1. Subgraph Analysis

2. Propagation Methods

a) Background

b) Normal Behavior

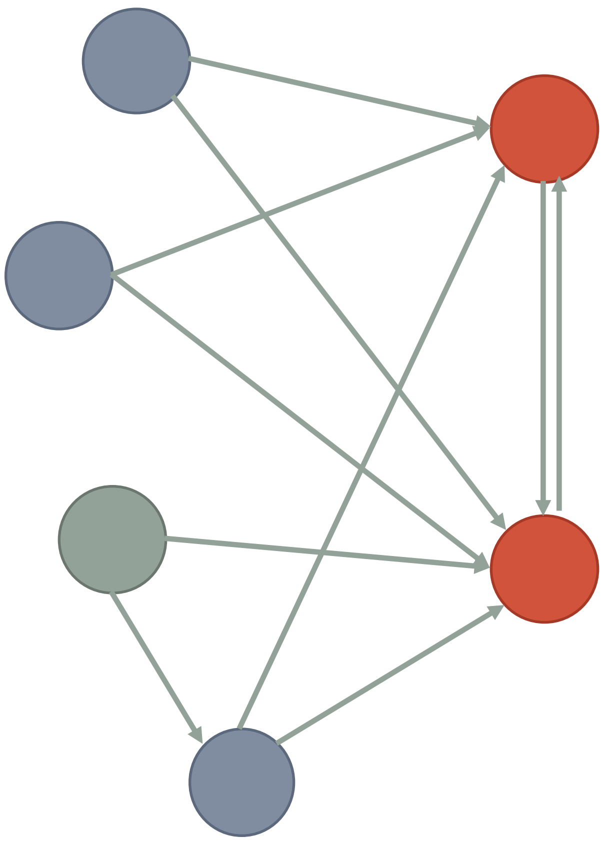
c) Abnormal Behavior

3. Latent Factor Models

Given a graph, how can we find the “important” nodes?

Given the web, how can we find authoritative webpages?

Authoritative nodes and hubs: HITS

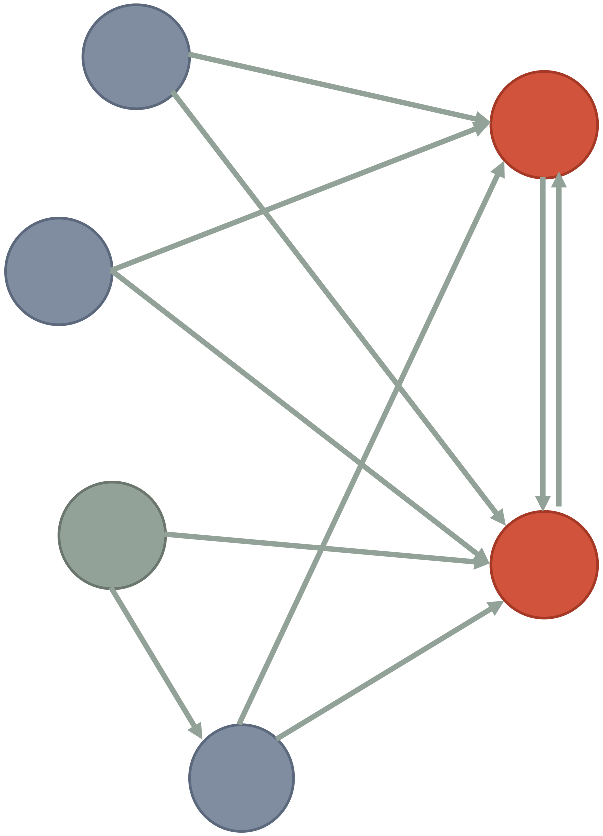


Authoritative nodes are pointed to by hubs

Hubs point to authoritative nodes

Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

Authoritative nodes and hubs: HITS



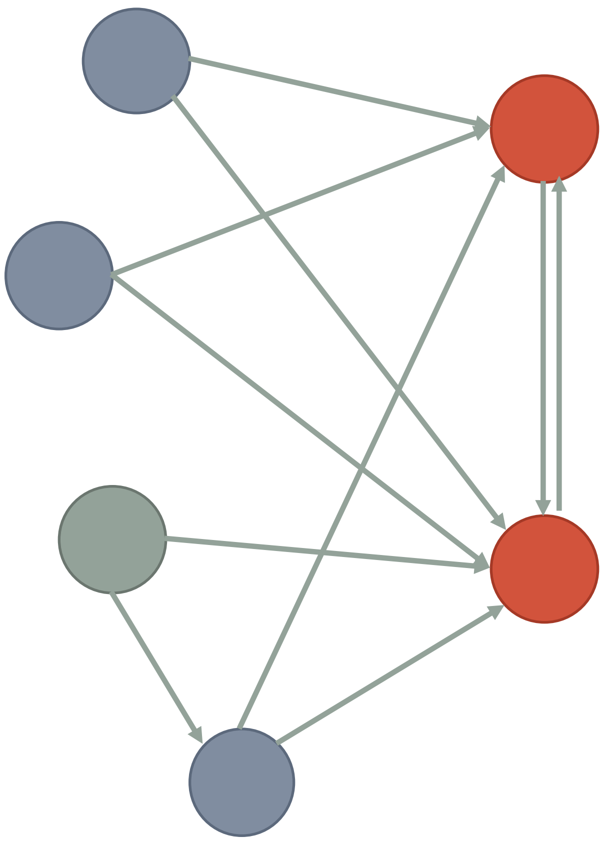
Authoritative nodes are pointed to by hubs

Hubs point to authoritative nodes

Seems circular but still solvable!

Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

Authoritative nodes and hubs: HITS



$$\text{Authoritativeness}(q) = \sum_{(p,q)\in E} \text{Hubness}(p)$$

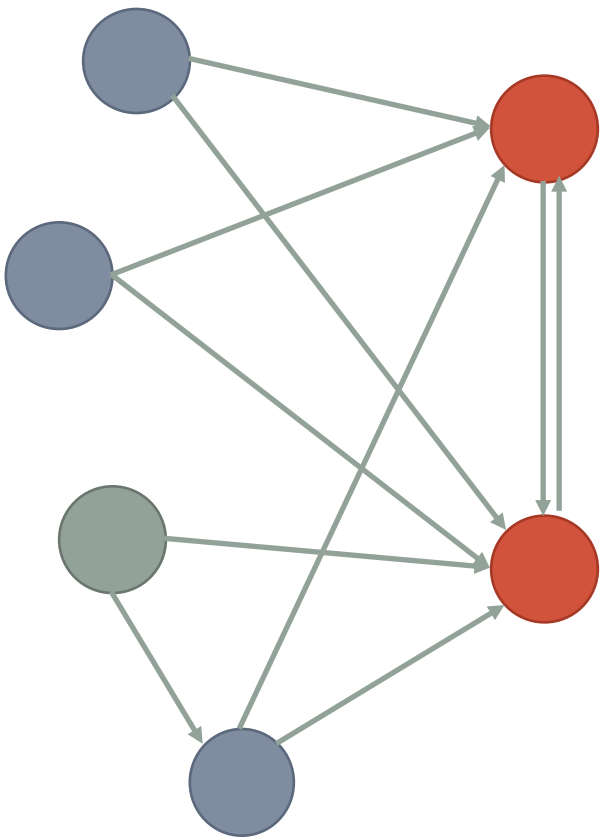
$$\text{Hubness}(p) = \sum_{(p,q)\in E} \text{Authoritativeness}(q)$$

Seems circular but still solvable!

*We also keep authoritativeness and hubness normalized

Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

Authoritative nodes and hubs: HITS



$$\text{Authoritativeness}(q) = \sum_{(p,q)\in E} \text{Hubness}(p)$$

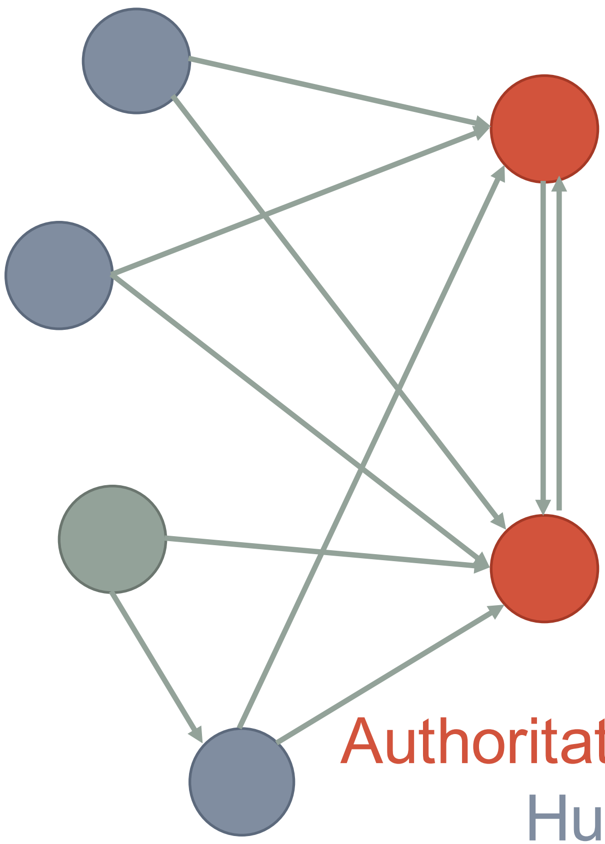
$$\text{Hubness}(p) = \sum_{(p,q)\in E} \text{Authoritativeness}(q)$$

Alternate updating both scores, repeatedly

*We also keep authoritativeness and hubness normalized

Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

Authoritative nodes and hubs: HITS



$$\text{Authoritativeness}(q) = \sum_{(p,q)\in E} \text{Hubness}(p)$$

$$\text{Hubness}(p) = \sum_{(p,q)\in E} \text{Authoritativeness}(q)$$

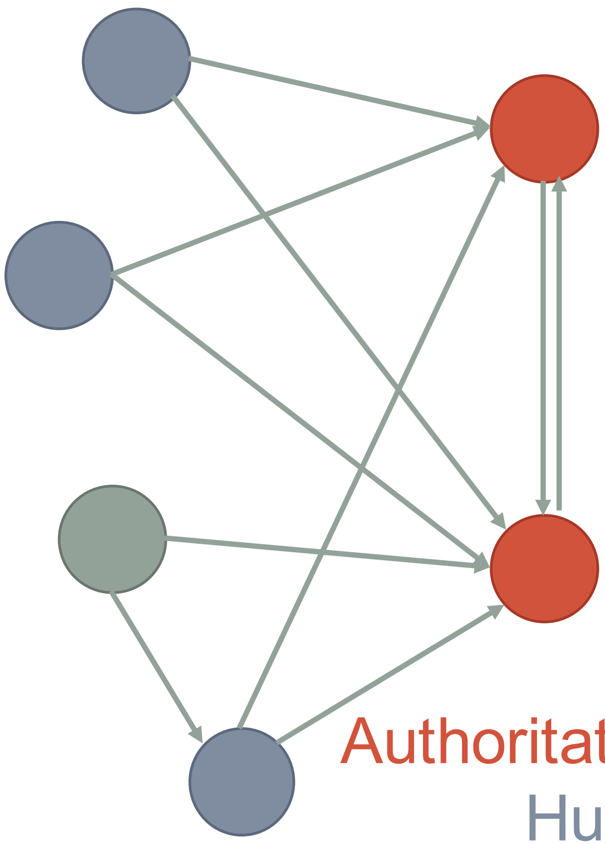
Assume graph adjacent matrix *A*

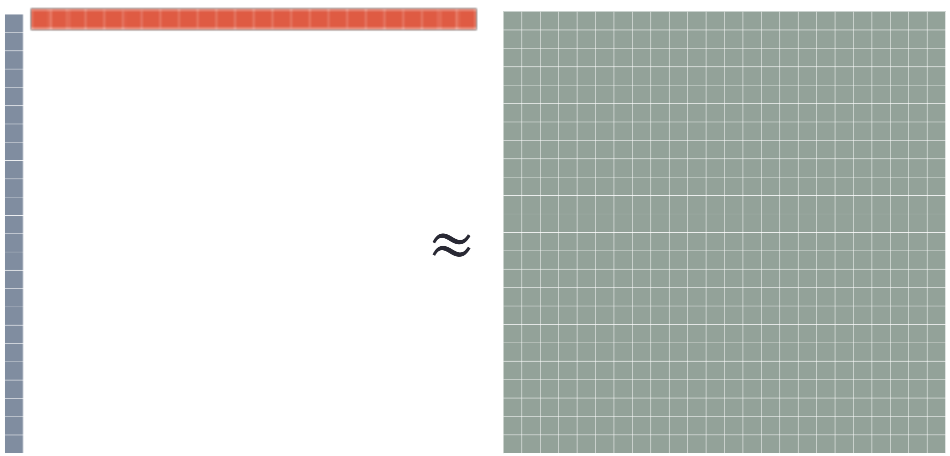
Authoritativeness is the first **left singular vector** of *A*

