Mining Large Graphs: Patterns, Anomalies, and Fraud Detection

Christos Faloutsos

CMU
Thank you!

- Alex Smola
- Aditi Chaudhuri
- Tiffany Kuan
Foils: at

http://www.cs.cmu.edu/~christos/TALKS/16-09-AMAZON/
Roadmap

• Introduction – Motivation
  – Why study (big) graphs?
• Part#1: Patterns in graphs
• Part#2: time-evolving graphs; tensors
• Conclusions
Graphs - why should we care?

- Recommendation systems
- App-stores + reviews
Graphs - why should we care?

~1B nodes (web sites)
~6B edges (http links)
‘YahooWeb graph’
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
- ....

- Many-to-many db relationship -> graph
Motivating problems

• P1: patterns? Fraud detection?

• P2: patterns in time-evolving graphs / tensors
Motivating problems

• P1: patterns? Fraud detection?
• P2: patterns in time-evolving graphs / tensors
Motivating problems

• P1: patterns? Fraud detection?
• P2: patterns in time-evolving graphs / tensors

* Robust Random Cut Forest Based Anomaly Detection on Streams Sudipto Guha, Nina Mishra, Gourav Roy, Okke Schrijvers, ICML’16
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
Solution# S.1

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

internet domains

log(degree) vs. log(rank)

att.com

ibm.com
Solution# S.1

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

internet domains

log(degree) vs. log(rank)

-0.82
Solution# S.1

- Q: So what?

![Diagram showing internet domains and log plots with values -0.82 and 0.82]
Q: So what?  
A1: # of two-step-away pairs:  
internet domains  

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

\[ \log(\text{rank}) - \log(\text{degree}) = -0.82 \]
Solution# S.1

• Q: So what?
  • A1: # of two-step-away pairs: \(100^2 \times N = 10\) Trillion

internet domains

\(= \text{friends of friends (F.O.F.)}\)

\[\log(\text{rank})\] \[\log(\text{degree})\]

\(\text{att.com}\)

\(\text{ibm.com}\)

-0.82
Solution# S.1

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

internet domains

friends of friends (F.O.F.)

\[ \log(\text{rank}) \]
\[ \log(\text{degree}) \]

\[ \text{att.com} \]

\[ \text{ibm.com} \]

\[-0.82\]

Amazon'16

(c) 2016, C. Faloutsos
Solution# S.1

• Q: So what?

• A1: # of two-step-away pairs: $O(d_{\text{max}}^2) \sim 10M^2$

internet domains

att.com

log(degree)

ibm.com

DCO @ CMU

~0.8PB -> a data center(!)

Gaussian trap

$\log(\text{degree})$

$\log(\text{rank})$

$\log(\text{degree})$

$-0.82$

att.com

ibm.com

~0.82

Amazon'16

(c) 2016, C. Faloutsos
Q: So what?

A1: # of two-step-away pairs: \(O(d_{\text{max}}^2) \sim 10M^2 \log(\text{rank}) \log(\text{degree})-0.82\)

internet domains

\(\sim 0.8\)PB -> a data center(!)

New algorithms -> Such patterns ->

\(~0.8\)PB -> \(~10M^2\)

Solution # S.1

Gaussian trap
Solution# S.2: Eigen Exponent $E$

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

$A \times x = \lambda \times x$

Rank of decreasing eigenvalue

Exponent = slope

$E = -0.48$
Roadmap

• Introduction – Motivation
• Part#1: Patterns in graphs
  – Patterns: Degree; Triangles
  – Anomaly/fraud detection
  – Graph understanding
• Part#2: time-evolving graphs; tensors
• 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]

X-axis: degree
Y-axis: mean # triangles
n friends $\rightarrow \sim n^{1.6}$ triangles
Triangle Law: Computations
[Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos) – \(O(d_{\text{max}}^2)\)

Q: Can we do that quickly?

