大数据分析

Big Graph Mining

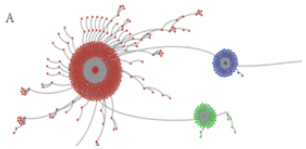
刘盛华

Graphs - why should we care?

>$10B; ~1B users

Graphs - why should we care?

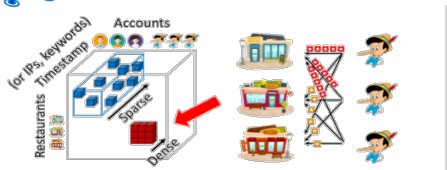
Internet Map
[lumeta.com]



Information Propagation
[J. Gao et al]

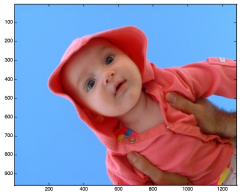
Graphs - why should we care?

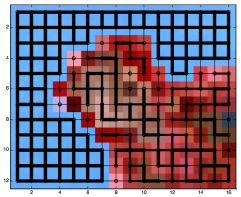
- People's relationships and behaviors



Graphs - why should we care?

- People's relationships and behaviors
- Graphs of images



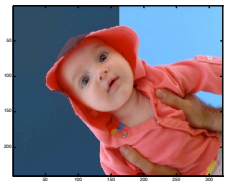


edge weight $e^{-\text{diff}(\text{pixel}i,\text{pixel}j)^2/t^2}$

Shi and Malik, IEEE Trans on PAMI vol. 22, no. 8, 2000

Graphs - why should we care?

- People's relationships and behaviors
- Graphs of images



Second Eigenvector's sparsest cut

Fourth Eigenvector's sparsest cut

Shi and Malik, IEEE Trans on PAMI vol. 22, no. 8 2000

Graphs - why should we care?

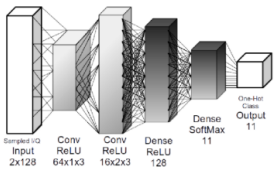
- People's relationships and behaviors
- Graphs of images
- Bank money transferring

2019/9/17

Graphs - why should we care?

- People's relationships and behaviors
- Graphs of images
- Bank money transferring
- Deep Neural Network (sparse decomposition)

稀疏分解

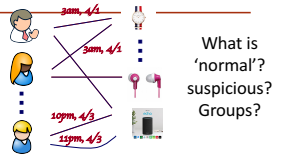


Reduce parameter redundancy using a *sparse decomposition*

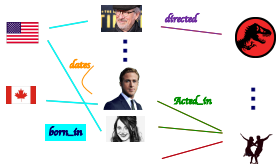
2019/9/17

What else?

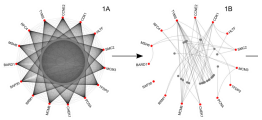
- Several other topics
  - recommendation system
  - Knowledge Graph (KG)
  - genetic graphs, protein networks and disease
  - … … 基因图谱，蛋白质网络和疾病
  - Many-to-many db relationship -> graph



What is 'normal'? suspicious? Groups?

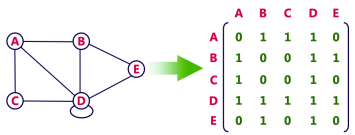


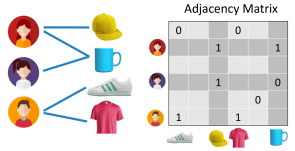
KG: What is 'normal'? Forecast? Spot errors?



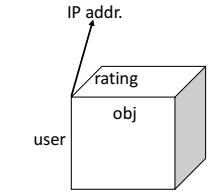
Protein Interactions

Graph, Matrix, and Tensor





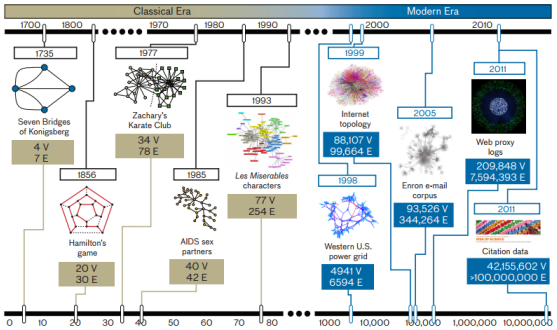
tensor with temporal mode



tensor with more than 3 modes

时间模张量

Overview of Graph Research



Nadya T. Bliss and Matthew C. Schmidt: Confronting the Challenges of Graphs and Networks, LINCOLN LABORATORY JOURNAL, VOL 20, No. 1, 2013

11

Outline

- Patterns in big graphs
- Dense block detection

大图形中的模式
密集块检测

- homogeneous graphs
- bipartite graphs
- multi-attribute graphs
- streaming graphs
- more topics

同构图
二部图
多属性图
流图
更多主题

Carnegie Mellon

中国科学院 CHINESE ACADEMY OF SCIENCES

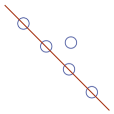
Part of the slides are borrowed from

Christos Faloutsos (CMU)

with permission

Motivating Patterns

- P: patterns? Fraud detection? 欺诈检测

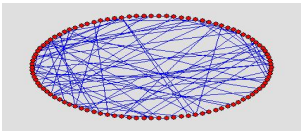


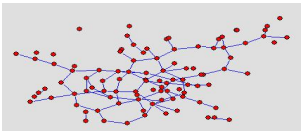
  - To spot anomalies (rarities), we have to discover patterns
  - Large datasets reveal patterns/anomalies that may be invisible otherwise…

大型数据集揭示的模式/异常可能是不可见的

Are real graphs random?

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





(generated with: pajek

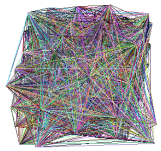
http://vlado.fmf.uni-lj.si/pub/networks/pajek/)

Pajek是大型复杂网络分析工具，
是用于研究目前所存在的各种复杂非线性网络的有力工具

Laws and patterns

- Are real graphs random?