Hubness is first **right singular vector** *A*

Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

Authoritative nodes and hubs: HITS



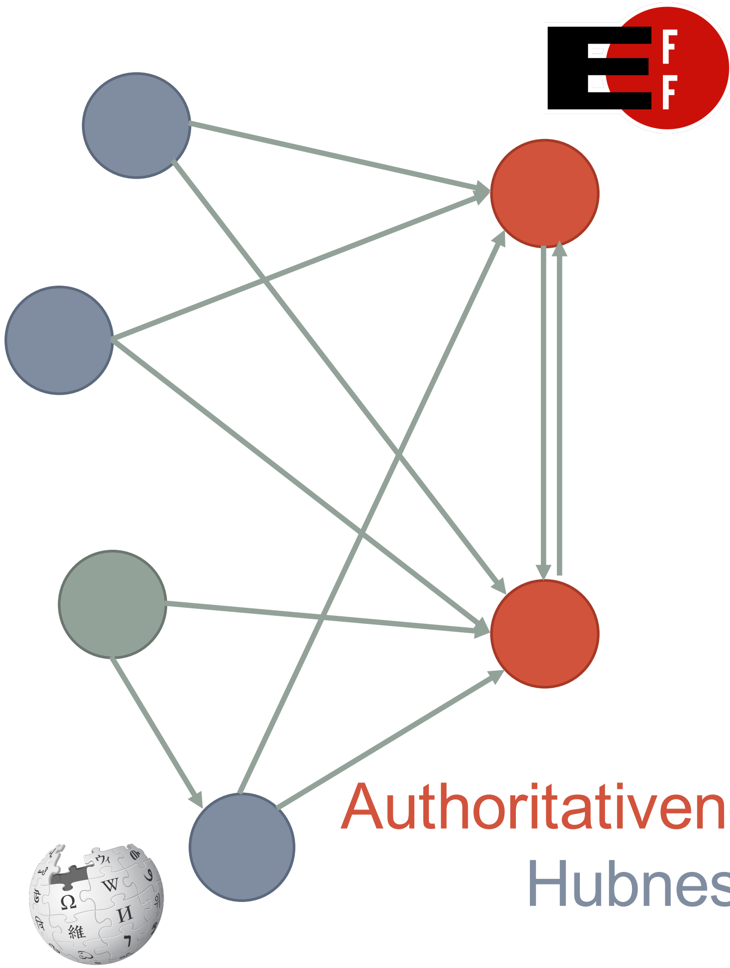


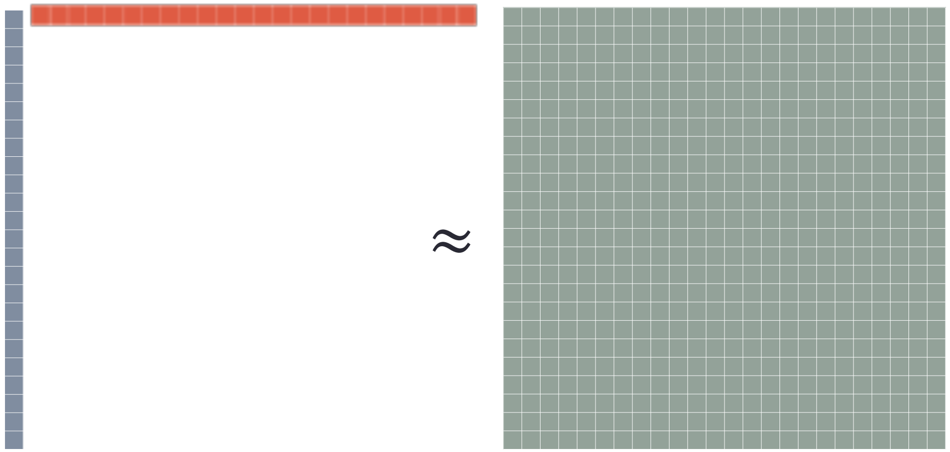
Authoritativeness is the first **left singular vector** of *A*

Hubness is first **right singular vector** *A*

Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

Authoritative nodes and hubs: HITS

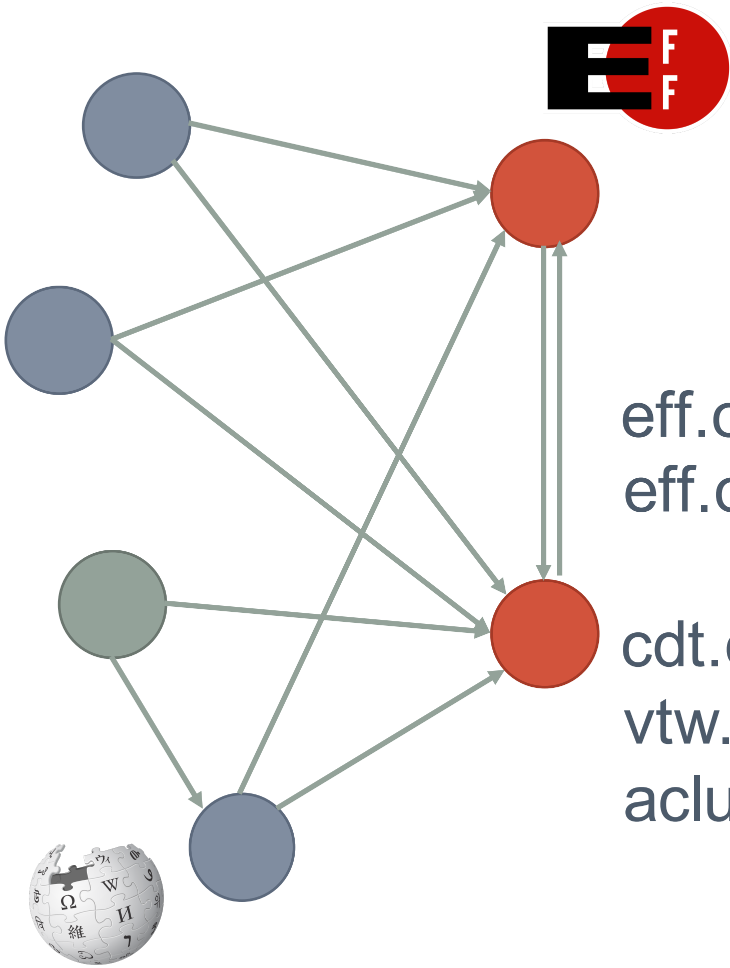




Authoritativeness is the first **left singular vector** of *A*
Hubness is first **right singular vector** *A*

Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

Authoritative nodes and hubs: HITS



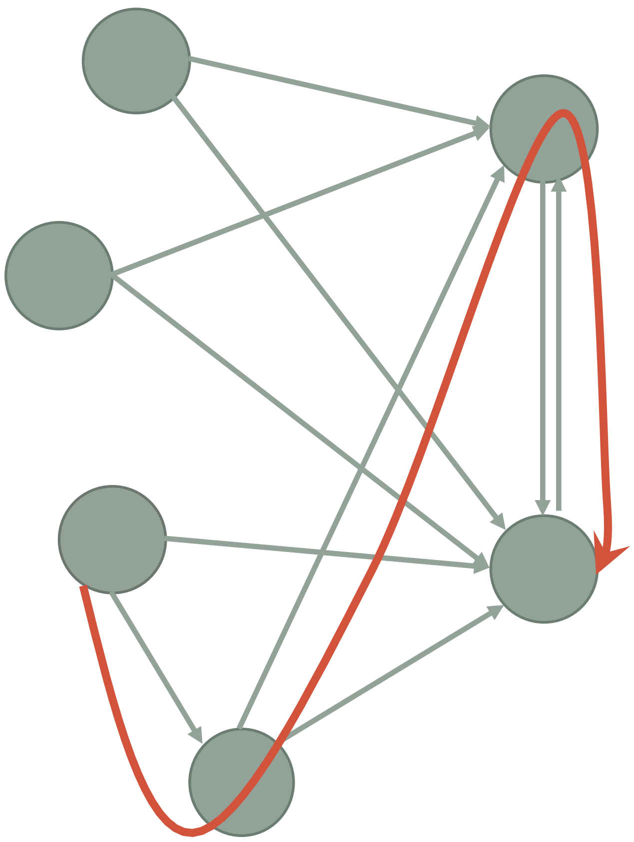
Restrict graph by query “censorship”

Authorities:

eff.org	Electronic Frontier Foundation
eff.org/blueribbon.html	EFF Blue Ribbon Campaign
cdt.org	Center for Democracy and Tech
vtw.org	Voters Telecommunication Watch
aclu.org	American Civil Liberties Union

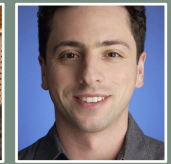
Authoritative Sources in a Hyperlinked Environment
Jon M. Kleinberg
JACM 1999