A:
Triangle Law: Computations
[Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos) – $O(d_{\text{max}}^2)$

Q: Can we do that quickly?
A: Yes!

$\#\text{triangles} = \frac{1}{6} \sum (\lambda_i^3)$
(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]
Triangle counting for large graphs?

Anomalous nodes in Twitter (~ 3 billion edges)
[U Kang, Brendan Meeder, +, PAKDD’11]
# MORE Graph Patterns

<table>
<thead>
<tr>
<th>Static</th>
<th>Unweighted</th>
<th>Weighted</th>
</tr>
</thead>
<tbody>
<tr>
<td>L02.</td>
<td>Triangle Power Law (TPL) [Tsourakakis `08]</td>
<td></td>
</tr>
<tr>
<td>L03.</td>
<td>Eigenvalue Power Law (EPL) [Siganos et al. `03]</td>
<td></td>
</tr>
<tr>
<td>L04. Community structure [Flake et al. <code>02, Girvan and Newman </code>02]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dynamic</th>
<th>Unweighted</th>
<th>Weighted</th>
</tr>
</thead>
<tbody>
<tr>
<td>L05.</td>
<td>Densification Power Law (DPL) [Leskovec et al. `05]</td>
<td>L11. Weight Power Law (WPL) [McGlohon et al. `08]</td>
</tr>
<tr>
<td>L06.</td>
<td>Small and shrinking diameter [Albert and Barabási <code>99, Leskovec et al. </code>05]</td>
<td></td>
</tr>
<tr>
<td>L07.</td>
<td>Constant size 2\textsuperscript{nd} and 3\textsuperscript{rd} connected components [McGlohon et al. `08]</td>
<td></td>
</tr>
<tr>
<td>L08.</td>
<td>Principal Eigenvalue Power Law ($\lambda_1$PL) [Akoglu et al. `08]</td>
<td></td>
</tr>
<tr>
<td>L09.</td>
<td>Bursty/self-similar edge/weight additions [Gomez and Santonia <code>98, Gribble et al. </code>98, Crovella and</td>
<td></td>
</tr>
</tbody>
</table>

---

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

<table>
<thead>
<tr>
<th>Static</th>
<th>Weighted</th>
</tr>
</thead>
<tbody>
<tr>
<td>L62.</td>
<td>Triangle Power Law (TPL) [Tsourakakis ’08]</td>
</tr>
<tr>
<td>L63.</td>
<td>Eigencentral Power Law (EPL) [Eigros et al. ’03]</td>
</tr>
<tr>
<td>L64.</td>
<td>Community structure [Blake et al. ’02, Girvan and Newman ’02]</td>
</tr>
<tr>
<td>L65.</td>
<td>Preferential Attachment (PA) [Barabási et al. ’99]</td>
</tr>
<tr>
<td>L66.</td>
<td>Small world [Watts and Strogatz ’98]</td>
</tr>
<tr>
<td>L67.</td>
<td>Preferential Attachment [Barabási et al. ’99]</td>
</tr>
<tr>
<td>L68.</td>
<td>Diameter [Zweig et al. ’08]</td>
</tr>
</tbody>
</table>

Dynamic

- L70. Triangle Power Law (TPL) [Tsourakakis ’08]
- L71. Eigencentral Power Law (EPL) [Eigros et al. ’03]
- L72. Community structure [Blake et al. ’02, Girvan and Newman ’02]
- L73. Preferential Attachment (PA) [Barabási et al. ’99]
- L74. Small world [Watts and Strogatz ’98]
- L75. Preferential Attachment [Barabási et al. ’99]
- L76. Diameter [Zweig et al. ’08]


Roadmap

- Introduction – Motivation
- Part#1: Patterns in graphs
  - Patterns
  - Anomaly / fraud detection
    - CopyCatch
    - Spectral methods (‘fBox’)
    - Belief Propagation
- Part#2: time-evolving graphs; tensors
- Conclusions
Fraud