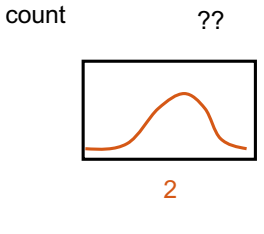
Laws and patterns



- Q: Are real graphs random?
- A: NO!!
  - S.0: Diameter ('6 degrees'; 'Kevin Bacon')
  - in- and out- degree distributions
  - other (surprising) patterns
- So, let's look at the data

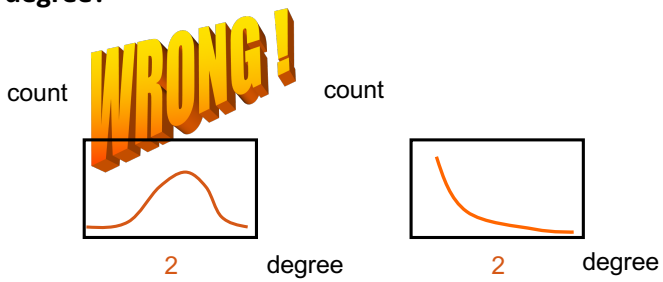
S.1 - degree distributions

- Q: avg degree is ~2 - what is the most probable degree?



S.1 - degree distributions

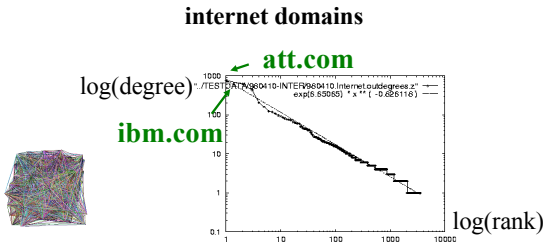
- Q: avg degree is ~2 - what is the most probable degree?



S.1 - degree distributions

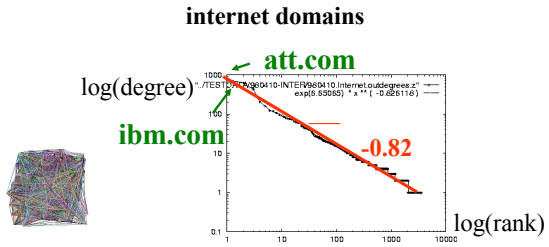
- **Power law** in the degree distribution [SIGCOMM99]

rank-degree plot



S.1 - degree distributions

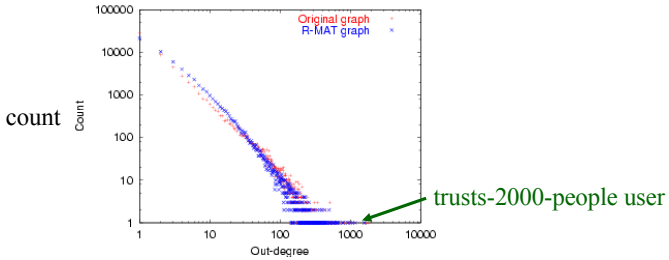
- Power law in the degree distribution [SIGCOMM99]



S.1 - degree distributions

- who-trusts-whom [Richardson + Domingos, KDD 2001]

epinions.com



(out) degree

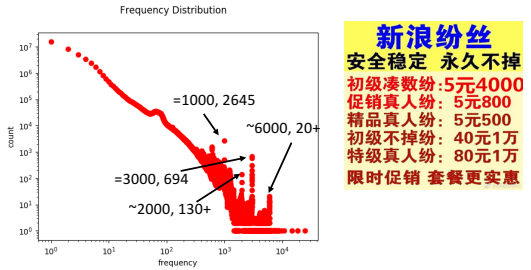
S.1- Skewed distributions

偏态分布

- Zipf
- 80-20
- Pareto
- Rich-get-richer
- Preferential attachment优先连接
- Matthew effect 马太效应
- CRP 反应蛋白
- ...

Anomaly detection 异常检测

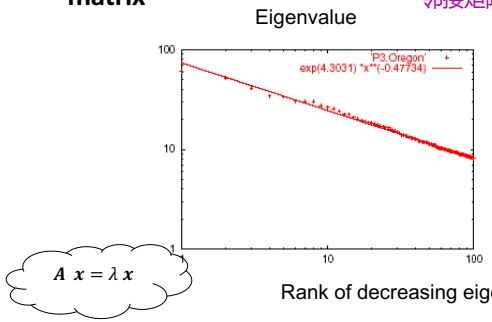
- A real-world log: user-msg



User degree dist

S.2: Eigen Exponent *E* 特征指数E

- A2: power law in the eigenvalues of the adjacency matrix
  邻接矩阵特征值指数法则



Exponent = slope
斜率

$E = -0.48$

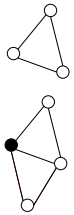
May 2001

S.3: Triangle 'Law' 三角形法则

- Real social networks have a lot of triangles

S.3: Triangle 'Law'

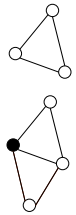
- Real social networks have a lot of triangles
  - Friends of friends are friends
- Any patterns? 2x friends -> 2x triangles ?



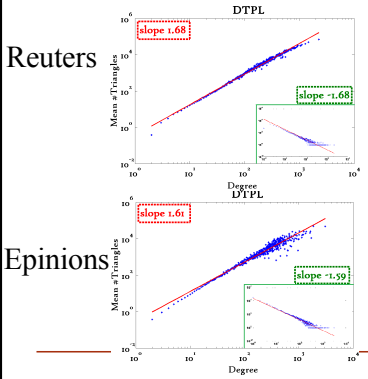
S.3: Triangle 'Law'

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

3x

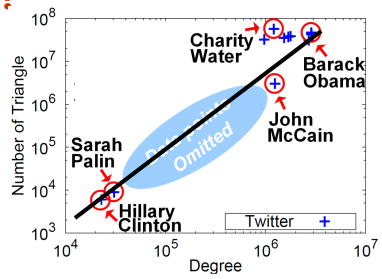


S.3: Triangle Law [Tsourakakis ICDM 2008]



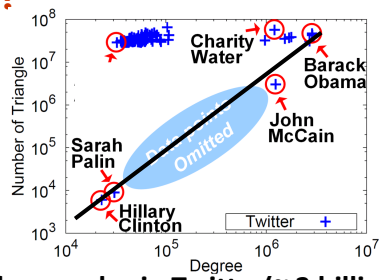
X-axis: degree
Y-axis: mean # triangles
n friends -> ~$n^{1.6}$ triangles

S.3: Triangle counting for large graphs?



