15-826: Multimedia Databases and Data Mining

Lecture #26: Graph mining - patterns

*Christos Faloutsos*

Must-read Material

- Jure Leskovec, Jon Kleinberg, Christos Faloutsos Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations, KDD 2005, Chicago, IL, USA

Must-read Material (cont’d)

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
- Problem#3: Scalability
- Conclusions

Graphs - why should we care?

- IR: bi-partite graphs (doc-terms)
- ... and more:
- web: hyper-text graph

Internet Map [lumeta.com]
Food Web [Martinez '91]
Friendship Network [Moody '01]
Linkedin, Facebook, Twitter

Graphs - why should we care?
Graphs - why should we care?

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

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - Static graphs
  - Weighted graphs
  - Time evolving graphs
- Problem#2: Tools
- Problem#3: Scalability
- Conclusions

Problem #1 - network and graph mining

- What does the Internet look like?
- What does FaceBook look like?
- What is 'normal'/'abnormal'?  
- which patterns/laws hold?
Problem #1 - network and graph mining

- What does the Internet look like?
- What does FaceBook look like?
- What is ‘normal’/‘abnormal’?
- which patterns/laws hold?
  - To spot anomalies (rarities), we have to discover patterns

Large datasets reveal patterns/anomalies that may be invisible otherwise…

Are real graphs random?

- random (Erdos-Renyi) graph – 100 nodes, avg degree = 2
- before layout
- after layout
- No obvious patterns

(generated with: pajek
http://vlado.fmf.uni-lj.si/pub/networks/pajek/)
Graph mining

- Are real graphs random?

Laws and patterns

- Are real graphs random?
- A: NO!!
  - Diameter
  - 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

log(degree) vs. log(rank)

<table>
<thead>
<tr>
<th>att.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibm.com</td>
</tr>
</tbody>
</table>
Solution# S.1

- Power law in the degree distribution [SIGCOMM99]

\[ \log(\text{rank}) = -0.82 \log(\text{degree}) \]

internet domains

Solution# S.2: Eigen Exponent $E$

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

\[ E = -0.48 \]

Exponent = slope

Rank of decreasing eigenvalue

- \[ \text{Mihail, Papadimitriou '02: slope is } \frac{1}{2} \text{ of rank exponent} \]

Exponent = slope

May 2001
But:
How about graphs from other domains?

More power laws:
- web hit counts [w/ A. Montgomery]

epinions.com
- who-trusts-whom
  [Richardson + Domingos, KDD 2001]
And numerous more

- # of sexual contacts
- Income [Pareto] – ‘80-20 distribution’
- Duration of downloads [Bestavros+]
- Duration of UNIX jobs (‘mice and elephants’)
- Size of files of a user
- ...
- ‘Black swans’

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - Static graphs
    - degree, diameter, eigen,
    - triangles
    - cliques
  - Weighted graphs
  - Time evolving graphs
- Problem#2: Tools

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?

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

X-axis: # of participating triangles
Y: count (~ pdf)
Triangle Law: #S.4
[Tsourakakis ICDM 2008]

X-axis: degree
Y-axis: mean # triangles

n friends -> ~$n^{1/2}$ triangles

---

Triangle Law: Computations
[Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos)
Q: Can we do that quickly?

A: Yes!

$\#\text{triangles} = \frac{1}{6} \text{Sum } (\lambda_i^3)$

(and, because of skewness (S2), we only need the top few eigenvalues!)
Triangle Law: Computations
[Tsourakakis ICDM 2008]
Wikipedia graph 2006-Nov-04
= 116M nodes = 77M edges

1000x+ speed-up, >90% accuracy

Triangle counting for large graphs?

Anomalous nodes in Twitter (~3 billion edges)
[U Kang, Brendan Meeder, +, PAKDD’11]
Any other ‘laws’?

Yes!

• Small diameter (~ constant!) –
  – six degrees of separation / ‘Kevin Bacon’
  – small worlds [Watts and Strogatz]

Any other ‘laws’?