• Given
  – Who ‘likes’ what page, and when

• Find
  – Suspicious users and suspicious products

Fraud

• Given
  – Who ‘likes’ what page, and when

• Find
  – Suspicious users and suspicious products

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

- Amazon’16

(c) 2016, C. Faloutsos
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. **Repeat** until convergence
Deployment at Facebook

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

3 months of *CopyCatch* @ Facebook

![Graph showing number of users caught over time]
Deployment at Facebook

Manually labeled 22 randomly selected clusters from February 2013
Deployment at Facebook

Fake acct

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

Manually labeled 22 randomly selected clusters from February 2013

(c) 2016, C. Faloutsos
Roadmap

• Introduction – Motivation
• Part#1: Patterns in graphs
  – Patterns
  – Anomaly / fraud detection
    • CopyCatch
    • Spectral methods (‘fBox’)
    • Belief Propagation
• Part#2: time-evolving graphs; tensors
• Conclusions
Problem: Social Network Link Fraud

Target: find “stealthy” attackers missed by other algorithms

Bipartite core

Clique
41.7M nodes
1.5B edges

Amazon'16
(c) 2016, C. Faloutsos
Problem: Social Network Link Fraud

Target: find “stealthy” attackers missed by other algorithms

Lekan Olawole Lowe @loweinc 26 Jul 09
Sign up free and Get 400 followers a day using http://tweeteradder.com

Lekan Olawole Lowe @loweinc 26 Jul 09
Get 400 followers a day using http://www.tweeterfollow.com

Takeaway: use reconstruction error between true/latent representation!

Roadmap

• Introduction – Motivation
• Part#1: Patterns in graphs
  – Patterns
  – Anomaly / fraud detection
    • CopyCatch
    • Spectral methods (‘fBox’, suspiciousness)
    • Belief Propagation
• Part#2: time-evolving graphs; tensors
• Conclusions
A General Suspiciousness Metric for Dense Blocks in Multimodal Data, Meng Jiang, Alex Beutel, Peng Cui, Bryan Hooi, Shiqiang Yang, and Christos Faloutsos, *ICDM*, 2015.
Suspicious Patterns in Event Data

Which is more suspicious?

20,000 Users
Retweeting same 20 tweets
6 times each
All in 10 hours

vs.

225 Users
Retweeting same 1 tweet
15 times each
All in 3 hours
All from 2 IP addresses

Answer: volume * $D_{KL}(p \parallel p_{\text{background}})$
## Suspicious Patterns in Event Data

### Retweeting: “Galaxy Note Dream Project: Happy Happy Life Traveling the World”

<table>
<thead>
<tr>
<th></th>
<th>User × tweet × IP × minute</th>
<th>Mass</th>
<th>Suspiciousness</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CROSSSPOT</strong></td>
<td>1 14×1×2×1,114</td>
<td>41,396</td>
<td>1,239,865</td>
</tr>
<tr>
<td></td>
<td>2 225×1×2×200</td>
<td>27,313</td>
<td>777,781</td>
</tr>
<tr>
<td></td>
<td>3 8×2×4×1,872</td>
<td>17,701</td>
<td>491,323</td>
</tr>
<tr>
<td><strong>HOSVD</strong></td>
<td>1 24×6×11×439</td>
<td>3,582</td>
<td>131,113</td>
</tr>
<tr>
<td></td>
<td>2 18×4×5×223</td>
<td>1,942</td>
<td>74,087</td>
</tr>
<tr>
<td></td>
<td>3 14×2×1×265</td>
<td>9,061</td>
<td>381,211</td>
</tr>
</tbody>
</table>

---

(c) 2016, C. Faloutsos
Roadmap

• Introduction – Motivation

• Part#1: Patterns in graphs
  – Patterns
  – Anomaly / fraud detection
    • CopyCatch
    • Spectral methods (‘fBox’)
    • Belief Propagation