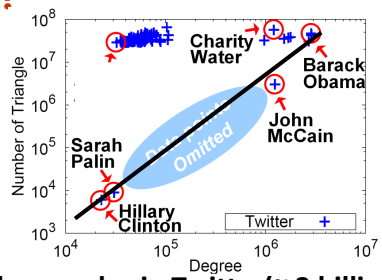
Anomalous nodes in Twitter(~ 3 billion edges)

S.3: Triangle counting for large graphs?



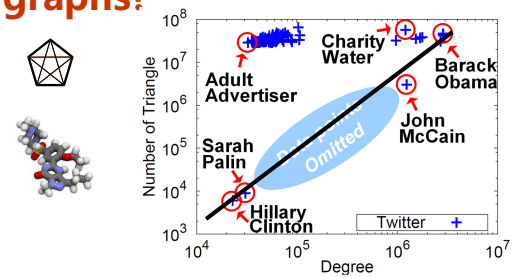
Anomalous nodes in Twitter(~ 3 billion edges)

S.3: Triangle counting for large graphs?



Anomalous nodes in Twitter(~ 3 billion edges)

S.3: Triangle counting for large graphs?



Anomalous nodes in Twitter(~ 3 billion edges)

Carnegie Mellon

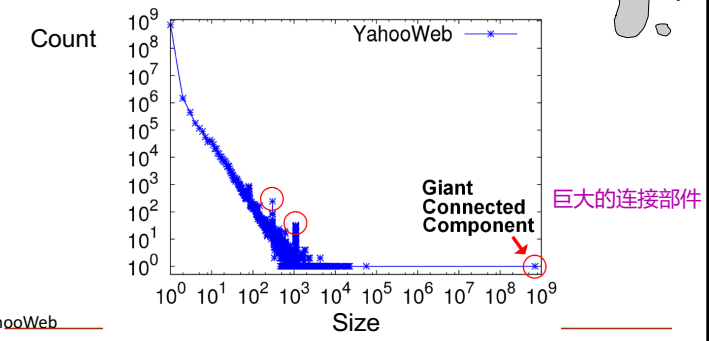
Generalized Iterated Matrix Vector Multiplication (GIMV)

广义迭代矩阵向量乘

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).

S.4: Conn. components

- **Connected Components**

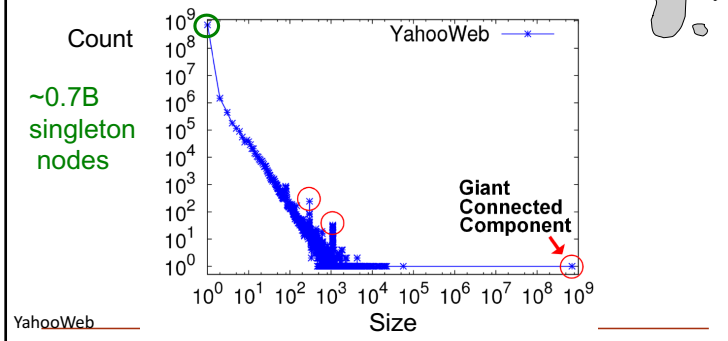


巨大的连接部件

YahooWeb

S.4: Conn. components

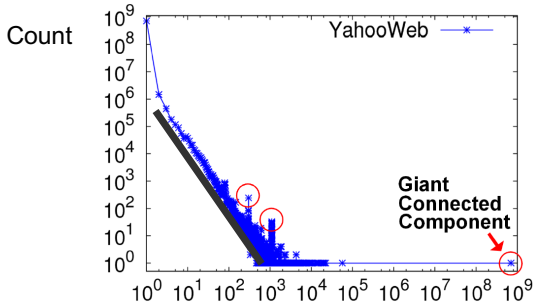
- **Connected Components**



YahooWeb

S.4: Conn. components

- Connected Components



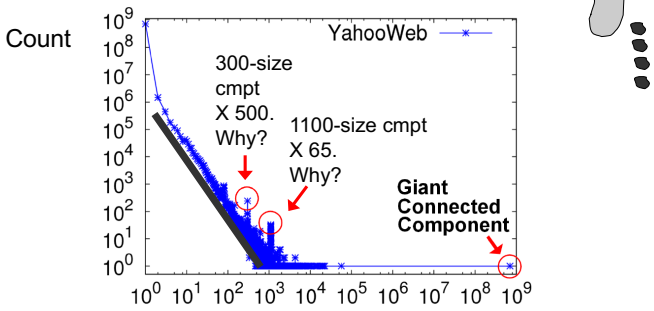
YahooWeb

U Kang+, PEGASUS: A Peta-Scale Graph Mining System - Implementation and Observations, ICDM 2009.

37

S.4: Conn. components

- Connected Components



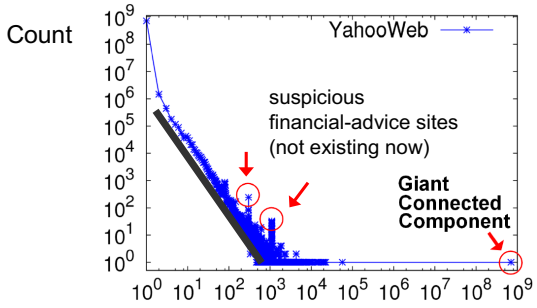
YahooWeb

U Kang+, PEGASUS: A Peta-Scale Graph Mining System - Implementation and Observations, ICDM 2009.

38

S.4: Conn. components

- Connected Components



YahooWeb

U Kang+, PEGASUS: A Peta-Scale Graph Mining System - Implementation and Observations, ICDM 2009.

39

Carnegie Mellon

Problem: Time evolution

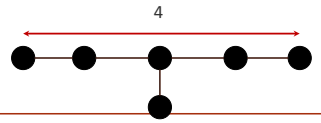
- with Jure Leskovec (CMU -> Stanford)

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

Jure Leskovec, Jon Kleinberg and Christos Faloutsos, *Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations*, KDD 2005 (Best Research paper award; test-of-time award).

