Carnegie Mellon

Anomaly detection in large graphs

Christos Faloutsos

CMU

www.cs.cmu.edu/~christos/TALKS/17-06-22-tencent/faloutsos_tencent_2017.pdf

Carnegie Mellon

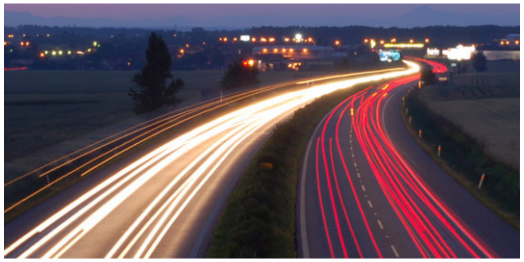
Thank you!

- Annette Jiang (IEEE)

- Evan Butterfield (IEEE)
- Tina Huang (Tencent)

Carnegie Mellon

Roadmap

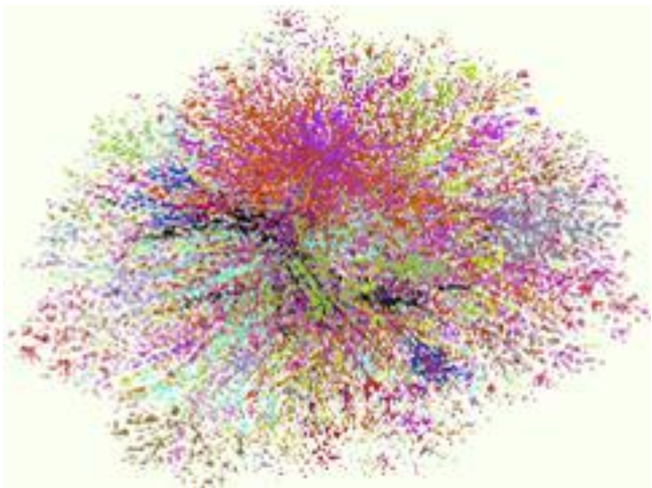


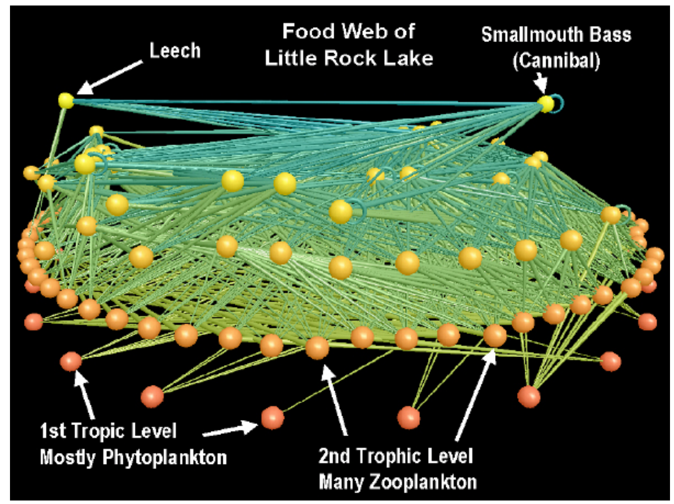
- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs; tensors
- Conclusions

Graphs - why should we care?

>$10B; ~1B users

Graphs - why should we care?





Internet Map
[lumeta.com]

Food Web
[Martinez '91]

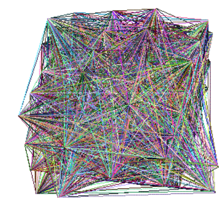
Graphs - why should we care?

- web-log ('blog') news propagation
- computer network security: email/IP traffic and anomaly detection
- Recommendation systems
- Who-bought-from-whom (ebay, Alibaba)
- ....

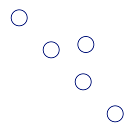
Many-to-many db relationship -> graph

Carnegie Mellon

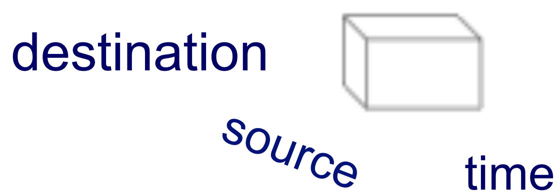
Motivating problems



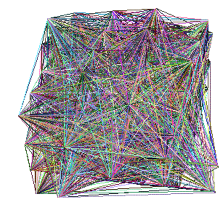
- P1: patterns? Fraud detection?



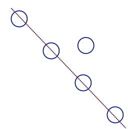
- P2: patterns in time-evolving graphs / tensors



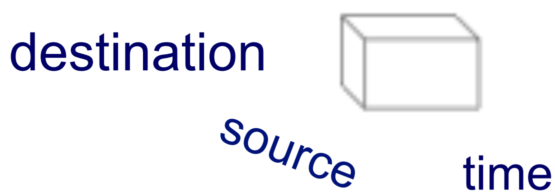
Motivating problems



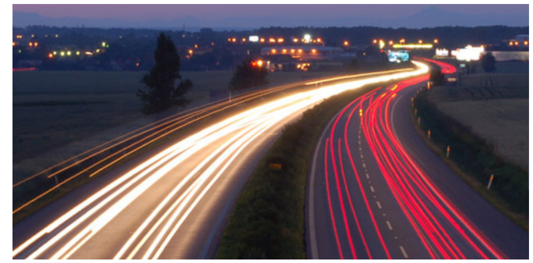
- P1: patterns? Fraud detection?

 Patterns  anomalies

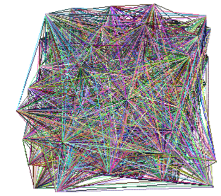
- P2: patterns in time-evolving graphs / tensors



Roadmap

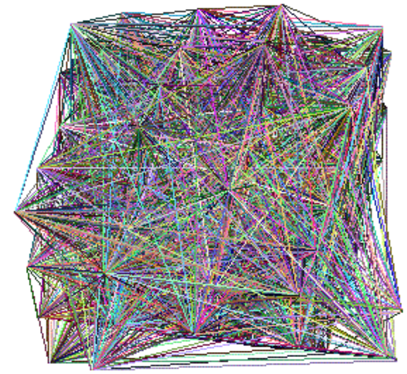


- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns & fraud detection
- Part#2: time-evolving graphs; tensors
- Conclusions



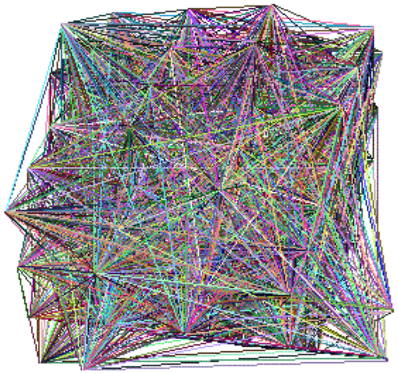
Part 1: Patterns, & fraud detection

Laws and patterns

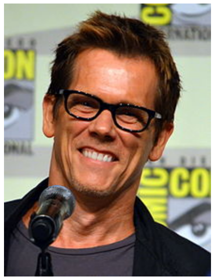


- Q1: Are real graphs random?

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



Carnegie Mellon

Solution# S.1

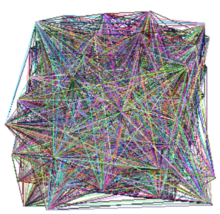
- Power law in the degree distribution [Faloutsos x 3 SIGCOMM99]

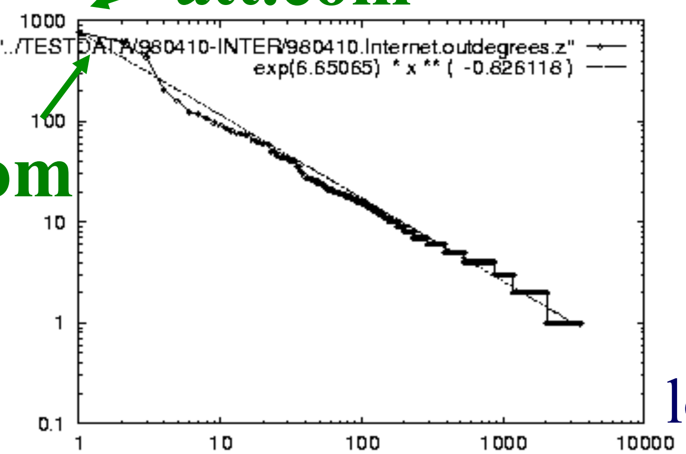
internet domains

att.com

log(degree)

ibm.com



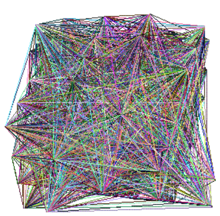


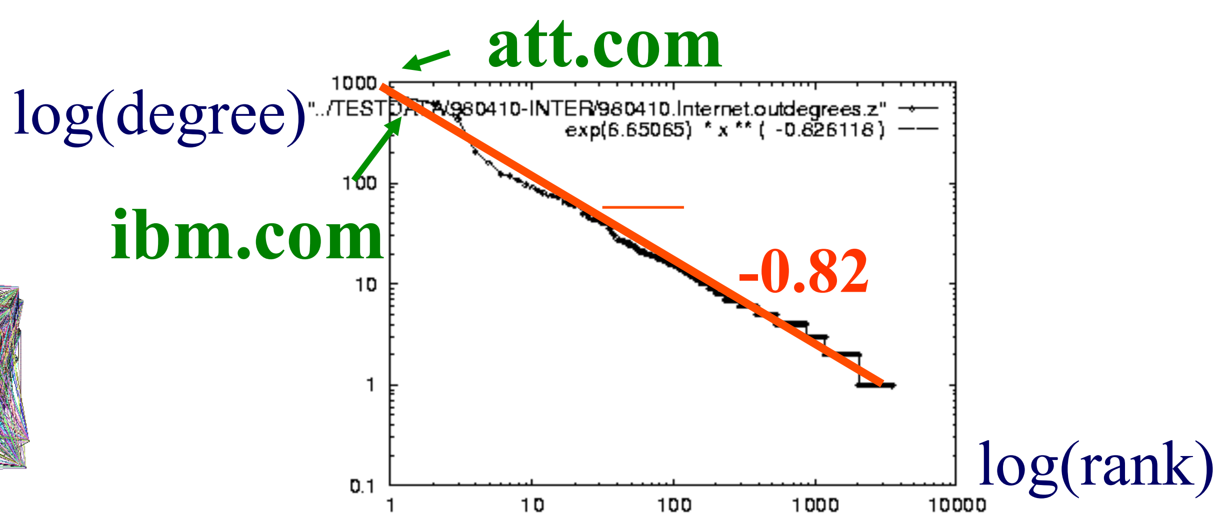
log(rank)

Solution# S.1

- Power law in the degree distribution [Faloutsos x 3 SIGCOMM99]

internet domains



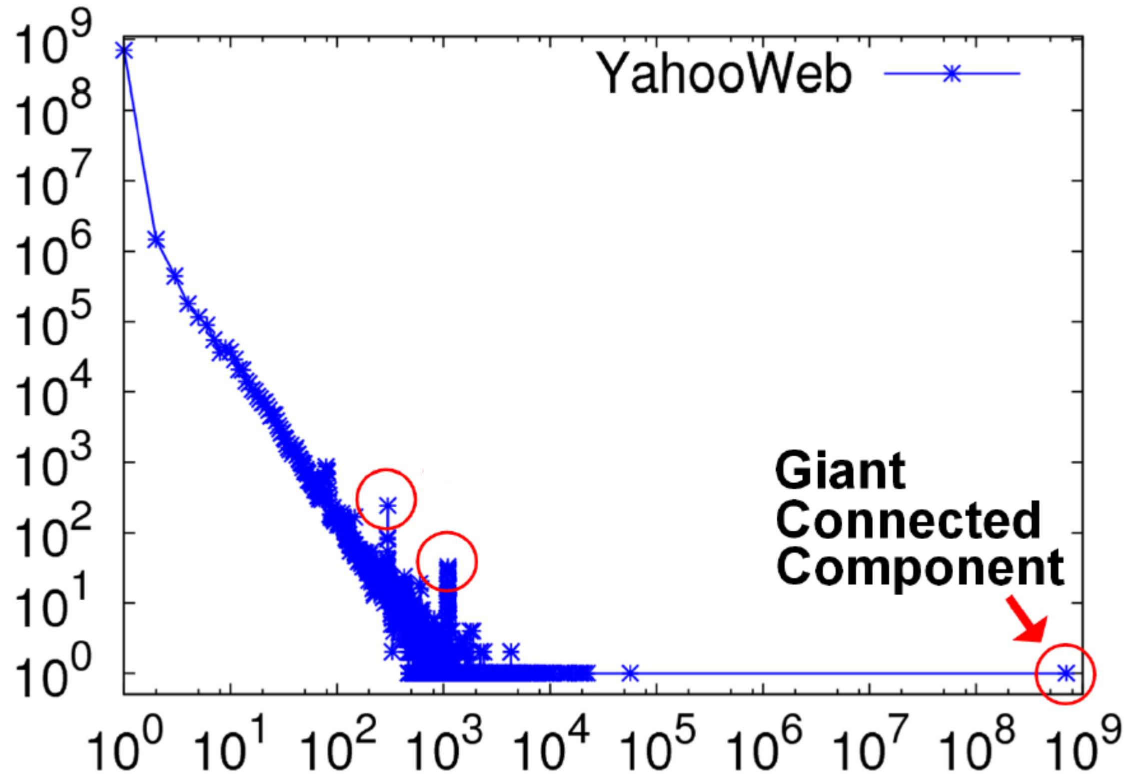


Carnegie Mellon

S2: connected component sizes

- Connected Components – 4 observations:

Count



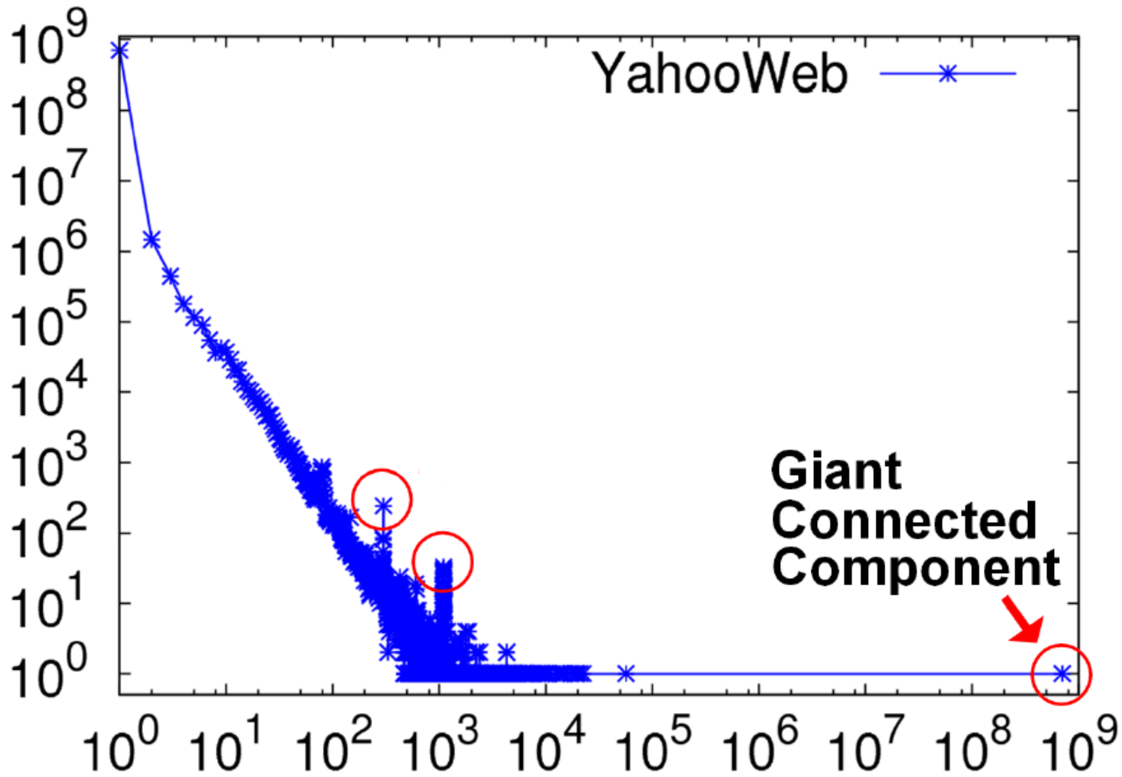
Size

1.4B nodes
6B edges

S2: connected component sizes

- Connected Components

Count



Size

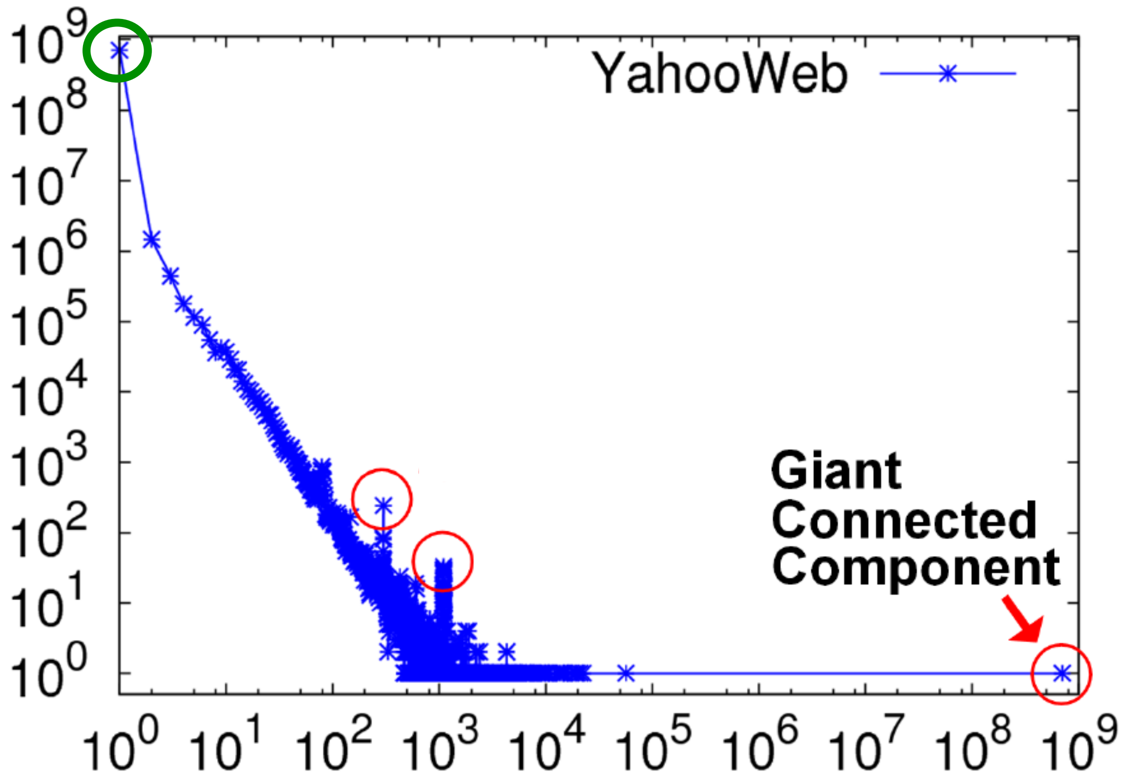
1) 10K x larger than next

S2: connected component sizes

- Connected Components

Count

2) ~0.7B singleton nodes



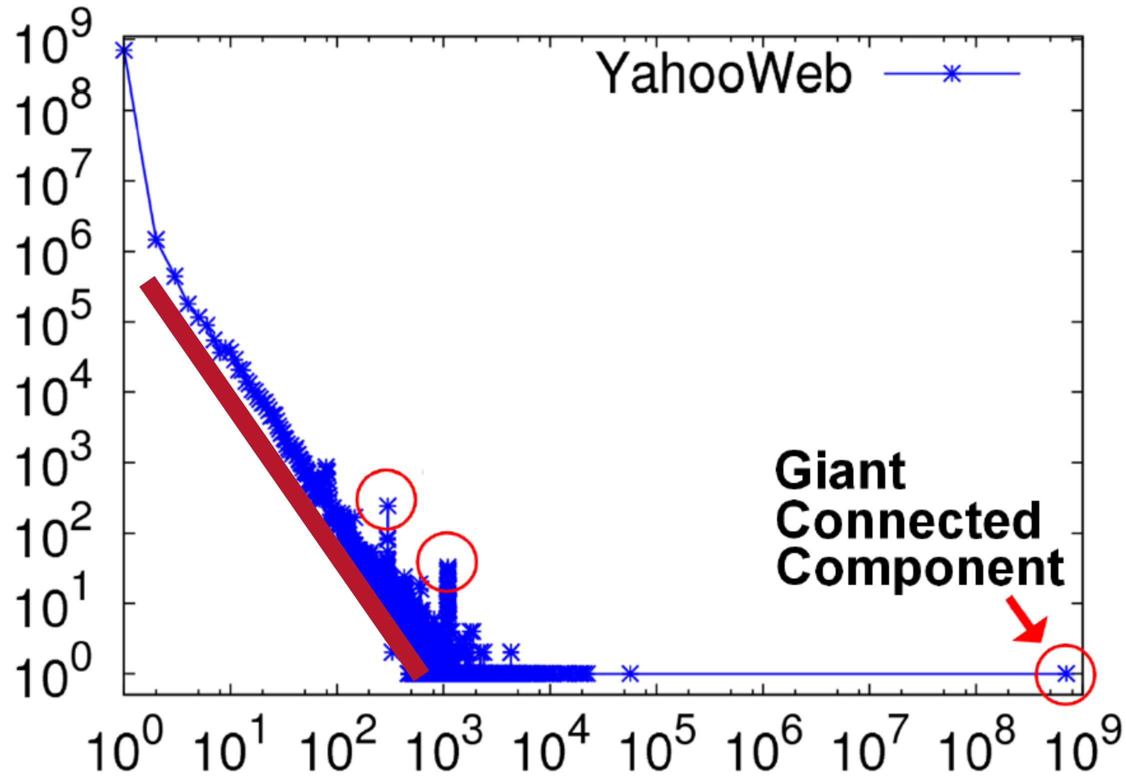
Size

S2: connected component sizes

- Connected Components

Count

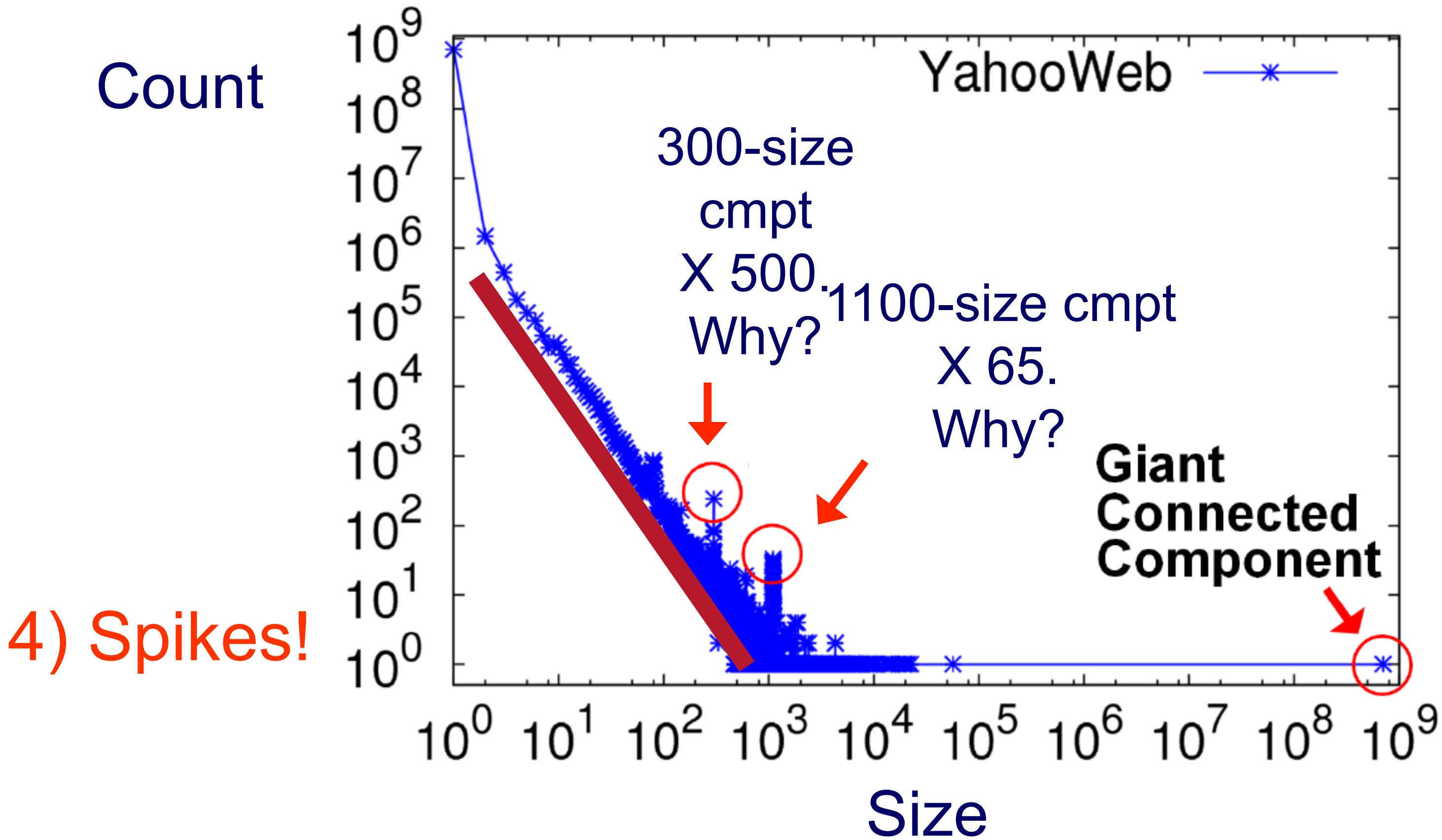
3) SLOPE!