• Part#2: time-evolving graphs; tensors

• Conclusions
E-bay Fraud detection

w/ Polo Chau & Shashank Pandit, CMU
[www’07]
E-bay Fraud detection
E-bay Fraud detection
E-bay Fraud detection - NetProbe

Inspection of a user's network to identify suspicious activity.
Popular press

And less desirable attention:
• E-mail from ‘Belgium police’ (‘copy of your code’?)
Roadmap

• Introduction – Motivation
• Part#1: Patterns in graphs
  – Patterns
  – Anomaly / fraud detection
    • CopyCatch
    • Spectral methods (‘fBox’)
    • Belief Propagation; fast computation & unification
• Part#2: time-evolving graphs; tensors
• Conclusions
Unifying Guilt-by-Association Approaches: Theorems and Fast Algorithms

Danai Koutra
U Kang
Hsing-Kuo Kenneth Pao

Tai-You Ke
Duen Horng (Polo) Chau
Christos Faloutsos

ECML PKDD, 5-9 September 2011, Athens, Greece
Problem Definition:
GBA techniques

Given: Graph; & few labeled nodes
Find: labels of rest (assuming network effects)
Are they related?

• RWR (Random Walk with Restarts)
  – google’s pageRank (‘if my friends are important, I’m important, too’)

• SSL (Semi-supervised learning)
  – minimize the differences among neighbors

• BP (Belief propagation)
  – send messages to neighbors, on what you believe about them
Are they related? YES!

- **RWR (Random Walk with Restarts)**
  - google’s pageRank (‘if my friends are important, I’m important, too’)
- **SSL (Semi-supervised learning)**
  - minimize the differences among neighbors
- **BP (Belief propagation)**
  - send messages to neighbors, on what you believe about them
# Correspondence of Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Matrix</th>
<th>Unknown</th>
<th>known</th>
</tr>
</thead>
<tbody>
<tr>
<td>RWR</td>
<td>$[I - cAD^{-1}]$</td>
<td>$x$</td>
<td>$(1-c)y$</td>
</tr>
<tr>
<td>SSL</td>
<td>$[I + a(D - A)]$</td>
<td>$x$</td>
<td>$y$</td>
</tr>
<tr>
<td>FABP</td>
<td>$[I + aD - c'A]$</td>
<td>$b_h$</td>
<td>$\phi_h$</td>
</tr>
</tbody>
</table>

---

Adjacency matrix:

$$
\begin{pmatrix}
1 & 1 & 1 \\
1 & d1 & d2 \\
d3 & d2 & 0 & 1 & 0 \\
\end{pmatrix}
$$

Final labels/beliefs:

$$
\begin{pmatrix}
0 & 1 & 0 \\
1 & 0 & 1 \\
0 & 1 & 0 \\
\end{pmatrix}
$$

Prior labels/beliefs:

$$
\begin{pmatrix}
0 \\
1 \\
1 \\
\end{pmatrix}
$$
Results: Scalability

FABP is **linear** on the number of edges.

Amazon'16
(c) 2016, C. Faloutsos
Summary of Part #1

• *many* patterns in real graphs
  – Power-laws everywhere
  – Gaussian trap
    • Avg << Max
  – 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; tensors
  – P2.1: time-evolving graphs
  – [P2.2: with side information (‘coupled’ M.T.F.)
  – Speed]
• 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 (e.g., side info)
  - Given subject – verb – object facts
    - And voxel-activity for each subject-word
  - Find patterns / anomalies
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]
• Conclusions
Answer to both: tensor factorization