• Bow-tie, for the web [Kumar+ ‘99]
• IN, SCC, OUT, ‘tendrils’
• disconnected components
Any other ‘laws’?

- power-laws in communities (bi-partite cores) [Kumar+, ’99]

- “Jellyfish” for Internet [Tauro+ ’01]
  - core: ~clique
  - ~5 concentric layers
  - many 1-degree nodes

EigenSpokes

EigenSpokes

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

\[ A = U \Sigma U^T \]
EigenSpokes

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

\[ A = U \Sigma U^T \]

EE plot:
- Scatter plot of scores of \( u_1 \) vs \( u_2 \)
- One would expect
  - Many points @ origin
  - A few scattered ~randomly
**EigenSpokes**

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

**EigenSpokes - pervasiveness**

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

**EigenSpokes - explanation**

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

So what?
- Extract nodes with high scores
- high connectivity
- Good “communities”
**Bipartite Communities!**

- patents from same inventor(s)
- 'cut-and-paste' bibliography!
- magnified bipartite community

---

**Outline**

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - Static graphs
    - degree, diameter, eigen,
    - triangles
    - cliques
  - Weighted graphs
  - Time evolving graphs
- Problem#2: Tools

---

**Observations on weighted graphs?**

- A: yes - even more ‘laws’!

M. McGlohon, L. Akoglu, and C. Faloutsos
*Weighted Graphs and Disconnected Components: Patterns and a Generator.*
*SIG-KDD 2008*
Observation W.1: Fortification

Q: How do the weights of nodes relate to degree?

More donors, more $?

$10
$5
$7

'Reagan'

'Clinton'

Observation W.1: fortification: Snapshot Power Law

- Weight: super-linear on in-degree
- Exponent ‘iw’: $1.01 < iw < 1.26$

More donors, even more $
Outline

• Introduction – Motivation
• Problem#1: Patterns in graphs
  – Static graphs
  – Weighted graphs
  – Time evolving graphs
• Problem#2: Tools
• …

Problem: Time evolution

• with Jure Leskovec (CMU -> Stanford)

• and Jon Kleinberg (Cornell – sabb. @ CMU)

T.1 Evolution of the Diameter

• Prior work on Power Law graphs hints at slowly growing diameter:
  – diameter ~ O(log N)
  – diameter ~ O(log log N)
• What is happening in real data?
T.1 Evolution of the Diameter

- Prior work on Power Law graphs hints at slowly growing diameter:
  - diameter $\sim O(\log N)$
  - diameter $\sim O(\log \log N)$
- What is happening in real data?
- Diameter shrinks over time

T.1 Diameter – “Patents”

- Patent citation network
- 25 years of data
- @1999
  - 2.9 M nodes
  - 16.5 M edges

T.2 Temporal Evolution of the Graphs

- $N(t)$ … nodes at time $t$
- $E(t)$ … edges at time $t$
- Suppose that
  $N(t+1) = 2 \times N(t)$
- Q: what is your guess for
  $E(t+1) = ? 2 \times E(t)$
T.2 Temporal Evolution of the Graphs

- \( N(t) \) … nodes at time \( t \)
- \( E(t) \) … edges at time \( t \)
- Suppose that \( N(t+1) = 2 \times N(t) \)
- Q: what is your guess for \( E(t+1) \)? \( \neq 2 \times E(t) \)
- A: over-doubled!
  - But obeying the "Densification Power Law"

T.2 Densification – Patent Citations

- Citations among patents granted
  - @1999
    - 2.9 M nodes
    - 16.5 M edges
  - Each year is a datapoint

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
  - Static graphs
  - Weighted graphs
  - Time evolving graphs
- Problem#2: Tools
  - ...
More on Time-evolving graphs

M. McGlohon, L. Akoglu, and C. Faloutsos
*Weighted Graphs and Disconnected Components: Patterns and a Generator.*
*SIG-KDD 2008*

[ Gelling Point ]

- Most real graphs display a gelling point
- After gelling point, they exhibit typical behavior. This is marked by a spike in diameter.