Size

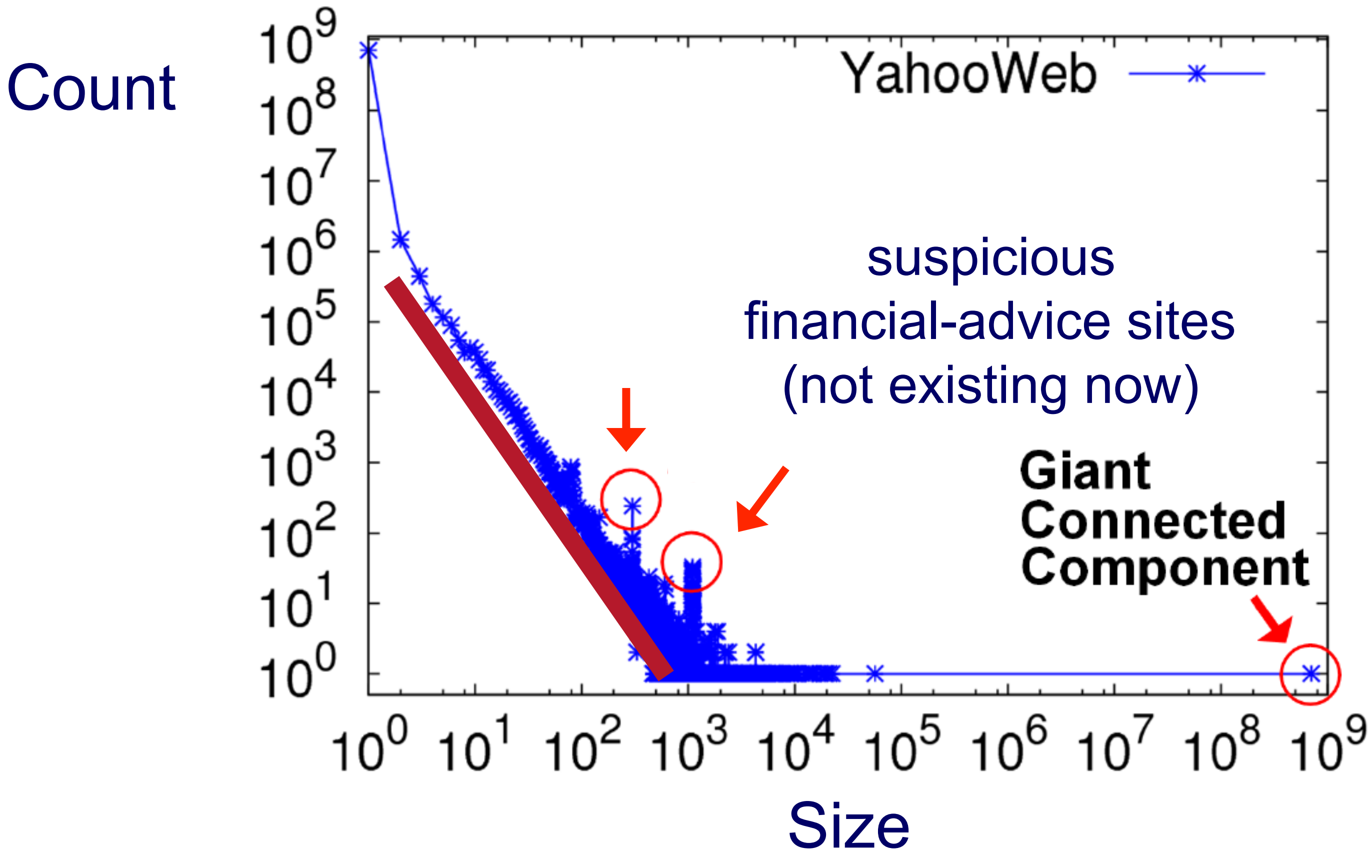
S2: connected component sizes

- Connected Components



S2: connected component sizes

- Connected Components



Carnegie Mellon

MORE Graph Patterns

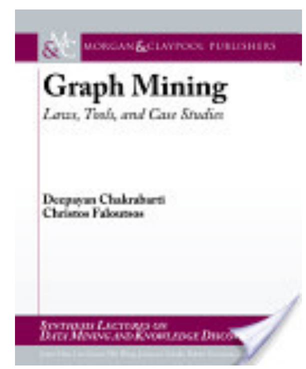
	Unweighted	Weighted
Static	**L01.** Power-law degree distribution [Faloutsos et al. \`99, Kleinberg et al. \`99, Chakrabarti et al. \`04, Newman \`04] **L02.** Triangle Power Law (TPL) [Tsourakakis \`08] **L03.** Eigenvalue Power Law (EPL) [Siganos et al. \`03] **L04.** Community structure [Flake et 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

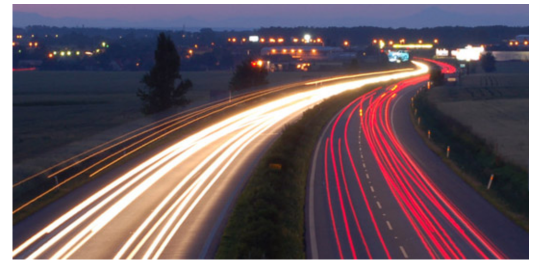
	Unweighted	Weighted
Static	L01. Power-law degree distribution [Faloutsos et al. `99, Kleinberg et al. `99, Chakrabarti et al. `04, Newman `04] L02. Triangle Power Law (TPL) [Tsourakakis `08] L03. Eigenvalue Power Law (EPL) [Siganos et al. `03] L04. Community structure [Flake et al. `02, Girvan and Newman `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.: Charu Aggarwal)



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

Roadmap

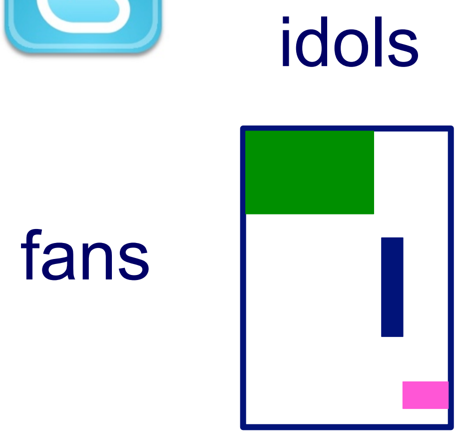


- Introduction – Motivation
- Part#1: Patterns in graphs
  - P1.1: Patterns
  - P1.2: Anomaly / fraud detection
    - No labels – spectral
    - With labels: Belief Propagation
- Part#2: time-evolving graphs; tensors
- Conclusions

Patterns  anomalies

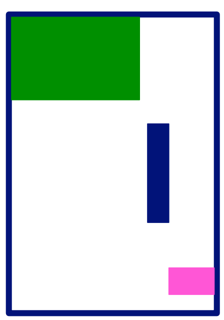
How to find 'suspicious' groups?

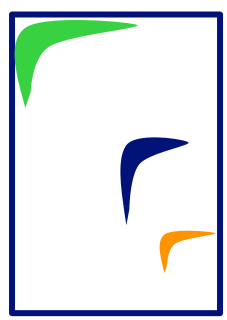
- 'blocks' are normal, right?



Except that:

- ‘blocks’ are normal, right?
- ‘hyperbolic’ communities are more realistic [Araujo+, PKDD’14]

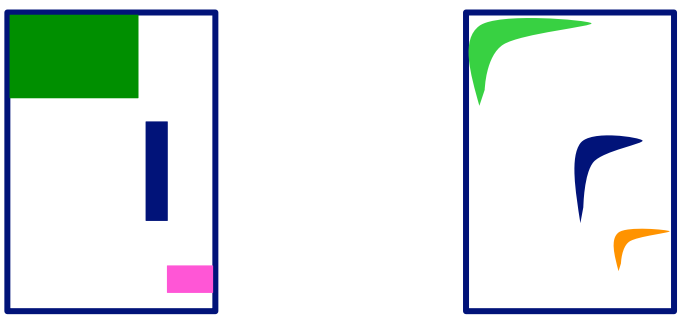




Except that:

- ‘blocks’ are usually suspicious
- ‘hyperbolic’ communities are more realistic [Araujo+, PKDD’14]

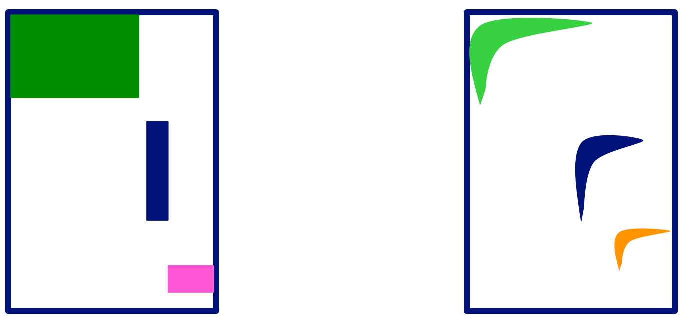
Q: Can we spot blocks, easily?



Except that:

- ‘blocks’ are usually suspicious
- ‘hyperbolic’ communities are more realistic [Araujo+, PKDD’14]

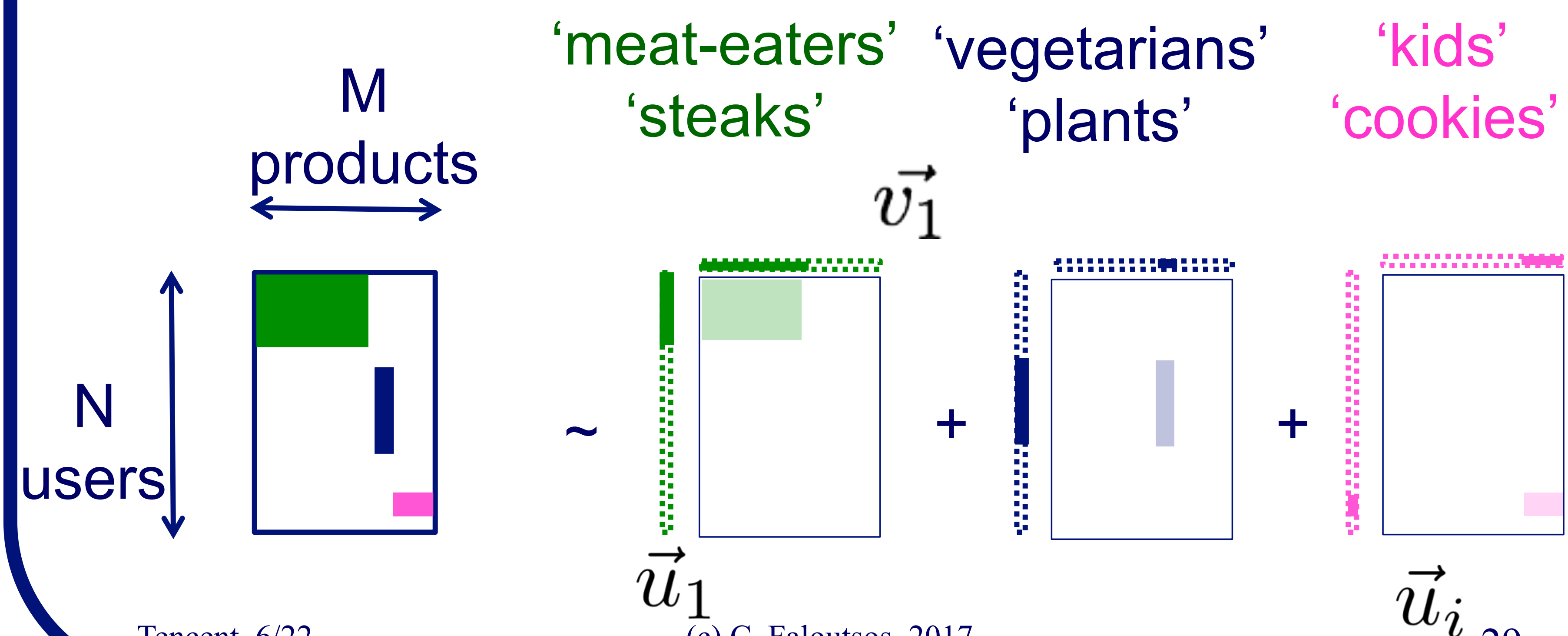
Q: Can we spot blocks, easily?
A: Silver bullet: SVD!



DETAILS

Crush intro to SVD

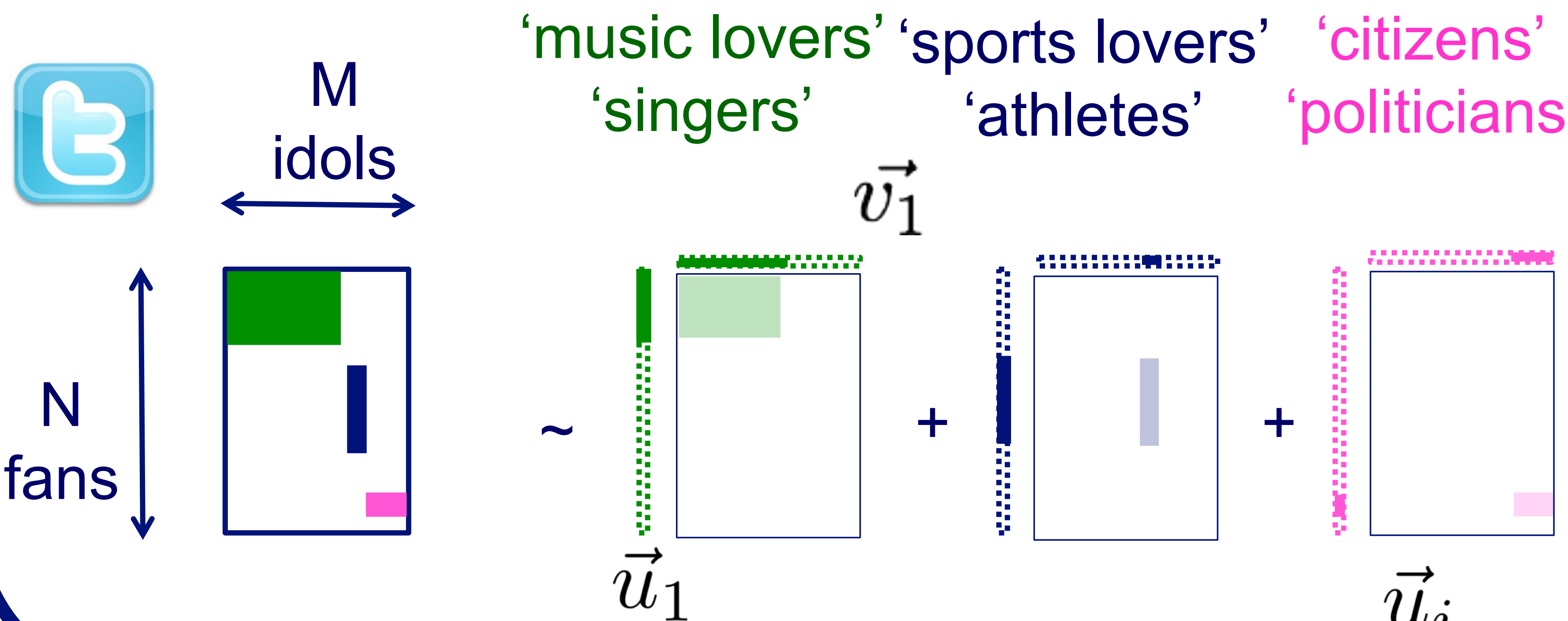
- Recall: (SVD) matrix factorization: finds blocks



DETAILS

Crush intro to SVD

- Recall: (SVD) matrix factorization: finds blocks

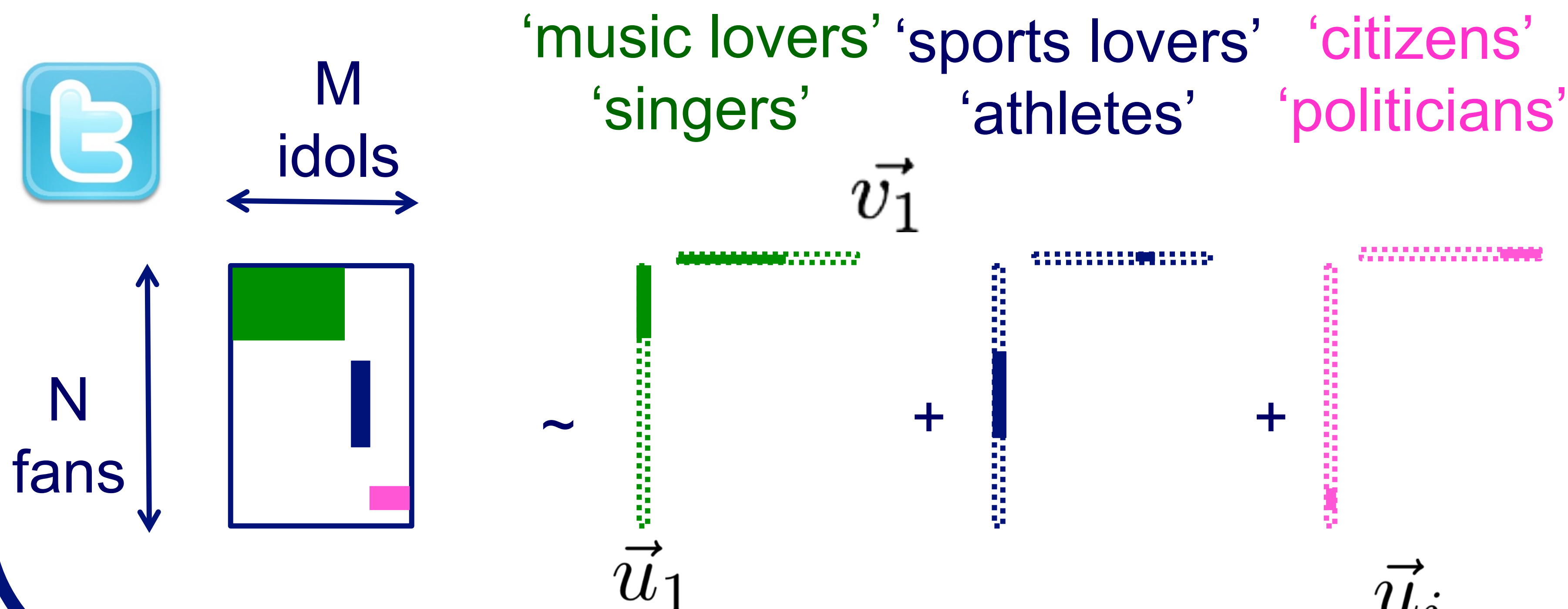


$\vec{v}_1$ $\vec{u}_1$ $\vec{u}_i$

DETAILS

Crush intro to SVD

- Recall: (SVD) matrix factorization: finds blocks



Carnegie Mellon

Inferring Strange Behavior from Connectivity Pattern in Social Networks

PAKDD'14



Meng Jiang, Peng Cui, Shiqiang Yang (Tsinghua)

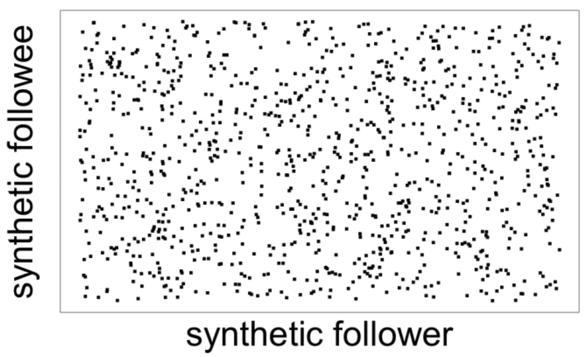
Alex Beutel, Christos Faloutsos (CMU)

Carnegie Mellon

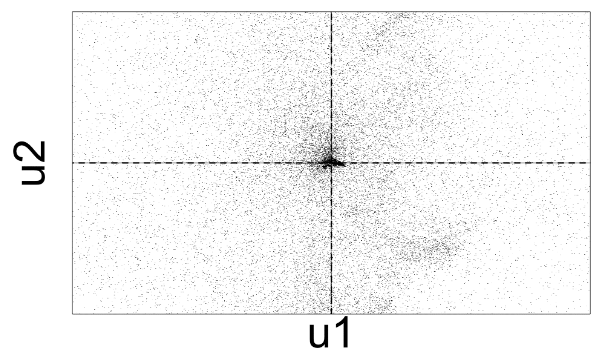
Lockstep and *Spectral Subspace Plot*

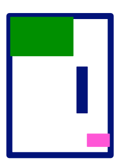
- Case #0: No lockstep behavior in random power law graph of 1M nodes, 3M edges
- Random ⟷ "Scatter"