(Simplified) PageRank

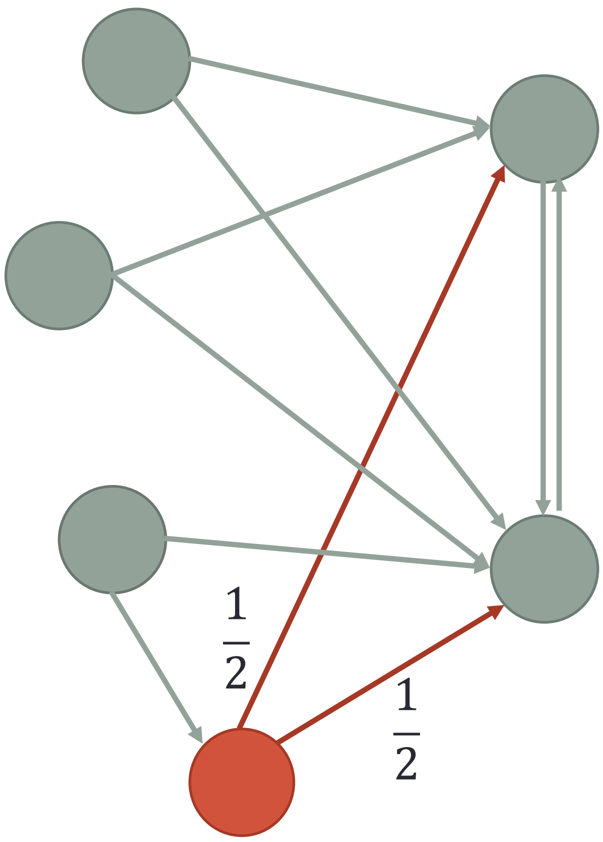


Analyze random walk in graph

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998



(Simplified) PageRank

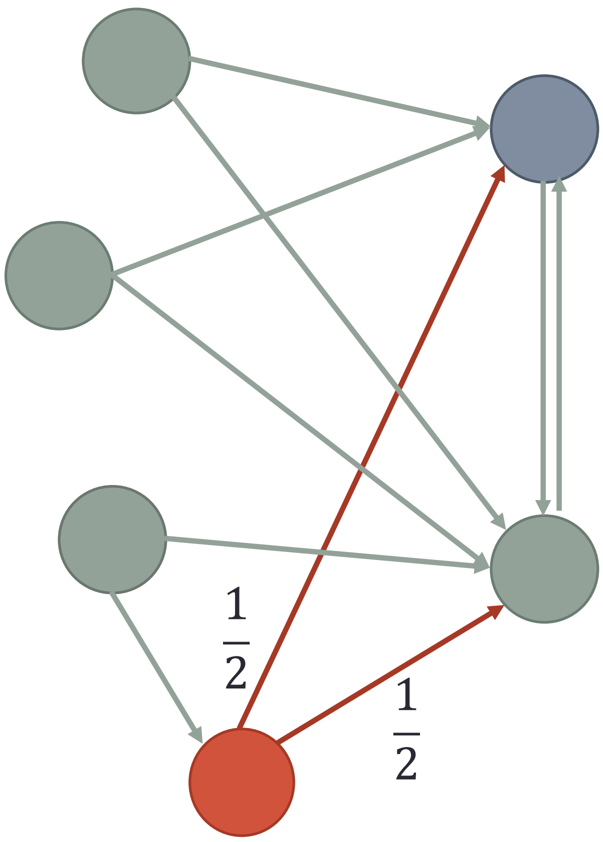


Analyze random walk in graph

From a node, take each outgoing edge with equal probability.

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

(Simplified) PageRank



Analyze random walk in graph

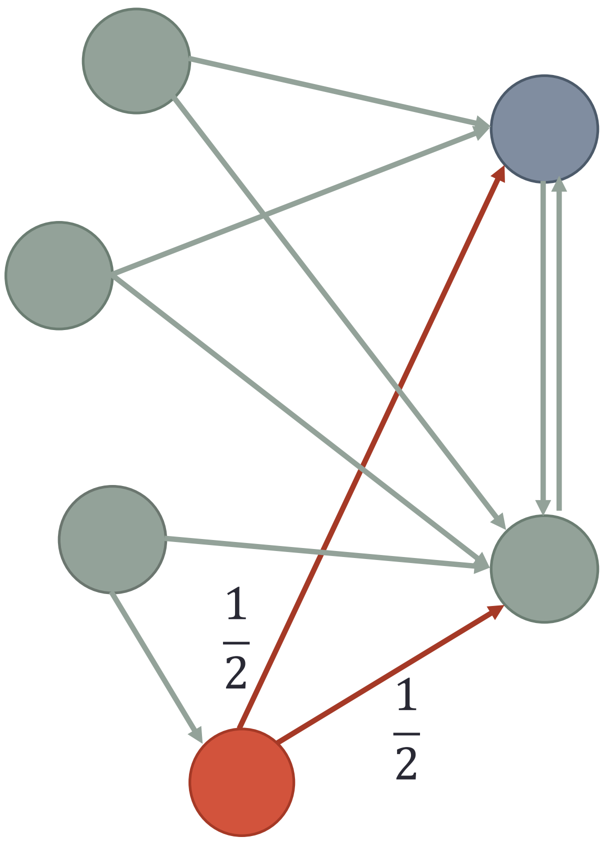
From a node, take each outgoing edge with equal probability.

$$\text{Rank}(p) = c \sum_{q \mid (p,q) \in E} \frac{\text{Rank}(q)}{\text{OutDegree}(q)}$$

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

(Simplified) PageRank

$$\text{Rank}(p) = c \sum_{q|(p,q)\in E} \frac{\text{Rank}(q)}{\text{OutDegree}(q)}$$

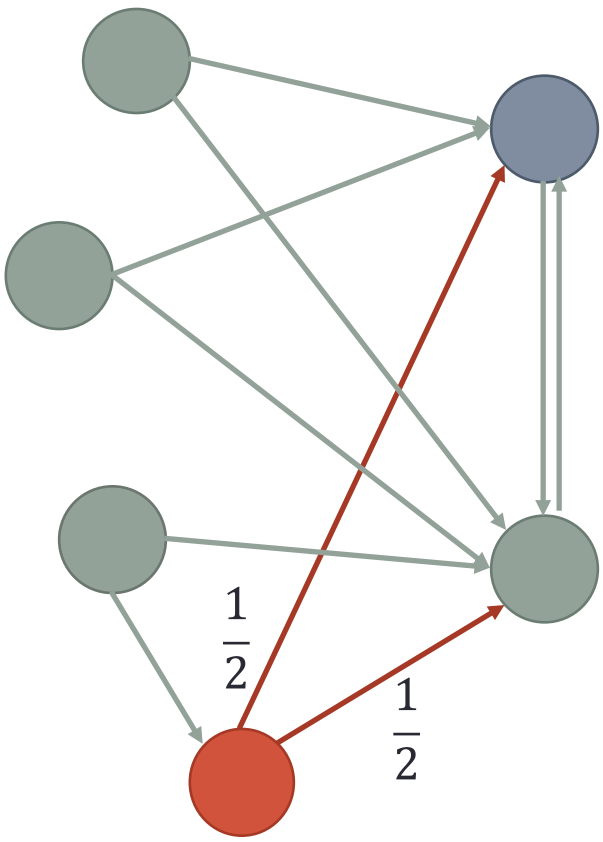


A' is a column normalized adjacency matrix

$$A'_{p,q} \begin{cases} \dfrac{1}{\text{OutDegree}(q)} & \text{if } (p,q) \in E \\ 0 & \text{if } (p,q) \notin E \end{cases}$$

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

(Simplified) PageRank



$$\text{Rank}(p) = c \sum_{q|(p,q)\in E} \frac{\text{Rank}(q)}{\text{OutDegree}(q)}$$

A' is a column normalized adjacency matrix

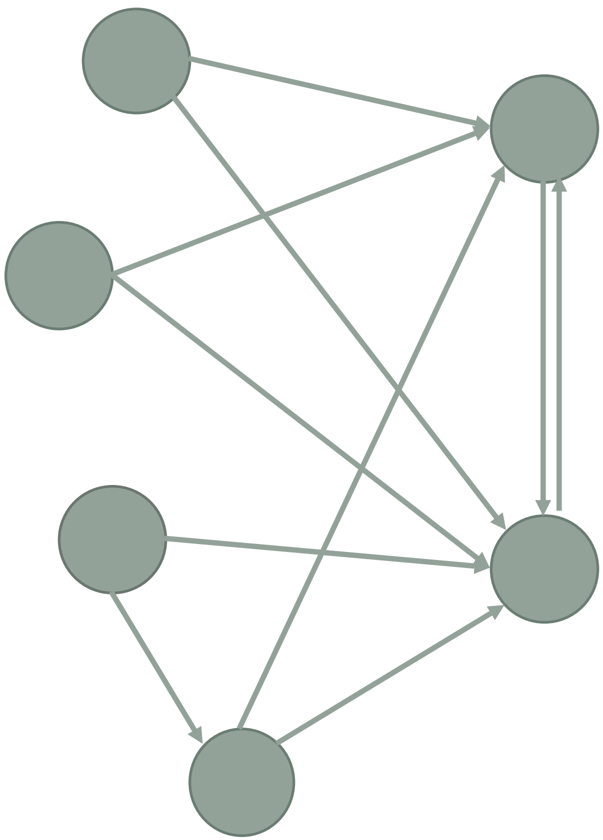
$$A'_{p,q} \begin{cases} \frac{1}{\text{OutDegree}(q)} & \text{if } (p,q) \in E \\ 0 & \text{if } (p,q) \notin E \end{cases}$$

$$\overrightarrow{\text{Rank}} = cA'\overrightarrow{\text{Rank}}$$

Rank is the first eigenvector of *cA'*

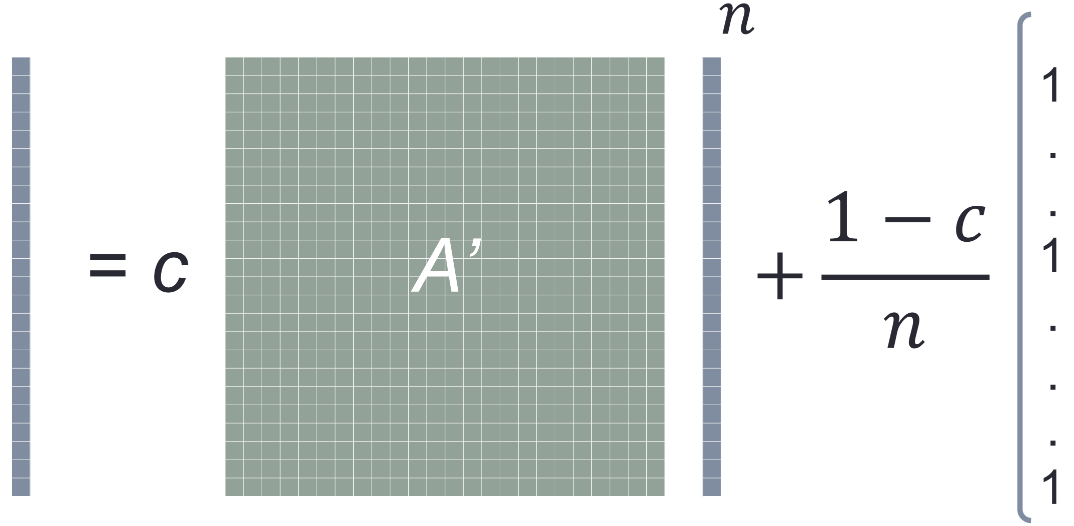
Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