- Recall: (SVD) matrix factorization: finds blocks

\[
\begin{align*}
\text{N users} & \quad \uparrow \quad \downarrow \quad \text{M products} \\
\text{~} & \quad \frac{\text{~}}{\text{~}} \\
\text{\textbf{\textcolor{green}{\textit{meat-eaters}}}} & \quad \textbf{\textcolor{yellow}{\textit{steaks}}} & \quad \textbf{\textcolor{blue}{\textit{vegetarians}}} & \quad \textbf{\textcolor{pink}{\textit{cooking}}} \\
\textbf{\textcolor{green}{\textit{plants}}} & \quad \textbf{\textcolor{blue}{\textit{kids}}} & \quad \textbf{\textcolor{pink}{\textit{cooking}}} \\
\end{align*}
\]

\[
\begin{align*}
\mathbf{U} & \approx \mathbf{A} \\
& = \mathbf{U}_1 \\
& + \mathbf{U}_{105}
\end{align*}
\]

Amazon'16
(c) 2016, C. Faloutsos
Answer to both: tensor factorization

• PARAFAC decomposition

\[ \text{subject} = \text{politicians} + \text{artists} + \text{athletes} \]
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

〜200 calls to EACH receiver on EACH day!

(c) 2016, C. Faloutsos
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!

Amazon'16

(c) 2016, C. Faloutsos
Anomaly detection in time-evolving graphs

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

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]
• Conclusions
Coupled Matrix-Tensor Factorization (CMTF)

User-attr.

Y

user

product

X

timestamp

a

b_1

b_F

c_1

c_F

da_1

da_F

Amazon'16

(c) 2016, C. Faloutsos
Coupled Matrix-Tensor Factorization (CMTF)
**Neuro-semantics**

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

• **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?**

- airplane
- dog
- questions
- nouns
- voxels
- persons

---

Amazon'16 (c) 2016, C. Faloutsos
Nouns
- Group 1: beetle, pants, bee
- Group 2: bear, cow, coat
- Group 3: glass, tomato, bell
- Group 4: bed, house, car

Questions
- Group 1: can it cause you pain? do you see it daily? is it conscious?
- Group 2: does it grow? is it alive? was it ever alive?
- Group 3: can you pick it up? can you hold it in one hand? is it smaller than a golfball?
- Group 4: does it use electricity? can you sit on it? does it cast a shadow?

Figure 4: Heat maps of brain activity in different groups. Group 1, which mostly refers to insects, such as bee or beetle. Additionally, Group 3 shows high activation in the Premotor Cortex.
Neuro-semantics

Small items -> Premotor cortex

Nouns
- glass
- tomato
- bell

Questions
- can you pick it up?
- can you hold it in one hand?
- is it smaller than a golfball?

Group 3

Premotor Cortex

Figure 4:
Small items -> Premotor cortex

Nouns
- glass
- tomato
- bell

Questions
- can you pick it up?
- can you hold it in one hand?
- is it smaller than a golfball?

Evangelos Papalexakis, Tom Mitchell, Nicholas Sidiropoulos, Christos Faloutsos, Partha Pratim Talukdar, Brian Murphy, Turbo-SMT: Accelerating Coupled Sparse Matrix-Tensor Factorizations by 200x, SDM 2014
Part 2: Conclusions

- Time-evolving / heterogeneous graphs -> tensors
- PARAFAC finds patterns
- (GigaTensor/HaTen2 -> fast & scalable)
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

Amazon'16
(c) 2016, C. Faloutsos
124
CONCLUSION#1 – Big data

- Patterns • Anomalies
- Large datasets reveal patterns/outliers that are invisible otherwise
CONCLUSION#2 – tensors

- powerful tool

= \begin{array}{c}
\text{1 caller} \\
\text{5 receivers} \\
\text{4 days of activity}
\end{array}
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
My goals wrt sabbatical @ amzn

• Win-win setting:
• For Amazon: state-of-the-art methods for anomaly detection
• For my students: focus on real-world research problems
My goals wrt sabbatical @ amzn

• Win-win setting:
• For Amazon: state-of-the-art methods for anomaly detection
• For my students: focus on real-world research problems