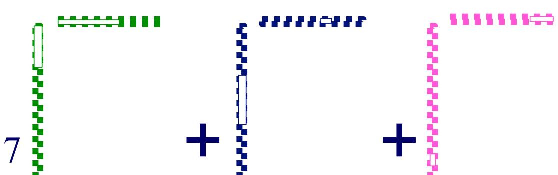
Adjacency Matrix



Spectral Subspace Plot



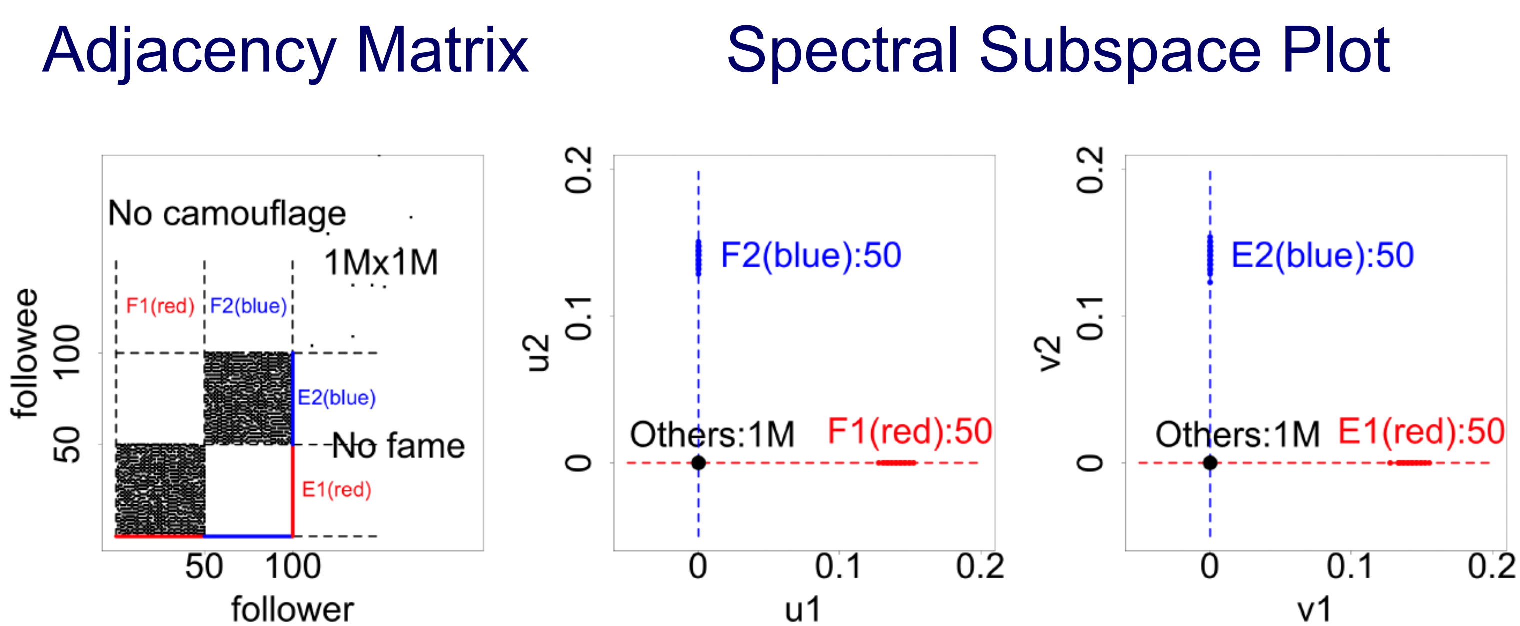




Carnegie Mellon

Lockstep and *Spectral Subspace Plot*

- Case #1: non-overlapping lockstep
- "Blocks" ⟷ "Rays"

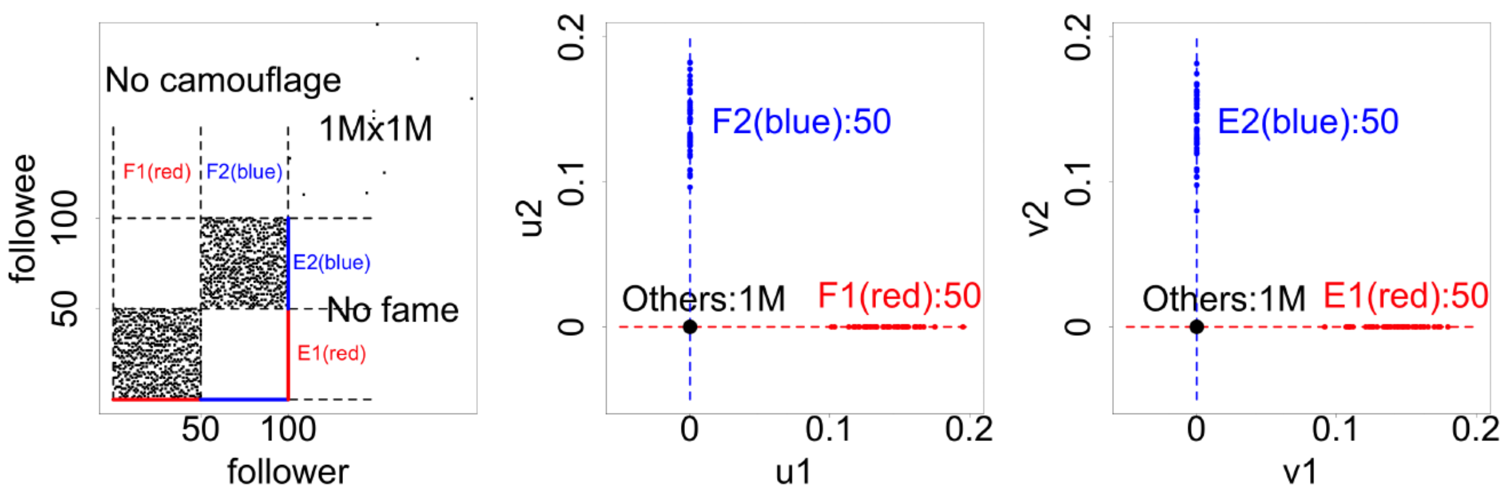


Rule 1 (short "rays"): two blocks, high density (90%), no "camouflage", no "fame"

Lockstep and *Spectral Subspace Plot*

- Case #2: non-overlapping lockstep
- "Blocks; low density" ⟷ Elongation

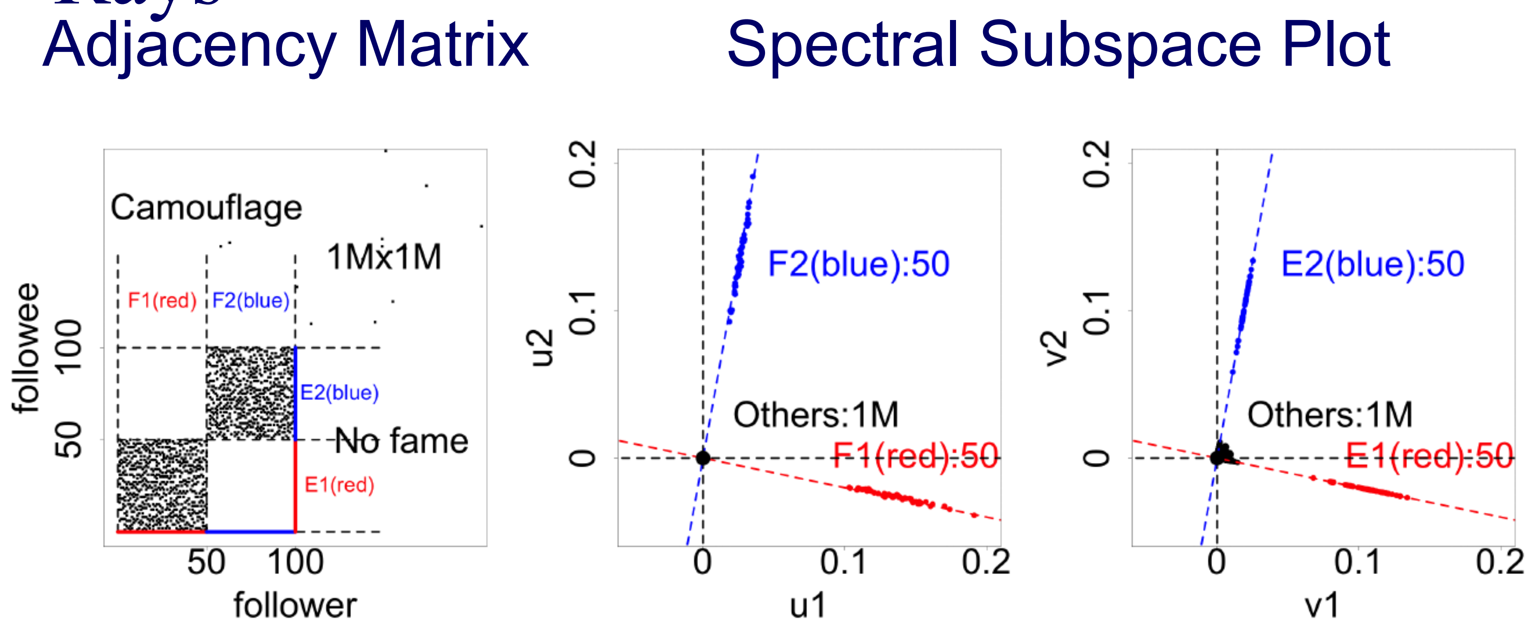
Adjacency Matrix — Spectral Subspace Plot



Rule 2 (long "rays"): two blocks, low density (50%), no "camouflage", no "fame"

Lockstep and *Spectral Subspace Plot*

- Case #3: non-overlapping lockstep
- "**Camouflage**" (or "Fame") ⟷ Tilting "Rays"

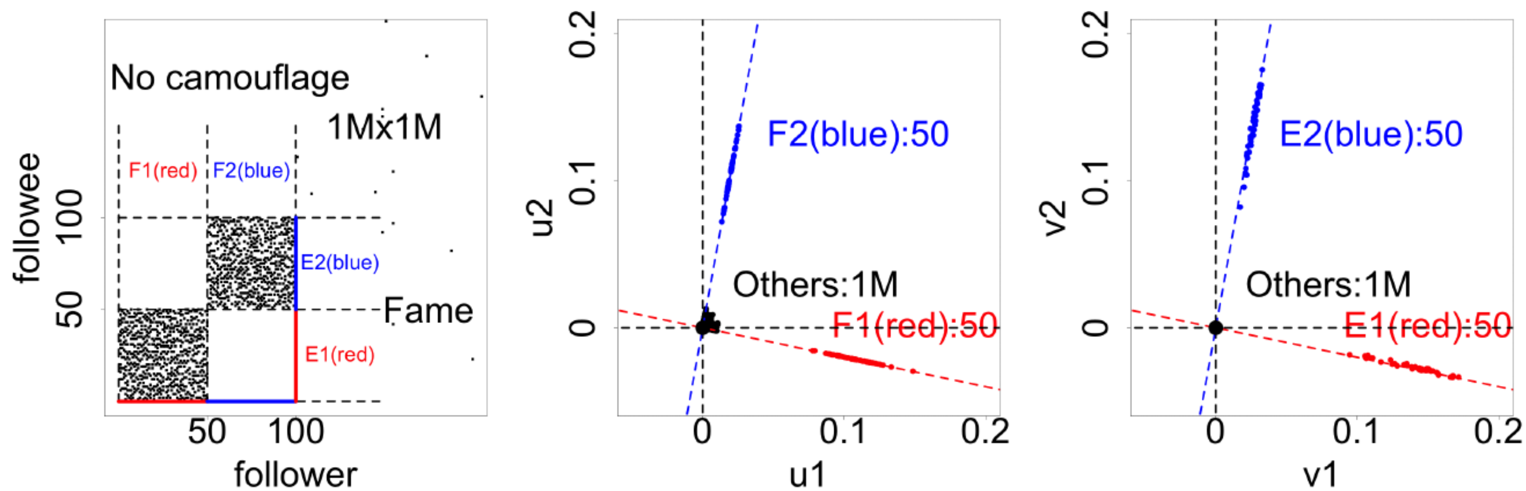


Rule 3 (tilting "rays"): two blocks, with "camouflage", no "fame"

Lockstep and *Spectral Subspace Plot*

- Case #3: non-overlapping lockstep
- "Camouflage" (or "**Fame**") ⟷ Tilting "Rays"

Adjacency Matrix / Spectral Subspace Plot



Rule 3 (tilting "rays"): two blocks, no "camouflage", with "fame"

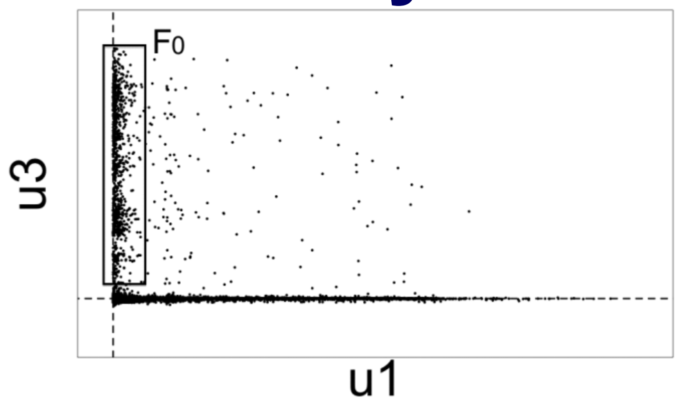
Carnegie Mellon

Dataset

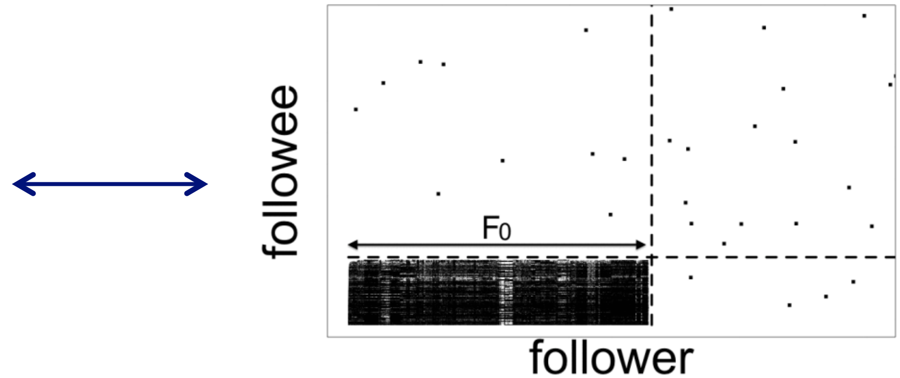
- Tencent Weibo
- 117 million nodes (with profile and UGC data)
- 3.33 billion directed edges

Real Data

"Rays"

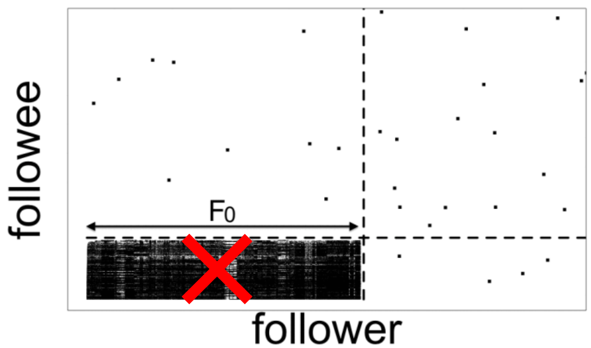


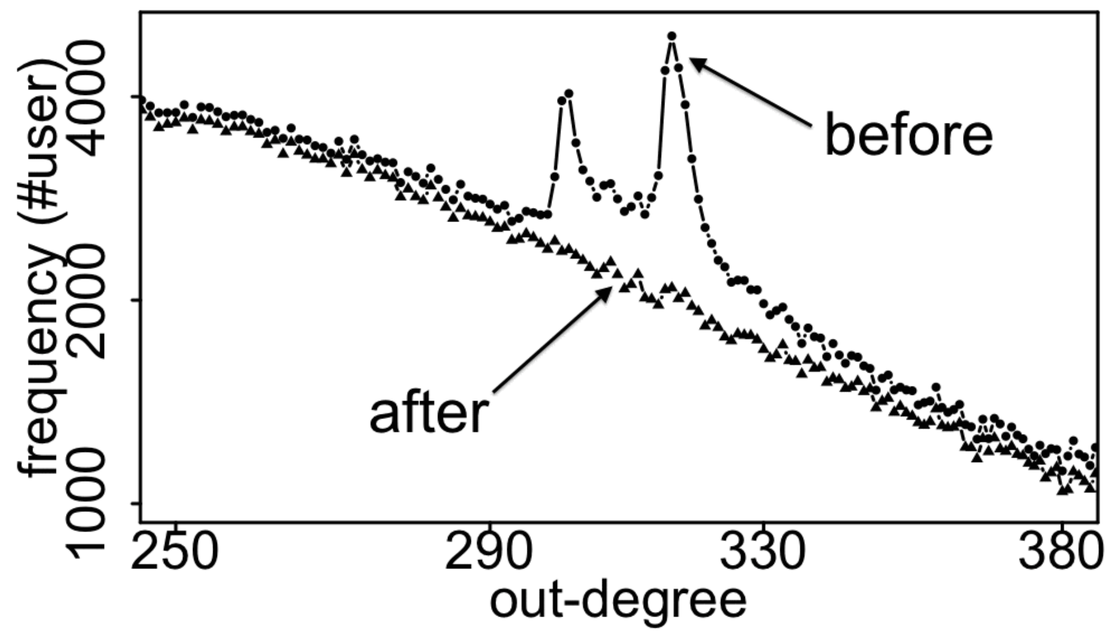
"Block"



Real Data

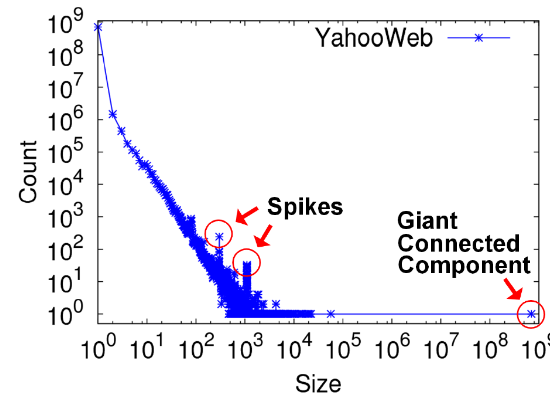
- Spikes on the out-degree distribution





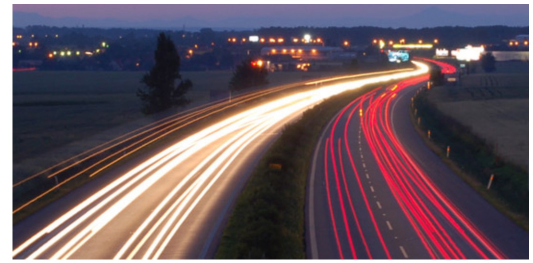
Summary of Part#1

- *many* patterns in real graphs
  - Power-laws everywhere
  - Long (and growing) list of tools for anomaly/ fraud detection



Patterns  anomalies

Roadmap



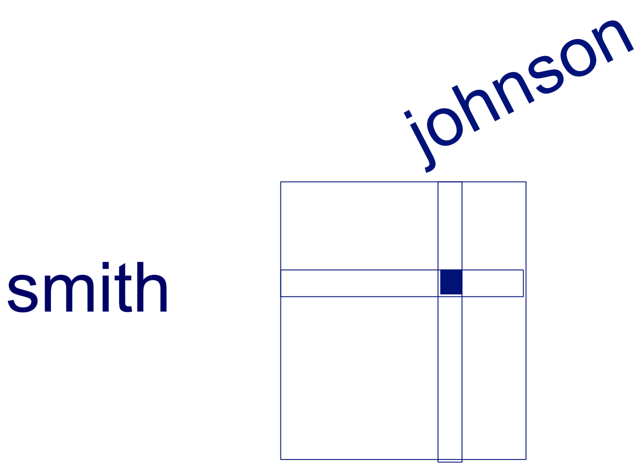
- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs
  - P2.1: tools/tensors
  - P2.2: other patterns
- Conclusions

Part 2: Time evolving graphs; tensors

Carnegie Mellon

Graphs over time -> tensors!

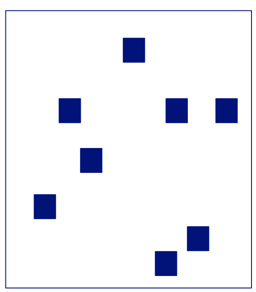
- Problem #2.1:
  - Given who calls whom, and when
  - Find patterns / anomalies



Carnegie Mellon

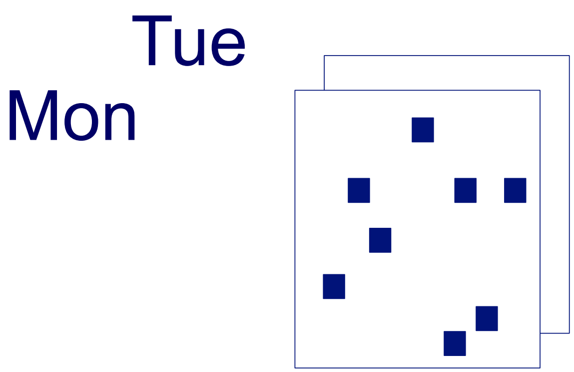
Graphs over time -> tensors!

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



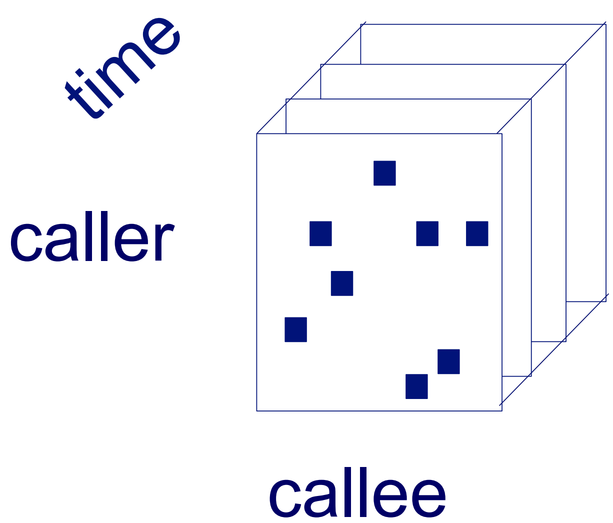
Graphs over time -> tensors!

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



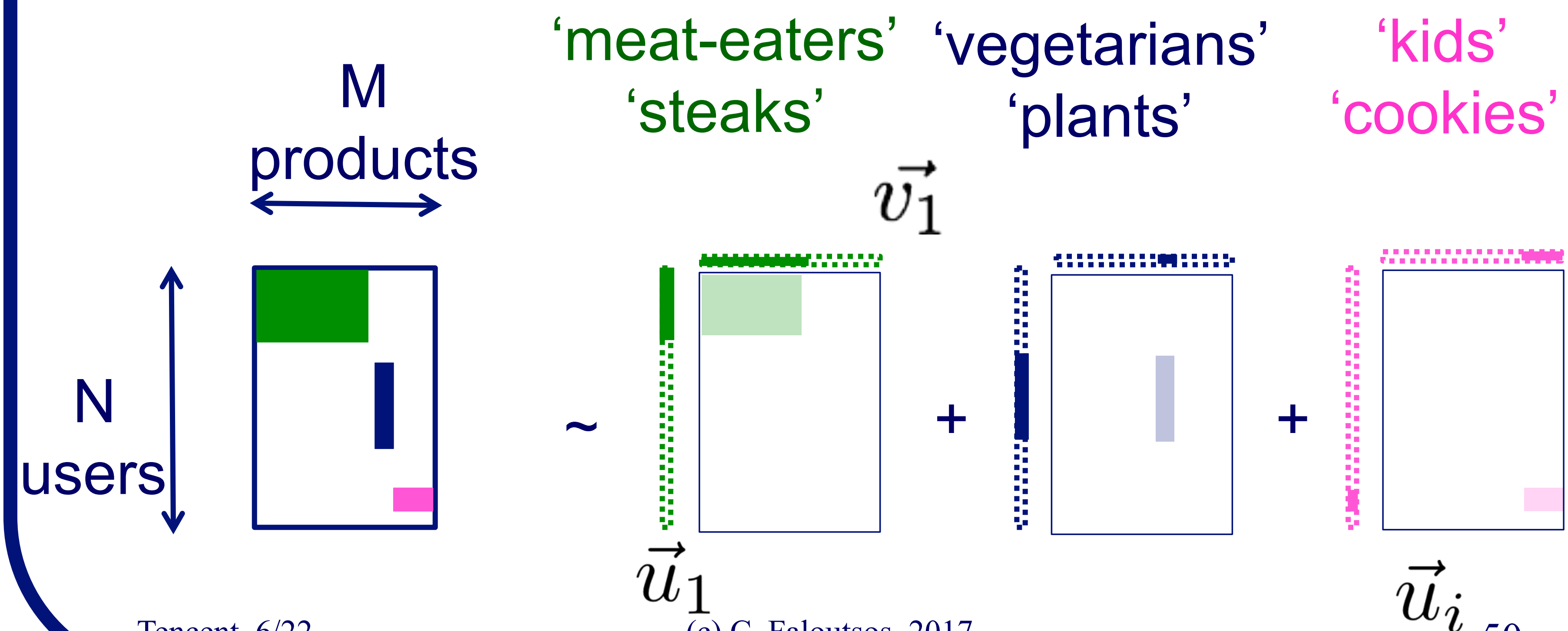
Graphs over time -> tensors!

