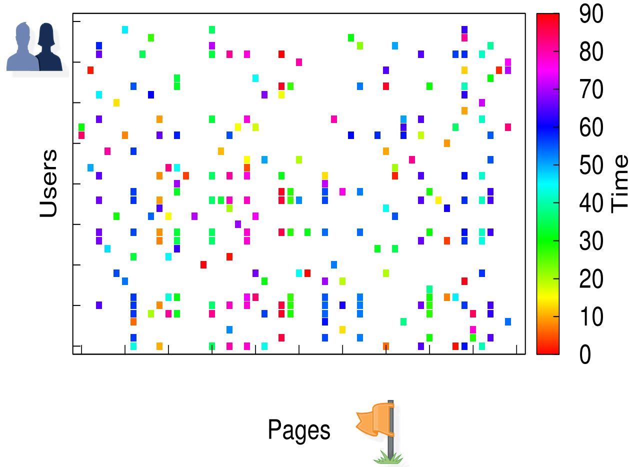


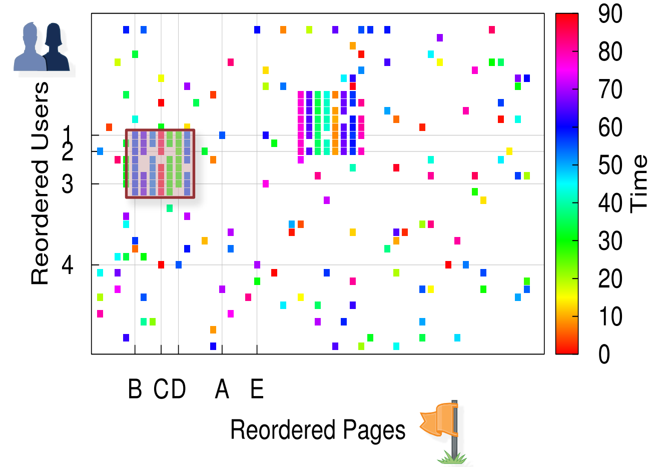
Graph Patterns and Lockstep Behavior

Our intuition

- Lockstep behavior: Same Likes, same time



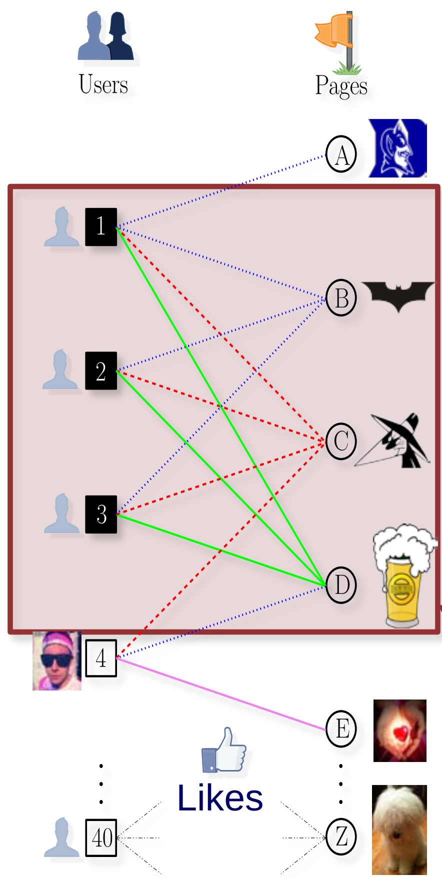


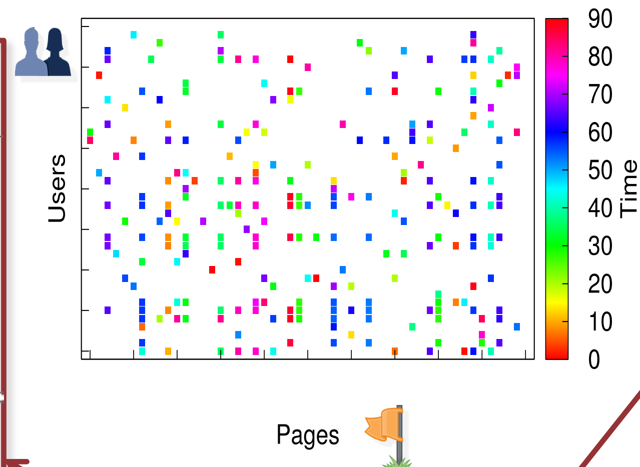


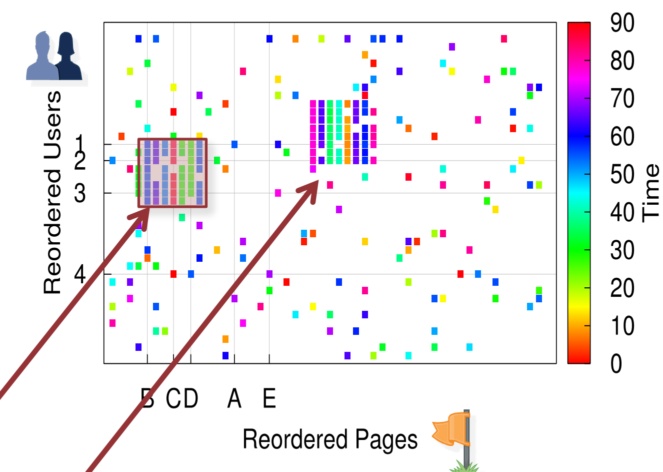
Graph Patterns and Lockstep Behavior

Our intuition

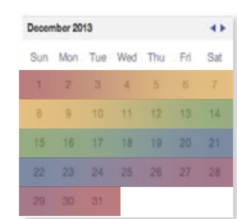
- Lockstep behavior: Same Likes, same time





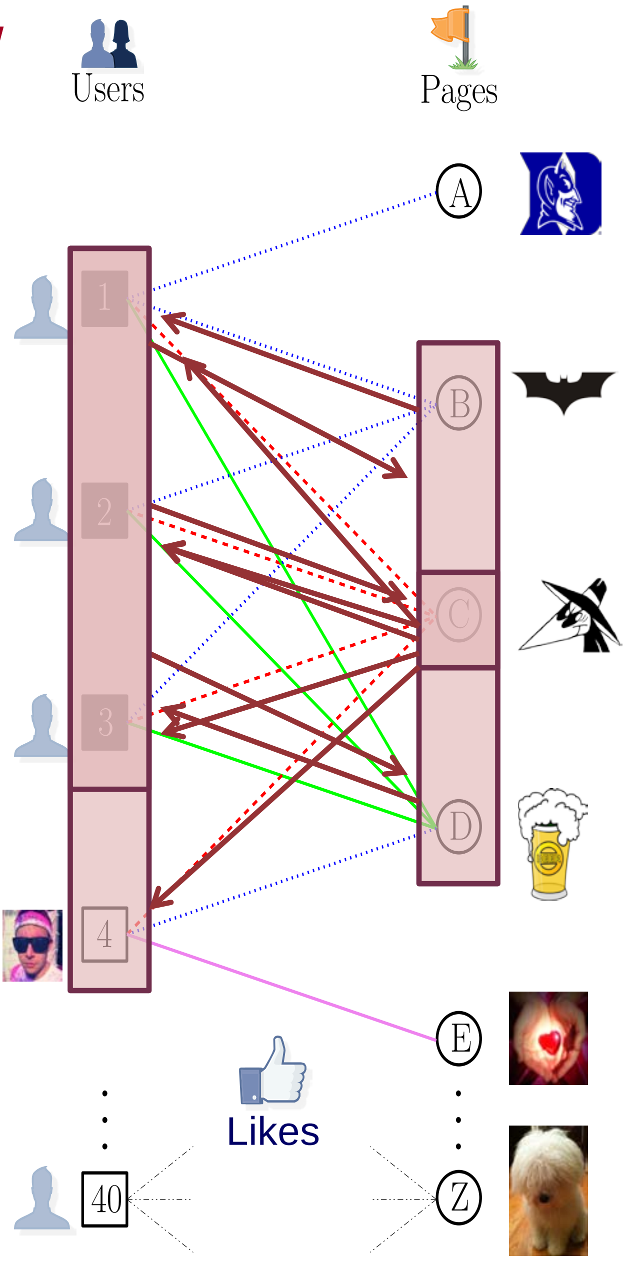


Suspicious Lockstep Behavior



MapReduce Overview

- Use Hadoop to search for many clusters in parallel:
  1. Start with randomly seed
  2. Update set of Pages and center Like times for each cluster
  3. Repeatuntil convergence

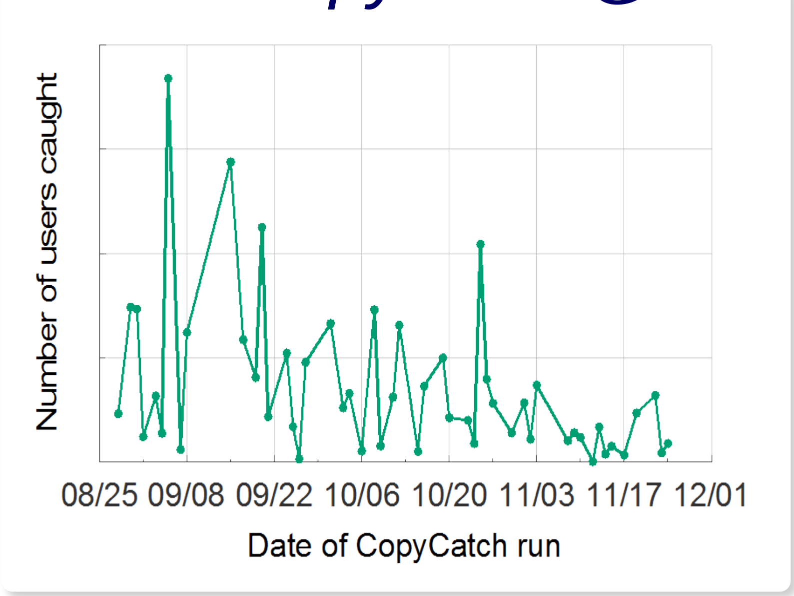


Deployment at Facebook

- *CopyCatch*runs regularly (along with many other security mechanisms, and a large Site Integrity team)

3 months of *CopyCatch*@ Facebook

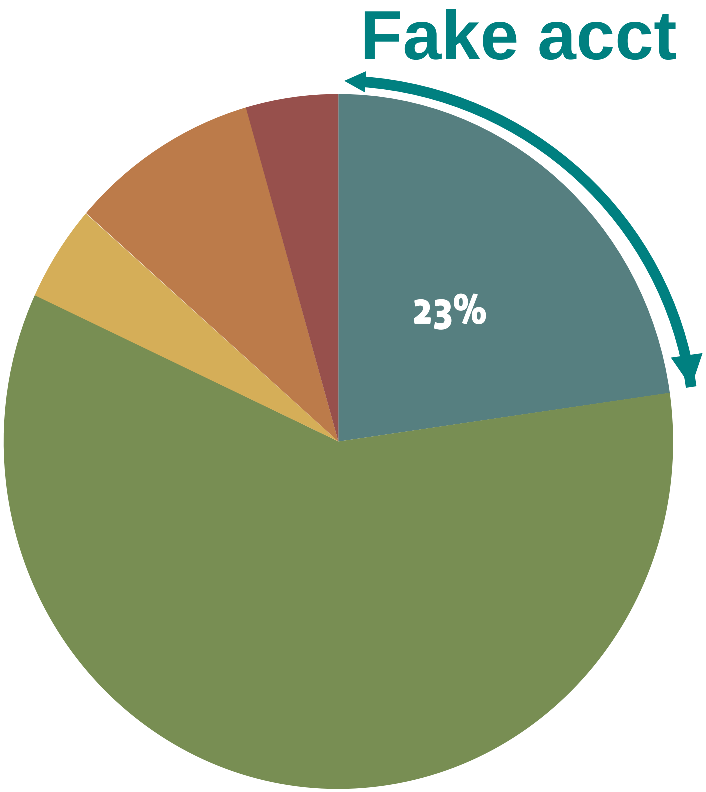
#users caught



time

Carnegie Mellon

Deployment at Facebook



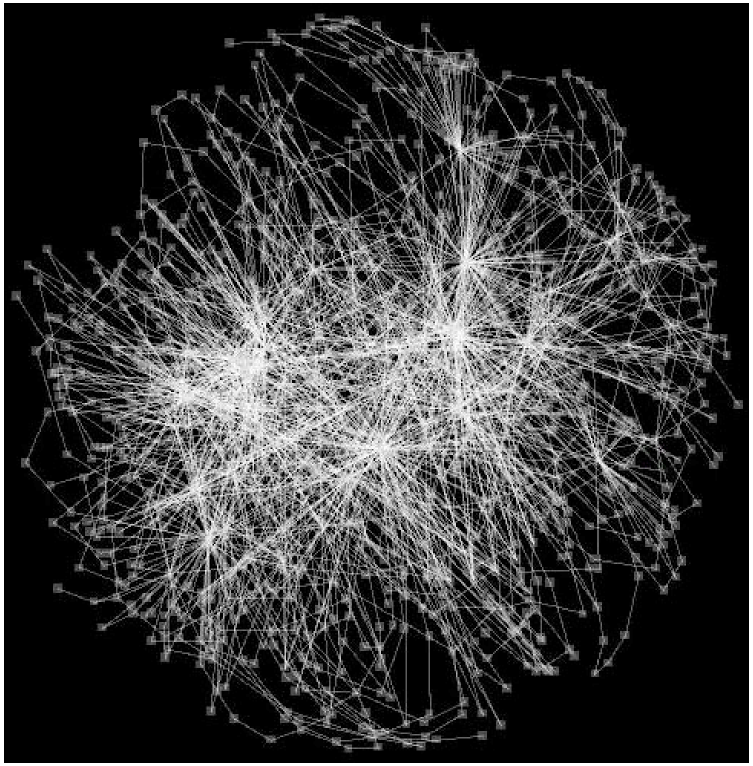
Most clusters (77%) come from **real** but **compromised** users

Manually labeled 22 randomly selected *clusters* from February 2013

Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
  - Patterns
  - Anomaly / fraud detection
  - Graph understanding
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Conclusions

Wikipedia - editors

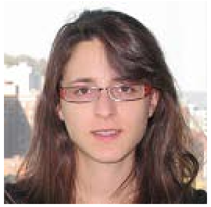


- Nodes: editors
- Edge A->B: 'A' changed 'B'

Any pattern?

VoG:Summarizing and Understanding Large Graphs

DanaiKoutra,



U Kang,

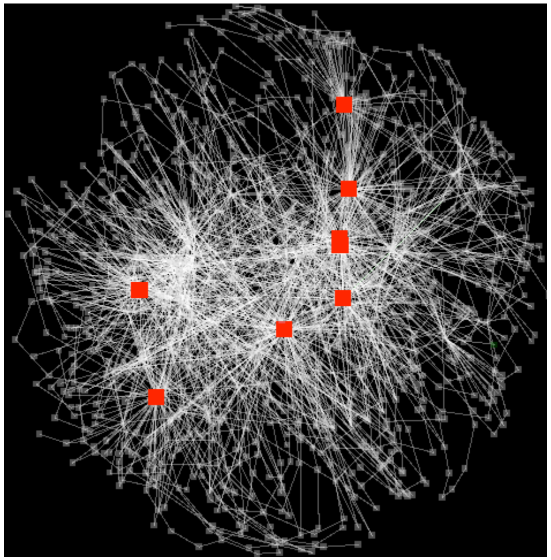
JillesVreeken,

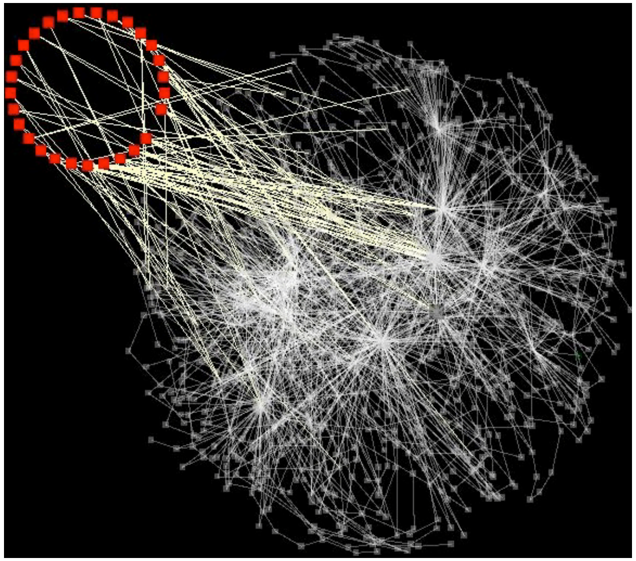
Christos Faloutsos.