Observation T.3: NLCC behavior

Q: How do NLCC’s emerge and join with the GCC?

(“NLCC” = non-largest conn. components)
- Do they continue to grow in size?
- or do they shrink?
- or stabilize?
Observation T.3: NLCC behavior

Q: How do NLCC’s emerge and join with the GCC?

("NLCC" = non-largest conn. components)
– Do they continue to grow in size?
– or do they shrink?
– or stabilize?

After the gelling point, the GCC takes off, but NLCC’s remain ~constant (actually, oscillate).
Timing for Blogs

- with Mary McGlohon (CMU->Google)
- Jure Leskovec (CMU->Stanford)
- Natalie Glance (now at Google)
- Mat Hurst (now at MSR)

[SDM’07]

T.4 : popularity over time

Post popularity drops-off – exponentially?

POWER LAW!

Exponent?
T.4: popularity over time

-1.6 exponent?
- close to -1.5: Barabasi’s stack model
- and like the zero-crossings of a random walk

Post popularity drops-off – exponentially?

POWER LAW!

-1.6 days after post


Figure 1: The correspondence patterns of Darwin and Einstein.

T.5: duration of phonecalls

*Surprising Patterns for the Call Duration Distribution of Mobile Phone Users*

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

PKDD 2010
'TLaC: Lazy Contractor'

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

Odds ratio =

\[
\text{Casualties(<x)} : \text{Survivors(\geq x)}
\]

\[== \text{power law}\]
Data Description

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

---

Outliers:

---

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - OddBall (anomaly detection)
  - Belief Propagation
  - Immunization
- Problem#3: Scalability
- Conclusions
Main idea

For each node,
• extract ‘ego-net’ (=1-step-away neighbors)
• Extract features (#edges, total weight, etc etc)
• Compare with the rest of the population

What is an egonet?
Selected Features

- $N_i$: number of neighbors (degree) of ego $i$
- $E_i$: number of edges in egonet $i$
- $W_i$: total weight of egonet $i$
- $\lambda_{max}$: principal eigenvalue of the weighted adjacency matrix of egonet $I$
Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  – OddBall (anomaly detection)
  – Belief Propagation
  – Immunization
- Problem#3: Scalability
- Conclusions
E-bay Fraud detection - NetProbe

Popular press

The Washington Post
Los Angeles Times

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

Outline
- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - OddBall (anomaly detection)
  - Belief Propagation – antivirus app
  - Immunization
- Problem#3: Scalability
- Conclusions
Polonium: Tera-Scale Graph Mining and Inference for Malware Detection

SDM 2011, Mesa, Arizona

Polo Chau
Machine Learning Dept
Carey Nachenberg
Vice President & Fellow
Jeffrey Wilhelm
Principal Software Engineer

Adam Wright
Software Engineer
Prof. Christos Faloutsos
Computer Science Dept

Polonium: The Data

- 60+ terabytes of data anonymously contributed by participants of worldwide Norton Community Watch program
- 50+ million machines
- 900+ million executable files

- Constructed a machine-file bipartite graph (0.2 TB+)
- 1 billion nodes (machines and files)
- 37 billion edges

Polonium: Key Ideas

- Use Belief Propagation to propagate domain knowledge in machine-file graph to detect malware
- Use “guilt-by-association” (i.e., homophily)
  - E.g., files that appear on machines with many bad files are more likely to be bad
- Scalability: handles 37 billion-edge graph
Polonium: One-Interaction Results

- True Positive Rate
  - % of malware correctly identified
- False Positive Rate
  - % of non-malware wrongly labeled as malware

84.9% True Positive Rate
1% False Positive Rate

Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
  - OddBall (anomaly detection)
  - Belief propagation
  - Immunization
- Problem#3: Scalability - PEGASUS
- Conclusions

Immunization and epidemic thresholds

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

• Given
  • a network,
  • k vaccines, and
  • the virus details
• Which nodes to immunize?
Q1: Immunization:
• Given a network, k vaccines, and the virus details
• Which nodes to immunize?