T.1 Evolution of the Diameter 直径演变

- Prior work on Power Law graphs hints at slowly growing diameter:
  - diameter ~ O(N $^{1/3}$)
  - diameter ~ O(log N)
  - diameter ~ O(log log N)
- What is happening in real data?

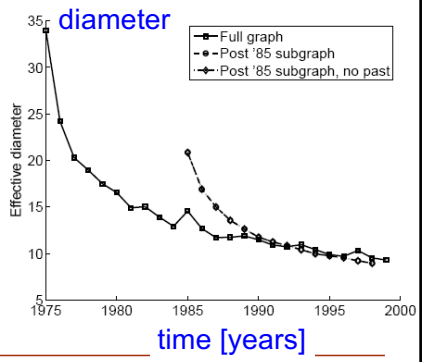


T.1 Evolution of the Diameter

- Prior work on Power Law graphs hints at slowly growing diameter:
  - diameter ~ O(N)
  - diameter ~ O(g N)
  - diameter ~ O(l 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 * N(t)
- Q: what is your guess for

  E(t+1) =? 2 * E(t)

T.2 Temporal Evolution of the Graphs

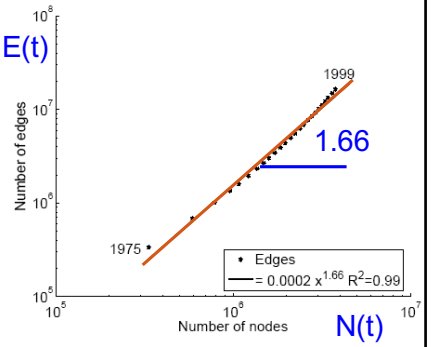
- N(t) … nodes at time t
- E(t) … edges at time t
- Suppose that

  N(t+1) = 2 * N(t)
- Q: what is your guess for

  E(t+1) = 2 * E(t)
- A: over-doubled!
  - But obeying the ``Densification Power Law'' 稠密化幂法则

T.2 Densification – Patent Citations

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



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]

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

Outline

- Patterns in big graphs
- Dense block detection
  - homogeneous graphs
  - bipartite graphs
  - multi-attribute graphs
  - streaming graphs
  - more topics

Methbot creates

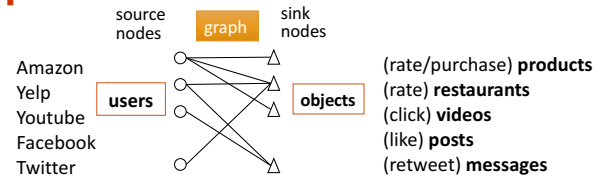
300 Million fake ”reviews” and clicks a day, earning

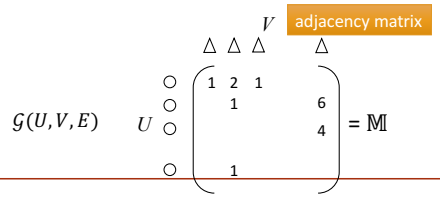
$5 million every day from them,

a report of WhiteOps (ad-fraud-detection company), Dec 2016

Methbot创建3亿个源节点的虚假“评论”
并每天点击，每天从中赚取500万美元

Abstract activities into bipartite Graph 将活动抽象成二部图





Problem of fraud detection

- **Given:**
  - **(user, object, timstamp, #stars)**
- **Find:**
  - **a group of suspicious users, and objects,** 找可疑用户和目标
- **To optimize:**
  - **the metric of suspiciousness from topology, rating time and scores.** 从拓扑结构、评分时间和分数来衡量可疑性。

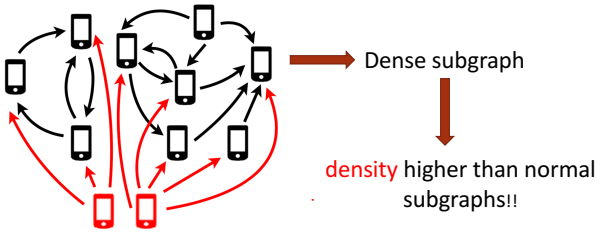
(user, object, timestamp, ~~#stars~~)
(user, object, ~~timestamp,~~ #stars)
(user, object~~, timestamp, #stars~~)

Why using graph to detect fraud?

- **Content can be cheated by NLP technology** 内容可以被NLP技术欺骗
- **Content is not available** 内容不可用
- **Graph is a good representation of**
  - **users reviewing/giving scores to objects**
  - **a user clicking a link, and watching a video**

图可以很好的代表：用户回看/打分；或者用户点击链接、看视频

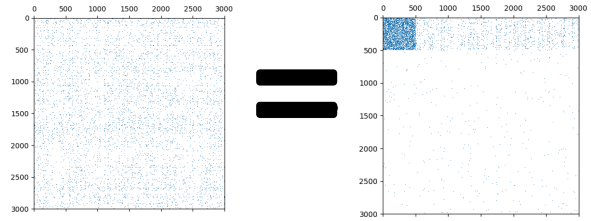
Density usually indicates unusual events 密度通常表示不寻常的事件



Eswaran, D., Faloutsos, C., Guha, S., Mishra, N. Spotlight: Detecting anomalies in streaming graphs. In: SIGKDD. pp. 1378–1386. ACM (2018)

How to find 'suspicious' groups? 可疑的

- 'blocks' are usually suspicious



对于欺诈检测，平均度密度，比密度值好

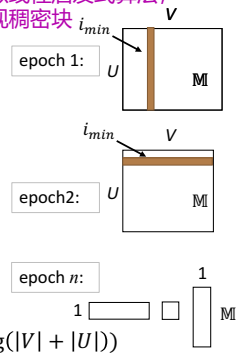
Average degree density works better than volume density for fraud detection

- **Volume density**
  - **Suppose**
    - **a fraudster has # of accounts: a**
    - **his goal is click b objects 200 times**
  - **Density:** $(b \cdot 200)/(a \cdot b) = 200/a$
  - **unlimited b does not increase density** 目标数b不影响密度
- **Average degree $g(\cdot)$: arithmetic / geometric** 算术/几何
  - **Arithmetic avg:** $(b \cdot 200)/(a + b)$
  - **Geometric avg:** $(b \cdot 200)/(\sqrt{ab})$