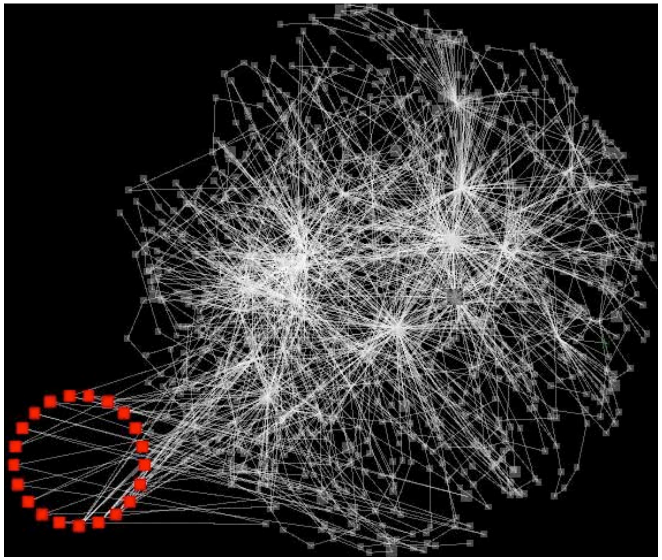
SDM 2014, Philadelphia, PA, April 2014.

Code: www.cs.cmu.edu/~dkoutra/CODE/vog.tar

VoG: Summarizing Wiki-controversy







top-8 star structures: admins, heavy wiki users, bots

warring factions changing each-other's edits. (Kiev vs Kiyv)

Ditto, between vandals

Carnegie Mellon

VoG: Summarizing Graphs using Rich Vocabularies

Main Ideas:

(1) Use `vocabulary' of subgraphtypes

(2) Minimum Description Length (MDL) and above vocabulary, to summarize graph

Summary of Part#1

- *many* patterns in real graphs
  - Power-laws everywhere
  - Gaussian trap
    - Avg<< Max

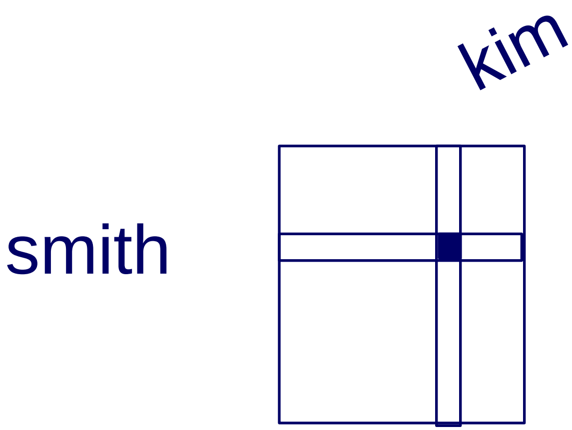
Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
  - P2.1: time-evolving graphs
  - P2.2: with side information (‘coupled’ M.T.F.)
  - Speed
- Part#3: Cascades and immunization
- Conclusions

Part 2:
Time evolving graphs; tensors

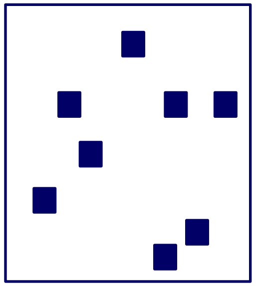
Graphs over time -> tensors!

- Problem #2.1:
  - Given who calls whom, and when
  - Find patterns / anomalies



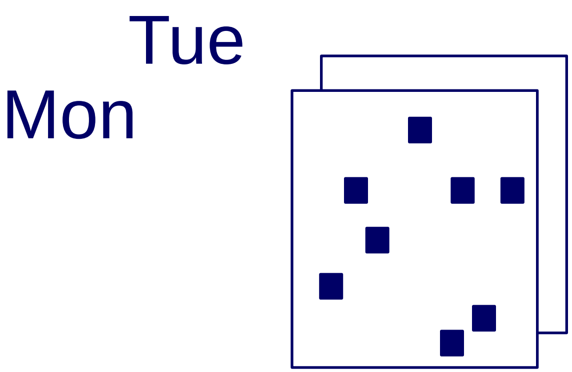
Graphs over time -> tensors!

- Problem #2.1:
  - Given who calls whom, and when
  - Find patterns / anomalies



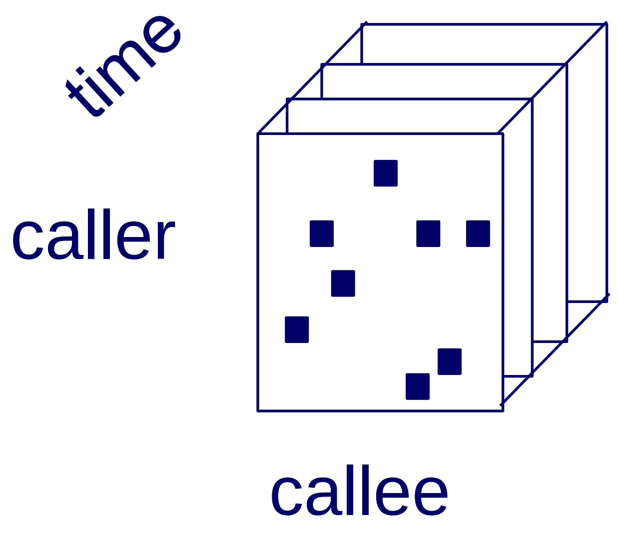
Graphs over time -> tensors!

- Problem #2.1:
  - Given who calls whom, and when
  - Find patterns / anomalies



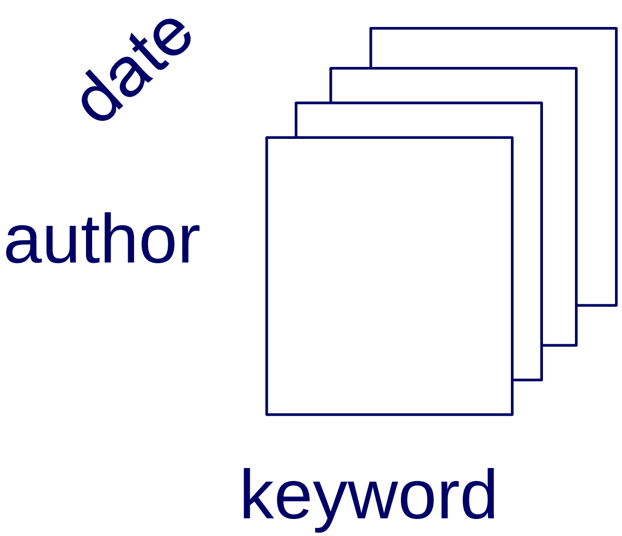
Graphs over time -> tensors!

- Problem #2.1:
  - Given who calls whom, and when
  - Find patterns / anomalies



Graphs over time -> tensors!

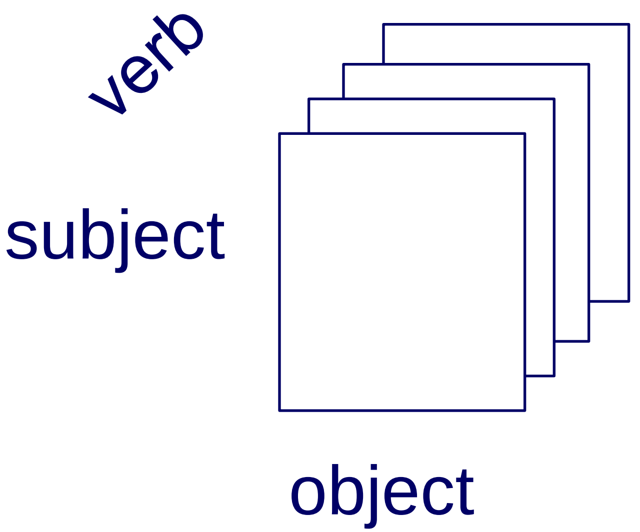
- Problem #2.1':
  - Given author-keyword-date
  - Find patterns / anomalies



MANY more settings, with >2 'modes'

Graphs over time -> tensors!

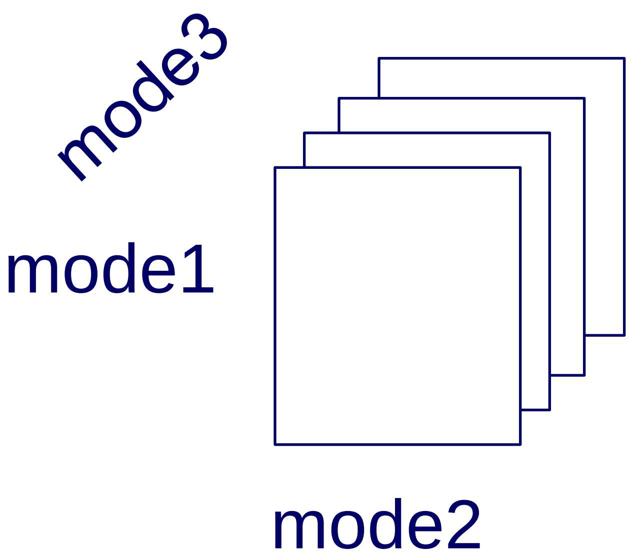
- Problem #2.1'':
  - Given subject – verb – object facts
  - Find patterns / anomalies



MANY more settings, with >2 'modes'

Graphs over time -> tensors!

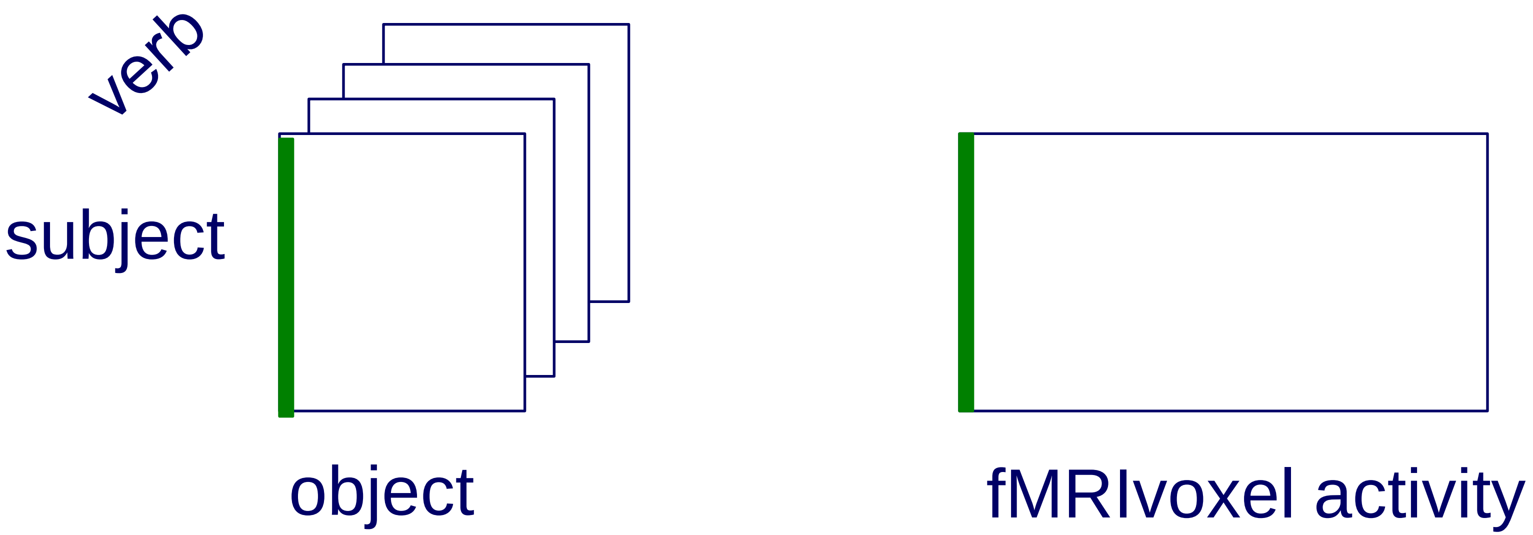
- Problem #2.1''':
  - Given <triplets>
  - Find patterns / anomalies



MANY more settings, with >2 'modes' (and 4, 5, etc modes)

Graphs & side info

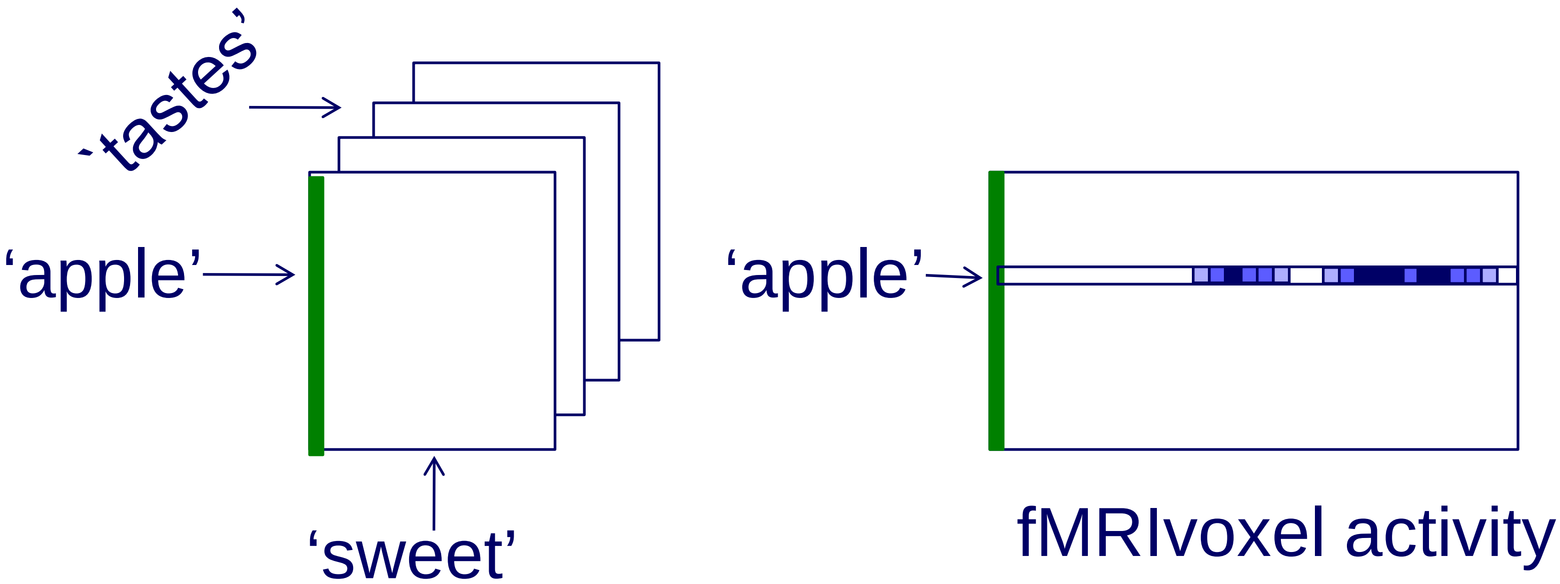
- Problem #2.2: coupled (eg., side info)
  - Given subject – verb – object facts
    - And voxel-activity for each subject-word
  - Find patterns / anomalies