PageRank



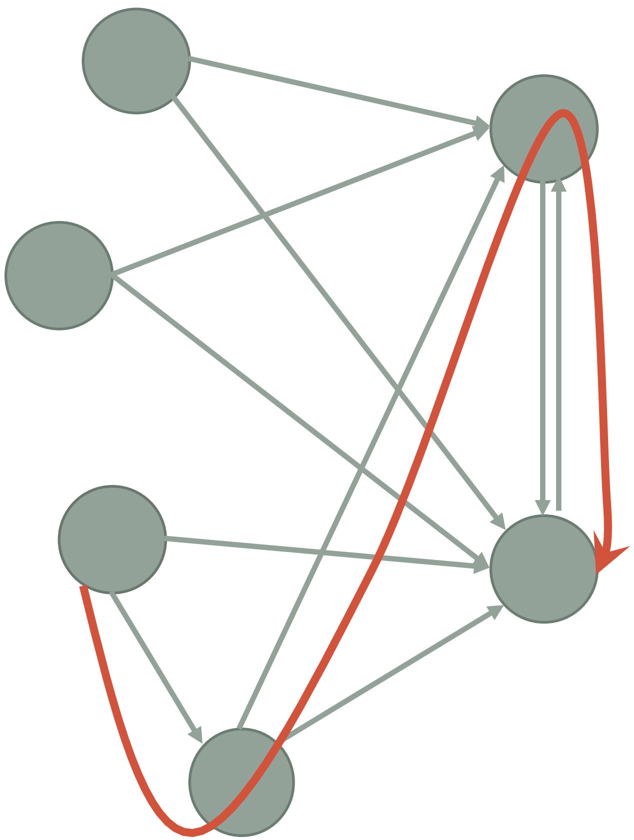
In "random walk," jump to new node randomly with probability (1-*c*)

$$\overrightarrow{\text{Rank}} = cA'\overrightarrow{\text{Rank}} + \frac{1-c}{n}\vec{\mathbf{1}}$$



Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

PageRank

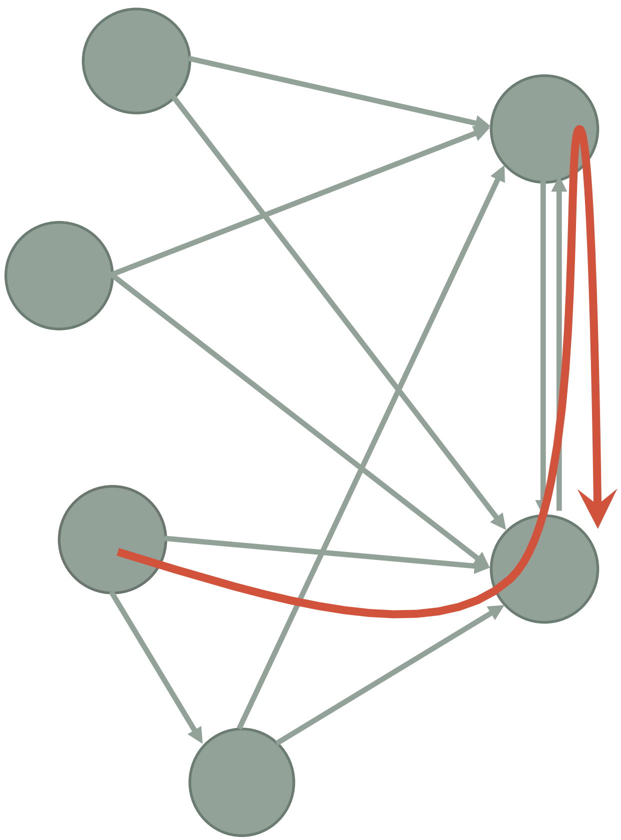


In "random walk," jump to new node randomly with probability (1-*c*)

$$\overrightarrow{\text{Rank}} = cA'\overrightarrow{\text{Rank}} + \frac{1-c}{n}\vec{\mathbf{1}}$$

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

PageRank

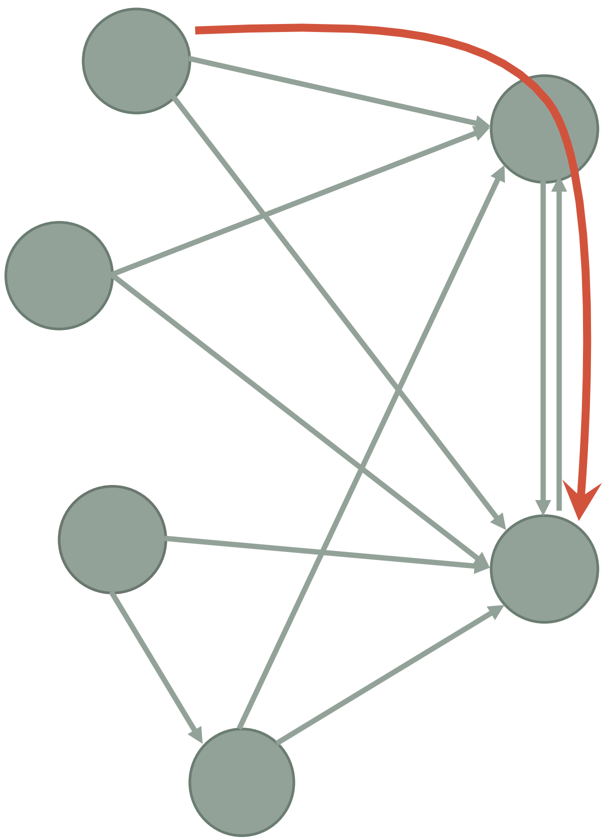


In “random walk,” jump to new node randomly with probability (1-*c*)

$$\overrightarrow{\mathrm{Rank}} = cA'\overrightarrow{\mathrm{Rank}} + \frac{1-c}{n}\vec{\mathbf{1}}$$

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

PageRank

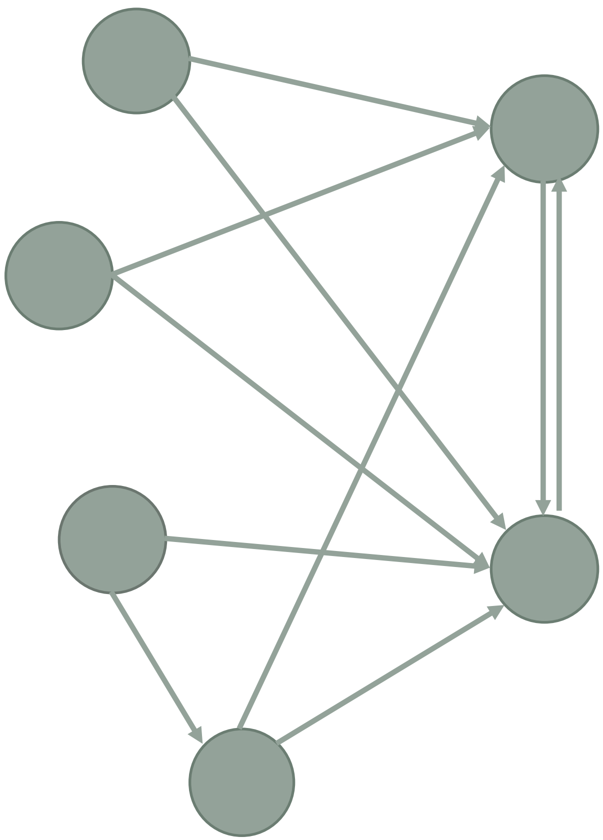


In “random walk,” jump to new node randomly with probability (1-*c*)

$$\overrightarrow{\mathrm{Rank}} = cA'\overrightarrow{\mathrm{Rank}} + \frac{1-c}{n}\vec{\mathbf{1}}$$

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

PageRank



In “random walk,” jump to new node randomly with probability (1-*c*)

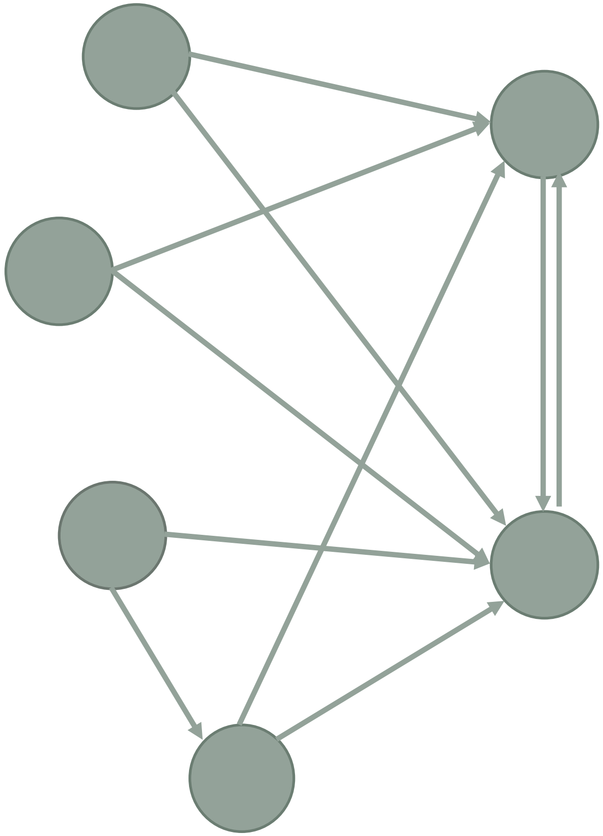
$$\overrightarrow{\text{Rank}} = cA'\overrightarrow{\text{Rank}} + \frac{1-c}{n}\vec{\mathbf{1}}$$

Rank is the first eigenvector of

$$cA' + \frac{1-c}{n}\vec{\mathbf{1}} \cdot \overrightarrow{\mathbf{1}^{\mathrm{T}}}$$

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

PageRank



Random Walk with Restarts

$$\overrightarrow{\text{Rank}} = cA'\overrightarrow{\text{Rank}} + \frac{1-c}{n}\vec{\mathbf{1}}$$

Rank is the first eigenvector of

$$cA' + \frac{1-c}{n}\vec{\mathbf{1}} \cdot \overrightarrow{\mathbf{1}^{\mathrm{T}}}$$

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

PageRank

Web Page	PageRank (average is 1.0)
Download Netscape Software	11589.00
http://www.w3.org/	10717.70
Welcome to Netscape	8673.51
Point: It's What You're Searching For	7930.92
Web-Counter Home Page	7254.97
The Blue Ribbon Campaign for Online Free Speech	7010.39
CERN Welcome	6562.49
Yahoo!	6561.80
Welcome to Netscape	6203.47
Wusage 4.1: A Usage Statistics System For Web Servers	5963.27
The World Wide Web Consortium (W3C)	5672.21
Lycos, Inc. Home Page	4683.31
Starting Point	4501.98
Welcome to Magellan!	3866.82
Oracle Corporation	3587.63