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



Answer : tensor factorization

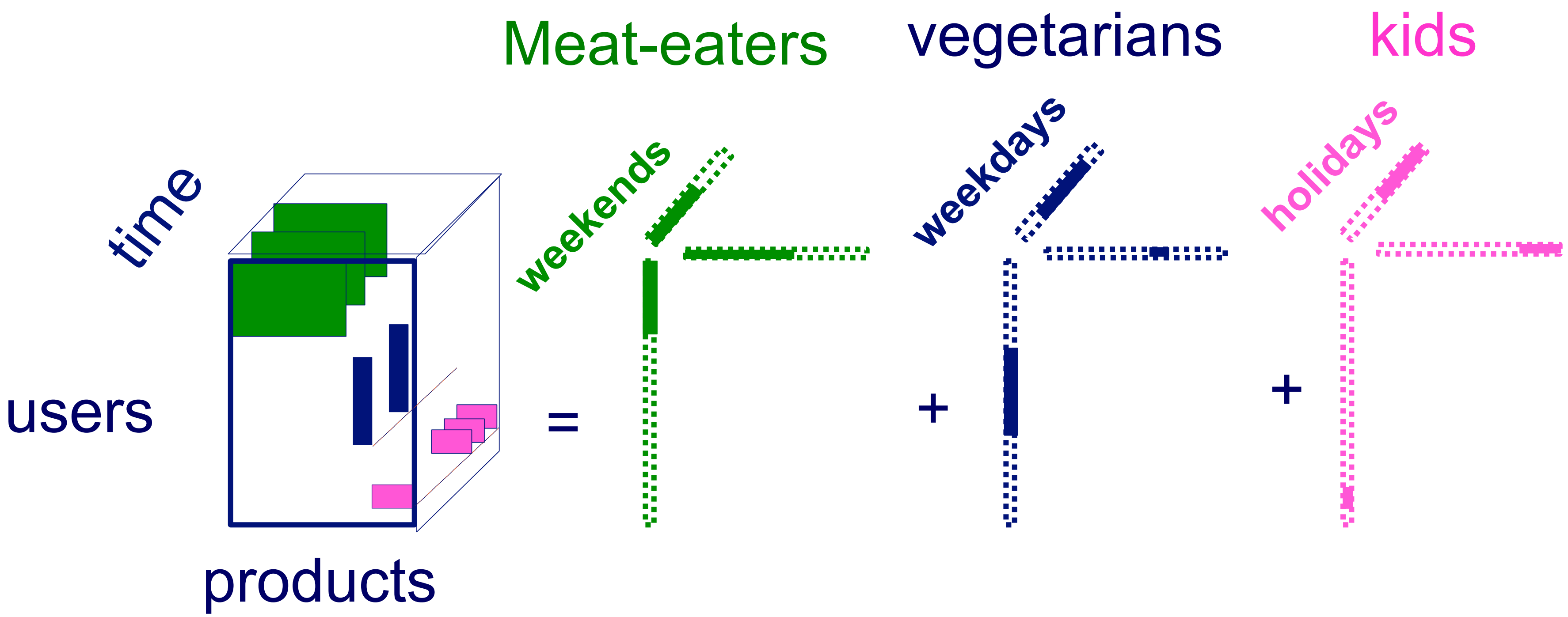
- Recall: (SVD) matrix factorization: finds blocks



Carnegie Mellon

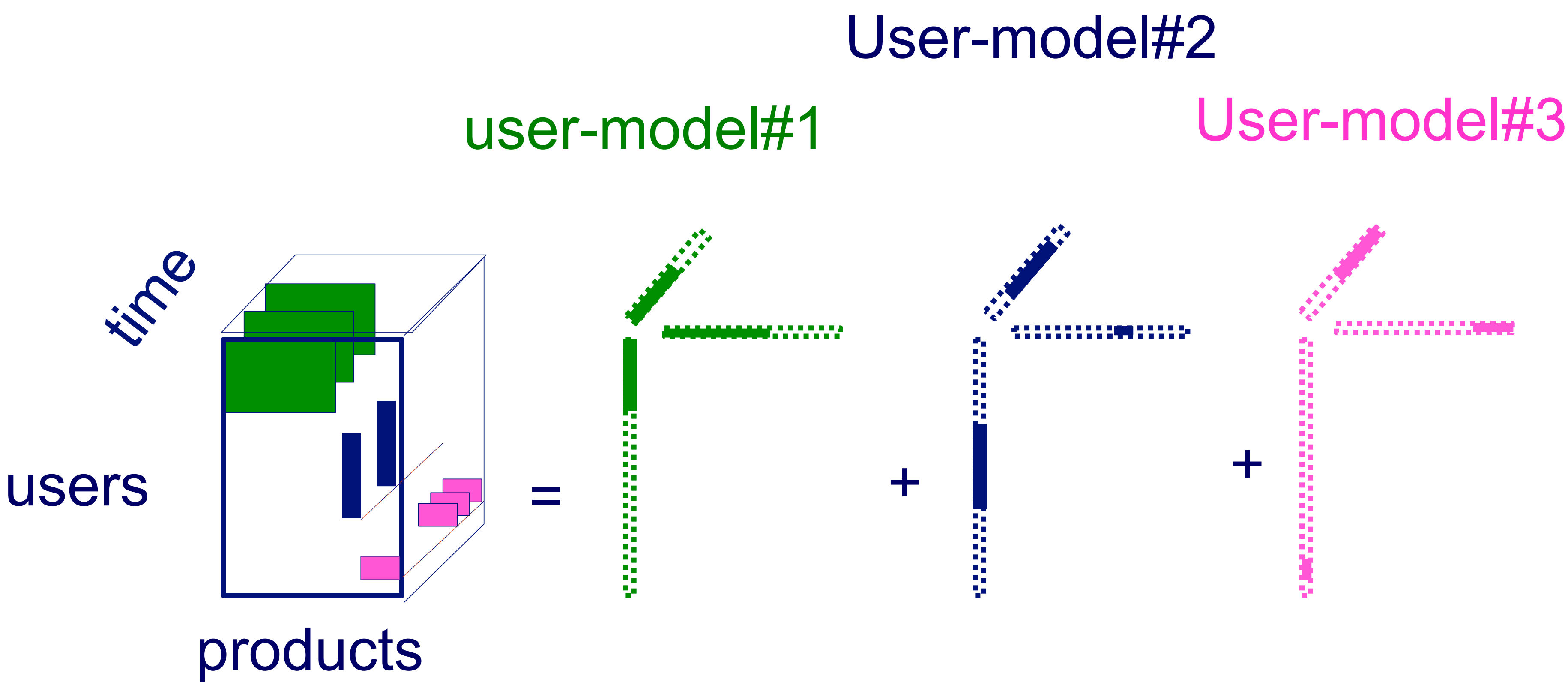
Answer: tensor factorization

- PARAFAC decomposition



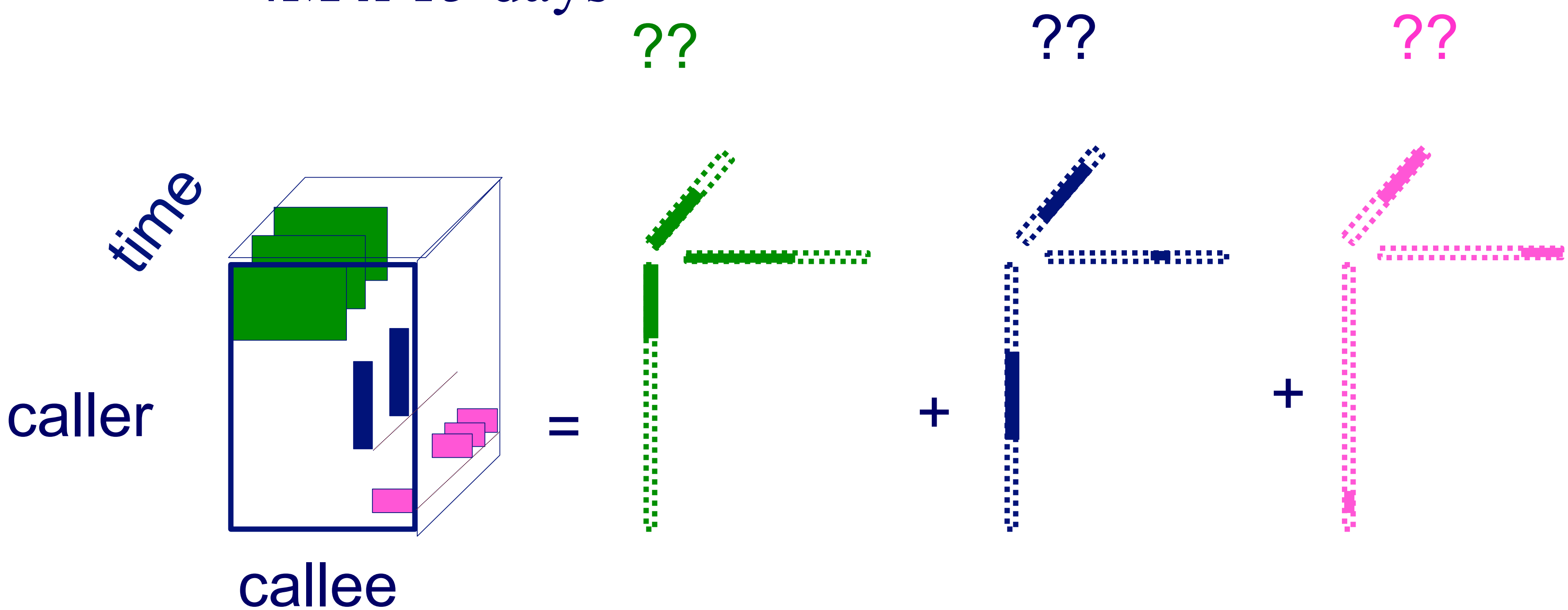
Answer: tensor factorization

- PARAFAC decomposition

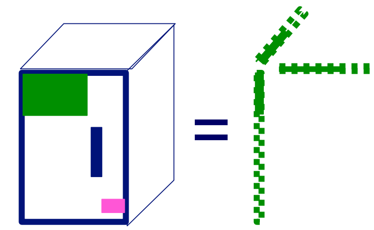


Answer: tensor factorization

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

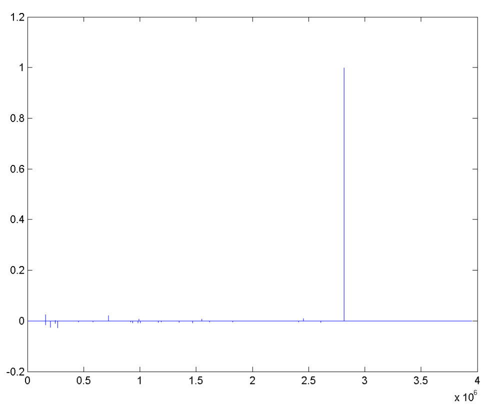


Anomaly detection in time-evolving graphs

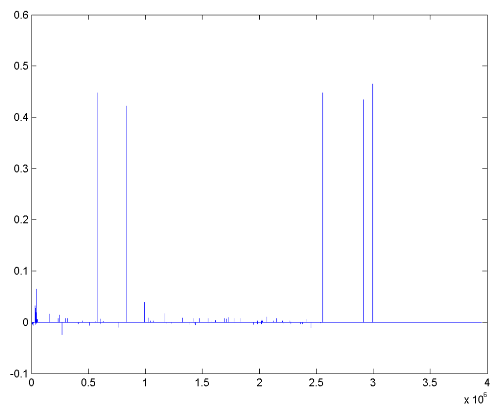


- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

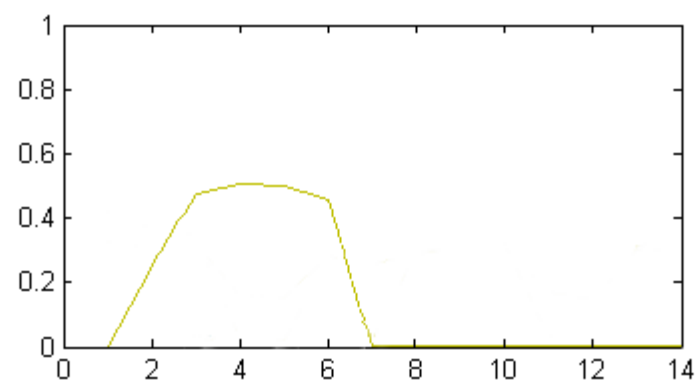
1 caller



5 receivers

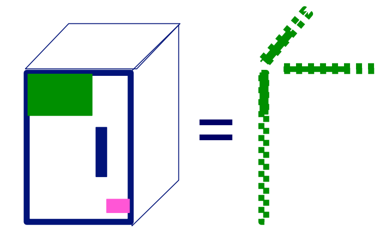


4 days of activity

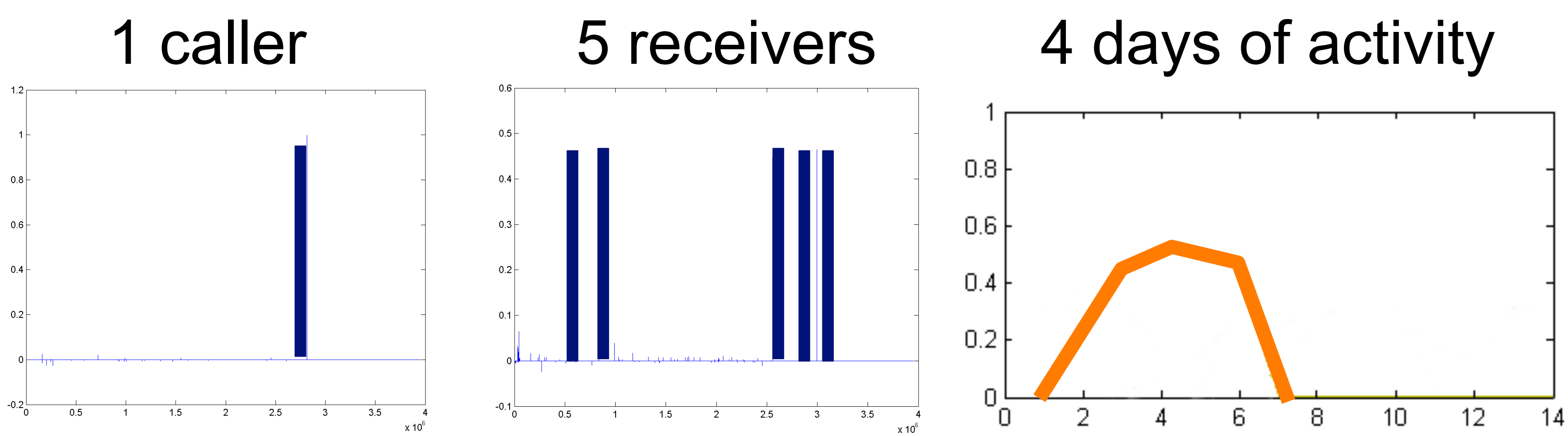


~200 calls to EACH receiver on EACH day!

Anomaly detection in time-evolving graphs

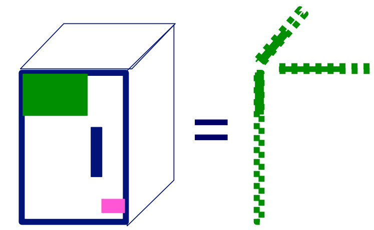


- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

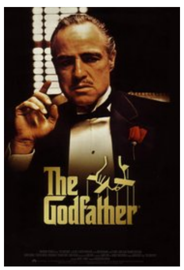


~200 calls to EACH receiver on EACH day!

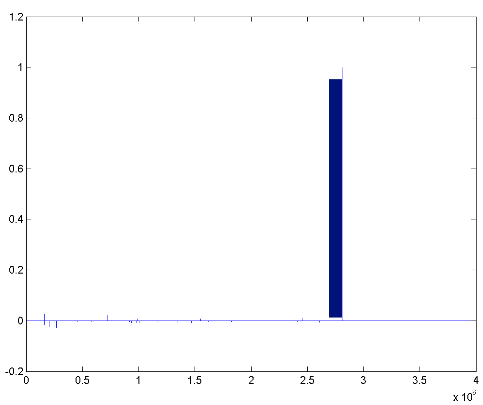
Anomaly detection in time-evolving graphs



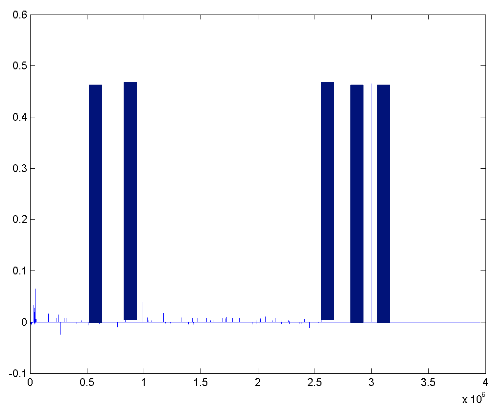
- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks



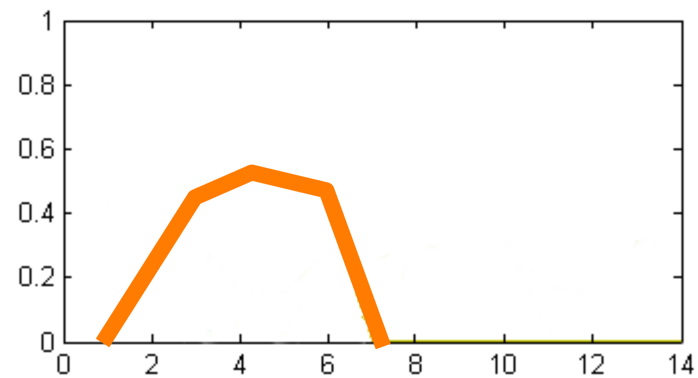
1 caller



5 receivers

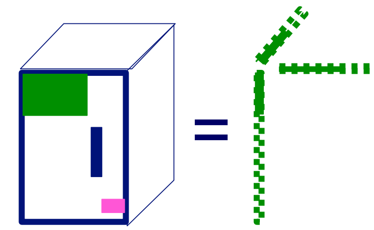


4 days of activity



~200 calls to EACH receiver on EACH day!

Anomaly detection in time-evolving graphs

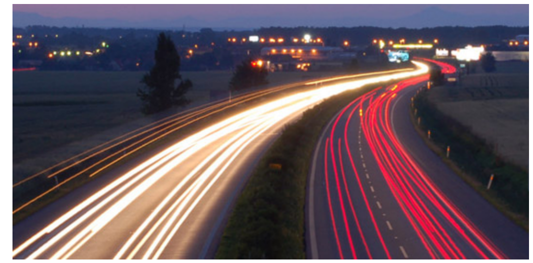


- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

Miguel Araujo, Spiros Papadimitriou, Stephan Günnemann, Christos Faloutsos, Prithwish Basu, Ananthram Swami, Evangelos Papalexakis, Danai Koutra. *Com2: Fast Automatic Discovery of Temporal (Comet) Communities*. PAKDD 2014, Tainan, Taiwan.

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs
  - P2.1: tools/tensors
  - P2.2: other patterns
    - inter-arrival time
    - ➡ Network growth
    - Group evolution
- Conclusions

Carnegie Mellon

Beyond Sigmoids: the NetTide Model for Social Network Growth and its Applications

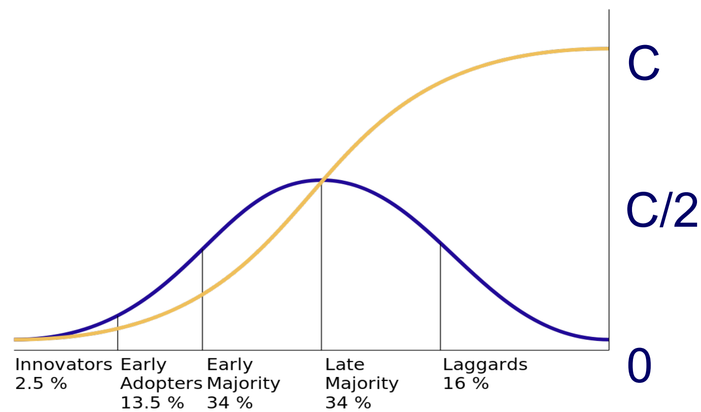
KDD'16

Chengxi Zang 臧承熙, Peng Cui, CF

PROBLEM: n(t) and e(t), over time?

- n(t): the number of nodes.
- e(t): the number of edges.
- E.g.:
  - How many members will have next month?
  - How many friendship links will have next year?

- Linear?
- Exponential?
- Sigmoid?