`apple tastes sweet'

Graphs & side info

- Problem #2.2: coupled (eg., side info)
  - Given subject – verb – object facts
    - And voxel-activity for each subject-word
  - Find patterns / anomalies



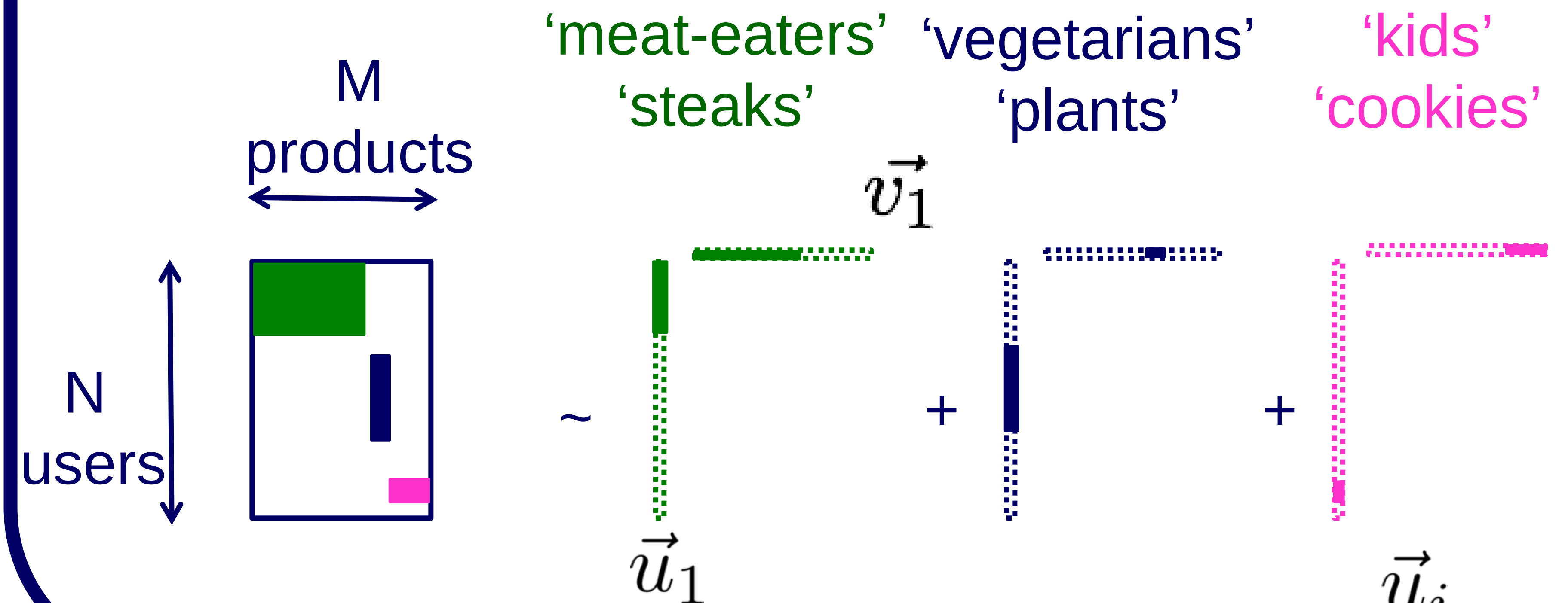
`apple tastes sweet'

Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
  - P2.1: time-evolving graphs
  - P2.2: with side information ('coupled' M.T.F.)
  - Speed
- Part#3: Cascades and immunization
- Conclusions

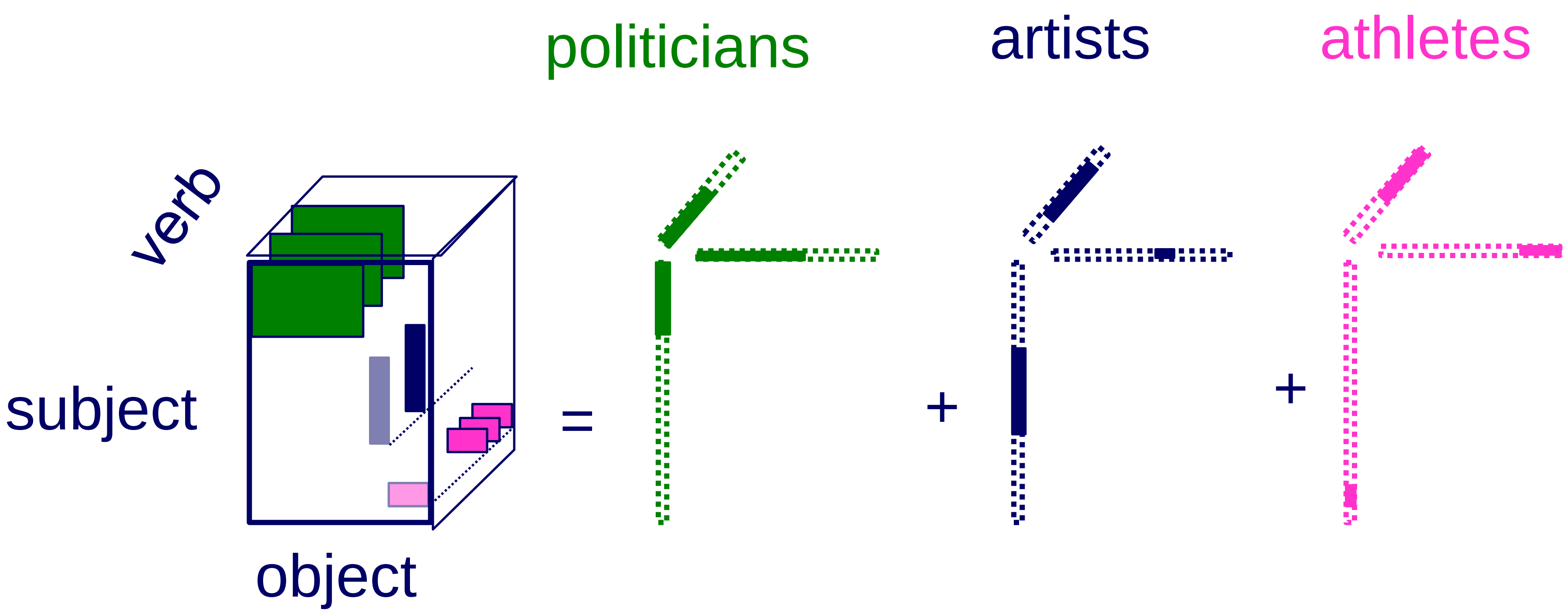
Answer to both: tensor factorization

- Recall: (SVD) matrix factorization: finds blocks



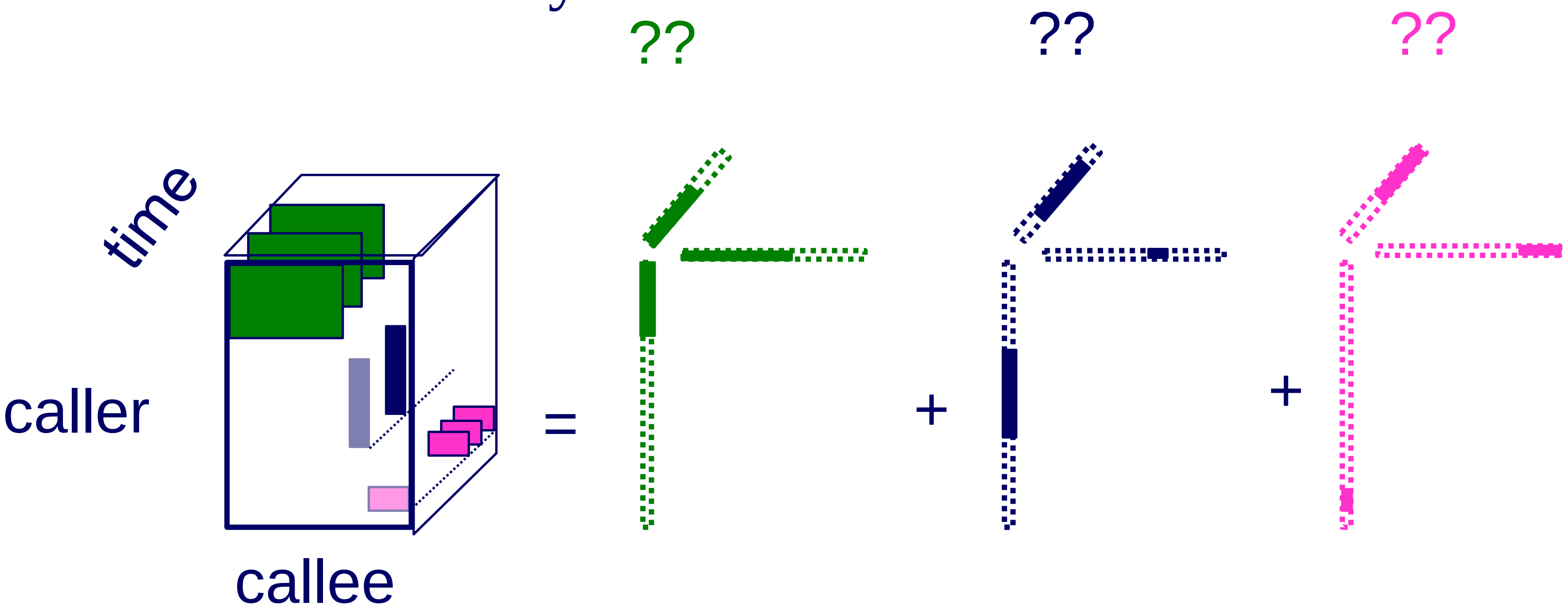
Answer to both: tensor factorization

- PARAFAC decomposition

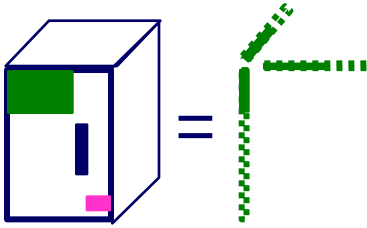


Answer: tensor factorization

- PARAFAC decomposition
- Results for who-calls-whom-when
  - 4M x 15 days

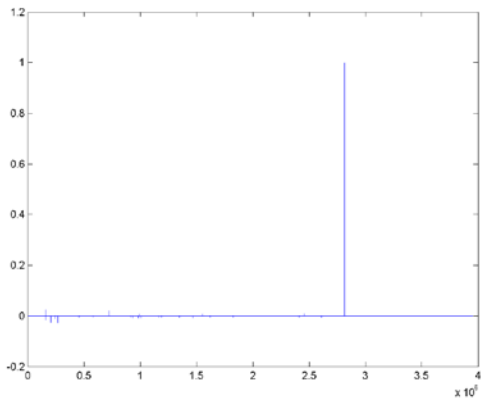


Anomaly detection in time-evolving graphs

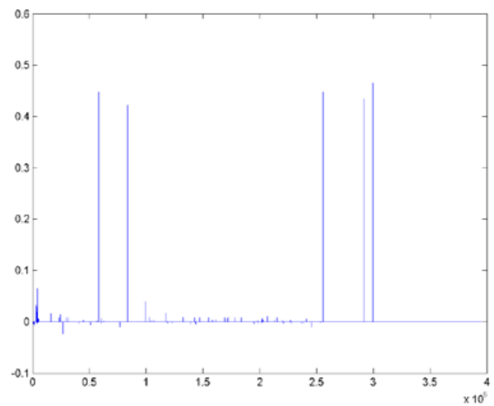


- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

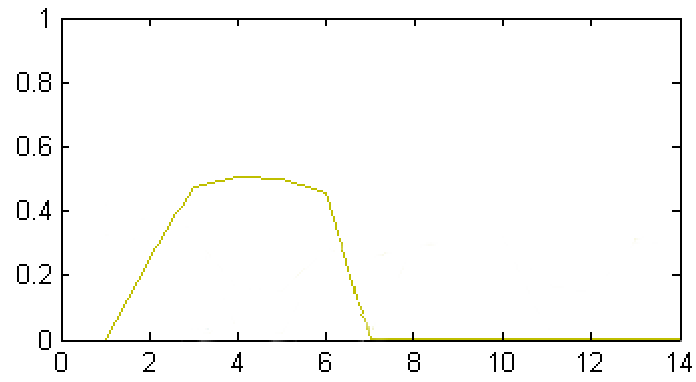
1 caller



5 receivers

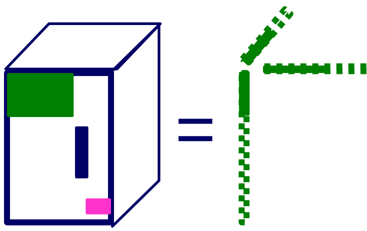


4 days of activity



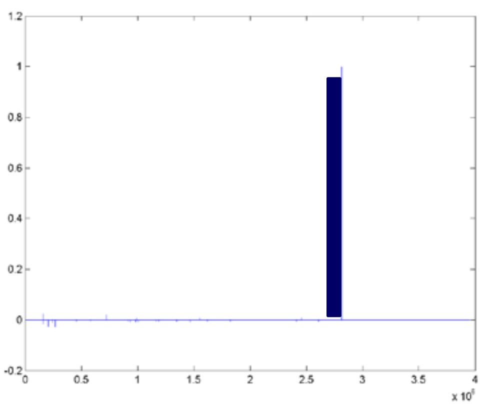
~200 calls to EACH receiver on EACH day!

Anomaly detection in time-evolving graphs

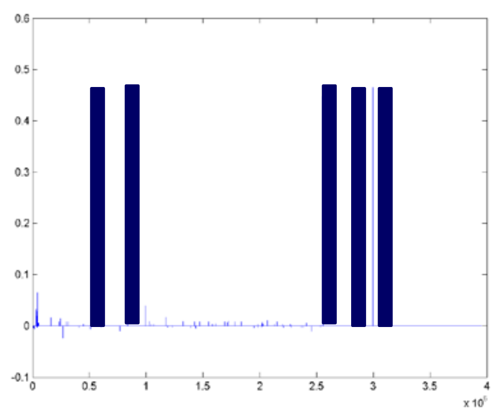


- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

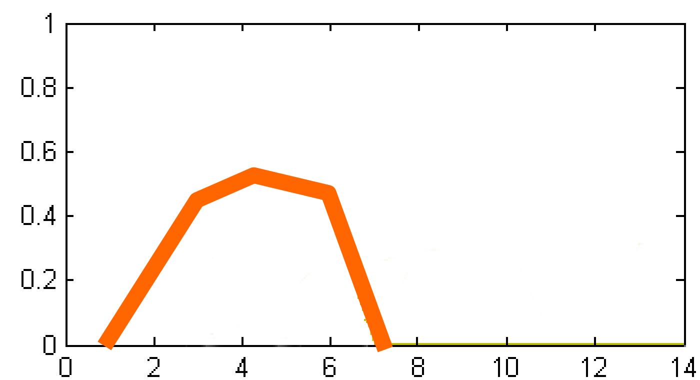
1 caller



5 receivers

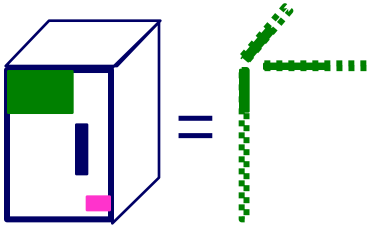


4 days of activity

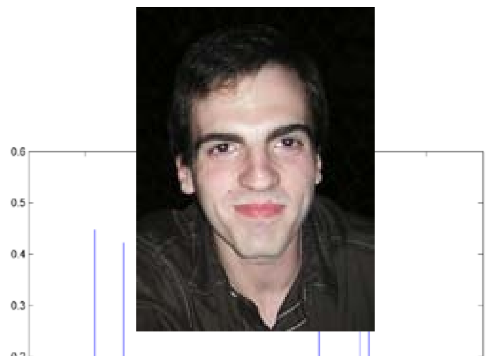


~200 calls to EACH receiver on EACH day!