[Asahiro et al, SWAT'96] [M Charikar, 2000] [B Hooi et al, KDD'16]

A near-linear heuristic algorithm to detect dense block 近似线性启发式算法，发现稠密块

- **Given: adjacency matrix $\mathbb{M}$**
- $X \leftarrow \{U, V\}$
- **While X is not empty**
  - $i_{min} \leftarrow \arg\min_{i \in X} \deg(i, X)$
  - $X \leftarrow X \setminus \{i_{min}\}$
  - **Keep X_{best} that has the best arithmetic avg degree $g(X_{best})$**
- **Return X_{best}**
- **Theoretical boundary:** $g(X_{best}) > \frac{1}{2} g(X_{opt})$
- **Time complexity with Priority Tree:** $O(|E| \log(|V| + |U|))$
- **Optimal algorithm needs** $O(|V|^2 \log^2 |V|)$



[A.V. Goldberg, Technical report, 1984]
[Asahiro et al, SWAT'96] [M Charikar, 2000] [B Hooi et al, KDD'16]

Theoretical Bounds

- The result will have a lower bond:

THEOREM 2. *Let $\mathcal{A}, \mathcal{B}$ be the set of users and objects returned by* FRAUDAR. *Then:*

$$g(\mathcal{A} \cup \mathcal{B}) \geq \frac{1}{2} g_{OPT}$$

where g_{OPT} is the maximum value of g, i.e.

$$g_{OPT} = \max_{\mathcal{A}', \mathcal{B}'} g(\mathcal{A}' \cup \mathcal{B}')$$

Proof

Define weight assigned to node v_i in S'as

$$w_i(\mathcal{S}) = \sum_{(v_j \in \mathcal{S}) \wedge ((v_i, v_j) \in \mathcal{E})} c_{ij} + \sum_{(v_j \in \mathcal{S}) \wedge ((v_j, v_i) \in \mathcal{E})} c_{ji}$$

where $c_{ij} (> 0)$ indicates the weight of edge $(v_i; v_j)$

Consider the optimal set S*. For each node $v_i \in S^*$, then $w_i(S') \geq g(S^*)$. Otherwise, removing a node with $w_i(S') < g(S^*)$ results in

$$g' = \frac{f(\mathcal{S}^*) - w_i(\mathcal{S}^*)}{|S^*| - 1} > \frac{f(\mathcal{S}^*) - g(\mathcal{S}^*)}{|S^*| - 1}$$
$$= \frac{f(\mathcal{S}^*) - f(\mathcal{S}^*)/|\mathcal{S}^*|}{|S^*| - 1} = g(\mathcal{S}^*),$$

which is a contradiction.

Proof , con.

Axiom 1. Let v_i be the node that FRAUDAR removes first among those in S^*, and let S' be the set before FRAUDAR removes v_i:
since $S' \supset S^*$, S' then $w_i(S') \geq w_i(S^*)$.

Axiom 2. Since FRAUDAR chooses to remove node v_i :
then : $\forall v_j \in S'$, $w_j(S') \geq w_i(S')$

Axiom 3. Since each term in $f(S')$ can be assigned to at most two nodes:
then summing over j gives $f(S') \geq \frac{|S'| w_i(S')}{2}$

Axiom 4. Since FRAUDAR returns the best solution that it encounters
Then $g(A \cup B) > g(S')$

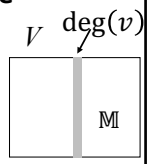
We conclude that :

$$g(\mathcal{A} \cup \mathcal{B}) \geq g(\mathcal{S}') = \frac{f(\mathcal{S}')}{|\mathcal{S}'|} \geq \frac{w_i(\mathcal{S}')}{2} \geq \frac{w_i(\mathcal{S}^*)}{2} \geq \frac{g(\mathcal{S}^*)}{2}.$$

M1. Fraudar: very popular products are less suspicious 非常受欢迎的产品不那么可疑

惩罚边的权重

- **Fraudar penalizes the weight of each edge**
  - **preprocess:** $e_{uv} \leftarrow 1/\log(\deg(v) + c) \cdot e_{uv}$,
    - **where** $e_{uv} = \mathbb{M}(u,v)$, c**=5**
  - **avg degree:** $g_{log}(X) = \frac{1}{|X|} \sum_{u,v \in X} e_{uv}$



- **Bounding Fraud** 边界欺诈
  - **Upper bound of fake edges that m_0 fraudsters can create for n_0 products:** m欺诈者可以为n产品创建的假边上限：

$$2(m_0 + n_0) \cdot g_{log}(X_{best}) \cdot \log(\frac{m_0}{\lambda} + c)$$

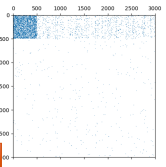
    - **where λ is the fraction of edges that an object has from fraudsters.**

其中l是一个物体从骗子那里得到的边的比例

Fraudar: bounding graph fraud in the face of camouflage [B Hooi et al, KDD'16]

60

Dense blocks and Maximum Rayleigh Quotient
稠密块和最大化瑞利商

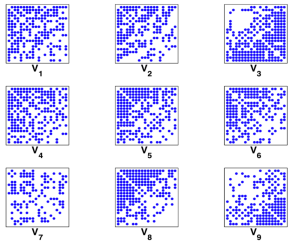
- Dense blocks in a matrix M
  - indicated by a selection vector
    $$x \in \{0,1\}^N$$
  - Arithmetic avg degree (density)
    $$f = \frac{\#edge}{\#nodes} = \frac{x^T M x}{x^T x}$$
    Rayleigh quotient
    $x =$ $\begin{bmatrix}1\\ \vdots \\ 1\\ 0\\ 0\\ \vdots\end{bmatrix}$
    
- Theorem on Rayleigh quotient

$$\lambda_1 = \max_x \frac{x^T M x}{x^T x}, \text{ and } x \text{ is the eigenvector of } \lambda_1$$

  - by relaxing x as non-zero real vector

M2. EigenSpoke: Spectral-based method
普方法

- Find dense groups of users by eigenvectors
  - 20 nodes with the highest magnitude projection along the first 9 singular vectors
    在前9个奇异向量上具有最大投影的20个节点



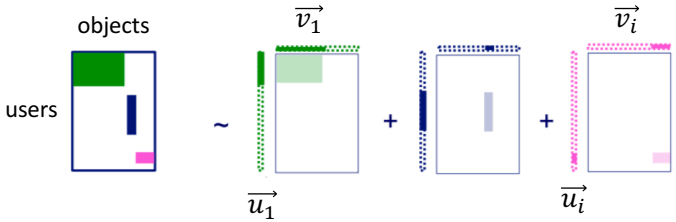
inducing sub-graphs contain near-cliques.