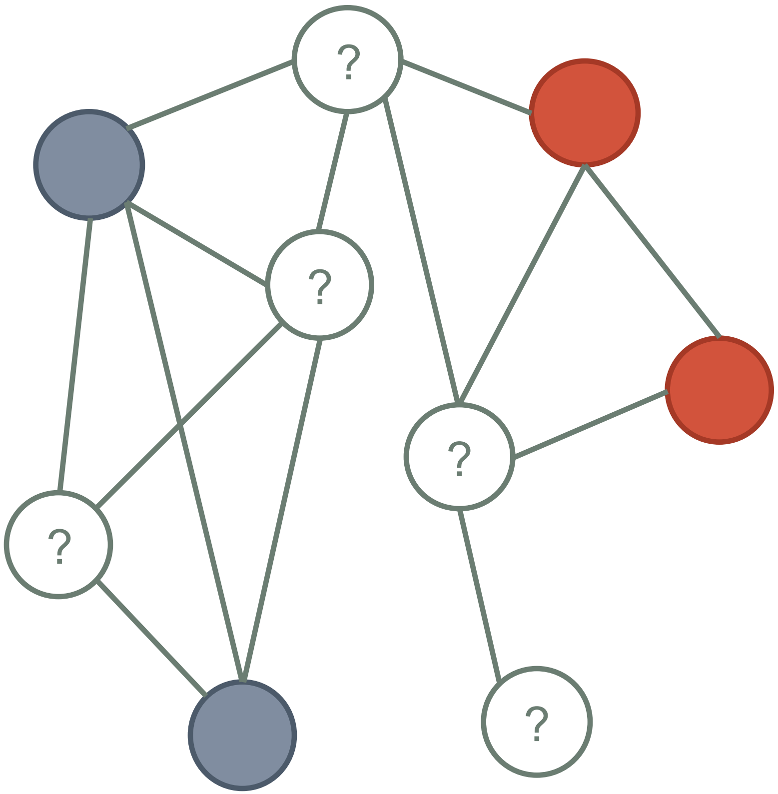
Table 1: Top 15 Page Ranks: July 1996

Anatomy of a Large-Scale Hypertextual Web Search Engine
Lawrence Page, Sergey Brin
WWW 1998

Practitioner's Guide

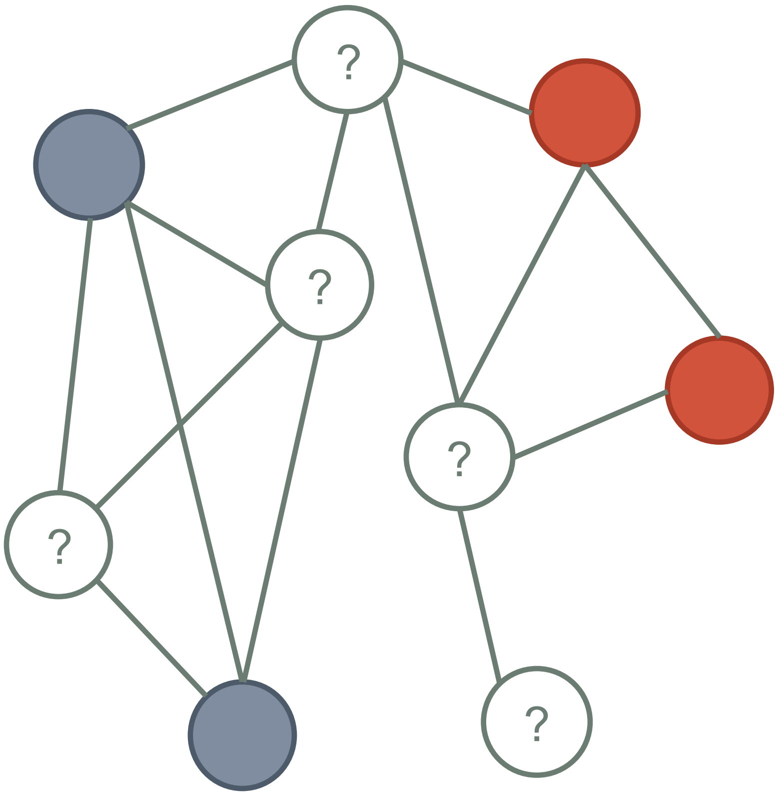
Method	Graph Type	Node Attributes	Edge Attributes	Seed Labels
HITS	Directed			
PageRank	Directed			Optional
Label Prop.	Undirected			✓
pMRF BP	Undirected			Preferred
EdgeExplain	Undirected	✓		✓

Semi-supervised Classification



Given a graph and labels for some nodes, can we learn the labels for the other nodes?

Semi-supervised Classification

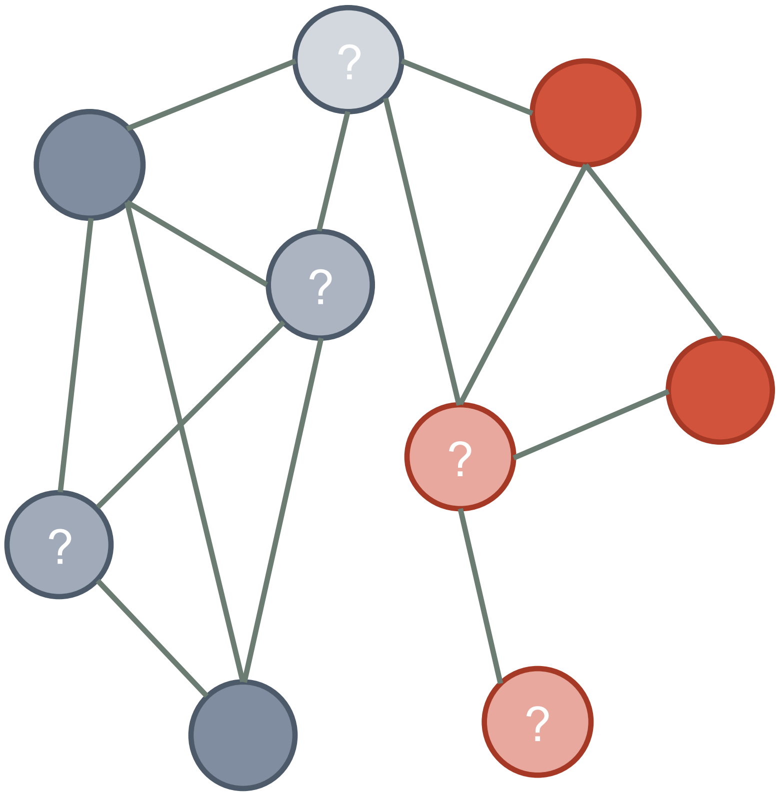


Given a graph and labels for some nodes, can we learn the labels for the other nodes?

Huge research area with many different formulations

(we will present only a few)

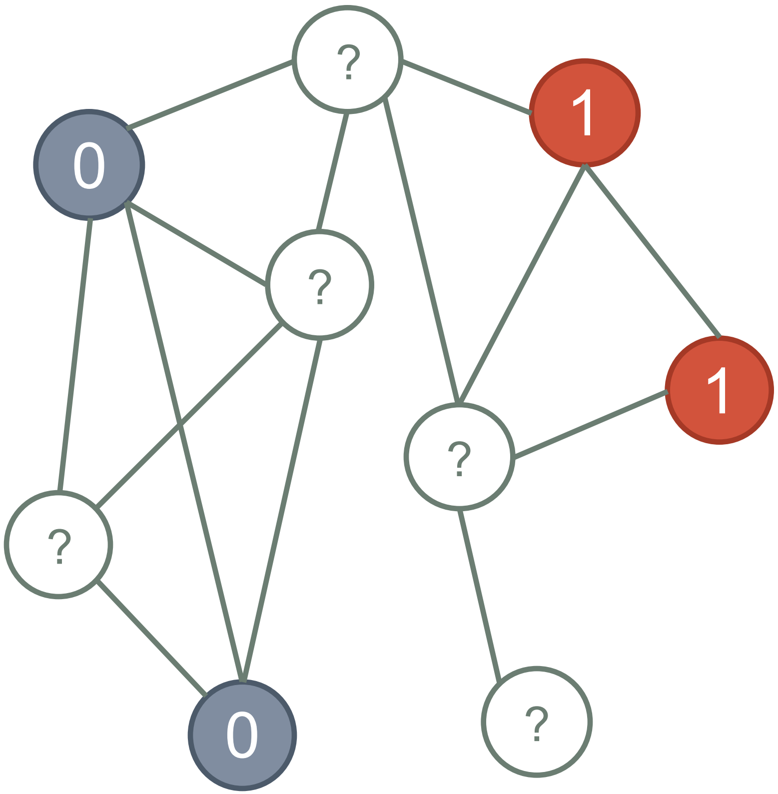
Semi-supervised Classification



Given a graph and labels for some nodes, can we learn the labels for the other nodes?

Generally, learn labels *X* so neighbors have the same label (homophily)

Semi-supervised Classification



$$\min_{f} \sum_{i,j} w_{i,j}(f(i) - f(j))^2$$

Edge weight $w_{i,j}$

$$f(i) = \frac{1}{\sum_j w_{i,j}} \sum_j w_{i,j} f(j)$$

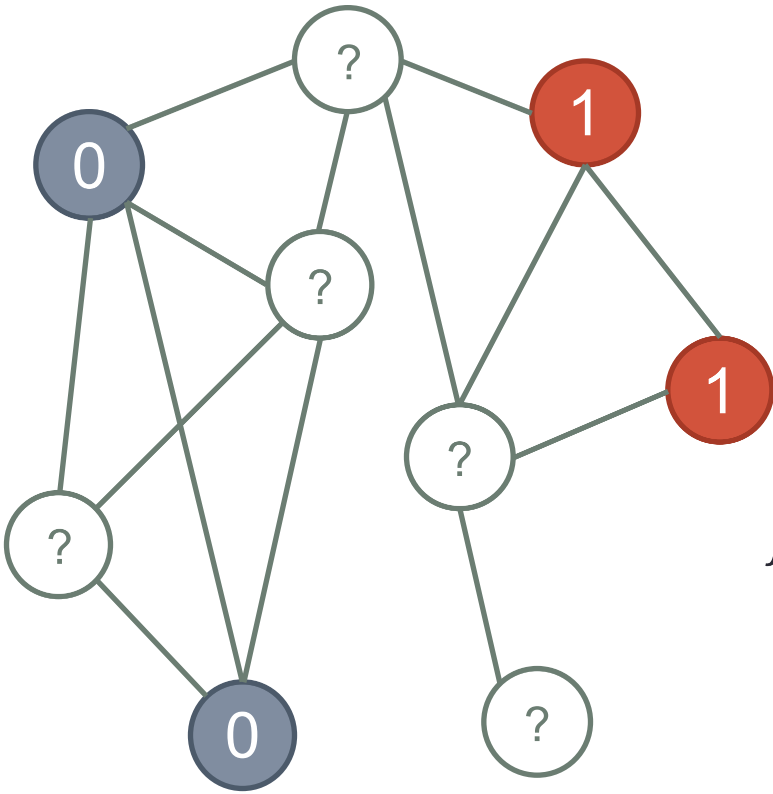
Can set label threshold for *f(i)*

Under this structure,
closed form solution given in paper

Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions
Xiaojin Zhu, Zoubin Ghahramani, John Lafferty
ICML 2003



Semi-supervised Classification

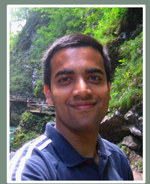


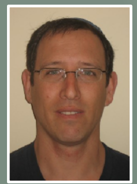
Can view as a random walk:

- Go to neighbor with probability proportional to $w_{i,j}$
- Return node-prior with probability p_v
- Return no guess with probability q_v