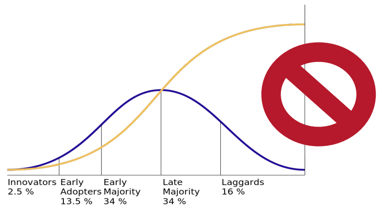
Carnegie Mellon

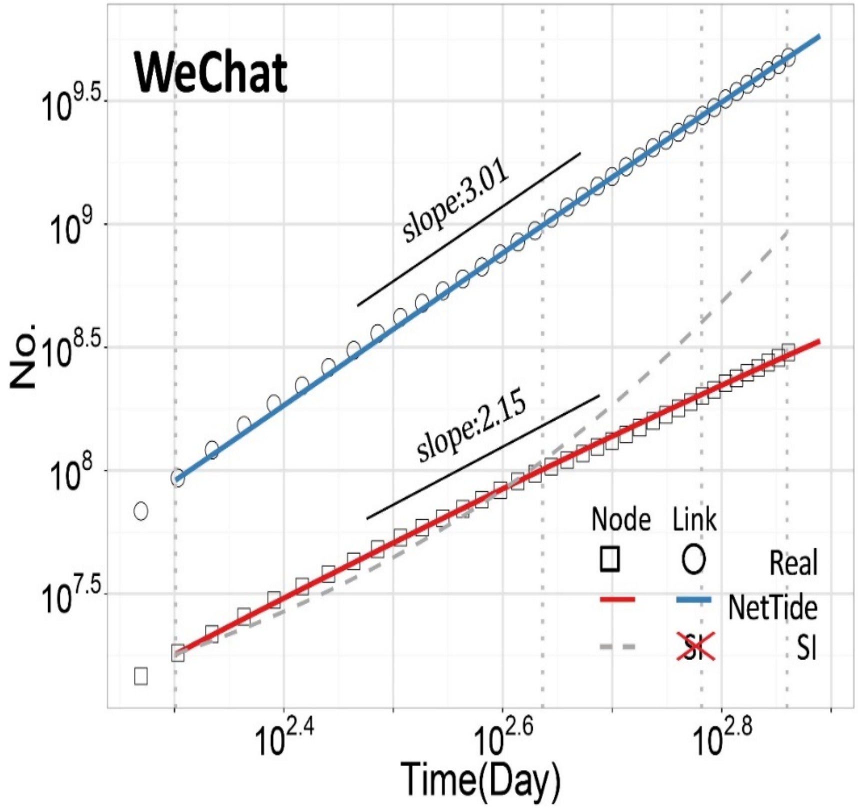
Datasets

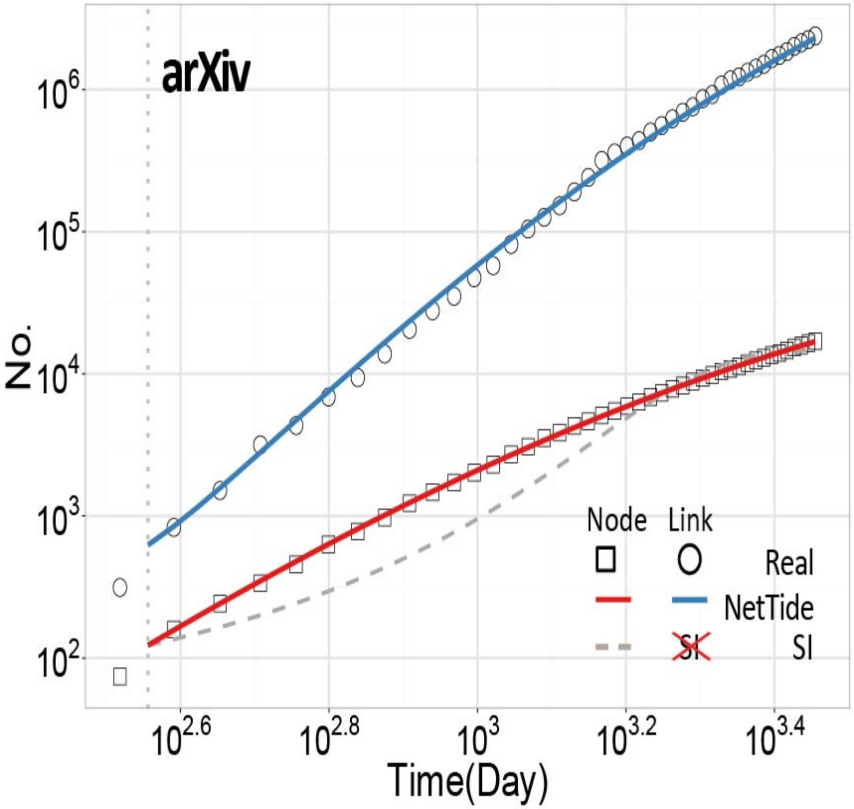
- **WeChat 2011/1-2013/1 300M nodes, 4.75B links**
- ArXiv 1992/3-2002/3 17k nodes, 2.4M links
- Enron 1998/1-2002/7 86K nodes, 600K links
- Weibo 2006 165K nodes, 331K links

Carnegie Mellon

A: Power Law Growth







Cumulative growth（Log-Log scale）

Carnegie Mellon

Details

Proposed: NetTide Model

- Nodes n(t)

$$\frac{dn(t)}{dt} = \frac{\beta}{t^{\theta}} n(t)(N - n(t))$$

- Links e(t)

$$\frac{de(t)}{dt} = \frac{\beta'}{t^{\theta}} n(t)\left(\alpha(n(t)-1)^{\gamma} - \frac{e(t)}{n(t)}\right) + 2\frac{dn(t)}{dt}$$

Details

NetTide-Node Model

$$\frac{dn(t)}{dt} = \frac{\beta}{t^{\theta}} n(t) \left(N - n(t)\right)$$

#nodes(t) → $n(t)$

Total population → N

- Intuition:
  - **Rich-get-richer**
  - Limitation
  - Fizzling nature

} = SI; ~Bass

Details

NetTide-Node Model

$$\frac{dn(t)}{dt} = \frac{\beta}{t^{\theta}} n(t) \left(N - n(t)\right)$$

#nodes(t)

Total population

- Intuition:
  - Rich-get-richer
  - **Limitation**
  - Fizzling nature

} = SI; ~Bass

Carnegie Mellon

Details

NetTide-Node Model

$$\frac{dn(t)}{dt} = \frac{\beta}{t^{\theta}} n(t) \left(N - n(t)\right)$$

#nodes(t)