Anomaly detection in time-evolving graphs



- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

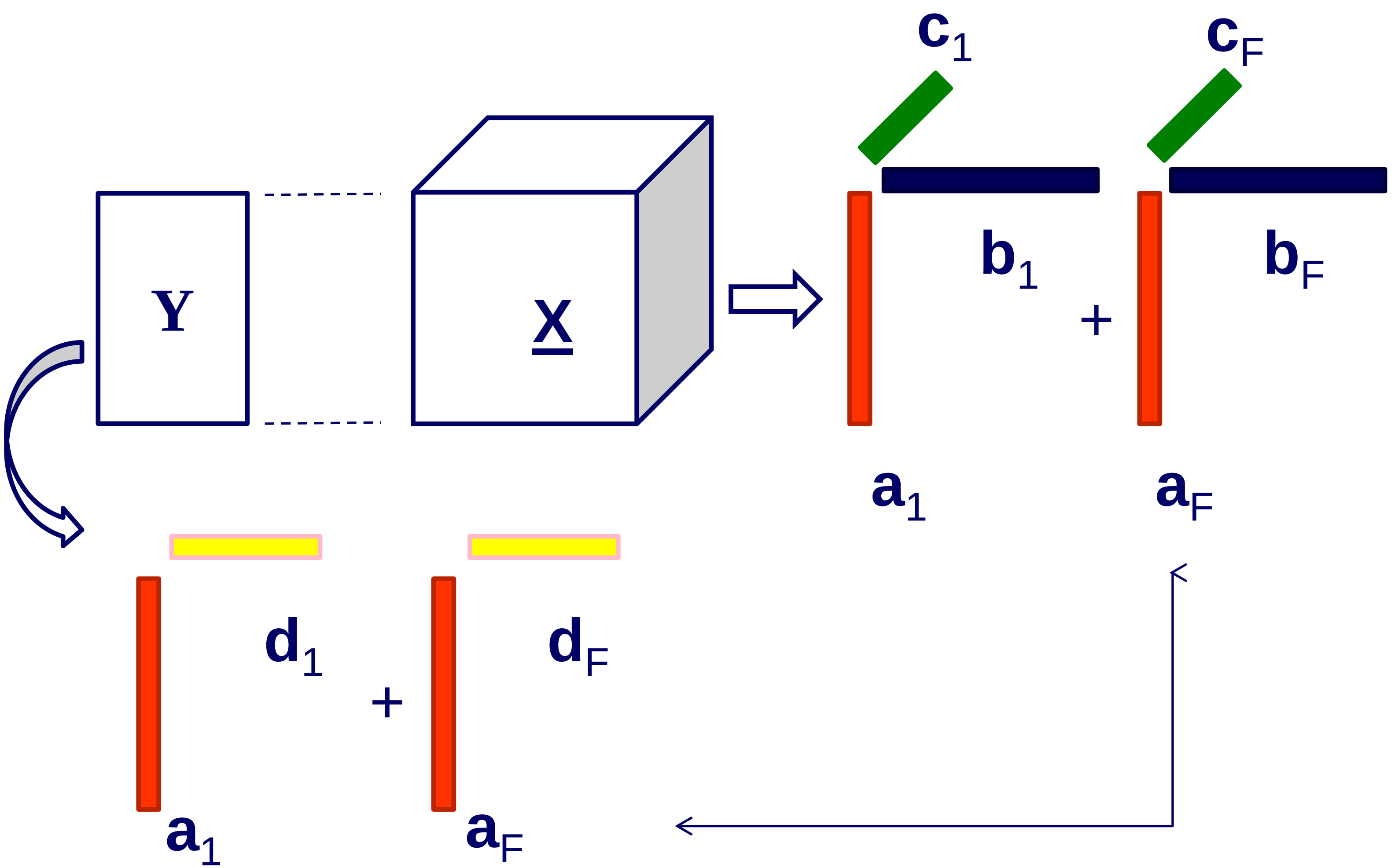


Miguel Araujo, Spiros Papadimitriou, Stephan Günnemann, Christos Faloutsos, PrithwishBasu, Ananthram Swami, EvangelosPapalexakis, DanaiKoutra. *Com2: Fast Automatic Discovery of Temporal (Comet) Communities*. PAKDD 2014, Tainan, Taiwan.

Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
  - P2.1: Discoveries @ phonecall network
  - P2.2: Discoveries in neuro-semantics
  - Speed
- Part#3: Cascades and immunization
- Conclusions

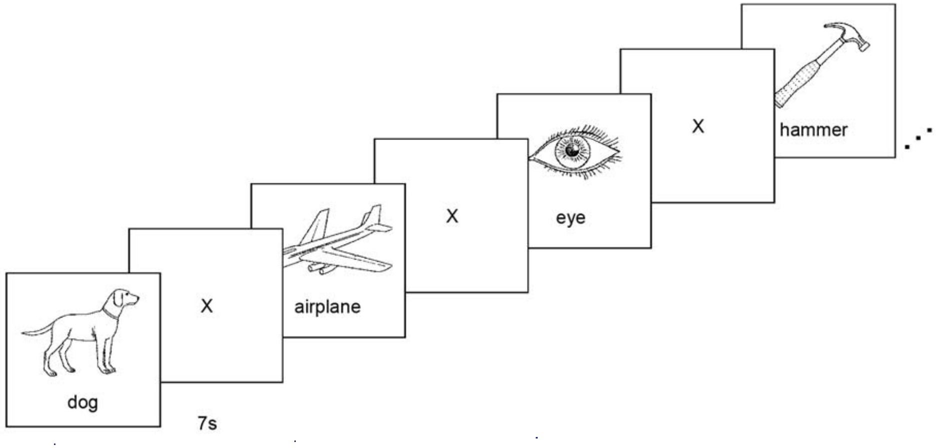
Coupled Matrix-Tensor Factorization (CMTF)



CarnegieMellon

Neuro-semantics

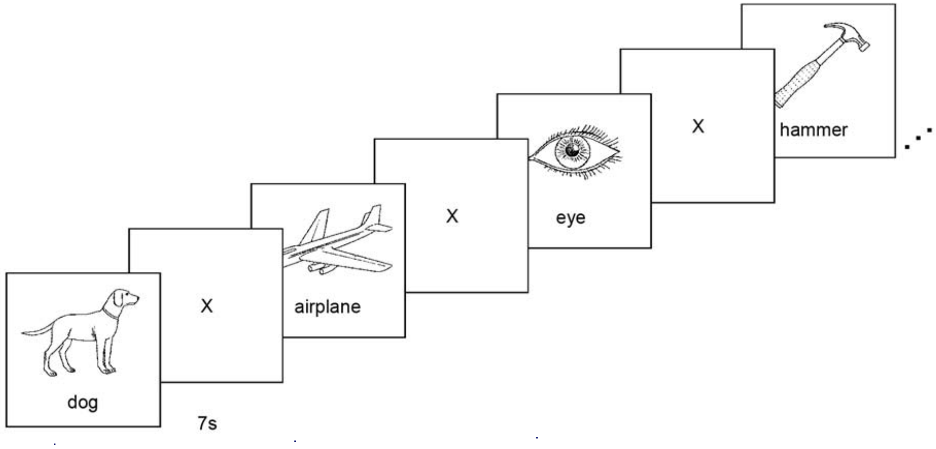
- **Brain Scan Data***
  - 9 persons
  - 60 nouns
- **Questions**
  - 218 questions
  - ‘is it alive?’, ‘can you eat it?’



*Mitchell et al. *Predicting human brain activity associated with the meanings of nouns*. Science,2008. Data@ www.cs.cmu.edu/afs/cs/project/theo-73/www/science2008/data.html

Neuro-semantics

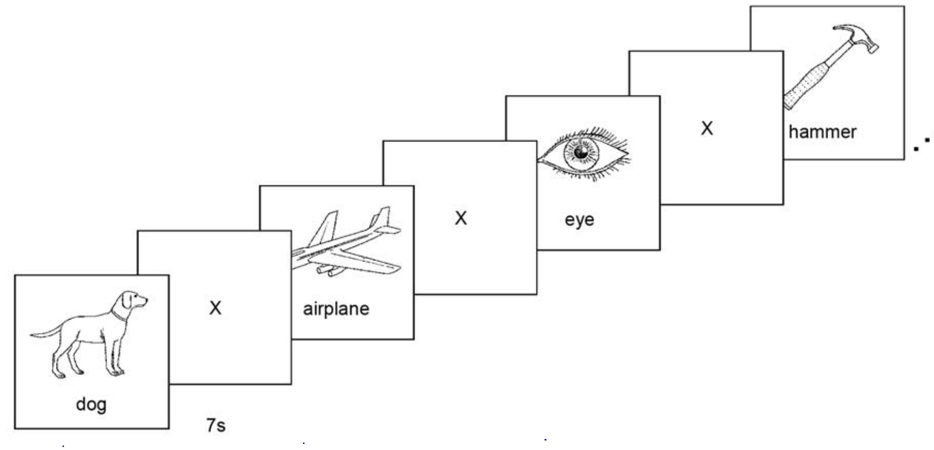
- **Brain Scan Data***
  - 9 persons
  - 60 nouns
- **Questions**
  - 218 questions
  - 'is it alive?', 'can you eat it?'



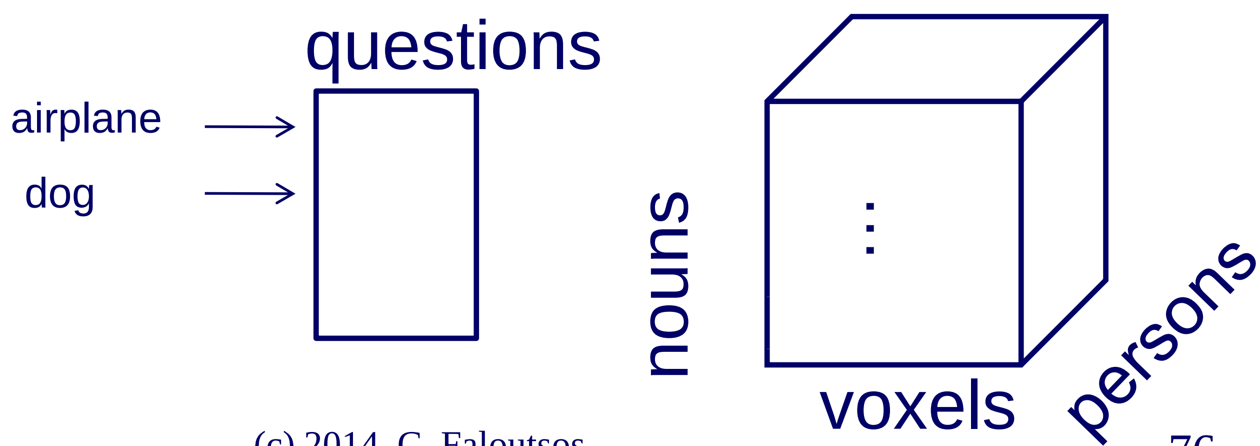
Patterns?

Neuro-semantics

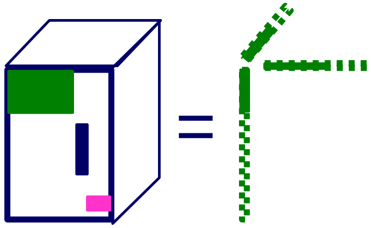
- **Brain Scan Data***
  - 9 persons
  - 60 nouns
- **Questions**
  - 218 questions
  - ‘is it alive?’, ‘can you eat it?’

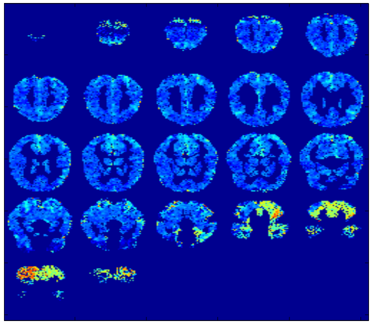


Patterns?

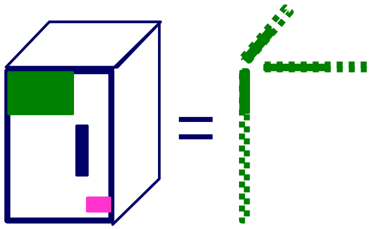


Neuro-semantics





Neuro-semantics



Small items -> Premotor cortex

Carnegie Mellon

Neuro-semantics

Small items -> Premotor cortex

EvangelosPapalexakis, Tom Mitchell, Nicholas Sidiropoulos, Christos Faloutsos, ParthaPratimTalukdar, Brian Murphy, *Turbo-SMT: Accelerating Coupled Sparse Matrix-Tensor Factorizations by 200x*, SDM 2014

Roadmap

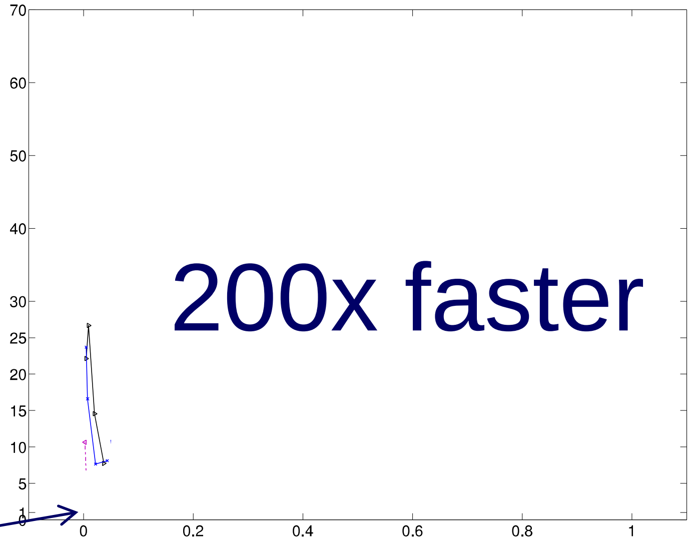
- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
  - P2.1: Discoveries @ phonecall network
  - P2.2: Discoveries in neuro-semantics
  - Speed
- Part#3: Cascades and immunization
- Conclusions

Speed of tensor/CMTF analysis

- Q1: Can we make it fast?
- Q2: Does it work for large, disk-based data?