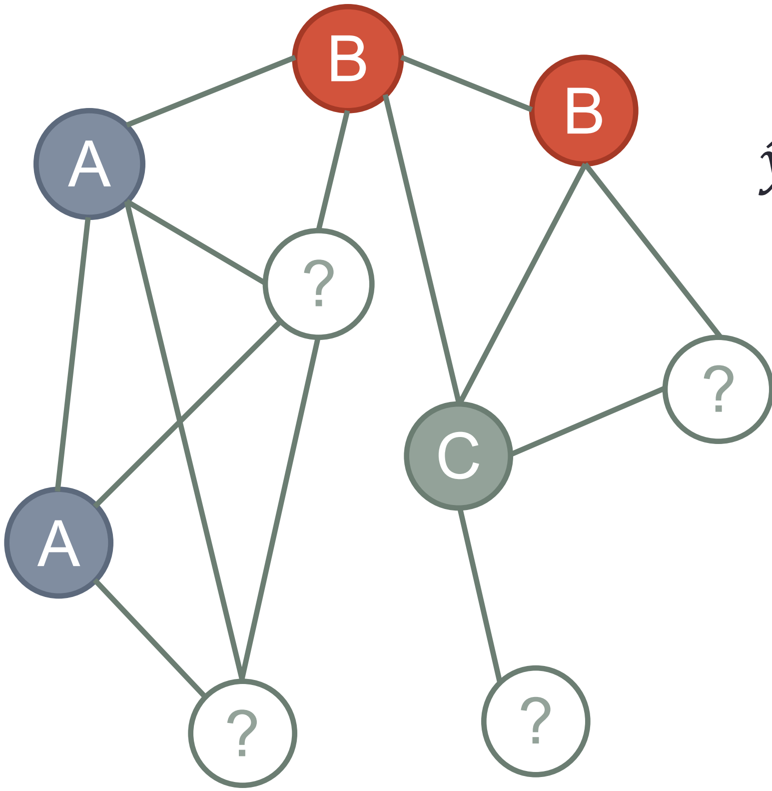
$$\hat{f}(i) = p_i f(i) + q_i \frac{1}{2} + \frac{1}{\sum_j w_{i,j}} \sum_j w_{i,j} f(j)$$

New Regularized Algorithms for Transductive Learning
Partha Pratim Talukdar and Koby Crammer
ECML/PKDD 2009





Semi-supervised Classification



$$\hat{y}_v = [p(A), p(B), p(C), p(\text{Don't Know})]$$

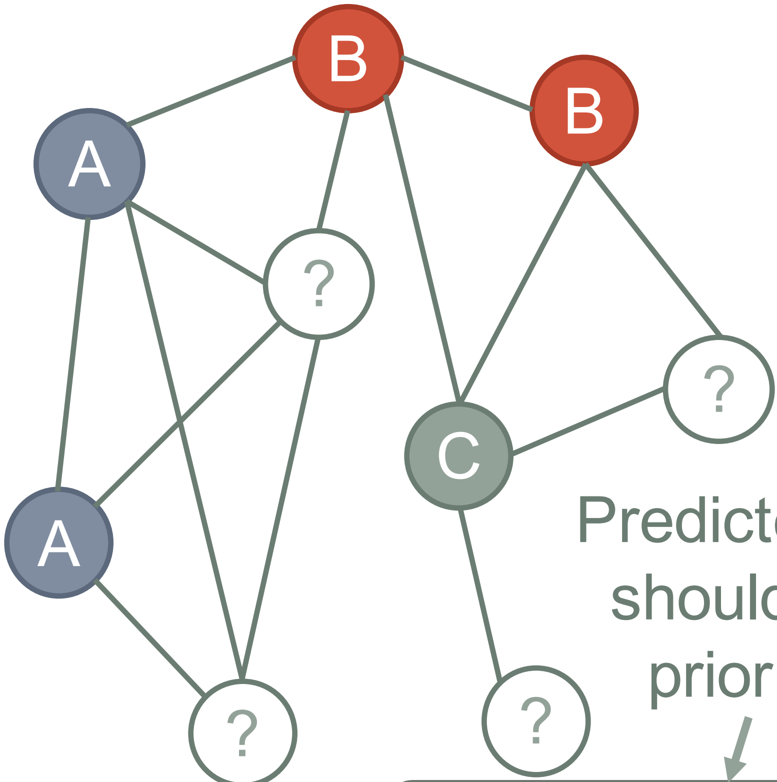
Observe prior labels y_v for some v

General prior $r = [0,0,0,1]$

$$\min_{\hat{y}} \sum_{v,l} p_v \left(y_{v,l} - \hat{y}_{v,l}\right)^2 + \sum_{u,v} w_{u,v} \|\hat{y}_u - \hat{y}_v\|^2 + \sum_v q_v \|\hat{y}_v - r\|^2$$

New Regularized Algorithms for Transductive Learning
Partha Pratim Talukdar and Koby Crammer
ECML/PKDD 2009

Semi-supervised Classification



$$\hat{y}_v = [p(A), p(B), p(C), p(\text{Don't Know})]$$

Observe prior labels y_v for some *v*

General prior $r = [0,0,0,1]$

Predicted labels should match prior labels

Neighbors should have similar labels

Regularization

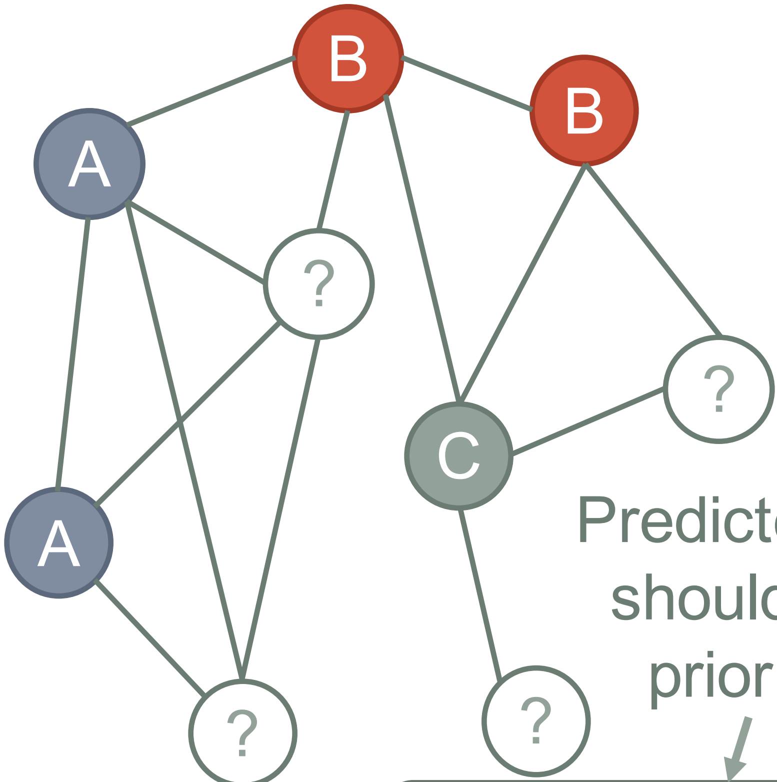
$$\min_{\hat{y}} \sum_{v,l} \boxed{p_v \left(y_{v,l} - \hat{y}_{v,l}\right)^2} + \sum_{u,v} \boxed{w_{u,v} \|\hat{y}_u - \hat{y}_v\|^2} + \sum_{v} \boxed{q_v \|\hat{y}_v - r\|^2}$$

New Regularized Algorithms for Transductive Learning

Partha Pratim Talukdar and Koby Crammer

ECML/PKDD 2009

Semi-supervised Classification



Can learn $\hat{y}$ through message passing of gradients

Predicted labels should match prior labels

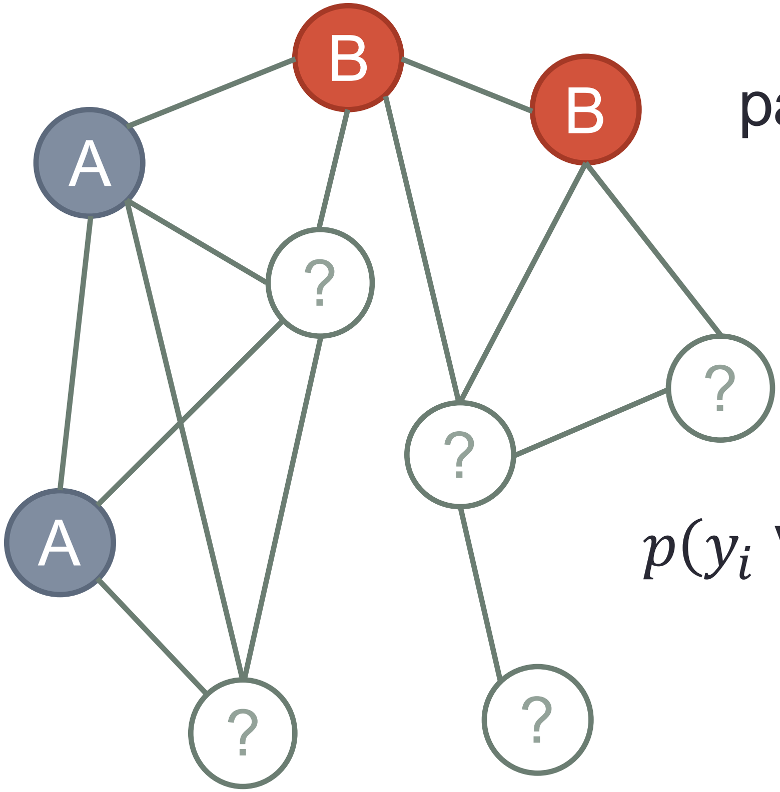
Neighbors should have similar labels

Regularization

$$\min_{\hat{y}} \sum_{v,l} \boxed{p_v \left(y_{v,l} - \hat{y}_{v,l}\right)^2} + \sum_{u,v} \boxed{w_{u,v} \|\hat{y}_u - \hat{y}_v\|^2} + \sum_{v} \boxed{q_v \|\hat{y}_v - r\|^2}$$

New Regularized Algorithms for Transductive Learning
Partha Pratim Talukdar and Koby Crammer
ECML/PKDD 2009

Semi-supervised Classifications: pMRF

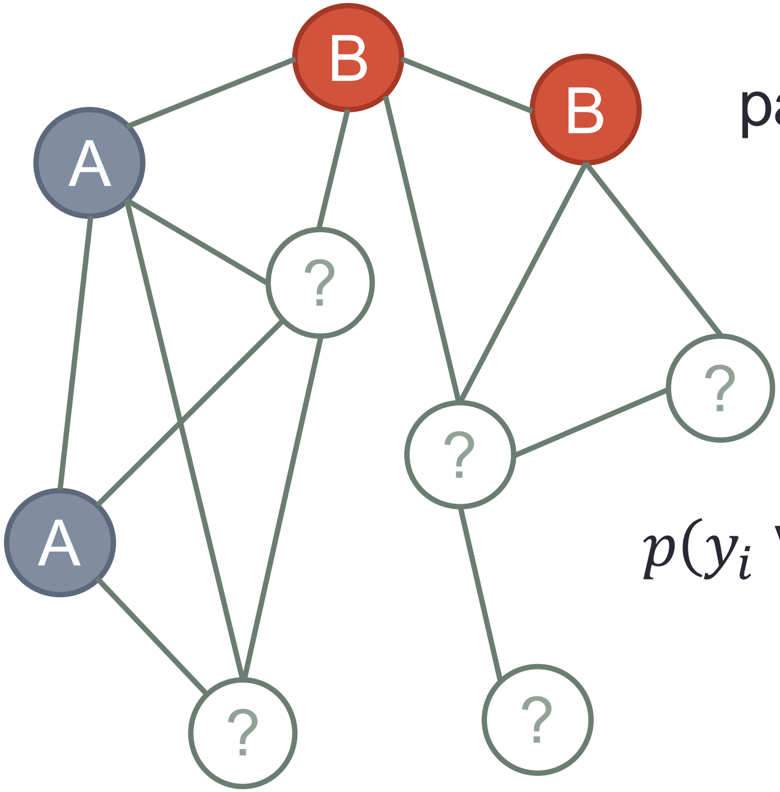


Probabilistic interpretation with pairwise Markov random fields (pMRF)