Prakash, B.A., Sridharan, A., Seshadri, M., Machiraju, S., Faloutsos, C. In PAKDD. Springer (2010)

SVD: Singular Vector

- (SVD) matrix factorization: finds blocks

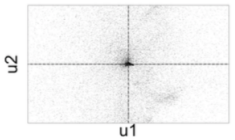
$$A = U\Sigma V^T$$



Neil Shah+, Spotting suspicious link behavior with fbox: An adversarial perspective, ICMD 2014

Dense block and *Spectral Subspace Plot*

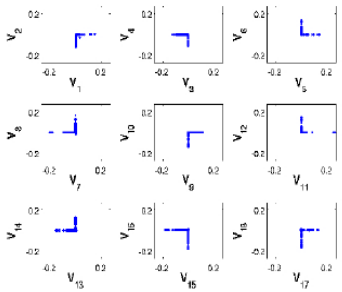
- Spectral Subspace Plot:
  - Scatter plot of scores of u1 vs u2
    u1和u2的分数的散点图
- One would expect
  - Many points @ origin
  - A few scattered ~randomly



Case #0: No lockstep behavior in random power law graph of 1M nodes, 3M edges

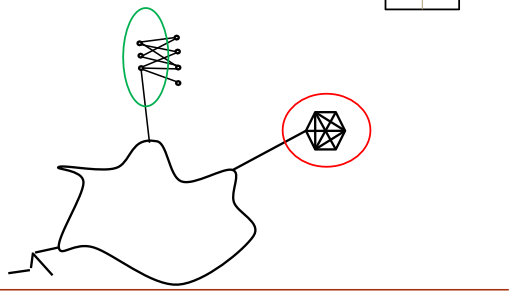
Lockstep and *Spectral Subspace Plot*

- Spectral Subspace Plot:
  - Scatter plot of scores of u1 vs u2
- One would expect
  - Many points @ origin
  - A few scattered ~ran[illegible]mly



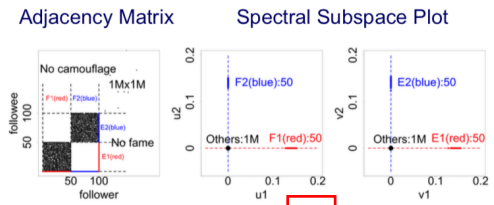
Spectral Subspace Plot - explanation

- Near-cliques, or near-bipartite-cores, loosely connected 近簇，或近二部核，松散连接



Dense block and *Spectral Subspace Plot*

- Case #1: non-overlapping dense block不重叠稠密块
- "Blocks" ⟺ "Rays" 块-》线



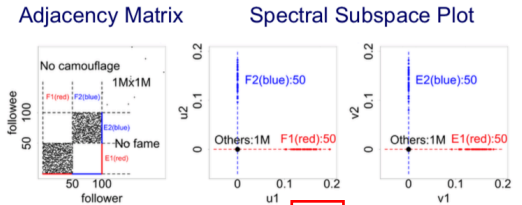
Rule 1 (short "rays"): two blocks, high density (90%), no "camouflage", no "fame"

伪装

[M. Jiang et al, PAKDD'14] [B. Aditya Prakash et al, PAKDD'10]

Dense block and *Spectral Subspace Plot*

- Case #2: non-overlapping lockstep
- "Blocks; low density" ⟺ Elongation 伸长线

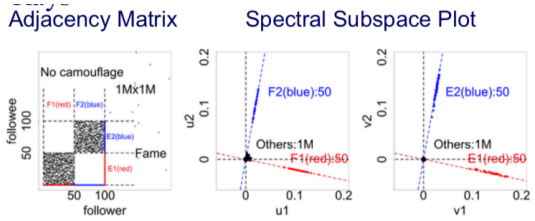


Rule 2 (long "rays"): two blocks, low density (50%), no "camouflage", no "fame"

[M. Jiang et al, PAKDD'14] [B. Aditya Prakash et al, PAKDD'10]

Dense block and *Spectral Subspace Plot*

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



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

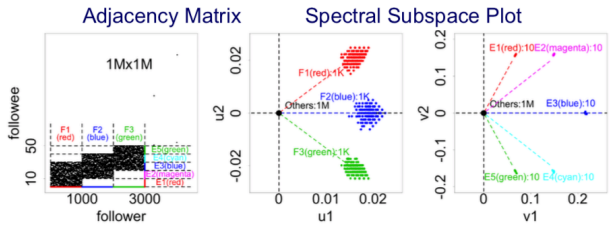
[M. Jiang et al, PAKDD'14] [B. Aditya Prakash et al, PAKDD'10]

69

Dense block and *Spectral Subspace Plot*

- Case #4: overlapping lockstep 重叠紧密块
- "Staircase" ⟺ "Pearls"

小于90度，块和点



Rule 4 ("pearls"): a "staircase" of three partially overlapping blocks.