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

Q2: will a virus take over?
• Flu-like virus (no immunity, ’SIS’)
• Mumps (life-time immunity, ’SIR’)
• Pertussis (finite-length immunity, ’SIRS’)

β: attack prob
δ: heal prob

A: depends on connectivity (avg degree? Max degree? variance? Something else?)
Epidemic threshold $\tau$

What should $\tau$ depend on?
- avg. degree? and/or highest degree?
- and/or variance of degree?
- and/or third moment of degree?
- and/or diameter?

• [Theorem] We have no epidemic, if

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

Proof: [Wang+03] (for SIS=flu only)
A2: will a virus take over?

- For all typical virus propagation models (flu, mumps, pertussis, HIV, etc)
- The only connectivity measure that matters, is $1/\lambda_1$
  - the first eigenvalue of the adj. matrix
  - [Prakash+, ‘10, arxiv]

Thresholds for some models

- $s = \text{effective strength}$
- $s < 1 : \text{below threshold}$

<table>
<thead>
<tr>
<th>Models</th>
<th>Effective Strength ((s))</th>
<th>Threshold (tipping point)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIS, SIR, SIRS, SEIR</td>
<td>$s = \lambda \cdot \left( \frac{\beta}{\delta} \right)$</td>
<td>$s = 1$</td>
</tr>
<tr>
<td>SIV, SEIV</td>
<td>$s = \lambda \cdot \left( \frac{\beta y}{\delta (\gamma + \theta)} \right)$</td>
<td>$s = 1$</td>
</tr>
<tr>
<td>$SI_1, I_2, V_1, V_2$ (H.I.V.)</td>
<td>$s = \lambda \cdot \left( \frac{\beta V_1 + \beta_0 V_2}{\nu_1 (\nu_1 + \nu_2)} \right)$</td>
<td>$s = 1$</td>
</tr>
</tbody>
</table>

A2: will a virus take over?

- Fraction of infected
- Graph: Portland, OR
  - 31M links
  - 1.5M nodes
- Time ticks
  - $10^0$ to $10^5$
- SIRS Infected (log-log)
  - Under1, Under2, Over1, Over2
  - Above: take-over
  - Below: exp. extinction

(c) 2011 C. Faloutsos
Q1: Immunization:

Given
• a network,
• k vaccines, and
• the virus details

Which nodes to immunize?

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

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

Outline

• Introduction – Motivation
• Problem#1: Patterns in graphs
• Problem#2: Tools
  – OddBall (anomaly detection)
  – Belief propagation
  – Immunization
• Problem#3: Scalability -PEGASUS
• Conclusions
Scalability

- Yahoo: 5Pb of data [Fayyad, KDD’07]
- Problem: machine failures, on a daily basis
- How to parallelize data mining tasks, then?

Outline – Algorithms & results

<table>
<thead>
<tr>
<th></th>
<th>Centralized</th>
<th>Hadoop/PEGASUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degree Distr.</td>
<td>old</td>
<td>old</td>
</tr>
<tr>
<td>Pagerank</td>
<td>old</td>
<td>old</td>
</tr>
<tr>
<td>Diameter/ANF</td>
<td>old</td>
<td>HERE</td>
</tr>
<tr>
<td>Conn. Comp</td>
<td>old</td>
<td>HERE</td>
</tr>
<tr>
<td>Triangles</td>
<td>done</td>
<td>HERE</td>
</tr>
<tr>
<td>Visualization</td>
<td>started</td>
<td></td>
</tr>
</tbody>
</table>