Observe labels for nodes $\mathcal{L}$ and infer labels for nodes in $\mathcal{U}$

$$p(y_i \; \forall i \in \mathcal{U}) = \frac{1}{Z} \prod_i \phi_i(y_i) \prod_{(i,j) \in E} \psi_{i,j}(y_i, y_j)$$

Semi-supervised Classifications: pMRF



Probabilistic interpretation with pairwise Markov random fields (pMRF)

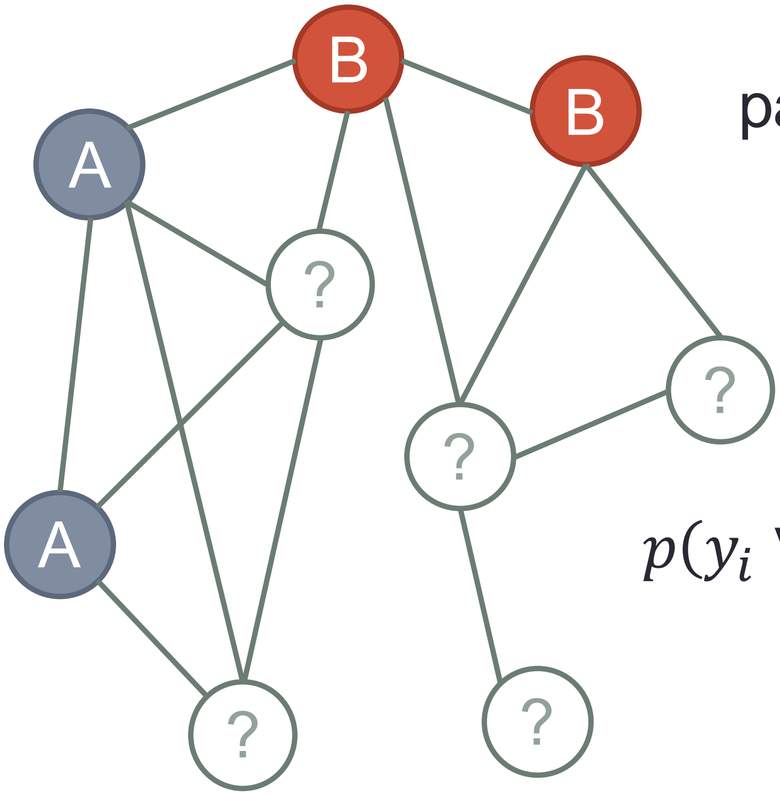
Observe labels for nodes $\mathcal{L}$ and infer labels for nodes in $\mathcal{U}$

$$p(y_i \; \forall i \in \mathcal{U}) = \frac{1}{Z} \prod_i \boxed{\phi_i(y_i)} \prod_{(i,j) \in E} \boxed{\psi_{i,j}(y_i, y_j)}$$

Prior belief for node *i*

Compatibility potentials between neighbors

Semi-supervised Classifications: pMRF



Probabilistic interpretation with pairwise Markov random fields (pMRF)

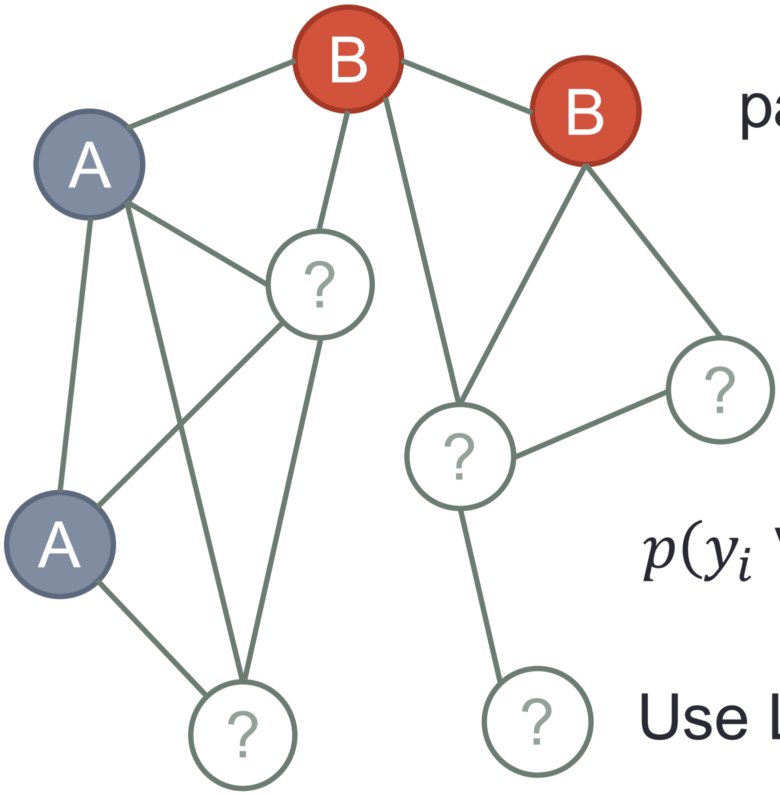
Observe labels for nodes $\mathcal{L}$ and infer labels for nodes in $\mathcal{U}$

$$p(y_i \ \forall i \in \mathcal{U}) = \frac{1}{Z} \prod_i \phi_i(y_i) \prod_{(i,j) \in E} \boxed{\psi_{i,j}(y_i, y_j)}$$

Compatibility potentials between neighbors

y_i \ y_j	A	B
A	0.8	0.2
B	0.2	0.8

Semi-supervised Classifications: pMRF



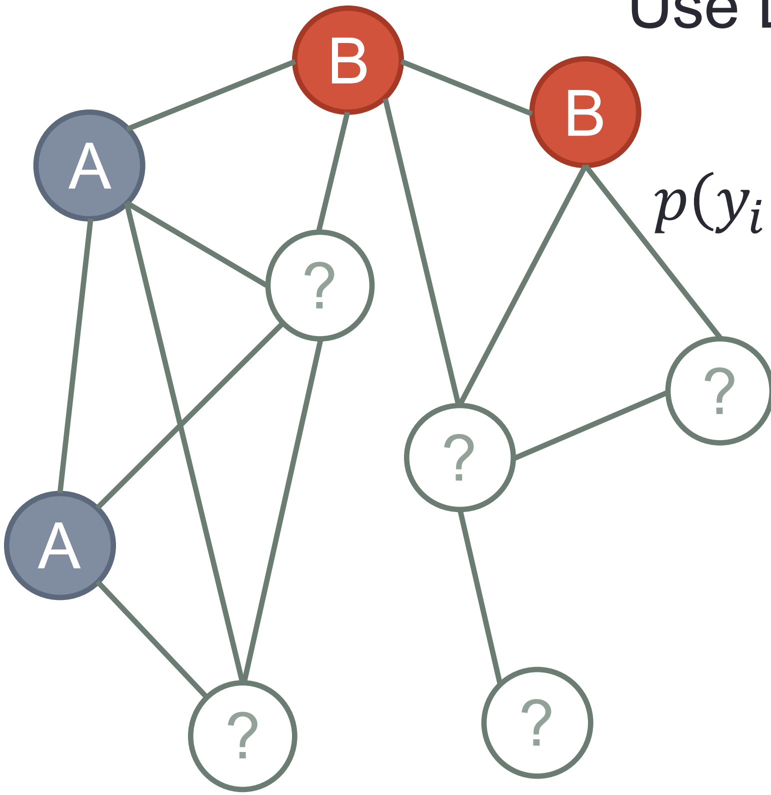
Probabilistic interpretation with pairwise Markov random fields (pMRF)

Observe labels for nodes $\mathcal{L}$ and infer labels for nodes in $\mathcal{U}$

$$p(y_i \; \forall i \in \mathcal{U}) = \frac{1}{Z} \prod_i \phi_i(y_i) \prod_{(i,j)\in E} \psi_{i,j}(y_i, y_j)$$

Use Loopy Belief Propagation [Pearl, 1982] to estimate most likely state.

Semi-supervised Classifications: pMRF

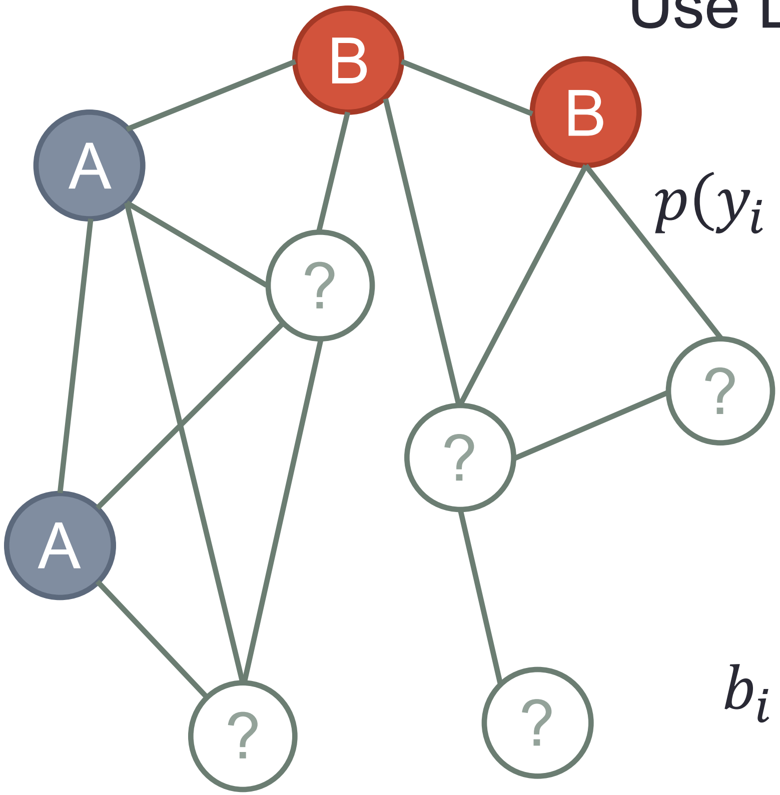


Use Loopy Belief Propagation [Pearl, 1982] to estimate most likely state.

$$p(y_i\ \forall i \in \mathcal{U}) = \frac{1}{Z}\prod_i \phi_i(y_i) \prod_{(i,j)\in E} \psi_{i,j}(y_i, y_j)$$

Iteratively send messages between nodes to learn state.

Semi-supervised Classifications: pMRF



Use Loopy Belief Propagation [Pearl, 1982] to estimate most likely state.

$$p(y_i \ \forall i \in \mathcal{U}) = \frac{1}{Z} \prod_i \phi_i(y_i) \prod_{(i,j) \in E} \psi_{i,j}(y_i, y_j)$$

Iteratively send messages between nodes to learn state.

Node y_i estimated by

$$b_i(y_i) \leftarrow \alpha \phi_i(y_i) \prod_{(i,j) \in E} m_{j \to i}(y_i)$$

Message from i to j given by

$$m_{i \to j}(y_j) \leftarrow \sum_{y_i} \phi_i(y_i) \psi_{i,j}(y_i, y_j) \prod_{(i,k) \in E | k \neq j} m_{k \to i}(y_i)$$

DETAILS!