阶梯状

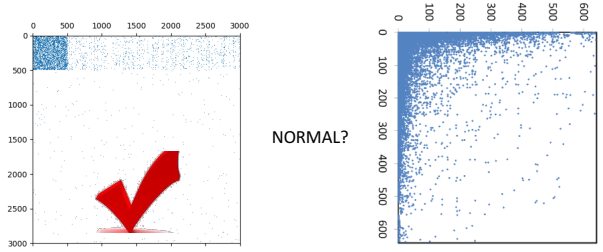
[M. Jiang et al, PAKDD'14] [B. Aditya Prakash et al, PAKDD'10]

70

M3. HoloScope: Hyperbolic community

双曲线

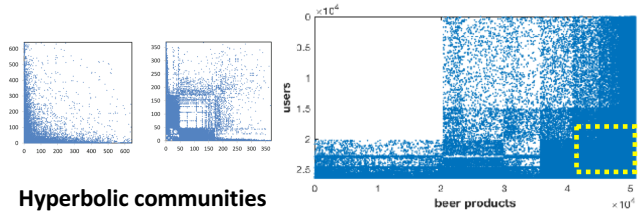
- 'hyperbolic' communities are more realistic [Araujo+, PKDD'14]



2019/9/17

71

Hyperbolic community exists in real graphs



Hyperbolic communities in YouTube friendship and Wikipedia articles [SNAP datasets]

Hyperbolic community In our BeerAdvocate data

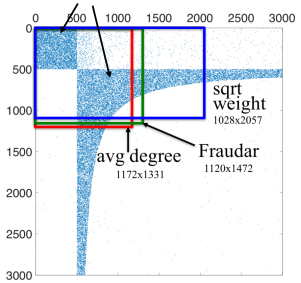
cross-association [D Chakrabarti et al, KDD'04]; Hyperbolic community detection [M Araujo et al, ECML-PKDD'14]; SNAP datasets: http://snap.stanford.edu/data/index.html

72

How can we avoid detecting the false positive hyperbolic block?

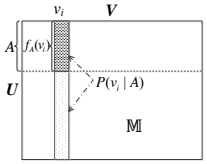
如何避免检测到假阳性的双曲块?

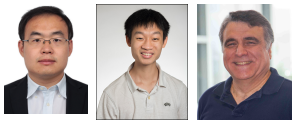
Penalize sink nodes in both blocks 在两个块中惩罚接收器节点



Synthetic data

HoloScope: Topology-and-Spike Aware Fraud Detection

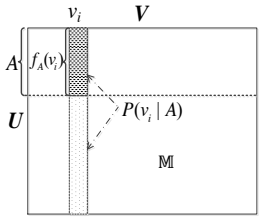




Shenghua Liu, Bryan Hooi, and Christos Faloutsos

Contrast suspiciousness in HoloScope

- $D(A,B) = \frac{\sum_{v_i \in B} f_A(v_i)}{|A|+|B|}$
  - $A \subset U, B \subset V$
  - $f_A(v_i) = \sum_{(u_j, v_i) \in E \wedge u_j \in A} \sigma_{ji} \cdot e_{ji}$, σ_{ji} **is edge weight**
- **Contrast susp:** $P(v_i \in B|A)$
  - **the conditional likelihood**



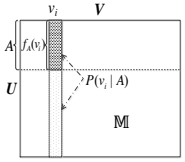
- **Objective:**

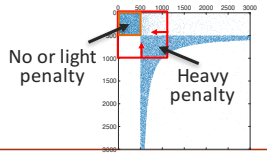
$$\max_A HS(A) := \mathbb{E}[D(A,B)]$$

$$= \frac{1}{|A| + \sum_{k \in V} P(v_k|A)} \sum_{i \in V} f_A(v_i) P(v_i|A)$$

Topology-aware (dense block) HS-α

- $P(v_i|A) = q(\alpha_i), \alpha_i = \frac{f_A(v_i)}{f_U(v_i)}$
  - **Scaling fun:** $q(x) = b^{x-1}$, $0 \le x \le 1$ **and constant** $b > 1$
- **users' susp score:**
  - $S(u_j \in A) = \sum_{u_j v_i \in E} \sigma_{ji} e_{ji} \cdot P(v_i|A)$

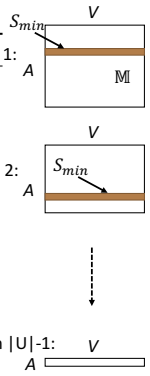




Algorithm HS-α considers topology

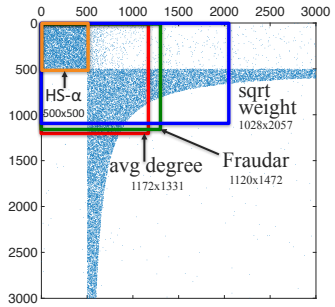
Algorithm 1 HS-α Algorithm.

Given adjacency matrix $\mathbb{M}$
Initialize:
$A = U$
$\mathcal{P}$= calculate contrast susp of all sink nodes given A
$\mathcal{S}$ = calculate susp scores of source nodes A.
MT = build priority tree of A with scores $\mathcal{S}$.
while A is not empty **do**
u = pop the source node of the minimum score from MT.
$A = A \setminus u$, delete u from A.
Update $\mathcal{P}$ with respect to new source nodes A.
Update MT with respect to new $\mathcal{P}$.
Keep A^* that has the largest objective $HS(A^*)$
end while
return A^* and $P(v|A^*), v \in V$.



HS-α can shrink the detection box over hyperbolic community

- **Synthetic data**
  - **Scaling fun:** $q(\alpha_i) = 128^{\alpha_i - 1}$
  - **b = 128**



The Complexity

- **Find burst and drop points of each sink node**
  - **cost $O(d_v)$, total cost $O(|E|)$**
- **Use framework of HS-α algorithm**

Algorithm 3 HS algorithm (unscalable).

Given bipartite multigraph $\mathcal{G}(U, V, E)$,
initial source nodes $A_0 \subset U$.
Initialize:
$A = A_0$
$\mathcal{P}$= calculate contrast suspiciousness given A_0
$\mathcal{S}$ = calculate suspiciousness scores of source nodes A.
MT = build priority tree of A with scores $\mathcal{S}$. ← $O(m_0 \log m_0), m_0 = |A_0|$
while A is not empty **do**
u = pop the source node of the minimum score from MT.
$A = A \setminus u$, delete u from A.
Update $\mathcal{P}$ with respect to new source nodes A. ← $O(d_u \cdot |A|)$
Update MT with respect to new $\mathcal{P}$. ← $O(d_u \cdot |A| \cdot \log m_0)$
Keep A^* that has the largest objective $HS(A^*)$
end while
return A^* and $P(v|A^*), v \in V$.

Time complexity

Algorithm 3 HS algorithm (unscalable).