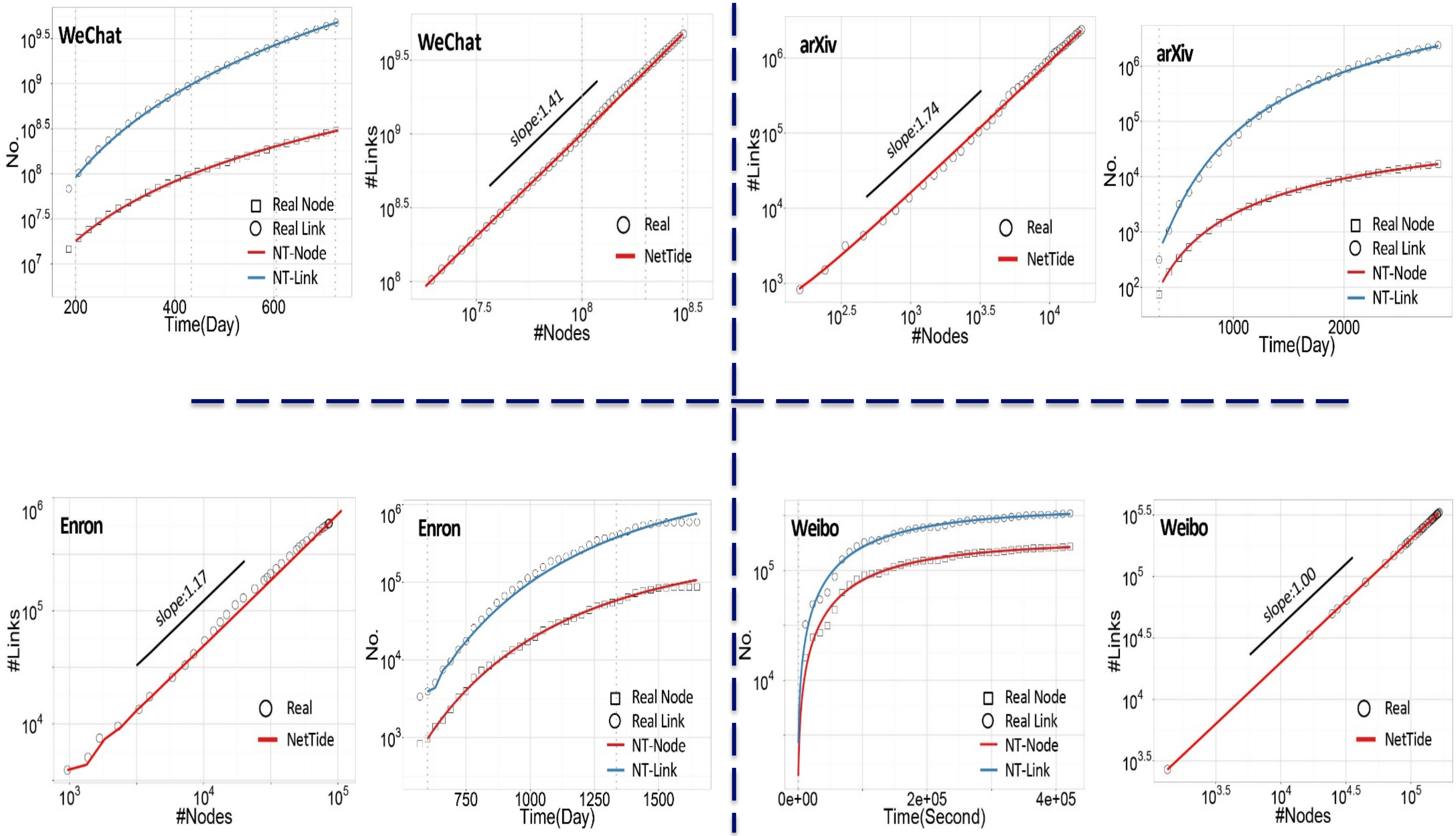
Total population

- Intuition:
  - Rich-get-richer
  - Limitation
  - **Fizzling nature**

= SI; ~Bass

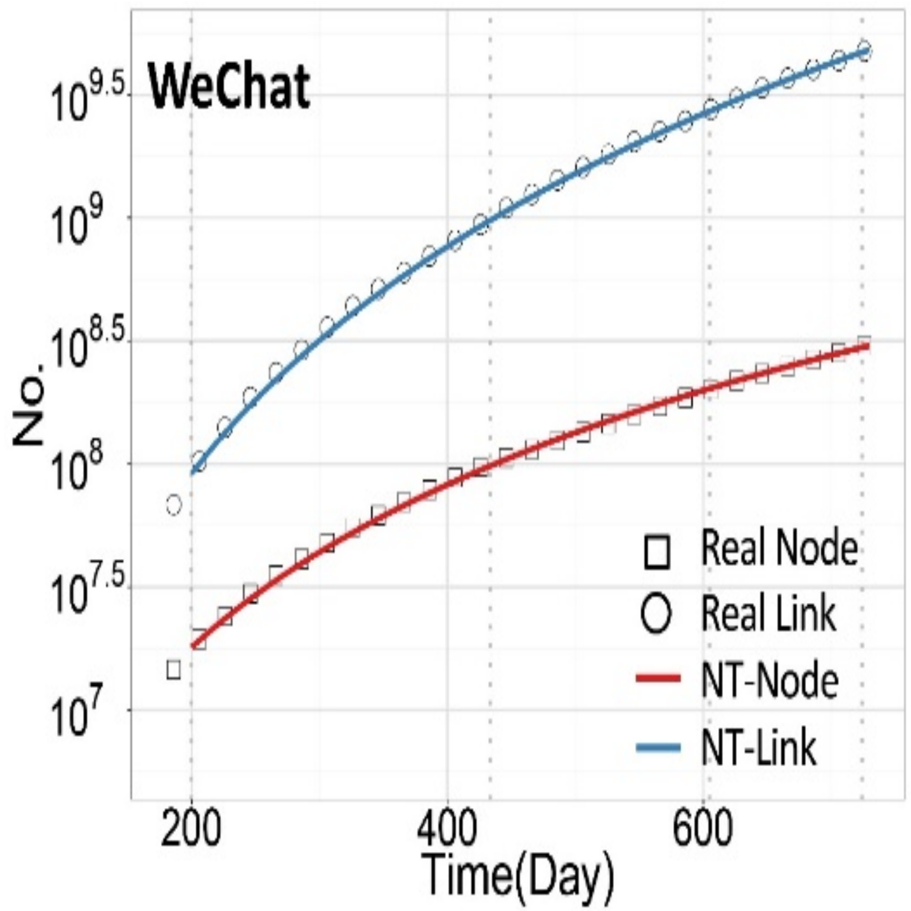
Carnegie Mellon

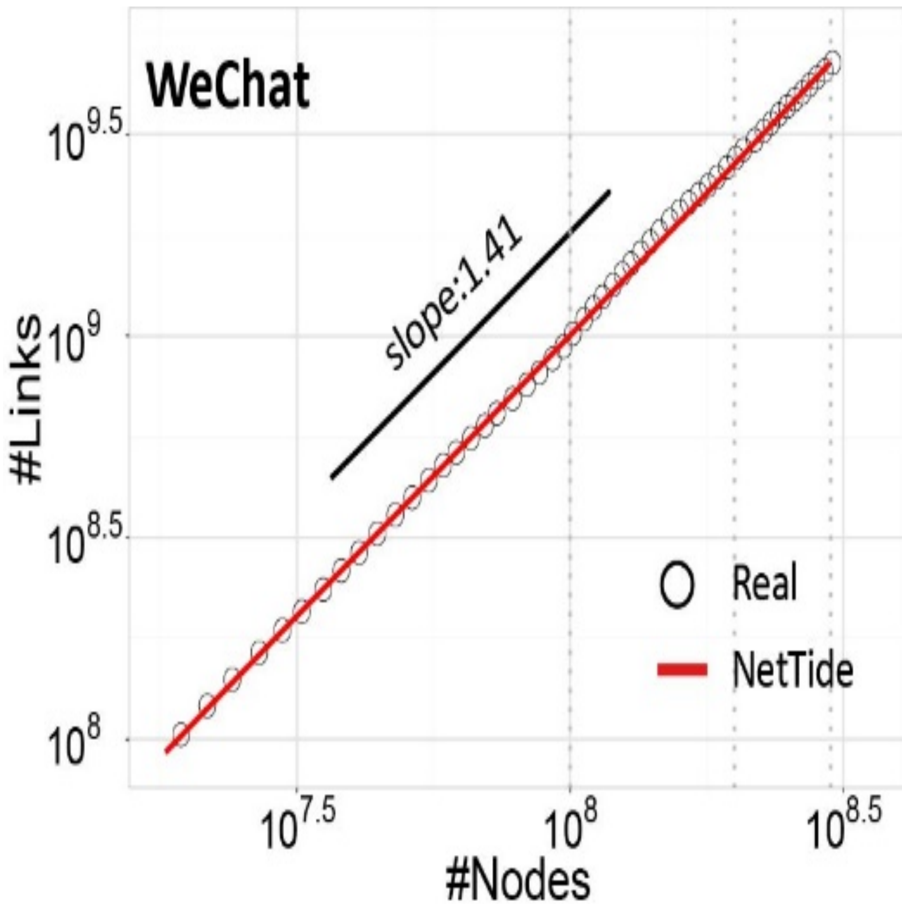
Results: Accuracy



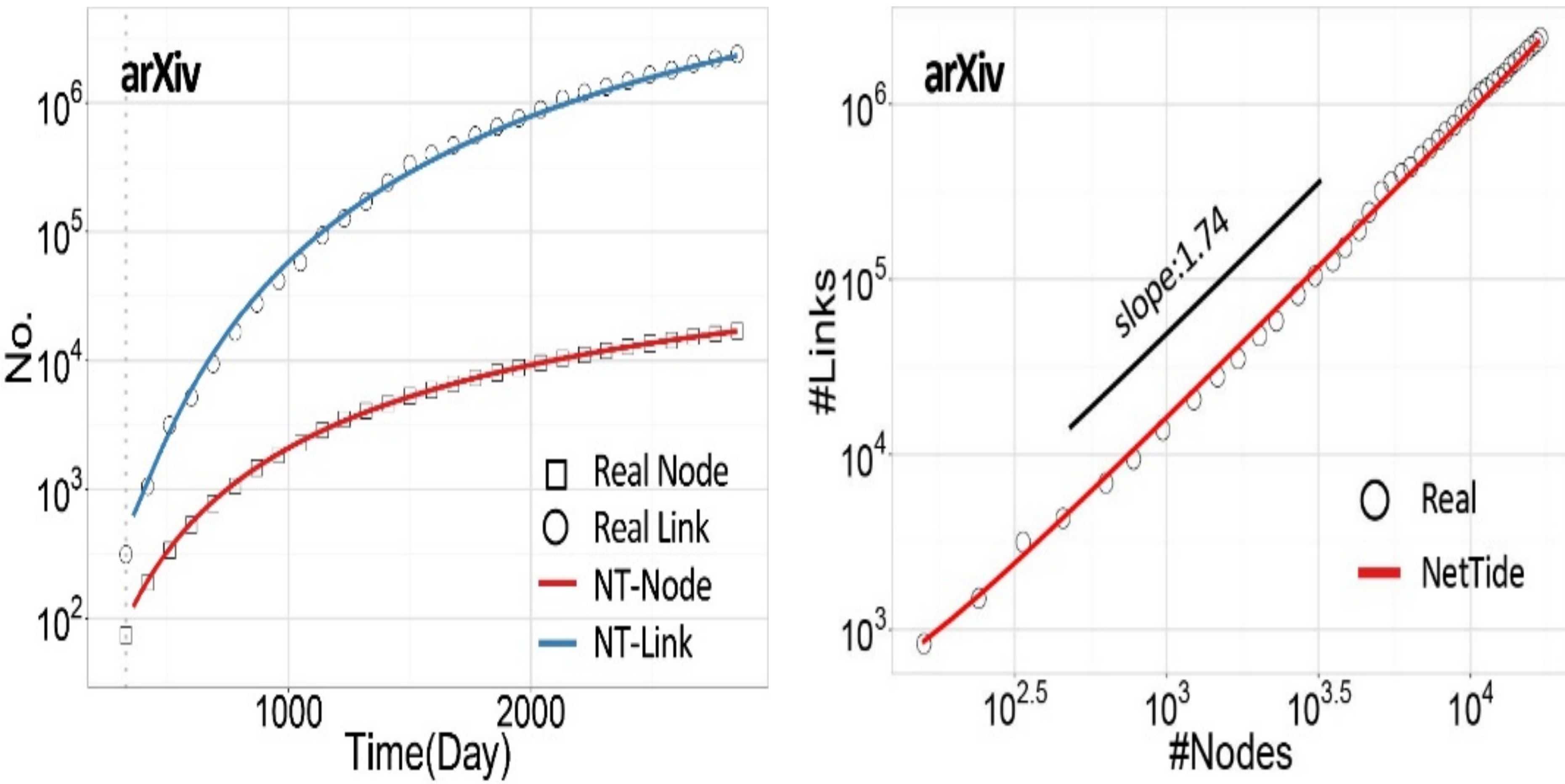
Carnegie Mellon

Results: Accuracy

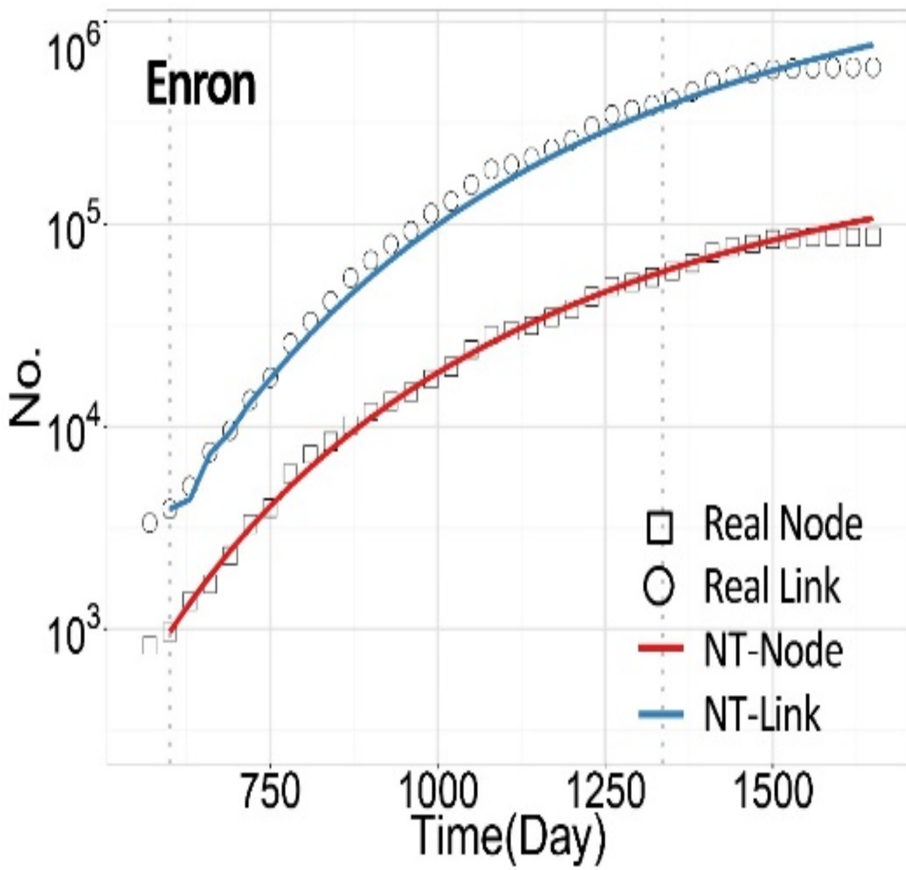


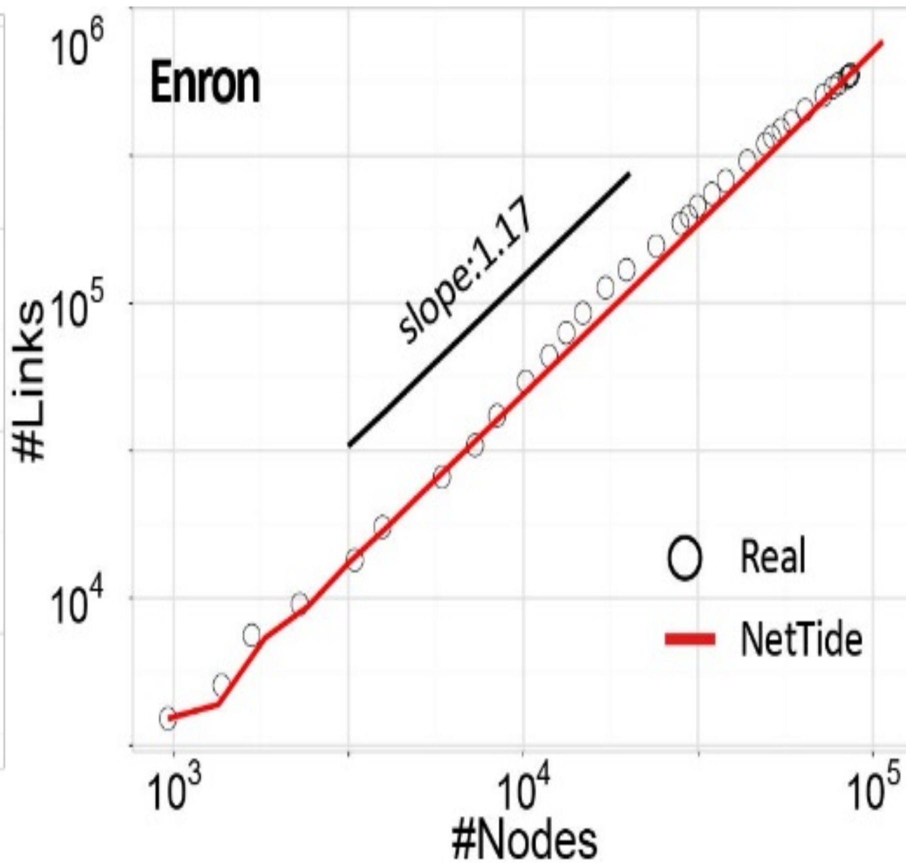


Results: Accuracy

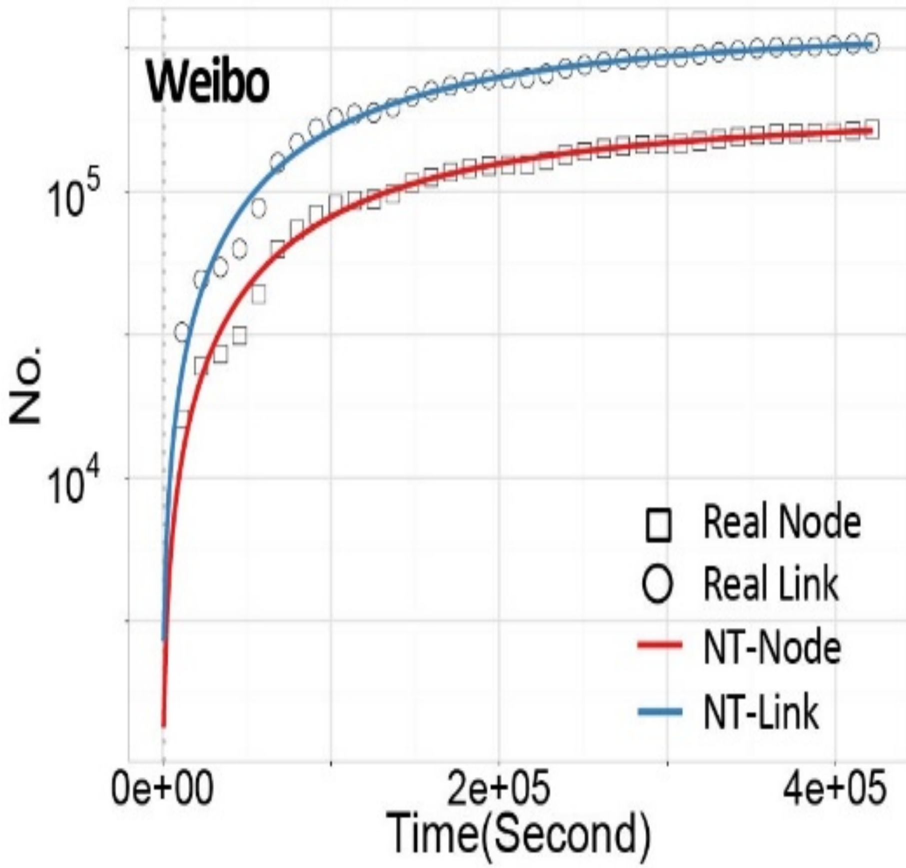


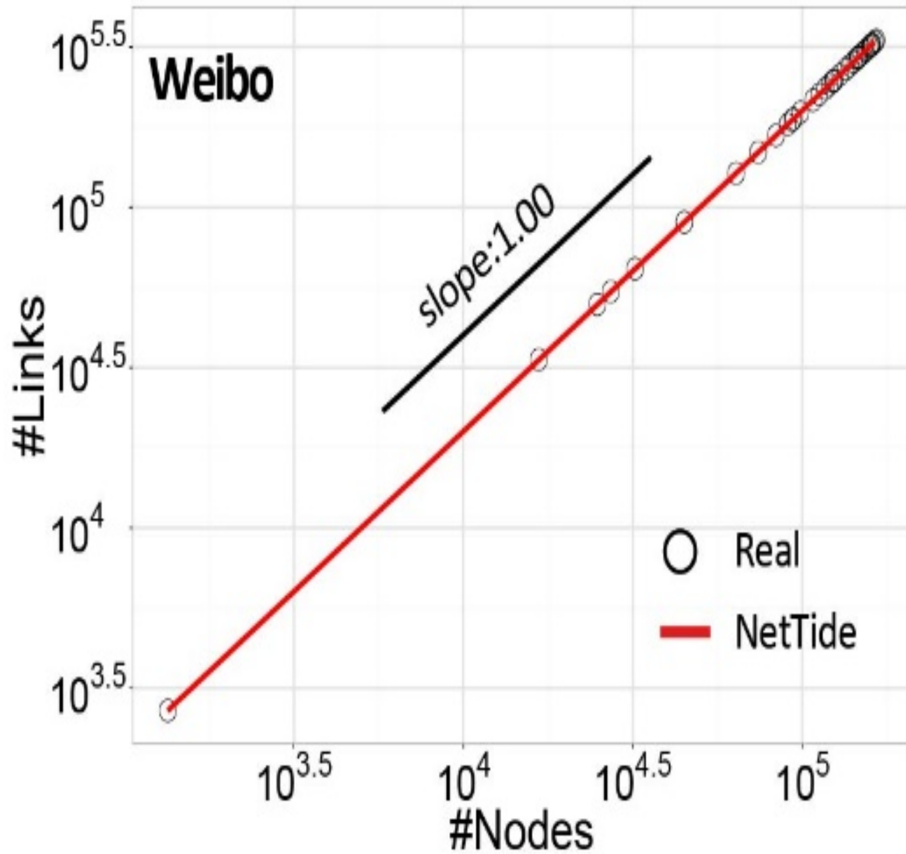
Results: Accuracy





Results: Accuracy

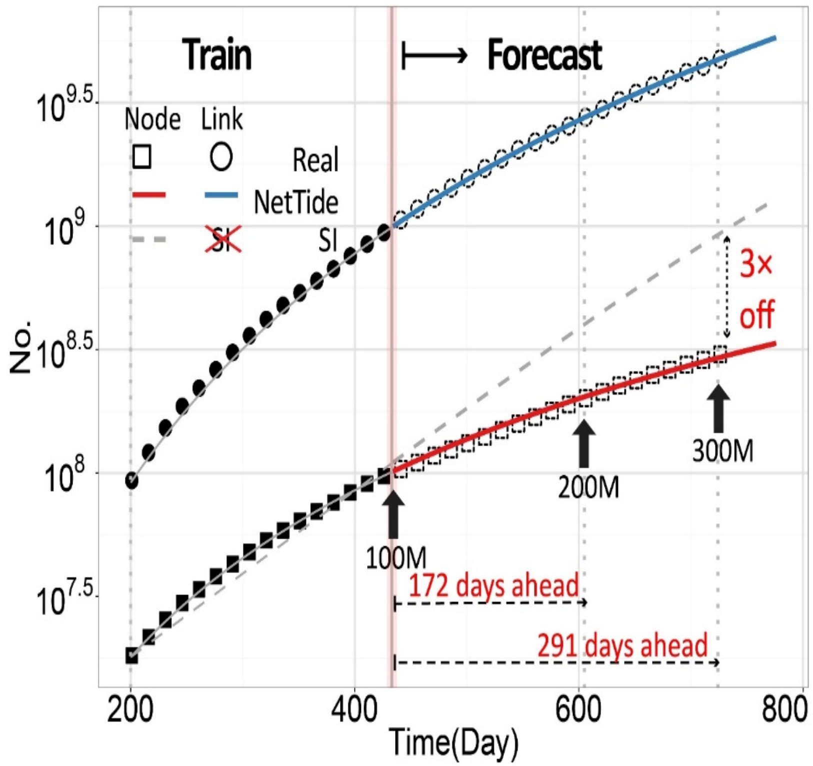


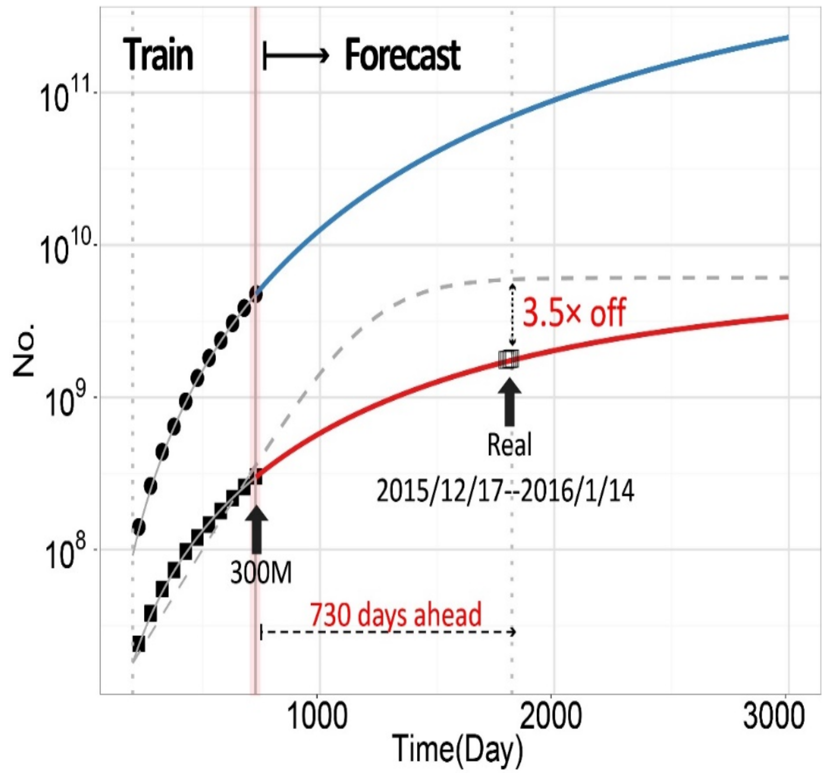


Results: Forecast

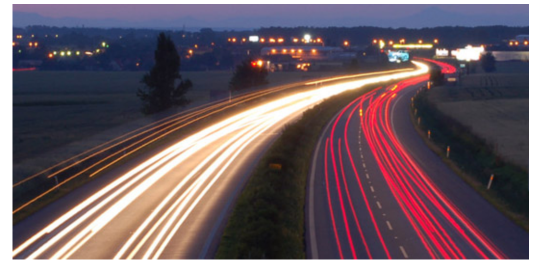
WeChat from 100 million
to 300 million

730 days ahead





Roadmap



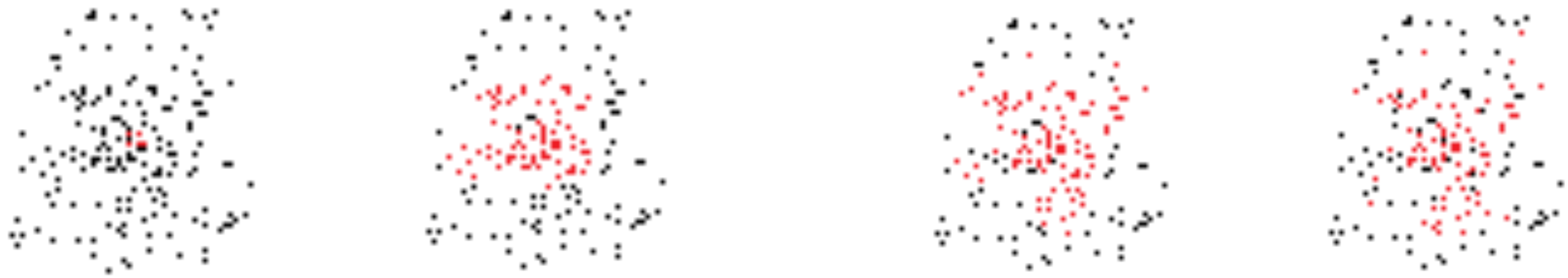
- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: time-evolving graphs
  - P2.1: tools/tensors
  - P2.2: other patterns
    - inter-arrival time
    - Network growth
    - Group evolution
- Conclusions

Come-and-Go Patterns of Group Evolution: A Dynamic Model

Tianyang Zhang, Peng Cui, Christos Faloutsos
Yunfei Lu, Hao Ye, Wenwu Zhu, Shiqiang Yang

KDD'16, San Francisco, CA

Social Group Dynamics – An open problem



1 hour
N = 5

3 days
N = 77

3 weeks
N = 98

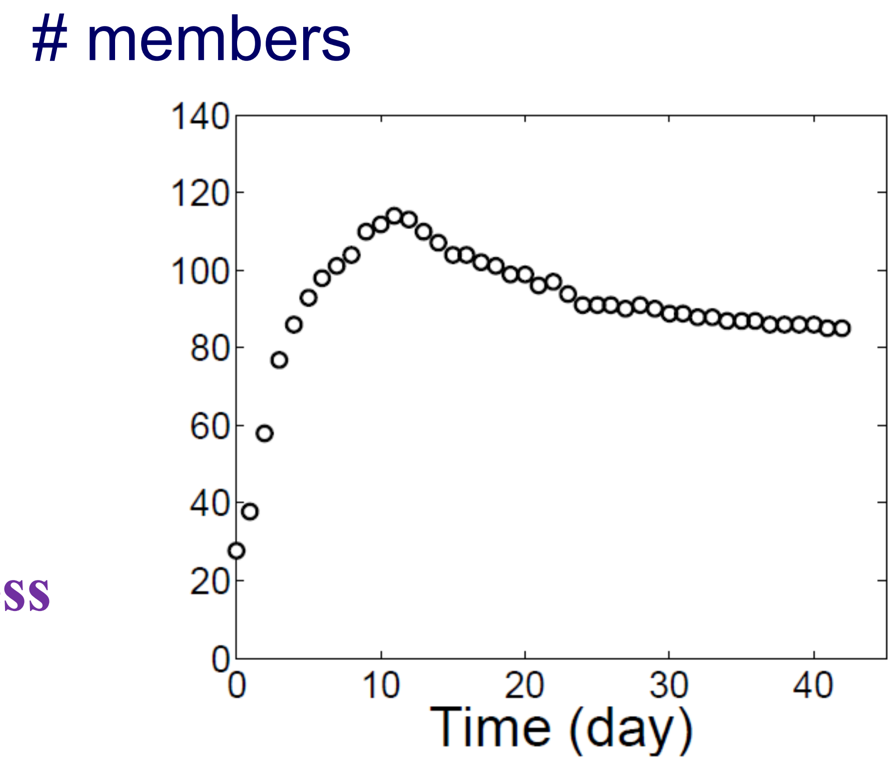
2 months
N = 83

- Will it grow larger or decline?
- Forecast group size after one month?

Carnegie Mellon

Our Problem: Group Evolution Process

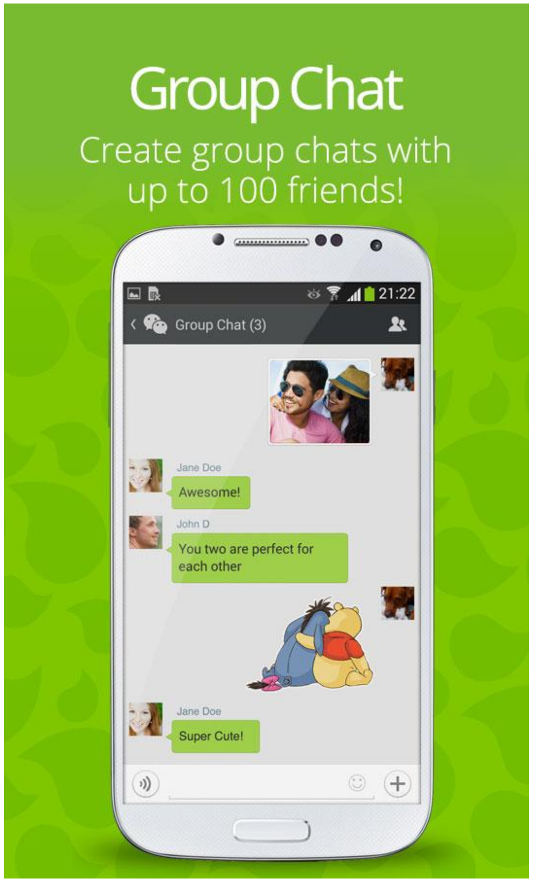
- Goals:
  - G1: Discover Patterns
  - G2: Reveal Mechanisms
  - G3: Model Evolution Process



Group Evolution Process

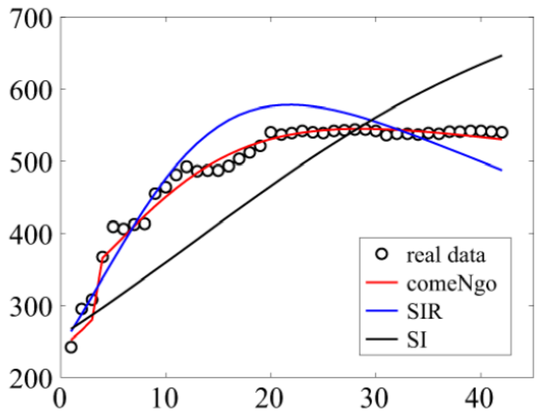
G1: Discover Patterns

- Wechat Group dataset
  - Largest social network in China
  - Sample 100K social groups
  - 42 days since established
  - 15M records
    - **Join / Quit log**
    - Temporal information

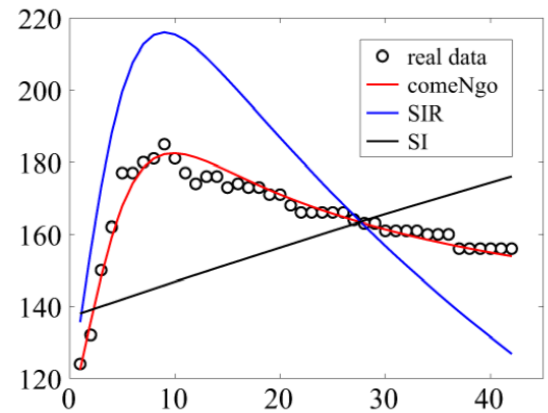


G1: Discover Patterns

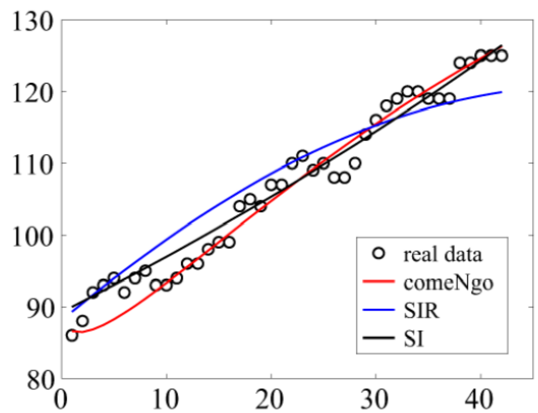
- Recurring Patterns



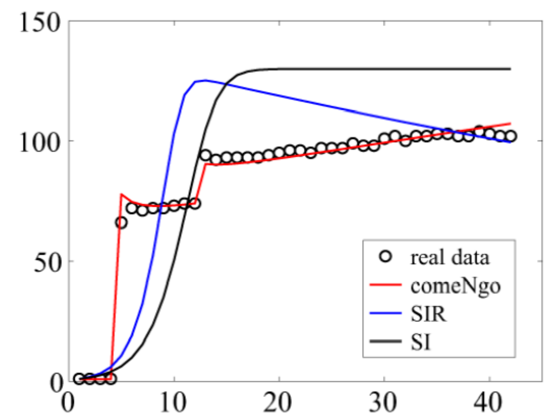
rise and stay



come and go



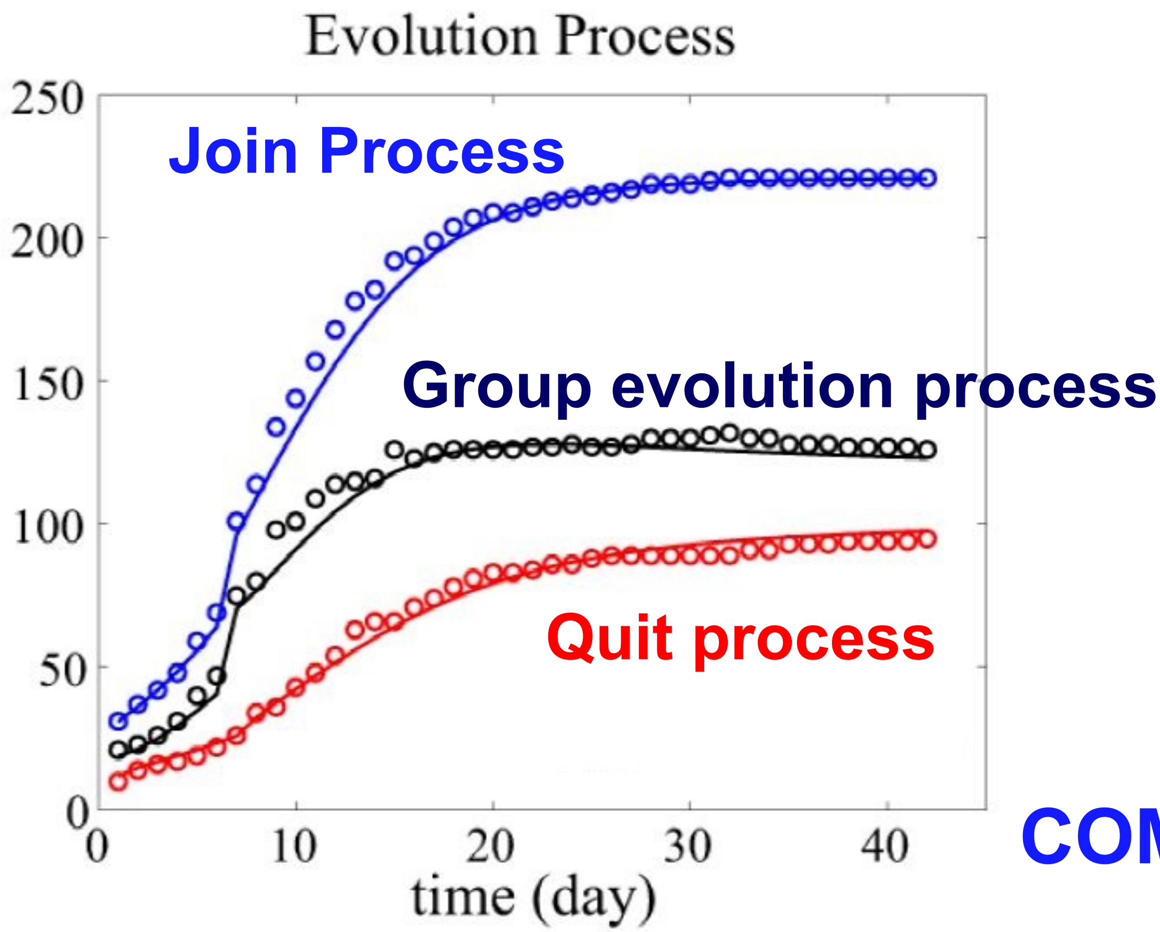
continuous increase



rocket increase

G2: Reveal Mechanisms

- Join/quit logs



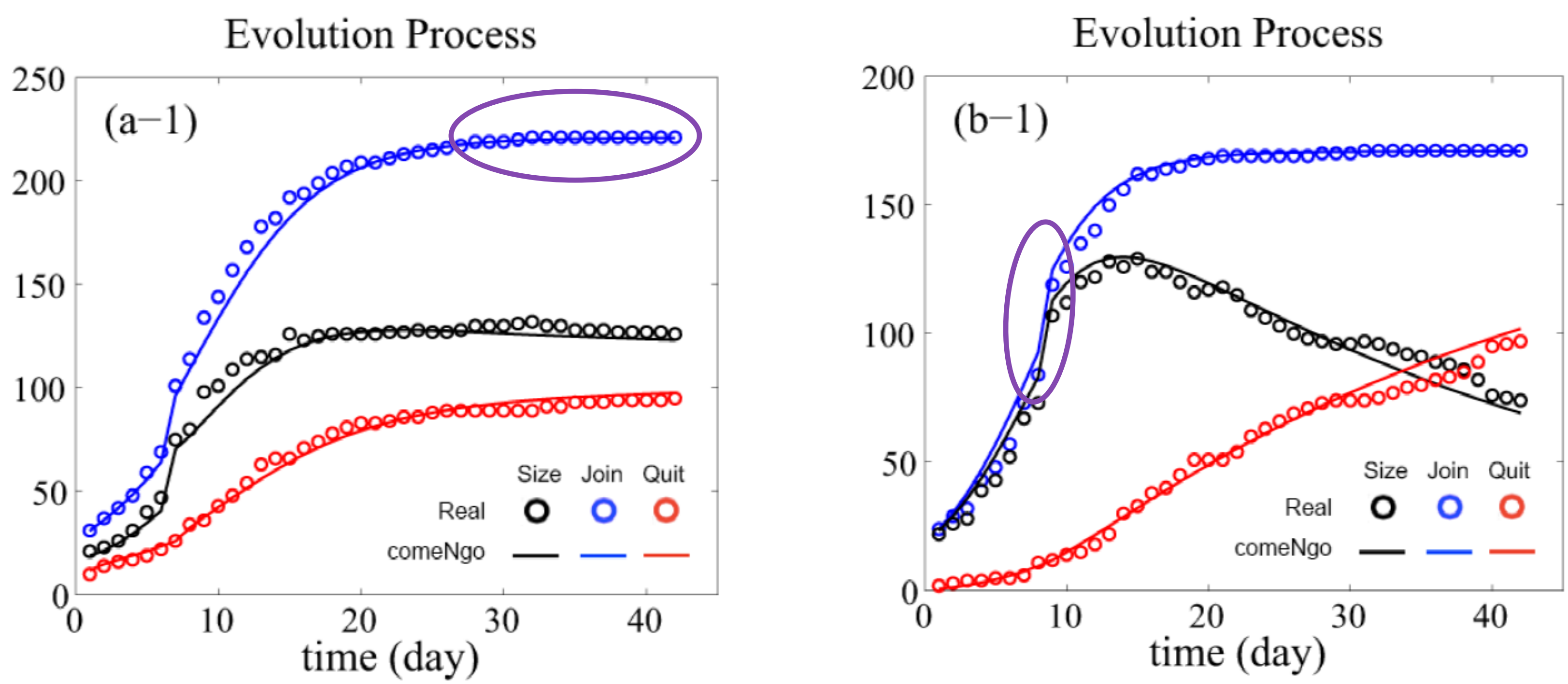
J(t)