A1: Turbo-SMT

Relative cost



Ideal

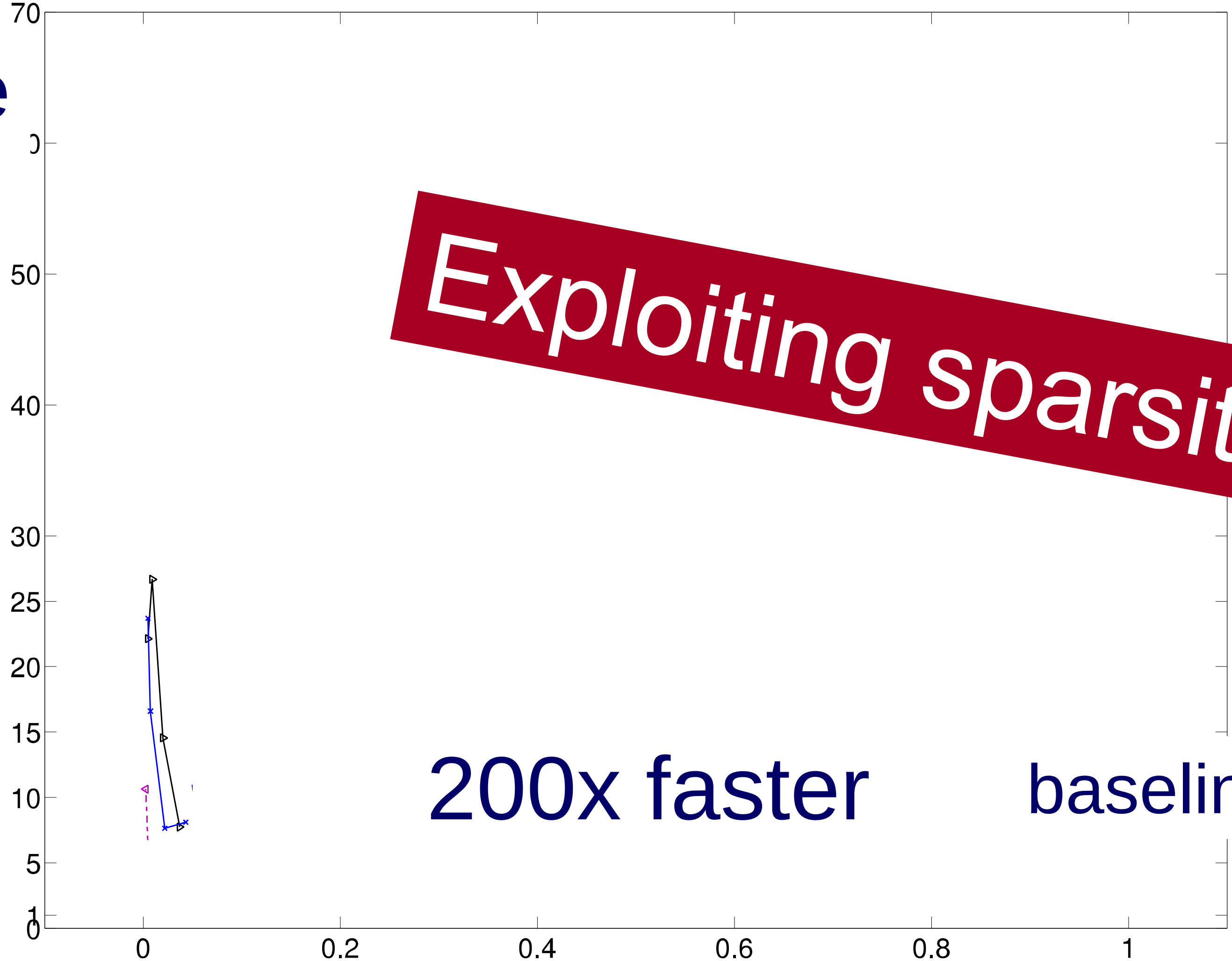
Relative run time

EvangelosPapalexakis, Tom Mitchell, Nicholas Sidiropoulos, Christos Faloutsos, Partha Pratim Talukdar, Brian Murphy, *Turbo-SMT: Accelerating Coupled Sparse Matrix-Tensor Factorizations by 200x*, SDM 2014

Carnegie Mellon

A1: Turbo-SMT

Relative cost



Exploiting sparsity

200x faster

baseline

Ideal

Relative run time

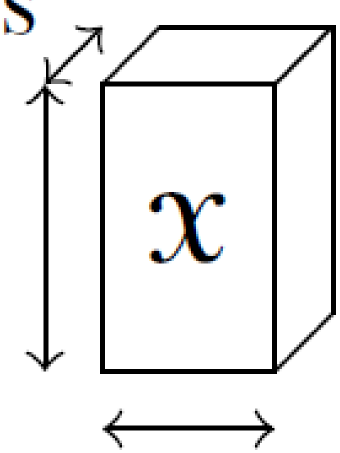
Q2: spilling to the disk?

Reminder: tensor (eg., Subject-verb-object)

144M non-zeros

48M verbs

26M subjects



26M objects

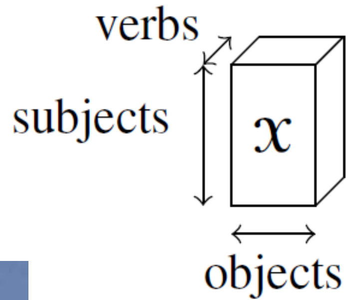
NELL (Never Ending Language Learner) @CMU

Carnegie Mellon

A2: GigaTensor

Reminder: tensor (eg., Subject-verb-object)

26M x 48M x 26M, 144M non-zeros

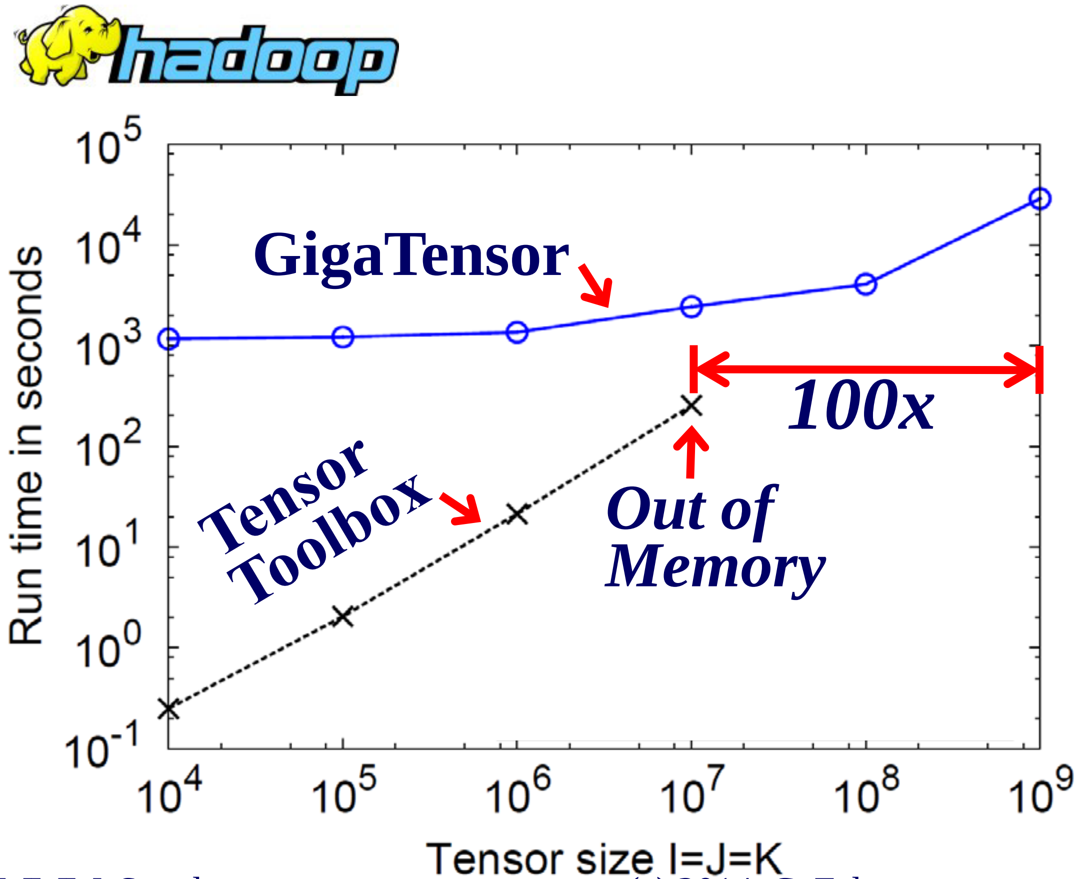


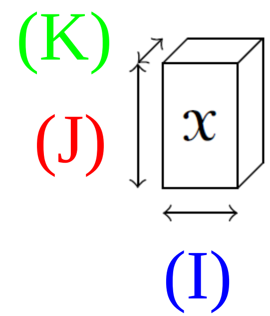
NELL (Never Ending Language Learner)@CMU

U Kang, Evangelos E. Papalexakis, AbhayHarpale, Christos Faloutsos, *GigaTensor: Scaling Tensor Analysis Up By 100 Times - Algorithms and Discoveries*, KDD'12

A2: GigaTensor

- GigaTensor solves ***100x*** larger problem

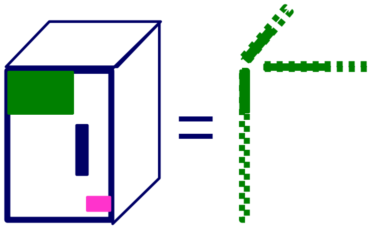




Number of nonzero = I / 50

Part 2: Conclusions

- Time-evolving / heterogeneous graphs -> tensors
- PARAFAC finds patterns
- Turbo-SMT; GigaTensor -> fast & scalable



Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Conclusions

Part 3: Cascades & Immunization

Why do we care?

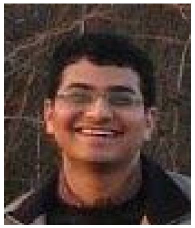
- Information Diffusion
- Viral Marketing
- Epidemiology and Public Health
- Cyber Security
- Human mobility
- Games and Virtual Worlds
- Ecology
- ....

Carnegie Mellon

Roadmap

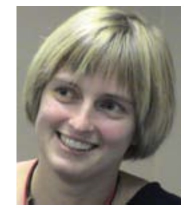
- A case for cross-disciplinarity
- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: Cascade analysis
  - (Fractional) Immunization
  - Epidemic thresholds
- Conclusions

Fractional Immunization of Networks



B. Aditya Prakash,

LadaAdamic,



Theodore Iwashyna (M.D.),

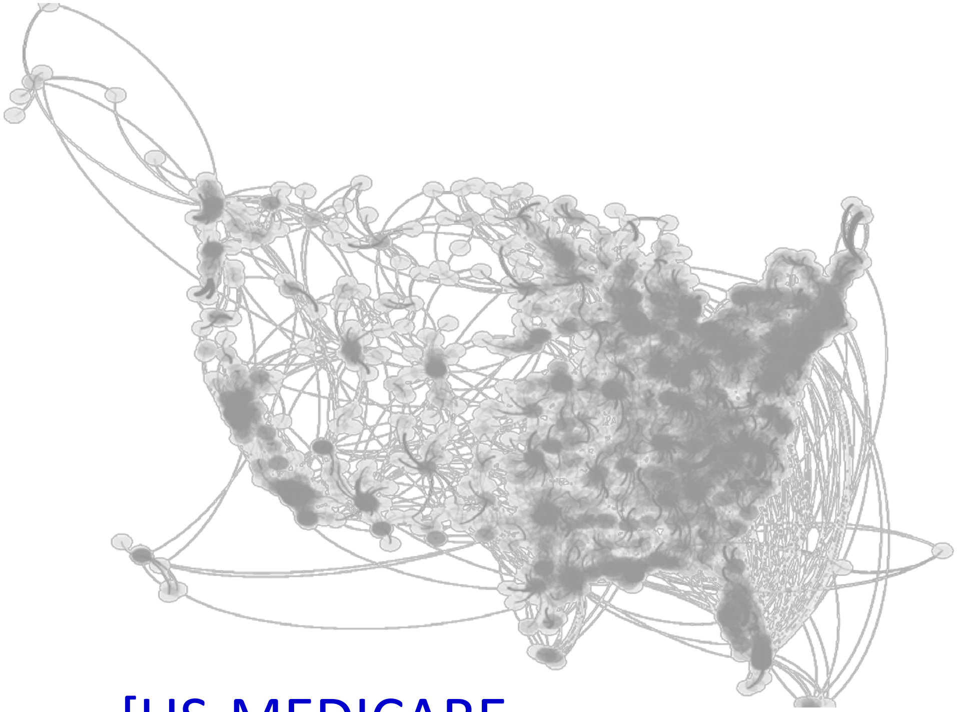
Hanghang Tong,

Christos Faloutsos

SDM 2013, Austin, TX

Whom to immunize?

- Dynamical Processes over networks



[US-MEDICARE NETWORK 2005]

- Each circle is a hospital
- ~3,000 hospitals
- More than 30,000 patients transferred

Problem: Given *k* units of disinfectant, whom to immunize?

Whom to immunize?

~6x fewer!

[US-MEDICARE NETWORK 2005]

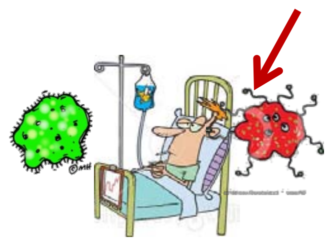
CURRENT PRACTICE

OUR METHOD

Hospital-acquired inf. : 99K+ lives, $5B+ per year

Fractional Asymmetric Immunization

Drug-resistant Bacteria
(like XDR-TB)