HADI for diameter estimation

- Radius Plots for Mining Tera-byte Scale Graphs U Kang, Charalampos Tsourakakis, Ana Paula Appel, Christos Faloutsos, Jure Leskovec, SDM’10
- Naively: diameter needs $O(N^2)$ space and up to $O(N^3)$ time – prohibitive (N~1B)
- Our HADI: linear on E (~10B)
  - Near-linear scalability wrt # machines
  - Several optimizations -> 5x faster
YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)
- Largest publicly available graph ever studied.
YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)

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

Radius

Count

10^1
10^2
10^3
10^4
10^5
10^6
10^7
10^8
10^9

0 5 10 15 20 25 30

14 (dir.)

19+? [Barabasi+]

-7 (undir.)

Radius

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

Q: Shape?

Effective Diameter = 7.62

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

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

(c) 2011 C. Faloutsos
YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)
• effective diameter: surprisingly small.
• Multi-modality: probably mixture of cores.
YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)
- effective diameter: surprisingly small.
- Multi-modality: probably mixture of cores.

Conjecture:
- ~7

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

Outline – Algorithms & results

<table>
<thead>
<tr>
<th></th>
<th>Centralized</th>
<th>Hadoop/PEGASUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degree Distr.</td>
<td>old</td>
<td>old</td>
</tr>
<tr>
<td>Pagerank</td>
<td>old</td>
<td>old</td>
</tr>
<tr>
<td>Diameter/ANF</td>
<td>old</td>
<td>HERE</td>
</tr>
<tr>
<td>Conn. Comp</td>
<td>old</td>
<td>HERE</td>
</tr>
<tr>
<td>Triangles</td>
<td></td>
<td>HERE</td>
</tr>
<tr>
<td>Visualization</td>
<td>started</td>
<td></td>
</tr>
</tbody>
</table>
Generalized Iterated Matrix Vector Multiplication (GIMV)

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

---

Example: GIM-V At Work

- Connected Components – 4 observations:
Example: GIM-V At Work

- Connected Components

1) 10K x larger than next

2) ~0.7B singleton nodes

3) SLOPE!
Example: GIM-V At Work

- Connected Components

4) Spikes!

Example: GIM-V At Work

- Connected Components

GIM-V At Work

- Connected Components over Time

LinkedIn: 7.5M nodes and 58M edges

Stable tail slope after the gelling point
Outline

- Introduction – Motivation
- Problem#1: Patterns in graphs
- Problem#2: Tools
- Problem#3: Scalability
- Conclusions

OVERALL CONCLUSIONS – low level:

- Several new patterns (fortification, shrinking diameter, triangle-laws, conn. components, etc)
- New tools:
  - anomaly detection (OddBall), belief propagation, immunization
- Scalability: PEGASUS / hadoop

OVERALL CONCLUSIONS – high level

- BIG DATA: Large datasets reveal patterns/outliers that are invisible otherwise
References


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


• Jimeng Sun, Yinglian Xie, Hui Zhang, Christos Faloutsos. Less is More: Compact Matrix Decomposition for Large Sparse Graphs, SDM, Minneapolis, Minnesota, Apr 2007.

• Jimeng Sun, Spiros Papadimitriou, Philip S. Yu, and Christos Faloutsos. GraphScope: Parameter-free Mining of Large Time-evolving Graphs ACM SIGKDD Conference, San Jose, CA, August 2007
References

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

References

• Hanghang Tong, Christos Faloutsos, and Jia-Yu Pan, Fast Random Walk with Restart and Its Applications, ICDM 2006, Hong Kong.

• Hanghang Tong, Christos Faloutsos, Center-Piece Subgraphs: Problem Definition and Fast Solutions, KDD 2006, Philadelphia, PA

References

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

References

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

www.cs.cmu.edu/~pegasus

Chau, Polo
Koutra, Danae
Prakash, Aditya

Akoglu, Leman
Kang, U
McGlohon, Mary
Tong, Hanghang

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

Faloutsos