I(t)

Q(t)

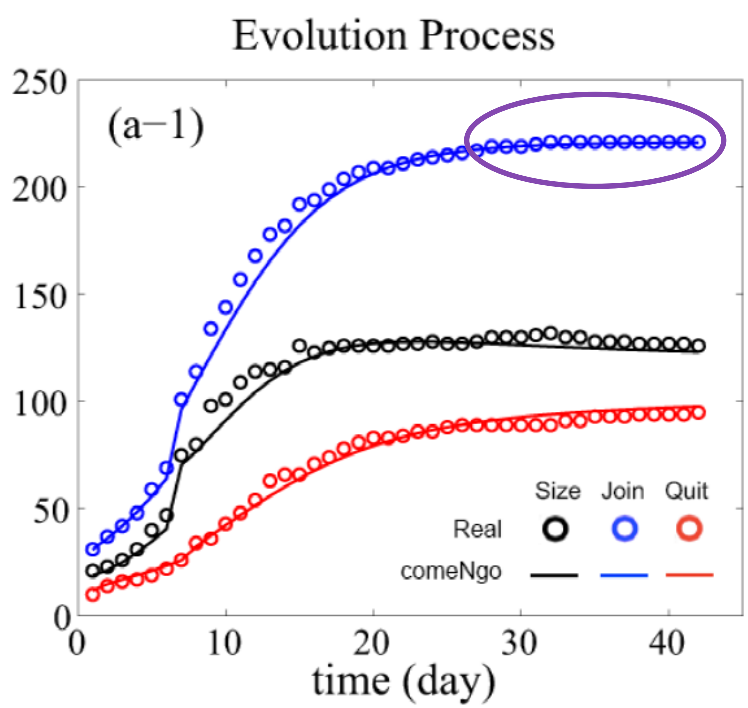
COMENGO

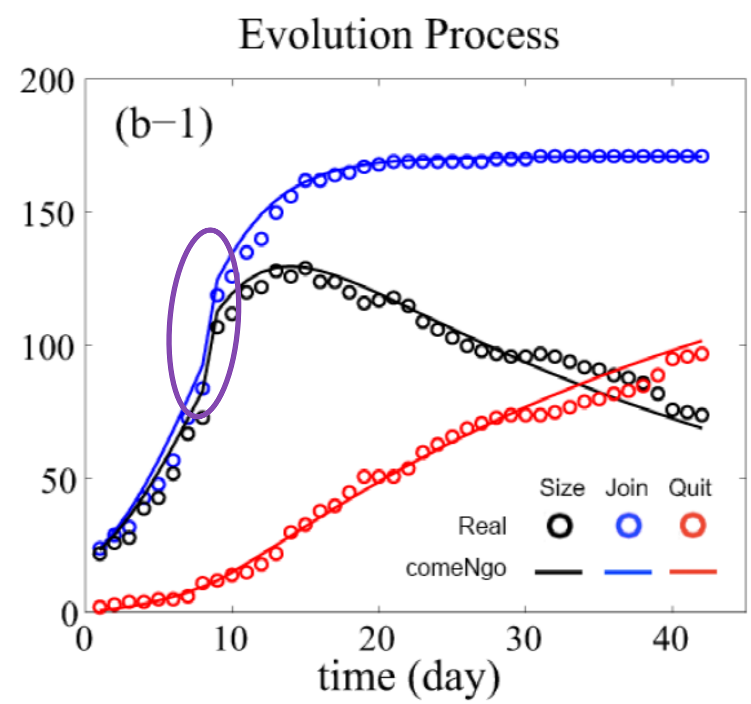
G2: Reveal Mechanisms



- **Q: Can we find (simple) equations, that can fit all these patterns (J(t), Q(t))?**

G2: Reveal Mechanisms

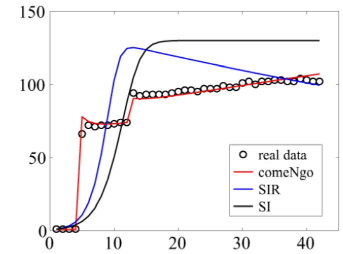




- **Q: Can we find (simple) equations, that can fit all these patterns (J(t), Q(t))?**
- **A: Yes!**

Details

G3-1: Dynamic Model – Join

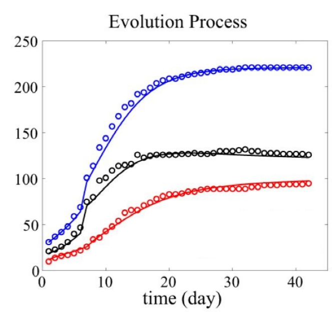


- **Join process**

diffusion growth | bursty growth

$$J'(t) = \frac{dJ}{dt} = \beta(N - J(t))I(t) + \sum \lambda_i \delta(t - t_i)$$

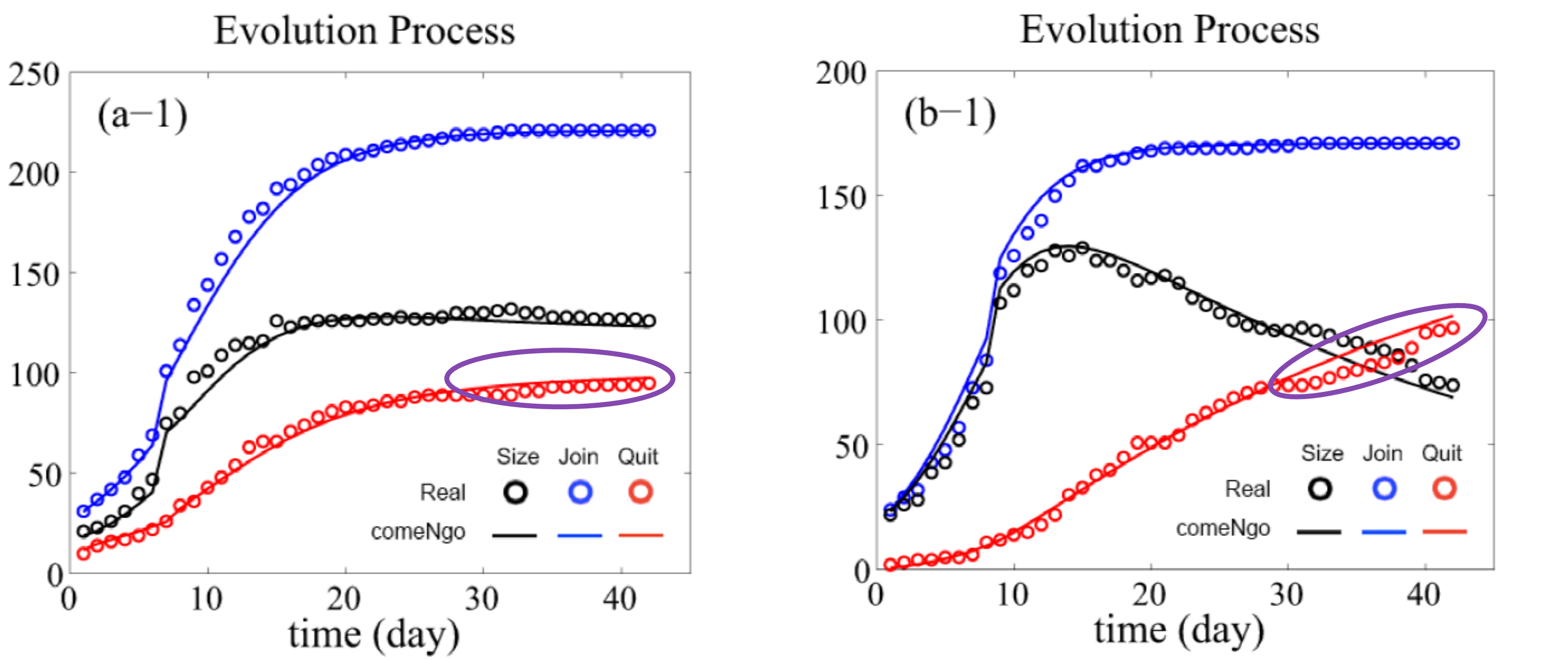
- β: attractiveness
- N: population
- $I(t)$: group size
- λ_i: effect strength
- t_i: time of external shock



Carnegie Mellon

Details

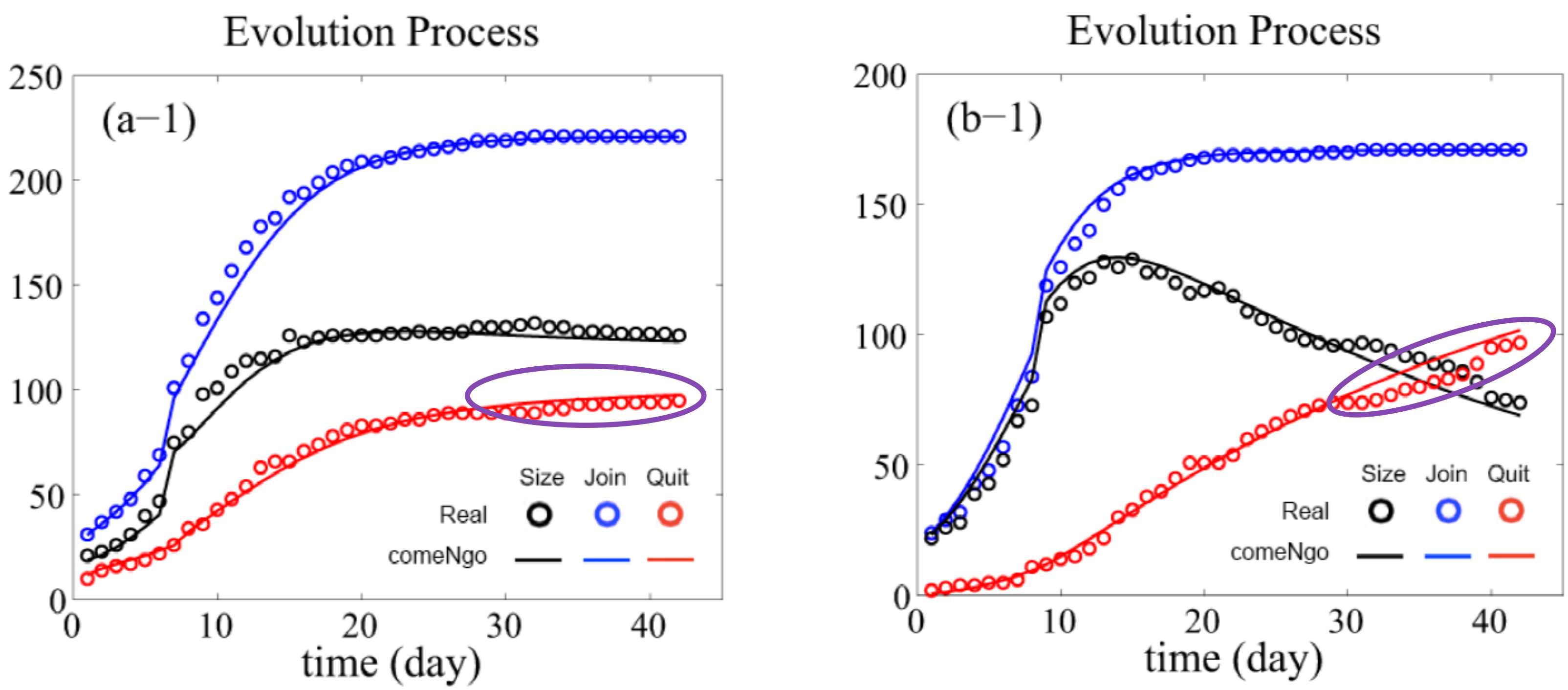
G2-2: Reveal Mechanisms – Quit



Stabilizing **declining**

Details

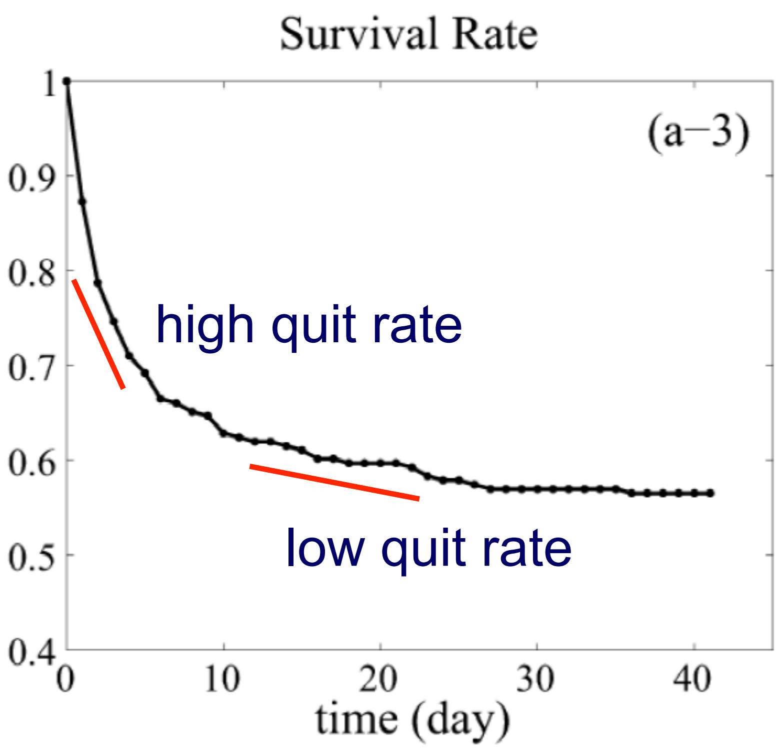
G2-2: Reveal Mechanisms – Quit

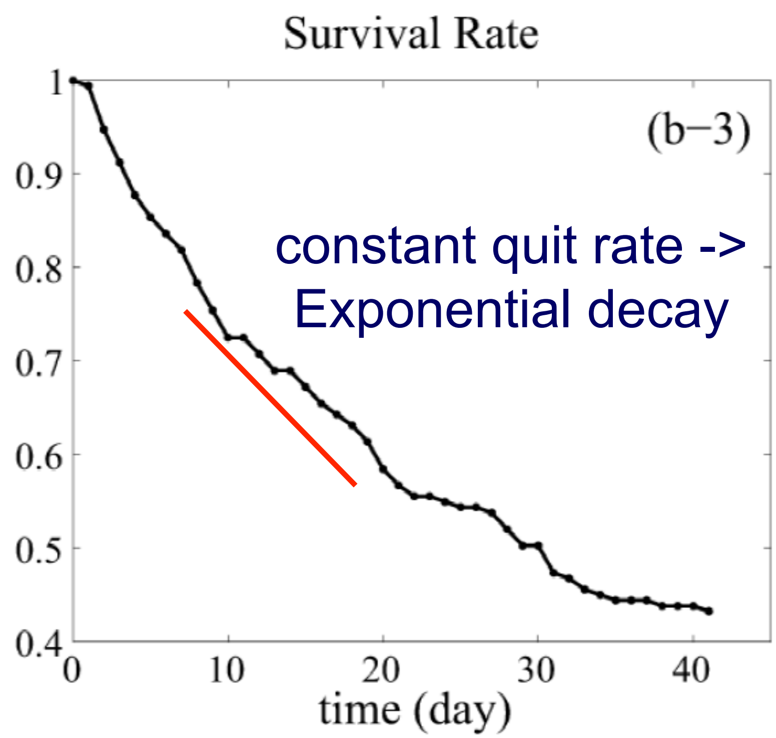


Q: Quitting: exponential (‘half life’ == SIR) ?

Details

G2-2: Reveal Mechanisms – Quit Process





Gradually stable

85% of all groups: Decay over holding time

Continuous decline

15% of all groups: Constant quit rate

Details

G3-2: Dynamic Model – Quit Process

- **Quit process – Quit rate:**

$$\gamma(\tau) = \gamma_0 \tau^{-\alpha}$$

α=0, exact exponential distribution
0< α<1, exponential like distribution
α>1, power-law distribution

Details

G3-2: Dynamic Model – Quit Process

Quit process

- Quit rate may decrease over holding time τ
- Power-law or Exponential distributed holding time

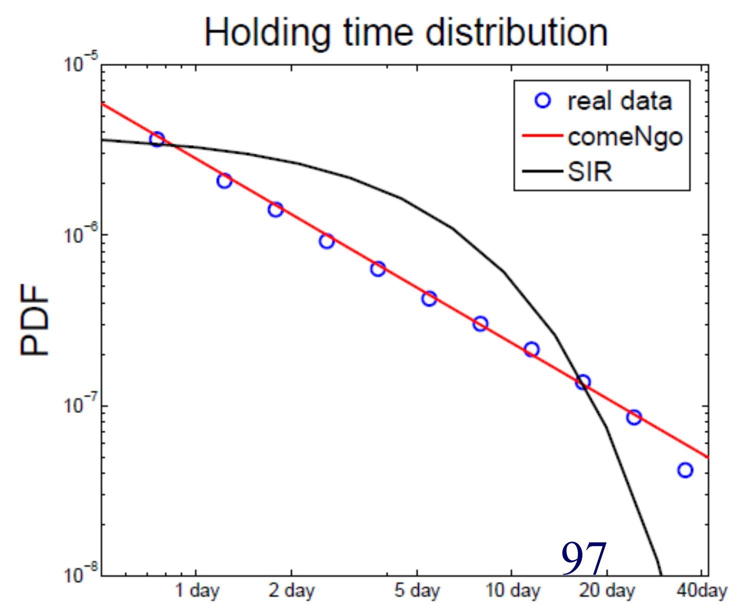
$$\gamma(\tau) = \gamma_0 \tau^{-\alpha}$$

Quit Rate

- α=0, exact exponential distribution
- 0< α<1, exponential like distribution
- α>1, power-law distribution

$$f(\tau) = c\tau^{-\alpha} \exp\left(\frac{\gamma_0 \tau^{1-\alpha}}{\alpha - 1}\right)$$

p.d.f of holding time τ



- γ_0: short time satisfaction degree
- α: long time dependence

Carnegie Mellon

G3: Dynamic Model - COMENGO

J(t) =? **Q(t)** =?

group size: $I(t) = J(t) - Q(t)$

join process: $J'(t) = \frac{dJ}{dt} = \beta(N - J(t))I(t) + \sum \lambda_i \delta(t - t_i)$

quit process: $Q'(t) = \frac{dQ}{dt} = \int_0^t J'(x) f(t-x) dx$

holding time: $f(\tau) = c\tau^{-\alpha} \exp(\frac{\gamma_0 \tau^{1-\alpha}}{\alpha - 1})$

G3: Dynamic Model - COMENGO

group size: $I(t) = J(t) - Q(t)$

join process: $J'(t) = \frac{dJ}{dt} = \beta(N - J(t))I(t) + \sum \lambda_i \delta(t - t_i)$

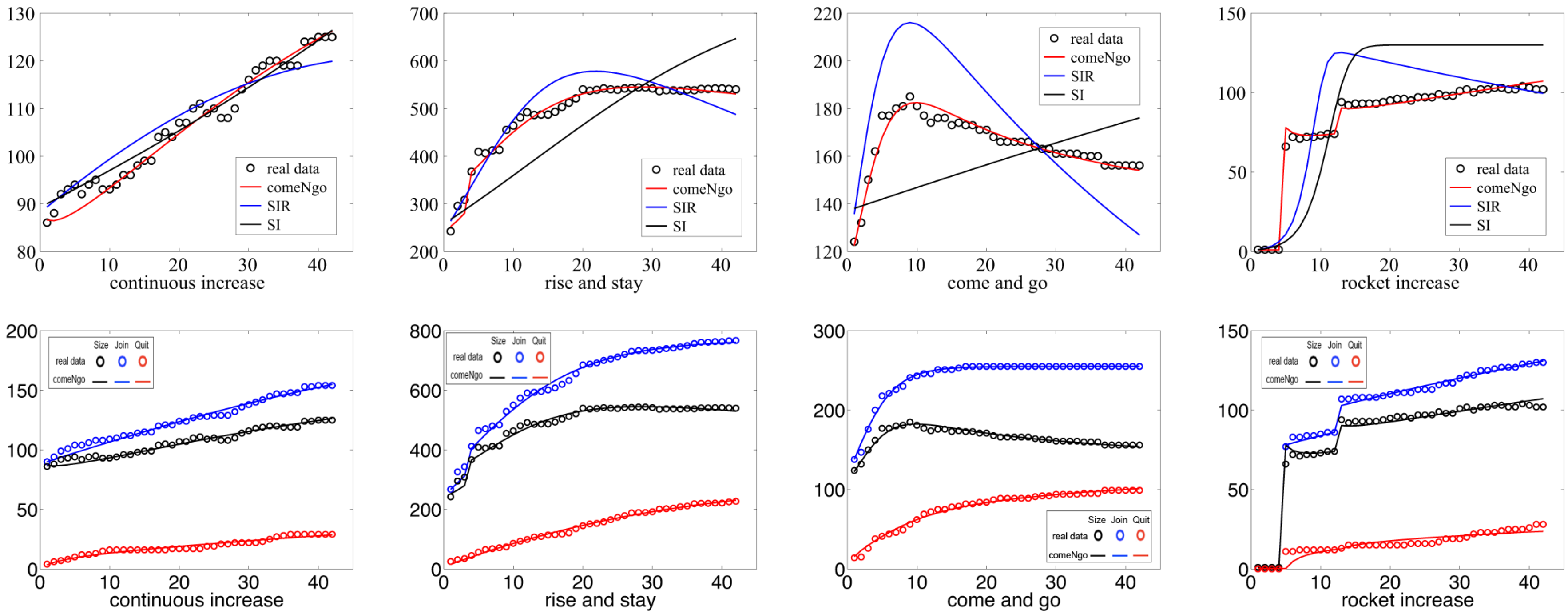
quit process: $Q'(t) = \frac{dQ}{dt} = \int_0^t J'(x) f(t-x) dx$

holding time: $f(\tau) = c\tau^{-\alpha} \exp(\frac{\gamma_0 \tau^{1-\alpha}}{\alpha - 1})$