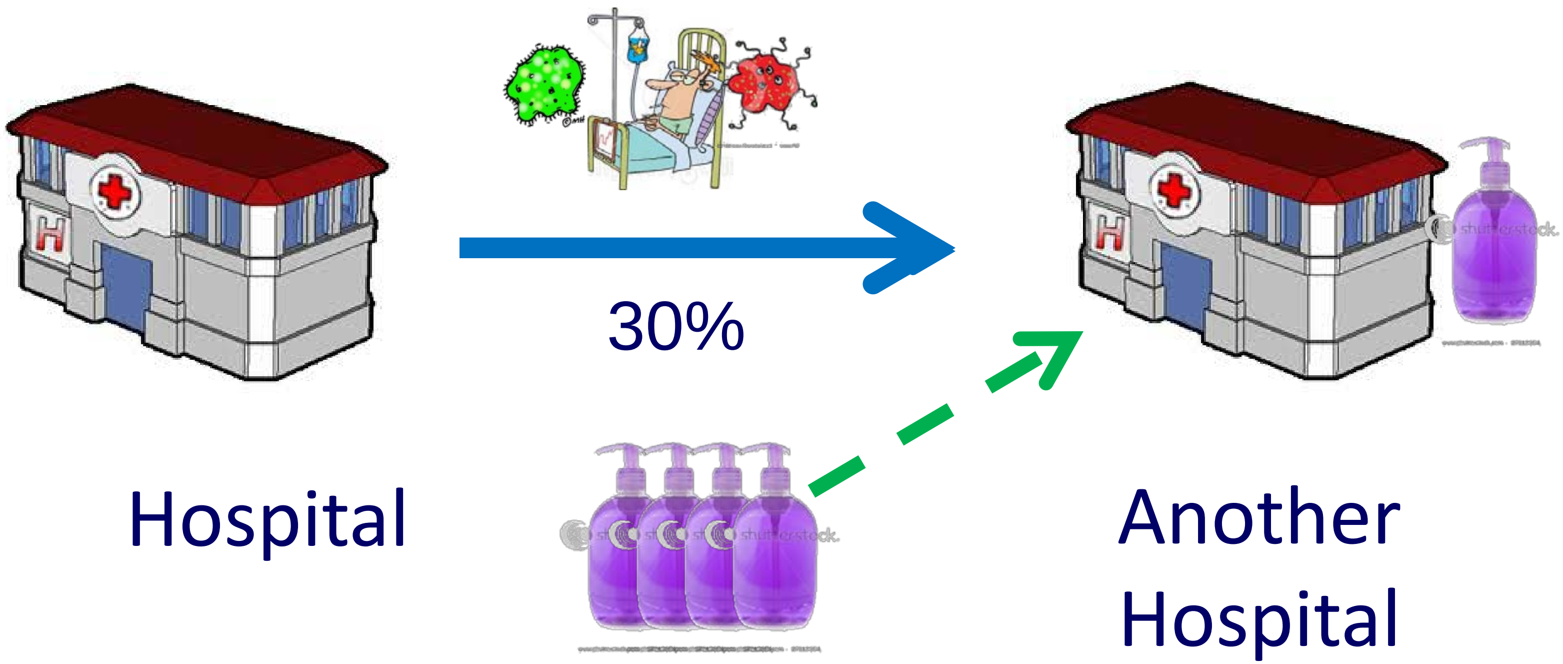


30%

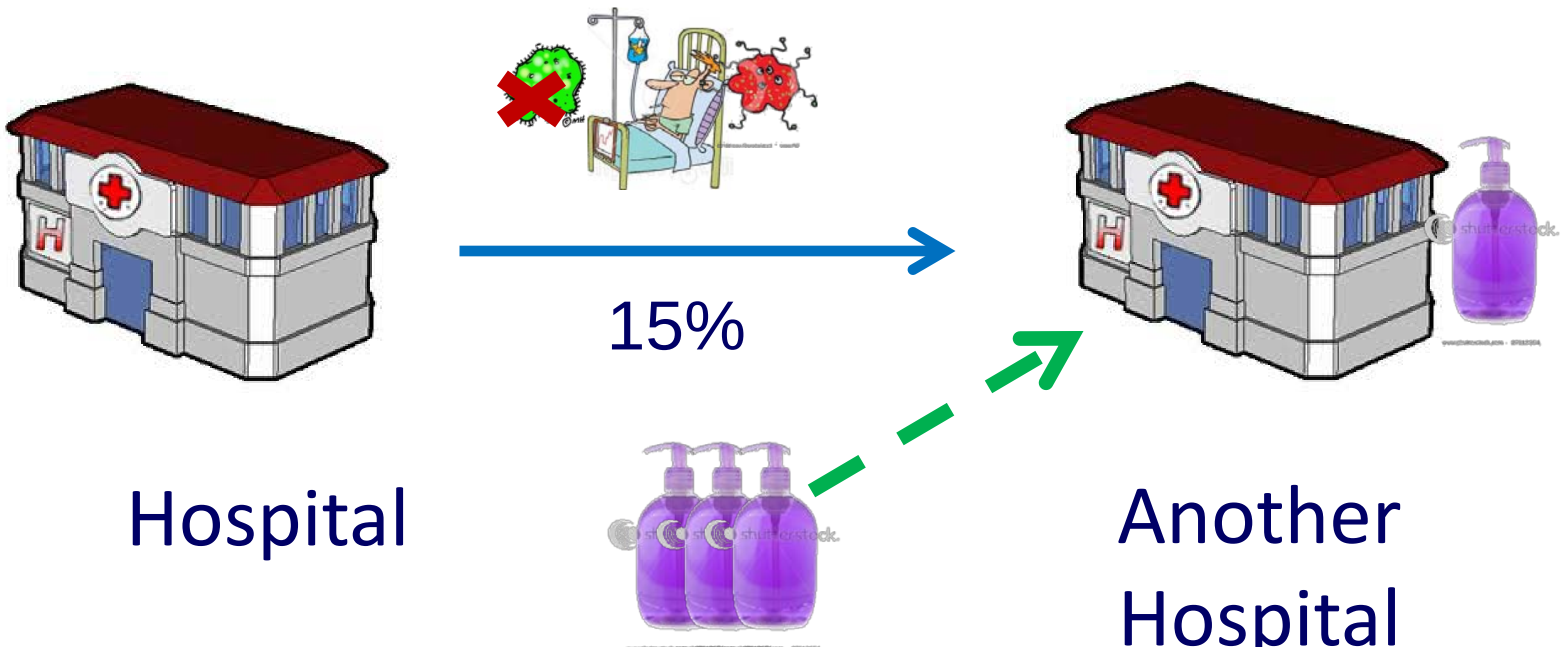
Hospital

Another Hospital

Fractional Asymmetric Immunization

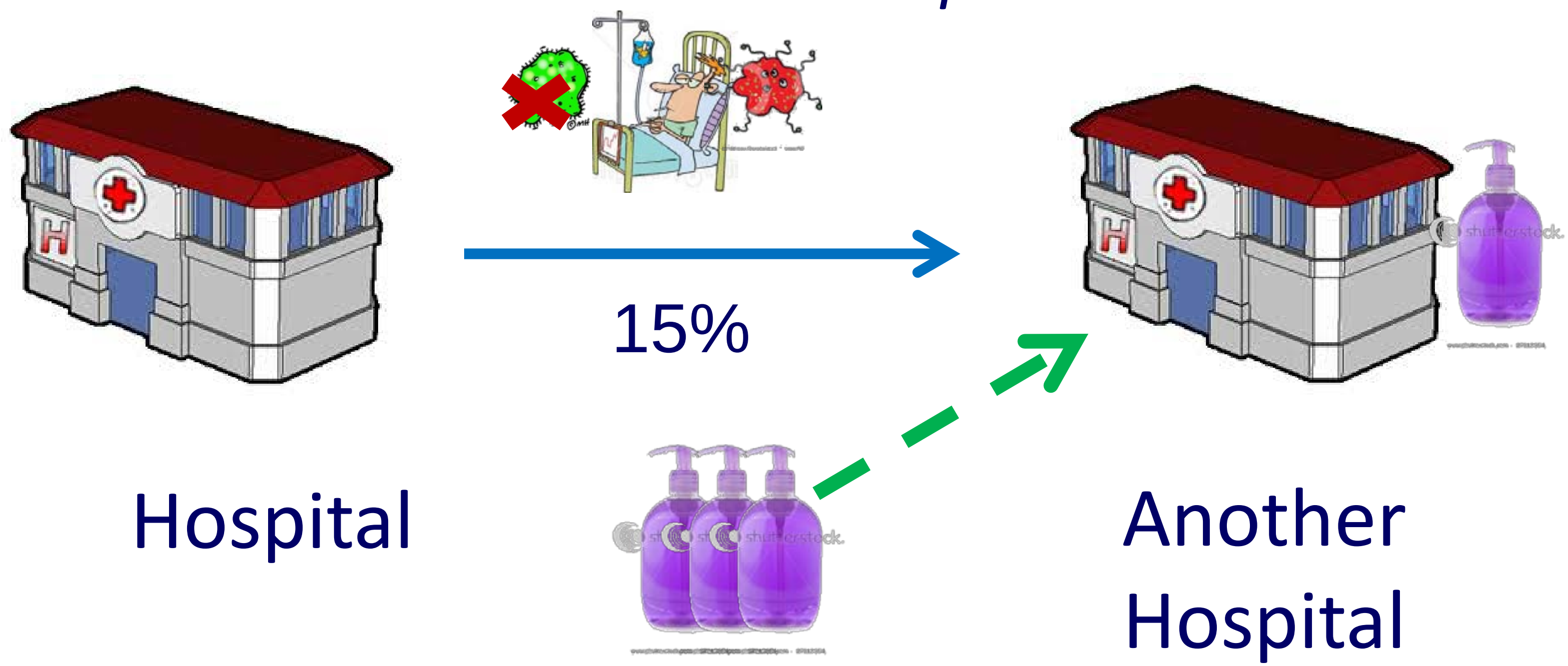


Fractional Asymmetric Immunization



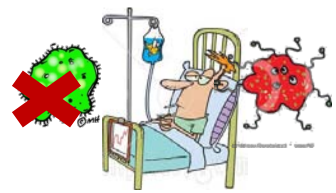
Fractional Asymmetric Immunization

***Problem**:*
Given k units of disinfectant,
distribute them
to maximize hospitals saved

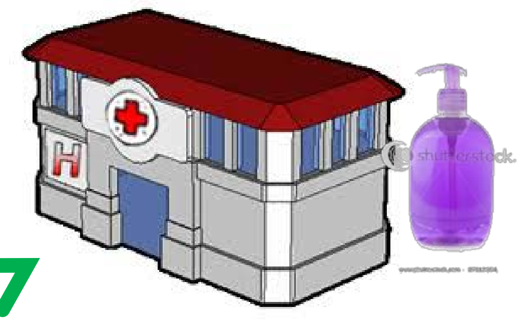


Fractional Asymmetric Immunization

***Problem**:*
Given k units of disinfectant,
distribute them
to maximize hospitals saved @ 365 days



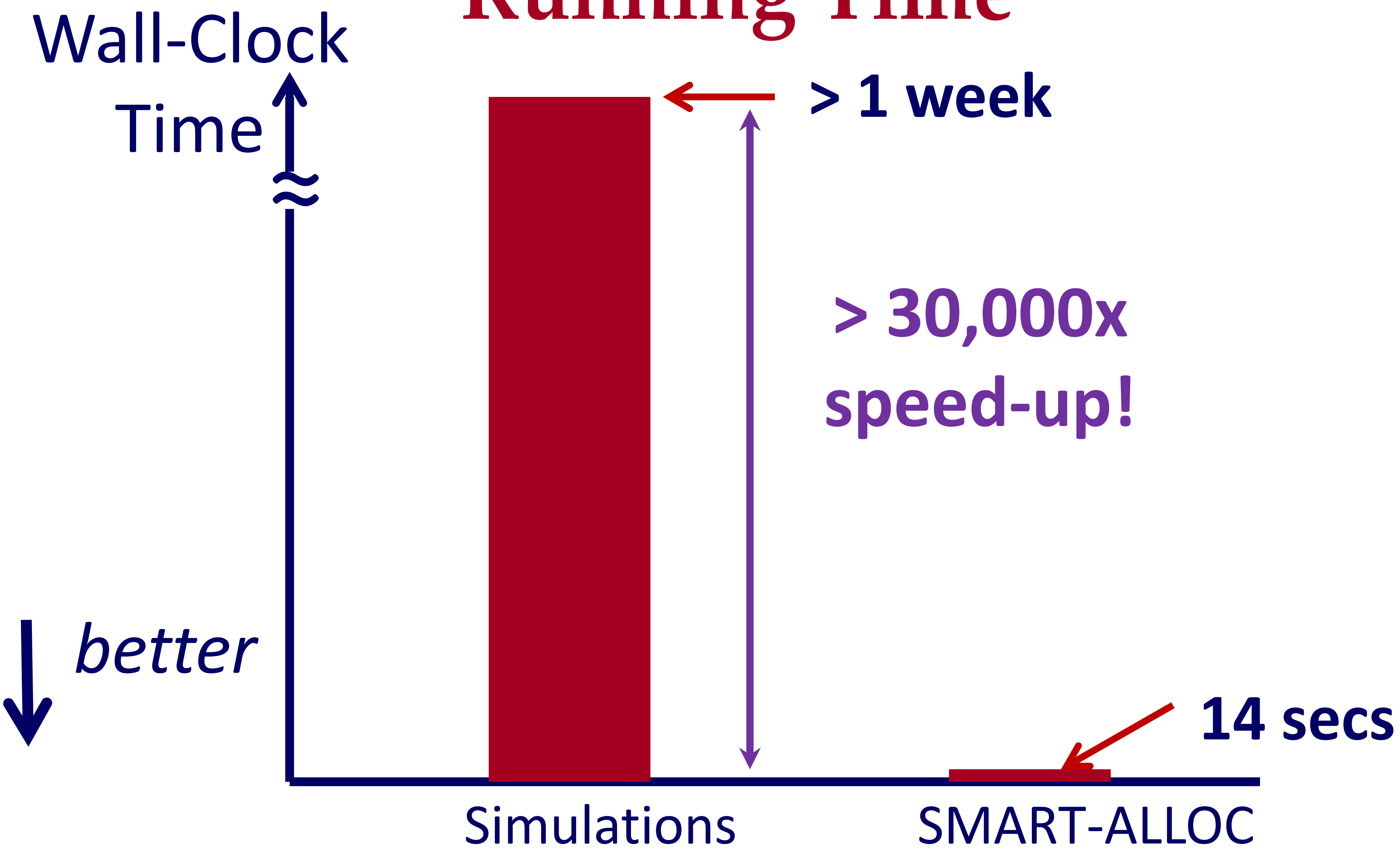
15%



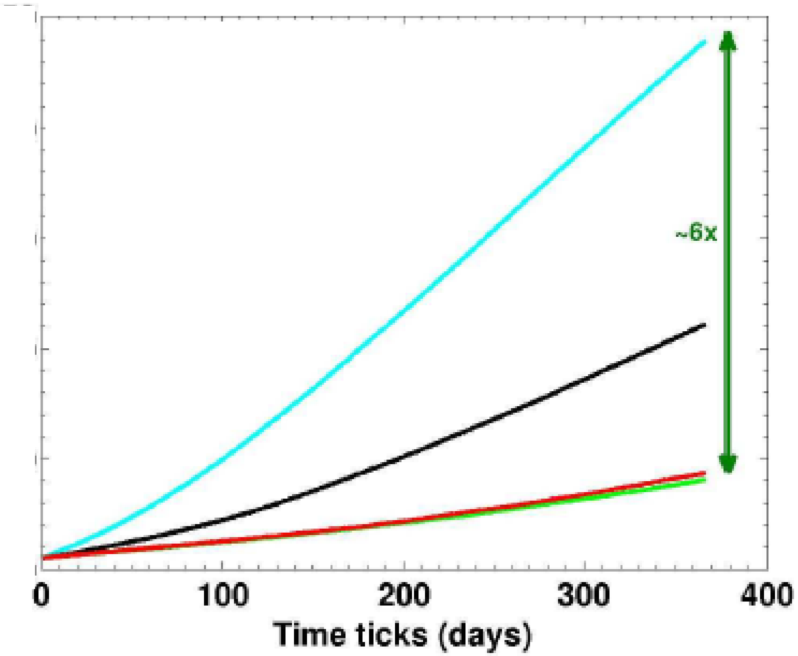
Hospital

Another Hospital

Running Time



Experiments



infected

uniform

better

SMART-ALLOC

K = 120

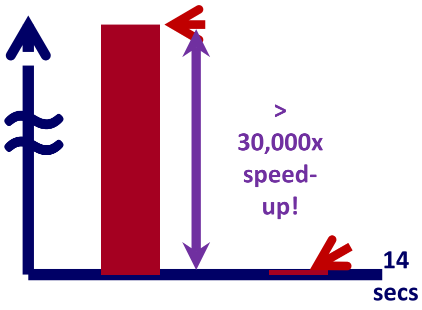
epochs

What is the 'silver bullet'?

A: Try to decrease connectivity of graph

Q: how to measure connectivity?

- Avg degree? Max degree?
- Std degree / avg degree ?
- Diameter?
- Modularity?
- 'Conductance' (~min cut size)?
- Some combination of above?



What is the 'silver bullet'?

A: Try to decrease connectivity of graph

Q: how to measure connectivity?

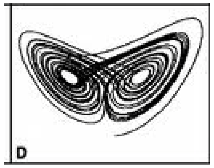
A: first **eigenvalue** of adjacency matrix

Avg degree
Max degree
Diameter
Modularity
'Conductance'

Q1: why??

(Q2: dfn& intuition of eigenvalue ?)

Carnegie Mellon



Why eigenvalue?

A1: ‘G2’ theorem and ‘**eigen-drop**’:

- For (almost) **any** type of virus
- For **any** network
- -> no epidemic, if small-enough first eigenvalue (λ_1) of *adjacency* matrix

Threshold Conditions for Arbitrary Cascade Models on Arbitrary Networks, B. Aditya Prakash, Deepayan Chakrabarti, Michalis Faloutsos, Nicholas Valler, Christos Faloutsos, ICDM 2011, Vancouver, Canada

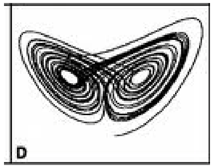
Why eigenvalue?

A1: ‘G2’ theorem and ‘**eigen-drop**’:

- For (almost) **any** type of virus
- For **any** network
- -> no epidemic, if small-enough first eigenvalue (λ_1) of *adjacency* matrix

- Heuristic: for immunization, try to min λ_1
- The smaller λ_1, the closer to extinction.

