**15-826: Multimedia Databases and Data Mining**

Lecture #28: 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

---

**Main outline**

- Introduction
- Indexing
- Mining
  - Graphs – patterns
  - Graphs – generators and tools
  - Association rules
  - …
Outline
• Introduction – Motivation
• Problem#1: Patterns in graphs
• Problem#2: Scalability
• Conclusions

Graphs - why should we care?
• IR: bi-partite graphs (doc-terms)

• web: hyper-text graph

• ... and more:
Outline

- Introduction – Motivation
- Problem #1: Patterns in graphs
  - Static graphs
  - Weighted graphs
  - Time evolving graphs
- Problem #2: 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?

- 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 ('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
  log(degree)
  log(rank)
  att.com
  ibm.com
Solution# S.1

- Power law in the degree distribution
  [SIGCOMM99]

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

internet domains

\(-0.82\)

Q: So what?

A1: # of two-step-away pairs:

\(100^2 \times N = 10\) Trillion

internet domains

friends of friends (F.O.F.)
**Solution S.1**

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

**Observation – big-data:**

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

- ~0.8PB -> a data center(!)

---

**Gaussian trap**

- Such patterns -> New algorithms
Observation – big-data:
• $O(N^2)$ algorithms are ~intractable - $N=1B$
  • $N^2$ seconds = 31B years
  • 1,000 machines

Observation – big-data:
• $O(N^2)$ algorithms are ~intractable - $N=1B$
  • $N^2$ seconds = 31B years
  • 1M machines

Observation – big-data:
• $O(N^2)$ algorithms are ~intractable - $N=1B$
  • $N^2$ seconds = 31B years
  • 10B machines ~ $10Trillion$

Observation – big-data:
• $O(N^2)$ algorithms are ~intractable - $N=1B$
  • $N^2$ seconds = 31B years
  • 10B machines ~ $10Trillion$

And parallelism might not help
Solution S.2: Eigen Exponent $E$

Exponent = slope

$E = -0.48$

May 2001

Eigenvalue

Rank of decreasing eigenvalue

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

Ax = $\lambda x$

But:

How about graphs from other domains?

More power laws:

- web hit counts [w/ A. Montgomery]

Web Site Traffic

Count (log scale)

Zipf

sites

users

in-degree (log scale)

May 2001

Exponent = slope

$E = -0.48$

May 2001

Eigenvalue

Rank of decreasing eigenvalue

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

15-826 (c) C. Faloutsos, 2017

15-826 (c) C. Faloutsos, 2017
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
  - Weighted graphs
  - Time evolving graphs

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]

- ASN
- HEP-TH
- Epinions

X-axis: # of participating triangles
Y: count (~ pdf)

Triangle Law: #S.4
[Tsourakakis ICDM 2008]

- SN
- Reuters
- Epinions

X-axis: degree
Y-axis: mean # triangles
n friends \( \sim n^{1.6} \) triangles

Faloutsos
Triangle Law: Computations
[Tsourakakis ICDM 2008]

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

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

#triangles = 1/6 \( \sum (\lambda_i^3) \)
(and, because of skewness (S2),
we only need the top few eigenvalues!

1000x+ speed-up, >90% accuracy

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]

(c) C. Faloutsos, 2017
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]

\[
\begin{align*}
\text{Log(count)} & = \text{Log(m)} \\
\text{n:1} & \\
\text{n:2} & \\
\text{n:3} & \\
\end{align*}
\]
Any other ‘laws’?

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

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)

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

15-826 (c) C. Faloutsos, 2017

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

15-826 (c) C. Faloutsos, 2017
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

Useful for fraud detection!
Outline

• Introduction – Motivation
• Problem#1: Patterns in graphs
  – Static graphs
    • degree, diameter, eigen,
    • Triangles
  – Weighted graphs
  – Time evolving graphs
• Problem#2: Scalability
• Conclusions

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 $?

‘Reagan’ $10
‘Clinton’ $5
$7
**Observation W.1: fortification:**

**Snapshot Power Law**

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

---

**More donors, even more $**

- $10$
- $5$

---

**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 $\sim O(N^{1/3})$
  - Diameter $\sim O(\log N)$
  - Diameter $\sim O(\log \log N)$
- What is happening in real data?
T.1 Evolution of the Diameter

- Prior work on Power Law graphs hints at \textit{slowly growing diameter}:
  - \([\text{diameter} \sim O(N^{1/3})]\)
  - \([\text{diameter} \sim \log N]\)
  - \([\text{diameter} \sim O(\log \log N)]\)
- What is happening in real data?
- Diameter \textit{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 \(E(t+1) =? 2 \times E(t)\)

- 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 \(E(t+1) =? 2 \times E(t)\)
- 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: Scalability
- Conclusions

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?

• 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?

Close to -1.5: Barabasi’s stack model

And like the zero-crossings of a random walk
-1.5 slope


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

Probably, power law (?)

No Power Law!
‘TLaC: Lazy Contractor’

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

Odds ratio = 
\[
\frac{\text{Casualties(<x)}}{\text{Survivors(\geq x)}}
\]

== power law

Log-logistic distribution

• CDF(t)/(1 - CDF(t)) == OR(t)
• For log-logistic: \( \log[\text{OR}(t)] = \beta + \rho \log(t) \)
Log-logistic distribution

- Logistic distribution: CDF -> sigmoid
- LOG-Logistic distribution:

\[ CDF(x) = \frac{1}{1 + \exp(-x)} \quad \text{CDF}(x) = \frac{1}{1 + \frac{1}{x}} \]

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: 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?
• A: map/reduce – hadoop (open-source clone)
  http://hadoop.apache.org/

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
Faloutsos

YahooWeb graph (120Gb, 1.4B nodes, 6.6 B edges)
- Largest publicly available graph ever studied.

Diameter: shrunk

7 degrees of separation (!)

~7 (undir.)

19+ [Barabasi+]

~1999, ~1M nodes

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

Q: Shape?

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

Radius Plot of GCC of YahooWeb.

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

Conjecture:
- EN
- DE
- BR

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:
  
  ![Graph](image)

  \( \text{Count} \) vs. \( \text{Size} \)

  1) 10K x larger than next
Example: GIM-V At Work

- Connected Components

<table>
<thead>
<tr>
<th>Size</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>~0.7B</td>
<td>singleton nodes</td>
</tr>
</tbody>
</table>

3) SLOPE!

Example: GIM-V At Work

- Connected Components

<table>
<thead>
<tr>
<th>Size</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>300-size cmpt X 500</td>
<td>Why?</td>
</tr>
<tr>
<td>1100-size cmpt X 65</td>
<td>Why?</td>
</tr>
</tbody>
</table>

4) Spikes!

Example: GIM-V At Work

- Connected Components

<table>
<thead>
<tr>
<th>Size</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>suspicious financial-advice sites (not existing now)</td>
</tr>
</tbody>
</table>
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
• DELETE
• Problem#2: Scalability
• Conclusions

OVERALL CONCLUSIONS – low level:

• Several new patterns (fortification, shrinking diameter, triangle-laws, conn. components, etc)
• Log-logistic distribution: ubiquitous
• 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


- 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, *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

(Project info)

www.cs.cmu.edu/~pegasus

Chau, Polo
Koutra, Danae
Prakash, Aditya
Akoglu, Leman
Kang, U
McGlohon, Mary
Tong, Hanghang