Carnegie Mellon

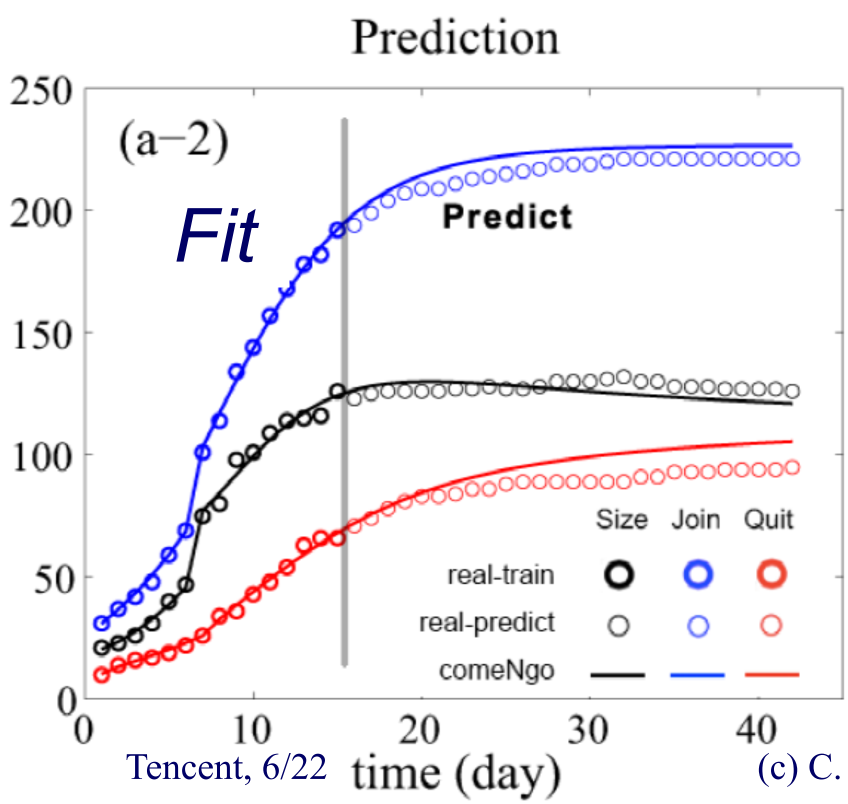
Experiment – Fitting Accuracy

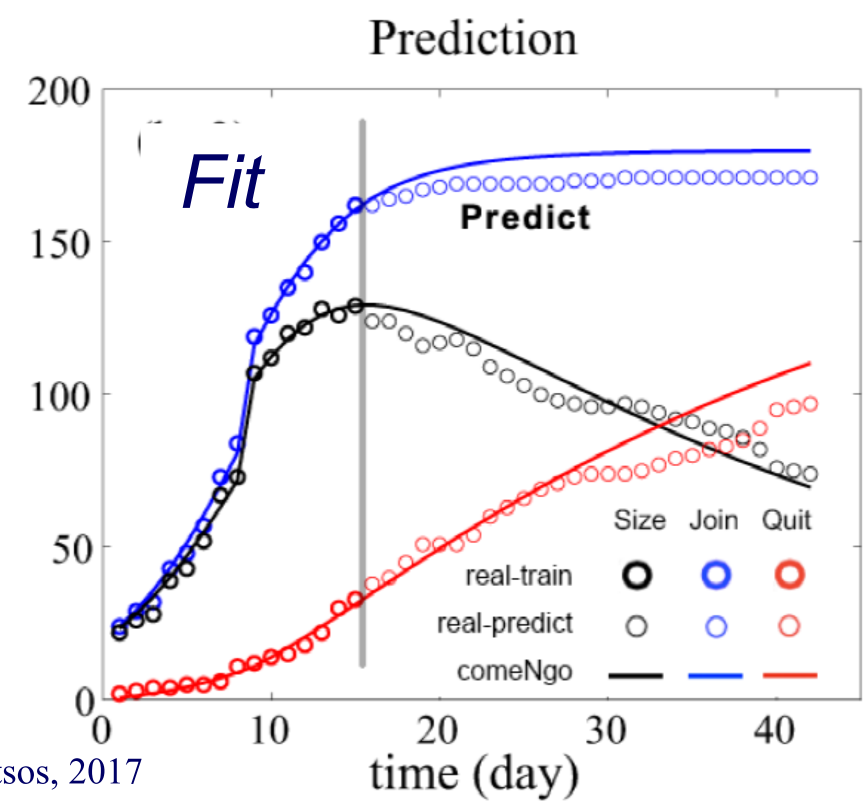
- Fits all different patterns
- Fit both join & quit process



Experiment – Predicting Power

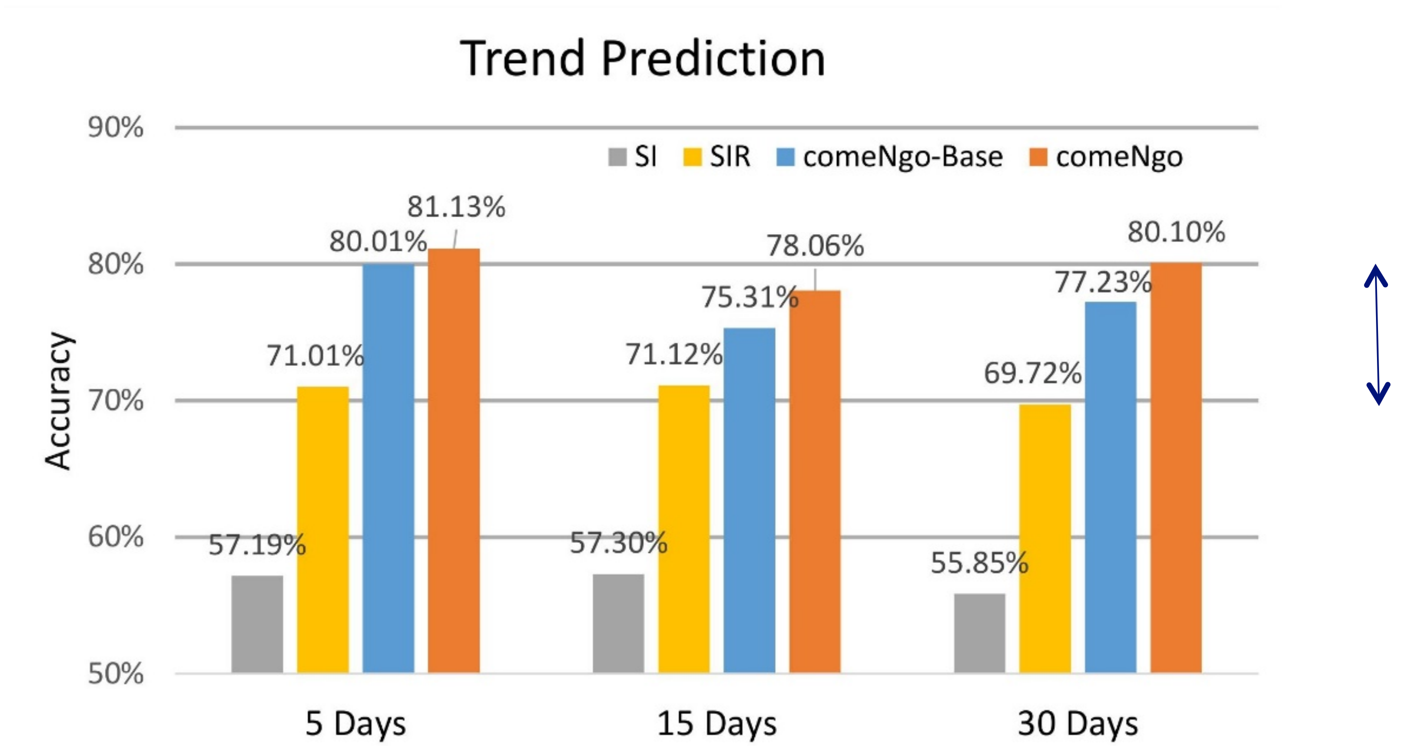
- Size prediction
  - Given early stage data, predict the group size in future





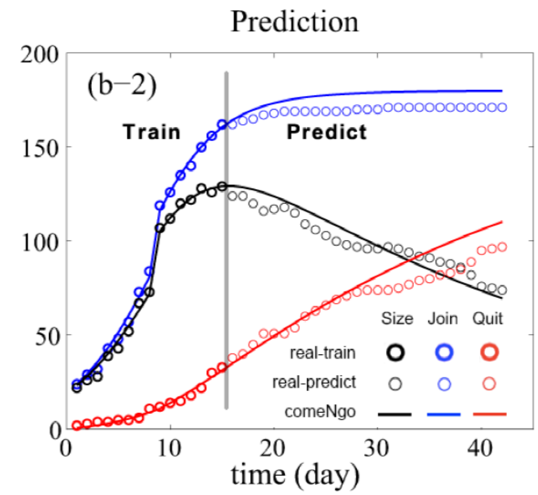
Experiment – Predicting Power

- **Trend prediction**
  - Given early stage data, predict whether the group will grow
  - **14.3%** better accuracy



Carnegie Mellon

Conclusions

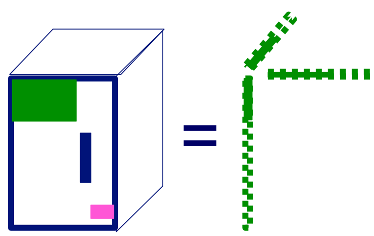


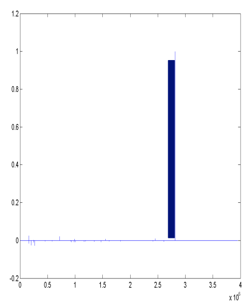
- ✓G1: Discover patterns
- ✓ G2: Reveal mechanisms
- ✓ G3: Novel unifying model
  - ✓Better accuracy
  - ✓predictive power

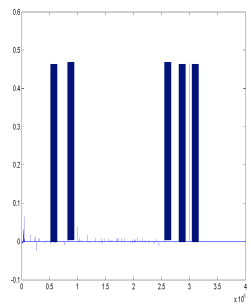
Tianyang Zhang
zhangty09@foxmail.com

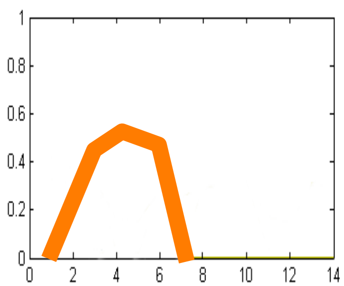
Part 2: Conclusions

- Time-evolving / heterogeneous graphs -> tensors
- PARAFAC finds patterns
- Surprising temporal patterns (P.L. growth, comeNgo group evolution)



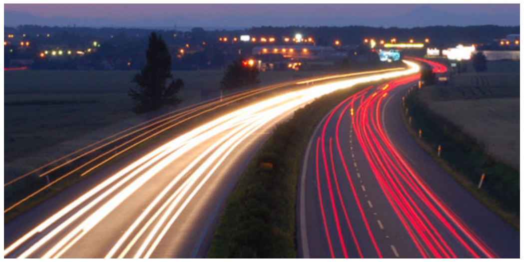






Carnegie Mellon

Roadmap



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

Thanks

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

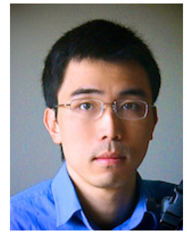
Thanks to: NSF IIS-0705359, IIS-0534205, CTA-INARC; Yahoo (M45), LLNL, IBM, SPRINT, Google, INTEL, HP, iLab

Cast

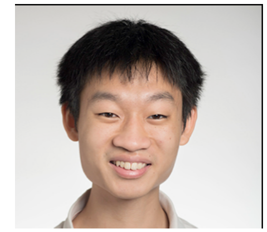

Akoglu, Leman


Araujo, Miguel


Beutel, Alex


Chau, Polo

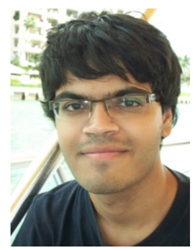

Eswaran, Dhivya

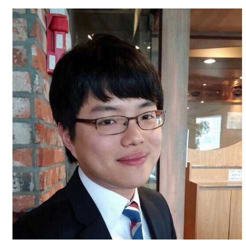

Hooi, Bryan


Kang, U


Koutra, Danai

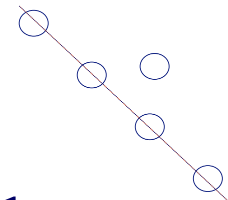

Papalexakis, Vagelis


Shah, Neil

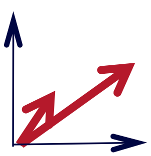

Shin, Kijung

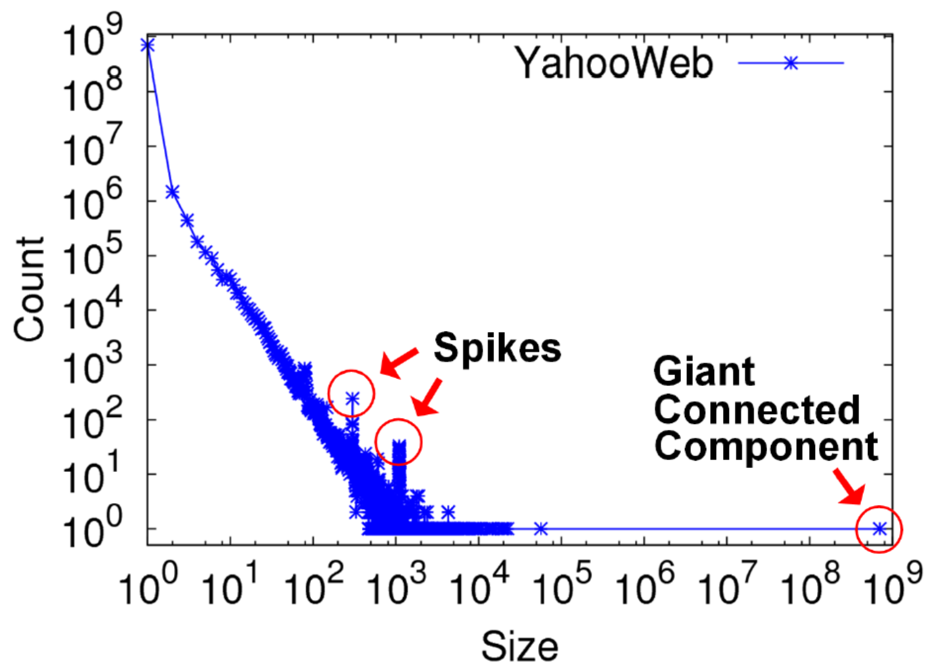

Song, Hyun Ah

CONCLUSION#1 – Big data



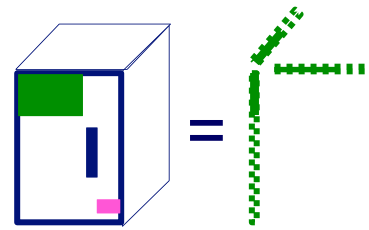
- **Patterns**  **Anomalies**
- **Large** datasets reveal patterns/outliers that are invisible otherwise

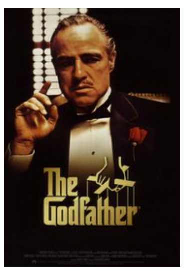


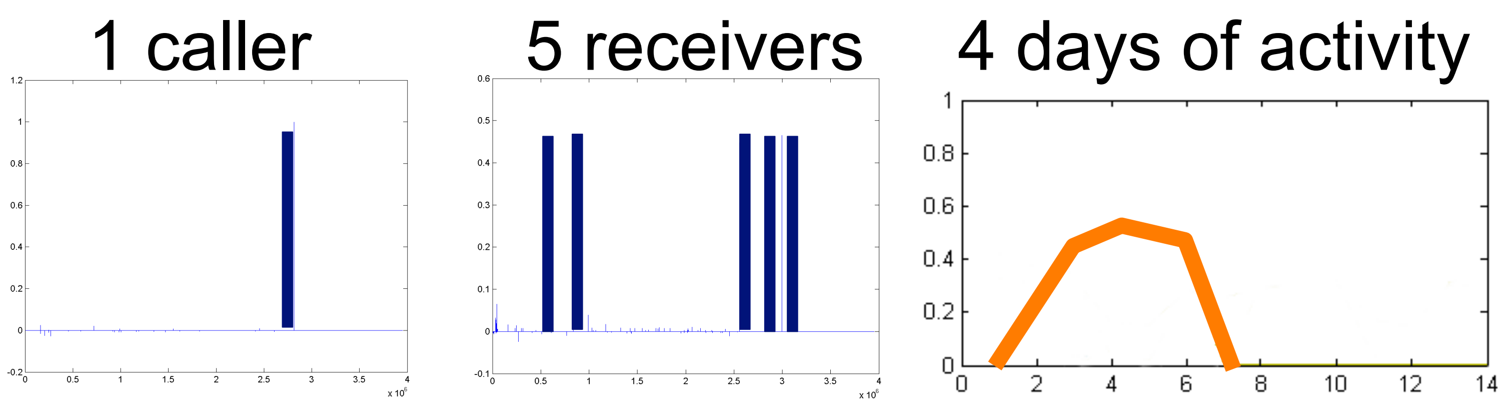


CONCLUSION#2 – tensors

- powerful tool

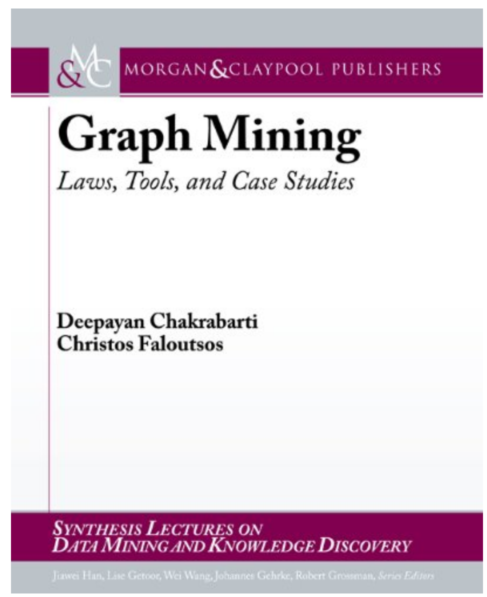






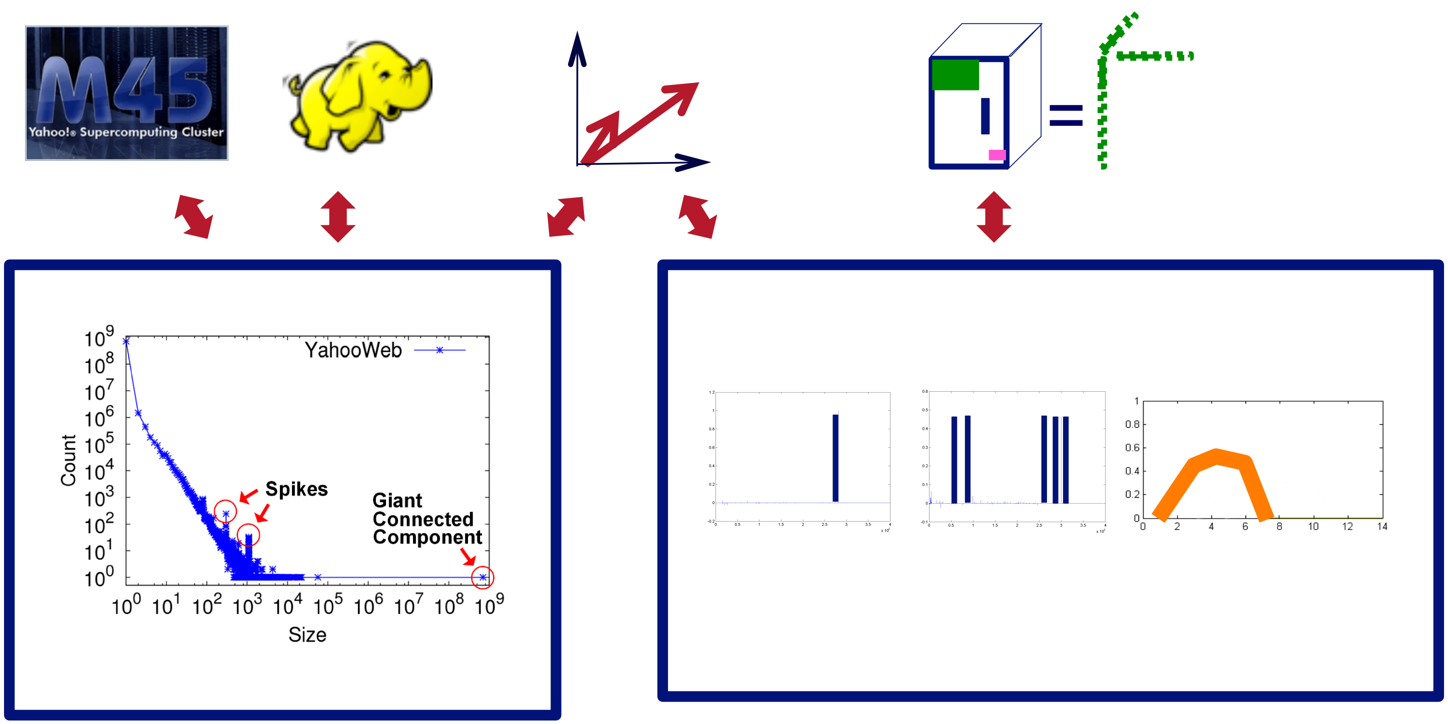
References

- 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:

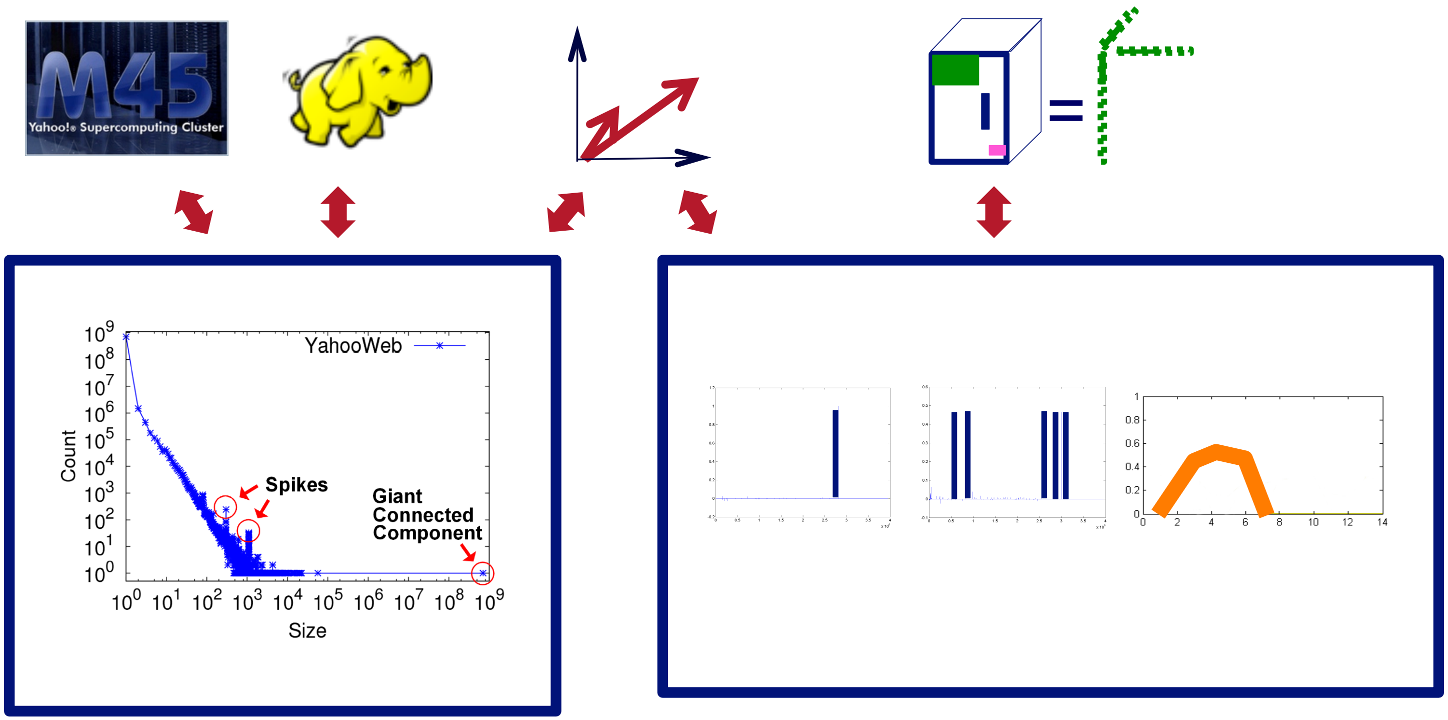
Cross-disciplinarity



Carnegie Mellon

Thank you!

Cross-disciplinarity



www.cs.cmu.edu/~christos/TALKS/17-06-22-tencent/faloutsos_tencent_2017.pdf