G2 theorem



Threshold Conditions for Arbitrary Cascade Models on Arbitrary Networks

B. Aditya Prakash, Deepayan Chakrabarti, Michalis Faloutsos, Nicholas Valler, Christos Faloutsos

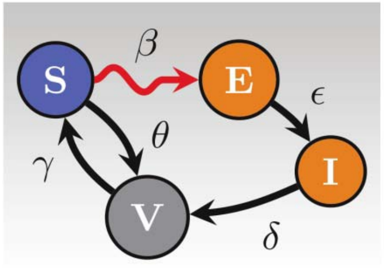
IEEE ICDM 2011, Vancouver

extended version, in arxiv
http://arxiv.org/abs/1004.0060

~10 pages proof

Our thresholds for some models

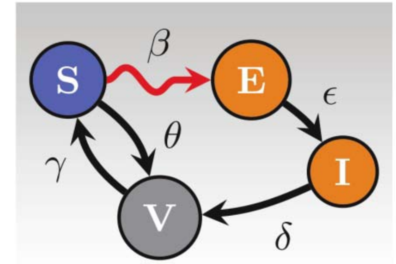
- *s = effective strength*
- *s< 1 : below threshold*



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

Our thresholds for some models

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



No immunity

Temp. immunity

w/ incubation

	e Strength	Threshold (tipping point)
SIS, SIR, SIRS, SEIR	$s = \lambda \left(\frac{\beta}{\delta}\right)$	$s = 1$
SIV, SEIV	$s = \lambda \,.\left(\frac{\beta\gamma}{\delta(\gamma+\theta)}\right)$	
$S_1 I_1 I_2 V_1 V_2$ (H.I.V.)	$s = \lambda \,.\left(\frac{\beta_1 v_2 + \beta_2 \varepsilon}{v_2(\varepsilon + v_1)}\right)$	

Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: Cascade analysis
  - (Fractional) Immunization
  - intuition behind λ_1
- Conclusions

Intuition for λ

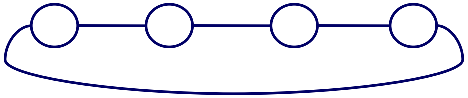
"Official" definitions:

- *Let **A** be the adjacency matrix. Then λ is the root with the largest magnitude of the characteristic polynomial of **A** [det(**A** – λ**I**)].*
- Also: $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$

Neither gives much intuition!

"Un-official" Intuition

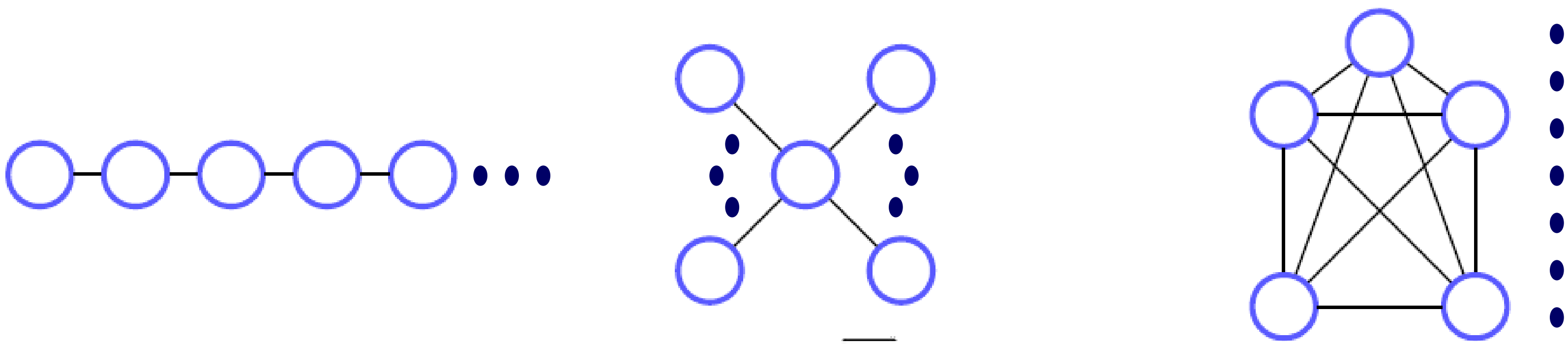
- For 'homogeneous' graphs, λ == *degree*



- λ ~ avg degree
  - done right, for skewed degree distributions

Largest Eigenvalue (λ)

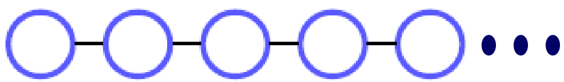
better connectivity $\longrightarrow$ higher λ



N = 1000 nodes

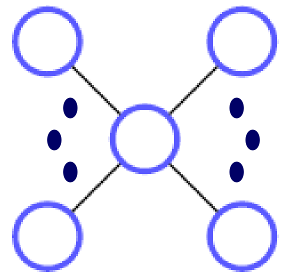
Largest Eigenvalue (λ)

better connectivity $\longrightarrow$ higher λ



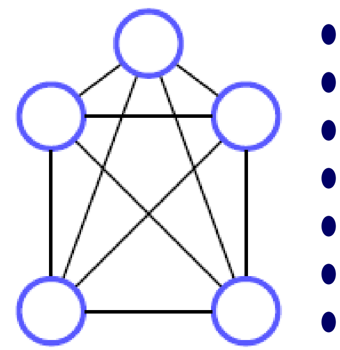
$\lambda \approx 2$

(a)Chain



$\lambda = \sqrt{N}$

(b)Star



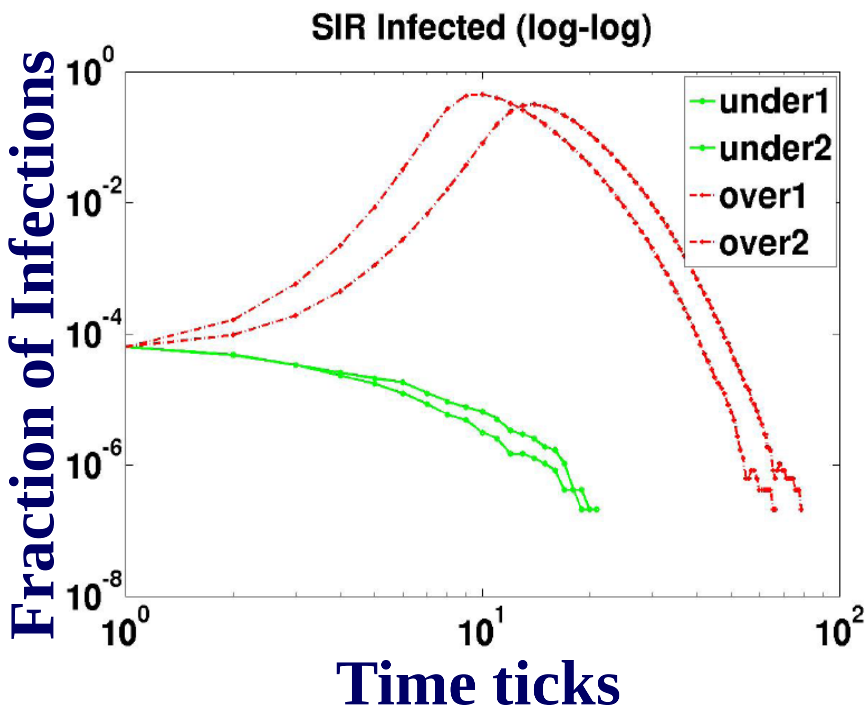
$\lambda = N\text{-}1$

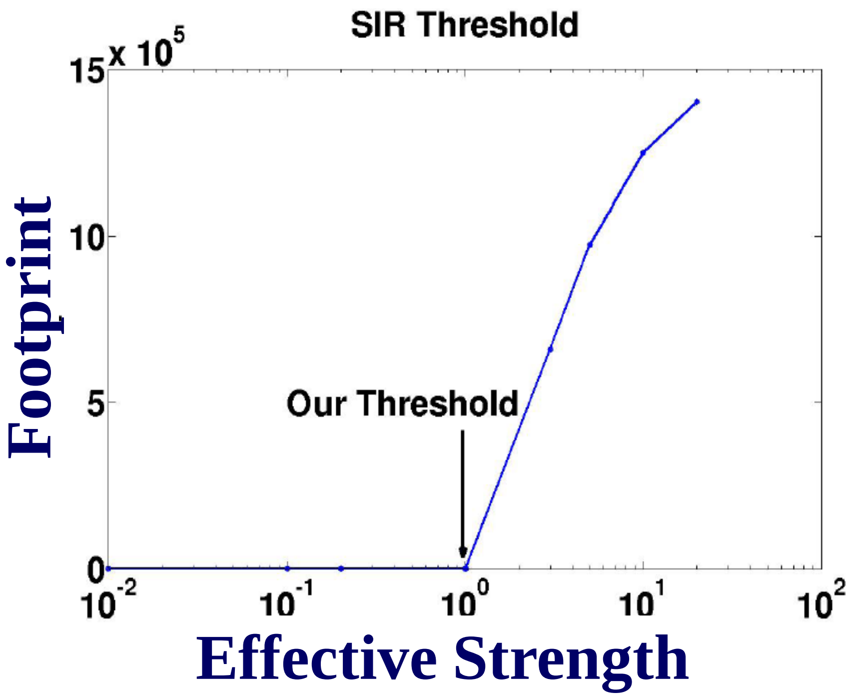
(c)Clique

$\lambda \approx 2$ — $\lambda = 31.67$ — $\lambda = 999$

N = 1000 nodes

Examples: Simulations – SIR (mumps)





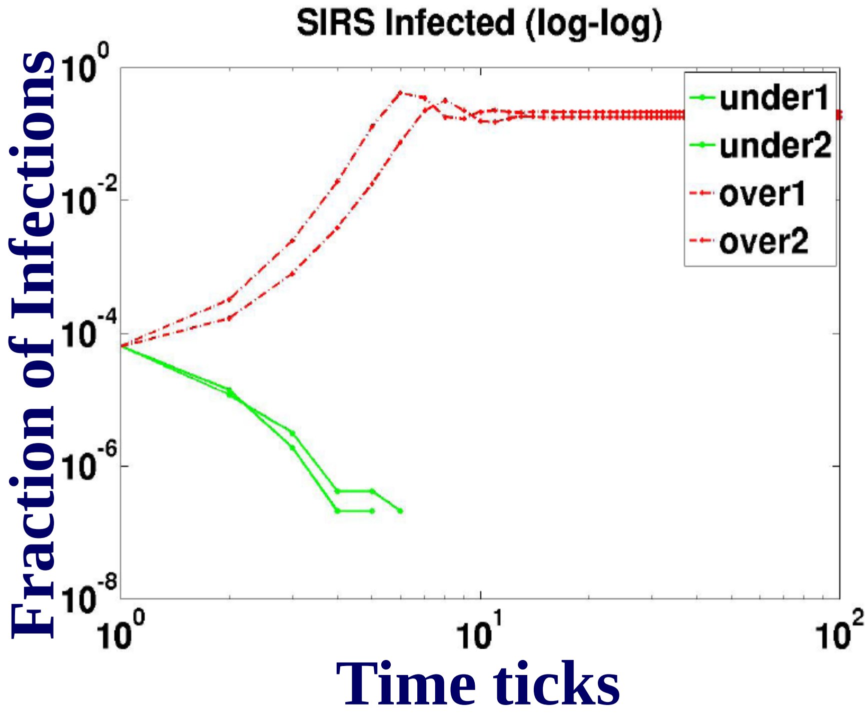
(a) Infection profile **(b) "Take-off" plot**

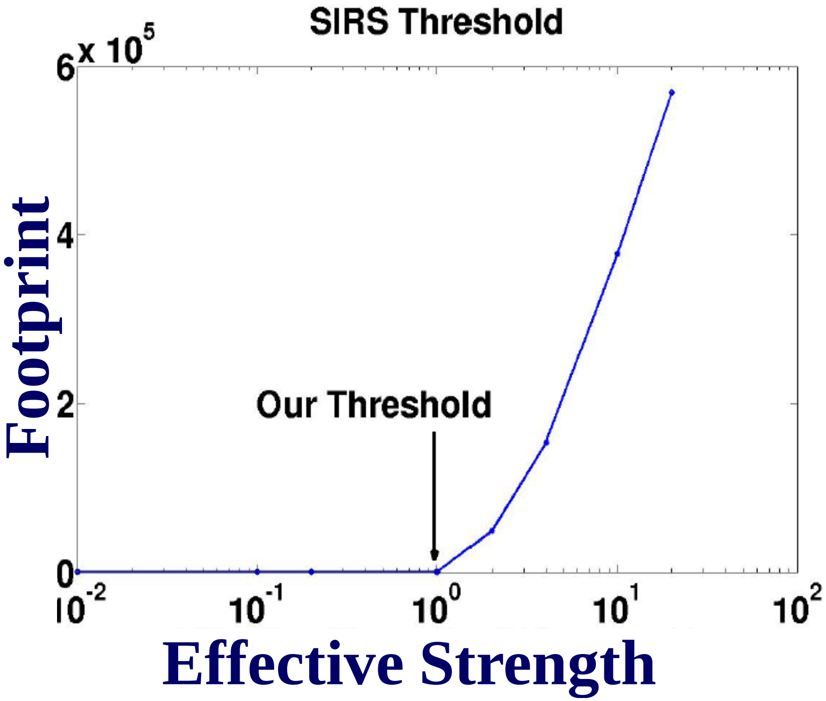
PORTLAND graph: *synthetic population,*

31 million links, 6 million nodes

Carnegie Mellon

Examples: Simulations – SIRS (pertusis)





(a) Infection profile **(b) "Take-off" plot**

PORTLAND graph: *synthetic population,*

31 million links, 6 million nodes

Part3: Immunization - conclusion

In (**almost any**) immunization setting,

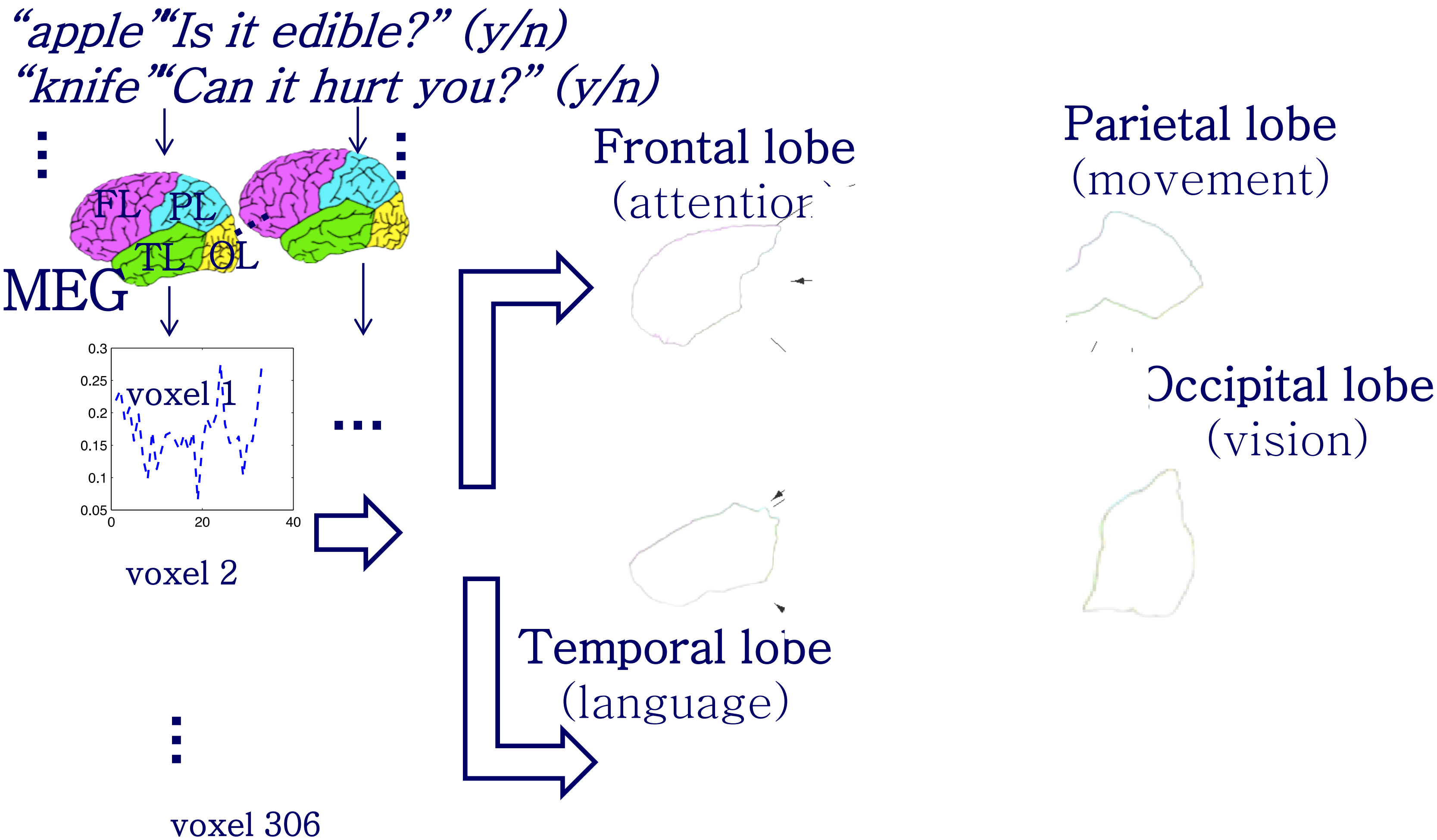
- Allocate resources, such that to
- **Minimize** λ_1
- (*regardless* of virus specifics)