Given bipartite multigraph $\mathcal{G}(U, V, E)$,
initial source nodes $A_0 \subset U$.
Initialize:
$A = A_0$
$\mathcal{P}$= calculate contrast suspiciousness given A_0
$\mathcal{S}$ = calculate suspiciousness scores of source nodes A.
MT = build priority tree of A with scores $\mathcal{S}$. ← $O(m_0 \log m_0), m_0 = |A_0|$
while A is not empty **do**
u = pop the source node of the minimum score from MT.
$A = A \setminus u$, delete u from A.
Update $\mathcal{P}$ with respect to new source nodes A. ← $O(d_u \cdot |A|)$
Update MT with respect to new $\mathcal{P}$. ← $O(d_u \cdot |A| \cdot \log m_0)$
Keep A^* that has the largest objective $HS(A^*)$
end while
return A^* and $P(v|A^*), v \in V$.

- **The time complexity is**
  - $\sum_{j=2,\cdots,m_0} O(d_j \cdot (j-1) \cdot \log m_0) = O(m_0 |E_0| \log m_0)$
  - **When $A_0 = U$, it is $O(|U||E| \log |U|)$**

Super quadratic # of nodes. Slow!

Scalable HS algorithm

- **Main idea: feed small groups of users $\widetilde{U}$ into** 对小用户组进行计算 ***GreedyShaving* Procedure (previous HS alg.)**

Algorithm 4 *FastGreedy* Algorithm for Fraud detection.

Given bipartite multigraph $\mathcal{G}(U, V, E)$.
$\mathbb{L}$ = get first several left singular vectors
for all $L^{(k)} \in \mathbb{L}$ **do**
  Rank source nodes U decreasingly on $L^{(k)}$
  $\tilde{U}^{(k)}$ = truncate $u \in U$ when $L_u^{(k)} \leq \frac{1}{\sqrt{|U|}}$
  $GreedyShaving$ with initial $\tilde{U}^{(k)}$.
end for
return the best A^* with maximized objective $HS(A^*)$,
  and the rank of $v \in V$ by $f_{A^*}(v) \cdot P(v|A^*)$.

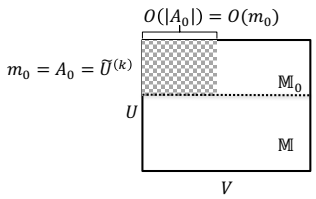
Scalable HS alg is sub-quadratic # of nodes

- **Theorem 2 (algorithm complexity)**

> Given $|V| = O(|U|)$ and $|E| = O(|U|^{\epsilon_0})$,
> the time complexity of *FastGreedy* is subquadratic,
> $o(|U|^2)$ in little-o notation,
> if $|\widetilde{U}^{(k)}| \leq |U|^{1/\epsilon}$, where $\epsilon > \max\{1.5, \frac{2}{3-\epsilon_0}\}$

- In real life graph, if $\epsilon_0 \leq 1.6$, so we can limit $|\widetilde{U}^{(k)}| \leq |U|^{1/1.6}$

Proof



skip proof

$m_0 = |\tilde{U}^{(k)}| \leq |U|^{1/\epsilon}$

- The total number of edges in $\mathbb{M}_0$ is

$$O(|E_0|) = O(m_0^2 + \frac{m_0 \cdot |E|}{|U|}) = O(|U|^{2/\epsilon} + |U|^{1/\epsilon - 1}|E|)$$

- So the HS algorithm complexity is

$$O(m_0|E_0|\log m_0) = O((|U|^{3/\epsilon} + |U|^{2/\epsilon - 1 + \epsilon_0})\log|U|)$$

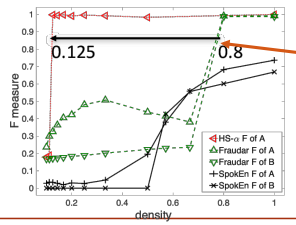
- Therefore, if $\epsilon > \max\{1.5, \frac{2}{3-\epsilon_0}\}$, the complexity is subqudratic $o(|U|^2)$. ($\exists \epsilon \leq 2$?)
- In real life graph, if $\epsilon_0 \leq 1.6$, so we can limit $|\widetilde{U}^{(k)}| \leq |U|^{1/1.6}$

$|E| = O(|U|^{\epsilon_0})$

Performance on injected labels

注入标签上的性能

- **Mimic fraudsters to inject edges, with different fraudulent density**
  - **BeerAdvocate Data**
  - **size : 26.5K x 50.8K, 1.07M, Jan 2008 – Nov 2011**

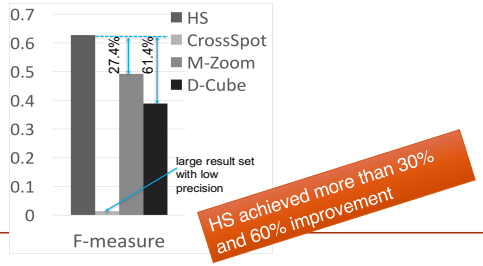


improved by contrast susp.

HS-α consider only topology (density)

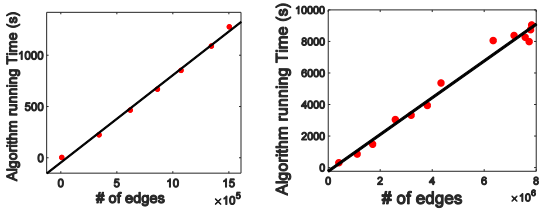
Performance on real labels from online system

- Sina Weibo is a microblog and Twitter-like website
  - 2.75 M users, 8.08 M messages, and 50.1 M edges in our data of Dec 2013



HS achieved more than 30% and 60% improvement

Scalability

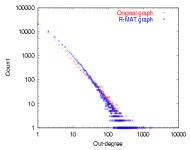


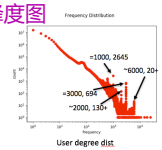
BeerAdvocate dataset Amazon Electronics dataset

HoloScope runs in near-linear time of # of edges

Take away

- Graphs are general representations for many to many relations 图可以表示关系
- Patterns: Skewed distributions
  - degree, … … 模式:偏态分布
  - "Gaussian trap" 高斯陷阱
- Anomalies: 异常
  - spike in degree plot 峰度图
  - dense blocks 稠密块





Questions?