- Conversely, in a market penetration setting
  - Allocate resources to
  - Maximize λ_1

Roadmap

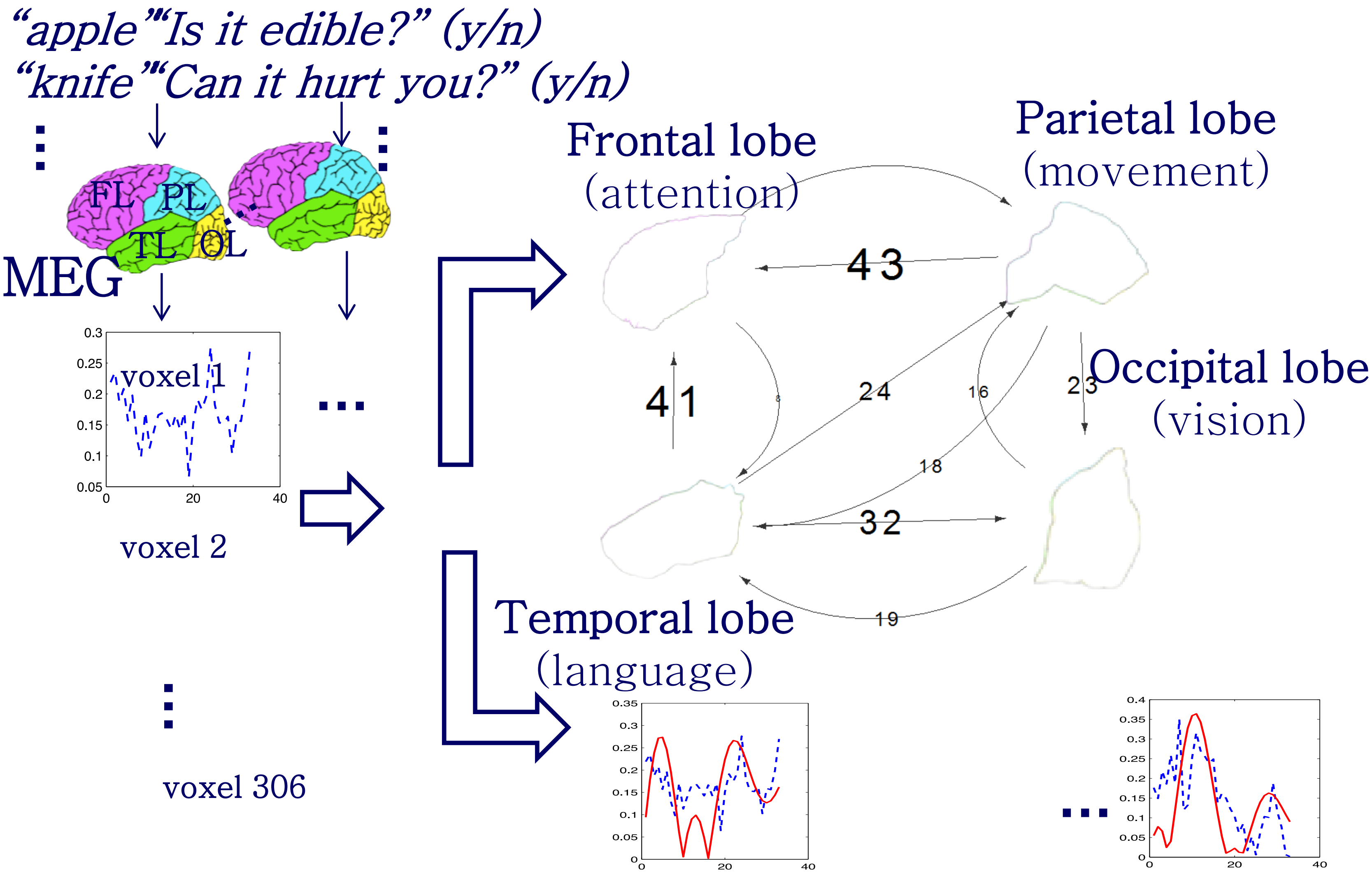
- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Future directions
- Conclusions

Brain connectivity



Carnegie Mellon

Brain connectivity



Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
- Part#3: Cascades and immunization
- Future directions
- Acknowledgements and Conclusions

Thanks

Disclaimer: All opinions are mine; not necessarily reflecting the opinions of the funding agencies

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

Project info: PEGASUS

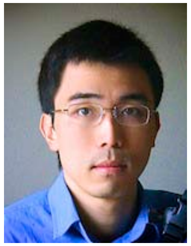
www.cs.cmu.edu/~pegasus

Results on large graphs: with Pegasus + hadoop + M45

Apache license

Code, papers, manual, video

Prof. U Kang



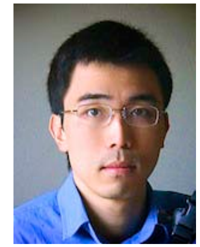
Prof. Polo Chau

Cast

Akoglu, Leman

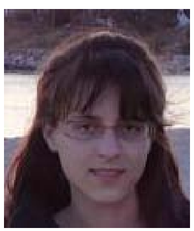
Araujo, Miguel

Beutel, Alex



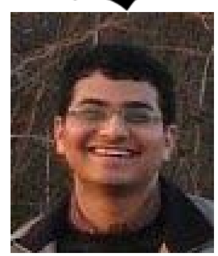
Chau, Polo

Kang, U



Koutra, Danai

Lee, Jay Yoon



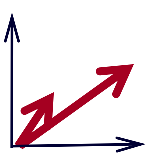
Prakash, Aditya

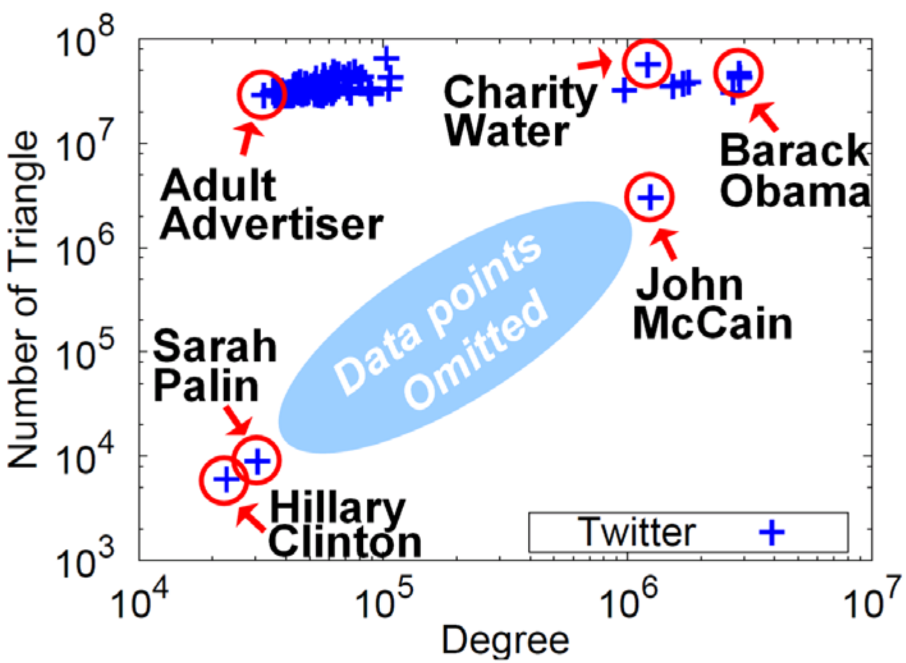
Papalexakis, Vagelis

Shah, Neil

CONCLUSION#1 – Big data

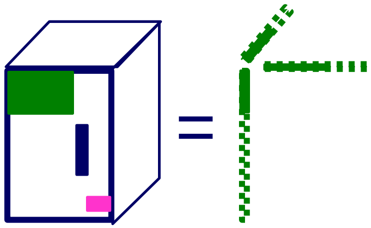
- **Large** datasets reveal patterns/outliers that are invisible otherwise





CONCLUSION#2 – tensors

- powerful tool



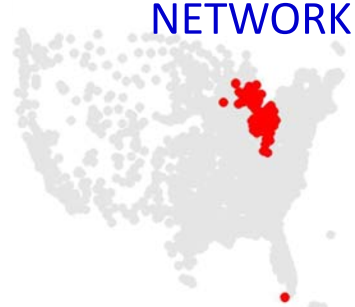
CONCLUSION#3 – eigen-drop

- Cascades & immunization: G2 theorem &**eigenvalue**

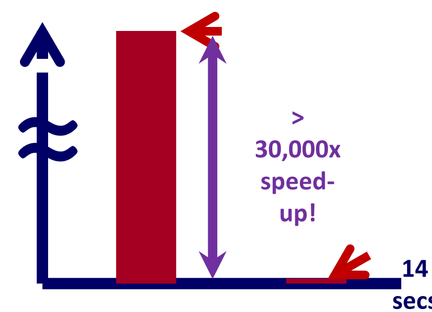
~6x fewer!

CURRENT PRACTICE

[US-MEDICARE NETWORK 2005]

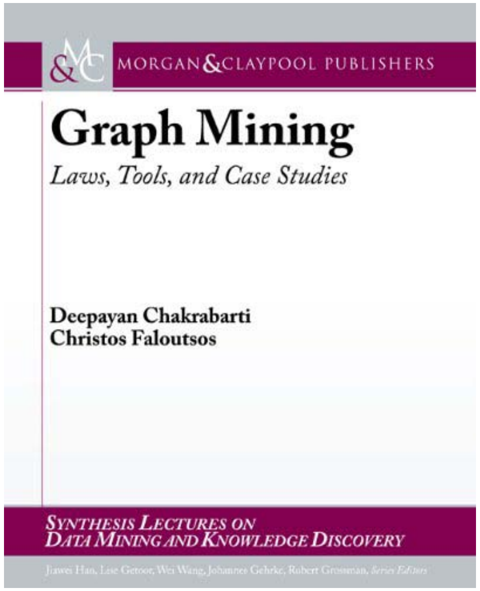


OUR METHOD



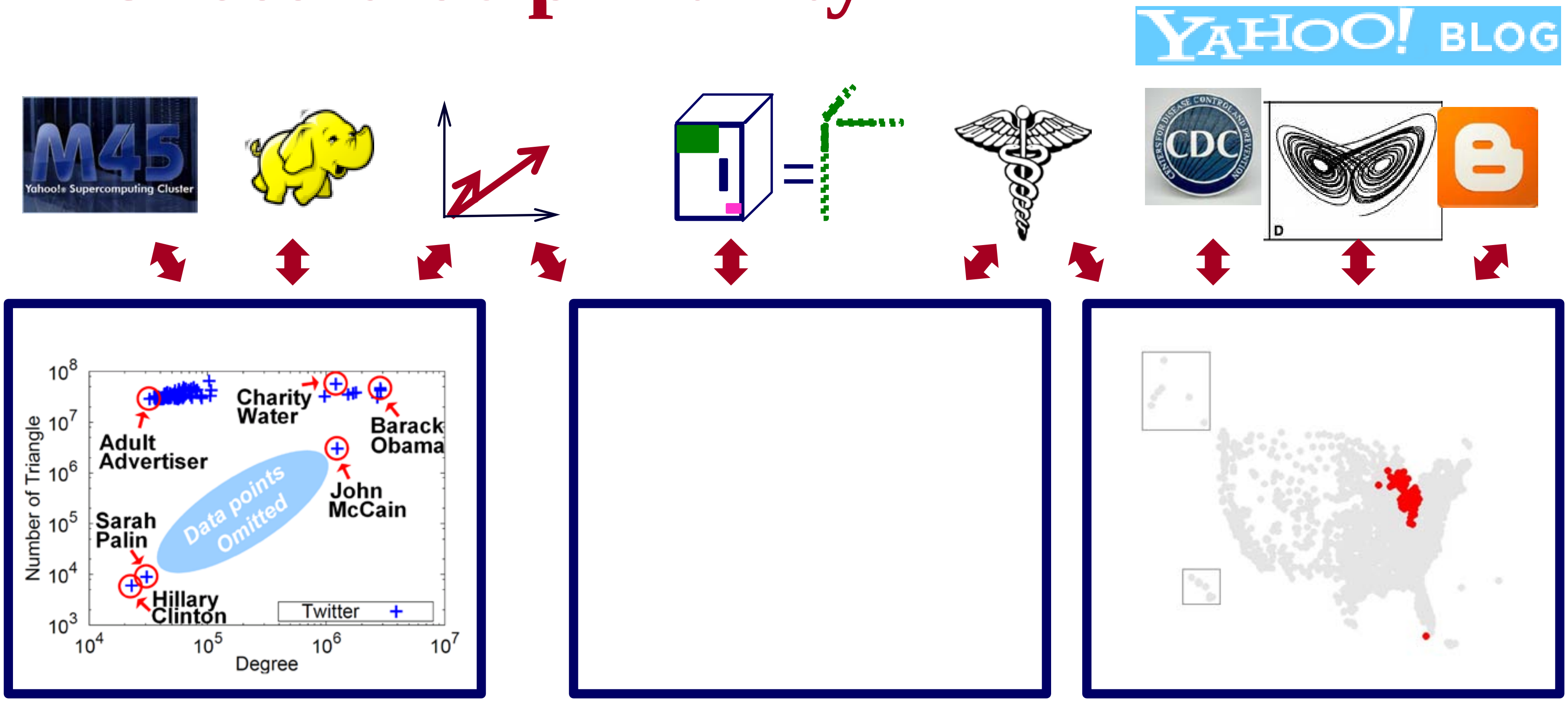
References

- D. Chakrabarti, C. Faloutsos: *Graph Mining – Laws, Tools and Case Studies*, Morgan Claypool 2012
- http://www.morganclaypool.com/doi/abs/10.2200/S00449ED1V01Y201209DMK006



TAKE HOME MESSAGE:

Cross-disciplinarity



Thank you! Questions?

Cross-disciplinarity