Unifying Propagation Methods

Method	Homophily	Heterophily	Convergence	Scalability
RWR	✓	✗	✓	✓
SSL	✓	?	✓	✓
BP	✓	✓	?	✓

Unifying Guilt-by-Association Approaches: Theorems and Fast Algorithms
Danai Koutra, Tai-You Ke, U Kang, Polo Chau,
Hsing-Kuo Kenneth Pao, Christos Faloutsos
ECML/PKDD 2011

Unifying Propagation Methods

Can approximate all methods by:

Method	matrix	unknown		known
RWR	$[\mathbf{I} - c\mathbf{A}\mathbf{D}^{-1}]\times$	$\boldsymbol{x}$	$=$	$(1-c)\ \boldsymbol{y}$
SSL	$[\mathbf{I} + \alpha(\mathbf{D} - \mathbf{A})]\times$	$\boldsymbol{x}$	$=$	$\boldsymbol{y}$
Gaussian BP $=$ SSL	$[\mathbf{I} + \alpha(\mathbf{D} - \mathbf{A})]\times$	$\boldsymbol{x}$	$=$	$\boldsymbol{y}$

1. Subgraph Analysis

2. Propagation Methods

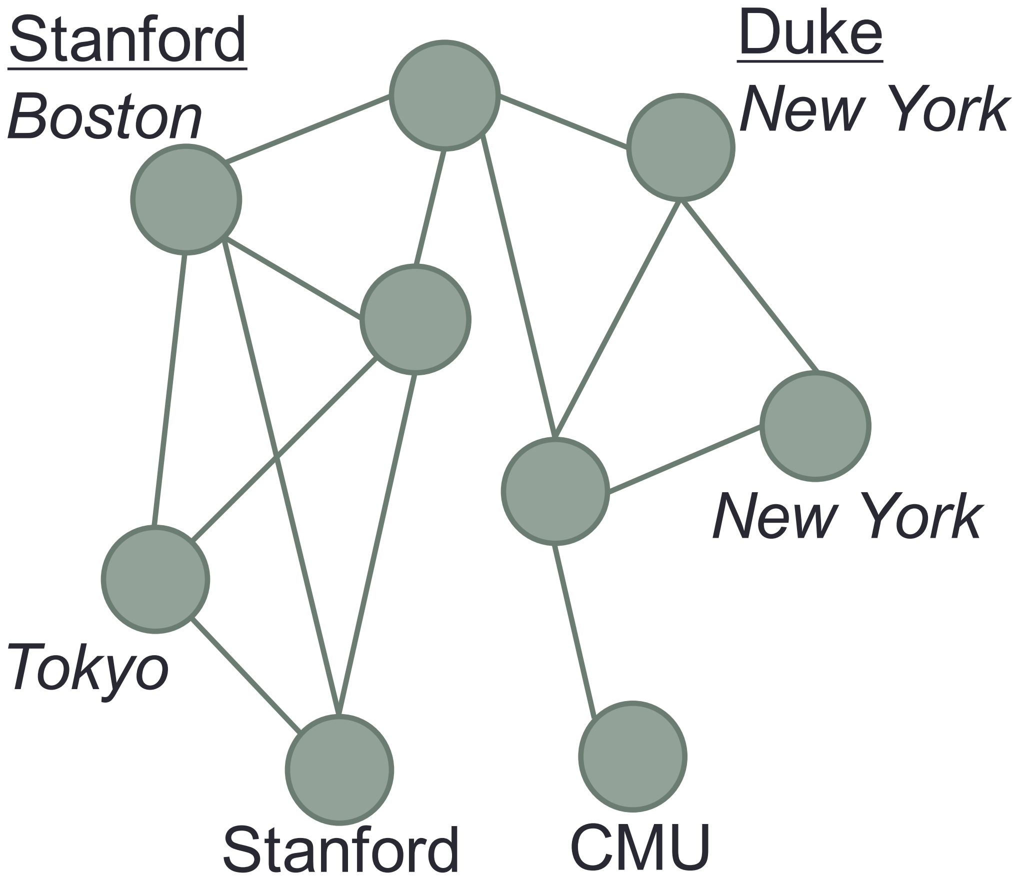
a) Background

b) Normal Behavior

c) Abnormal Behavior

3. Latent Factor Models

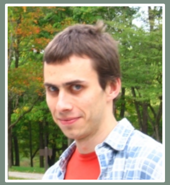
Explain friendships to infer user properties



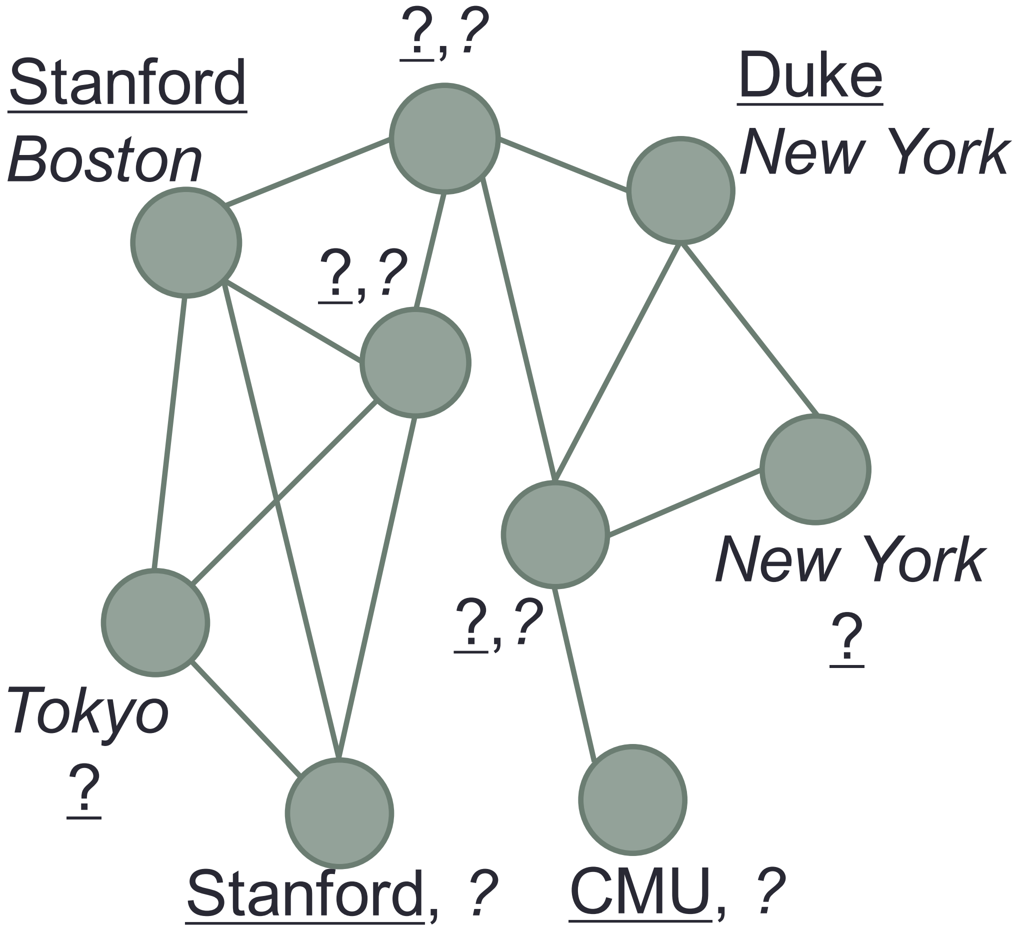
If we have incomplete labels of many types, can we infer the rest of the labels?

Idea:
Adjacent nodes only need to be agree on one label

Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak, Jonathan Chang, Sofus A. Macskassy
ICML 2014

Explain friendships to infer user properties

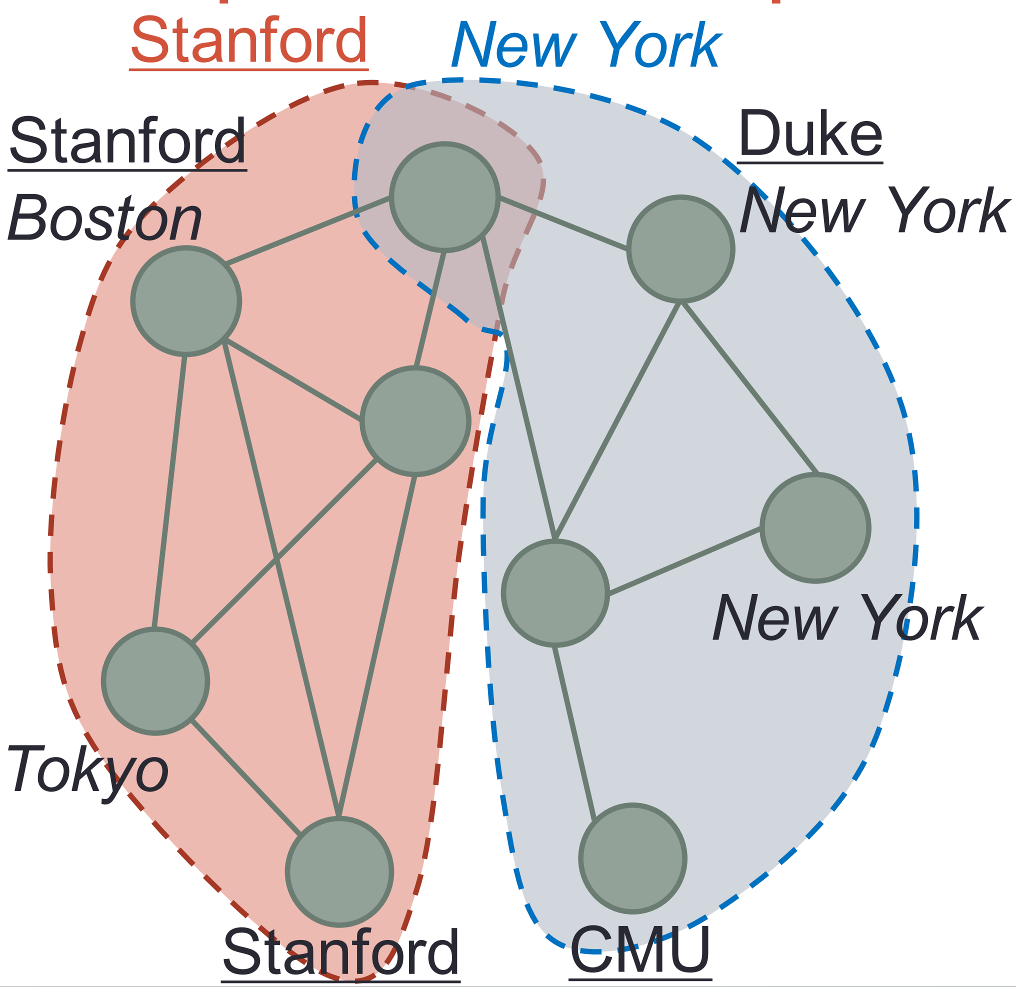


If we have incomplete labels of many types, can we infer the rest of the labels?

Idea:
Adjacent nodes only need to be agree on one label

Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak, Jonathan Chang, Sofus A. Macskassy
ICML 2014

Explain friendships to infer user properties

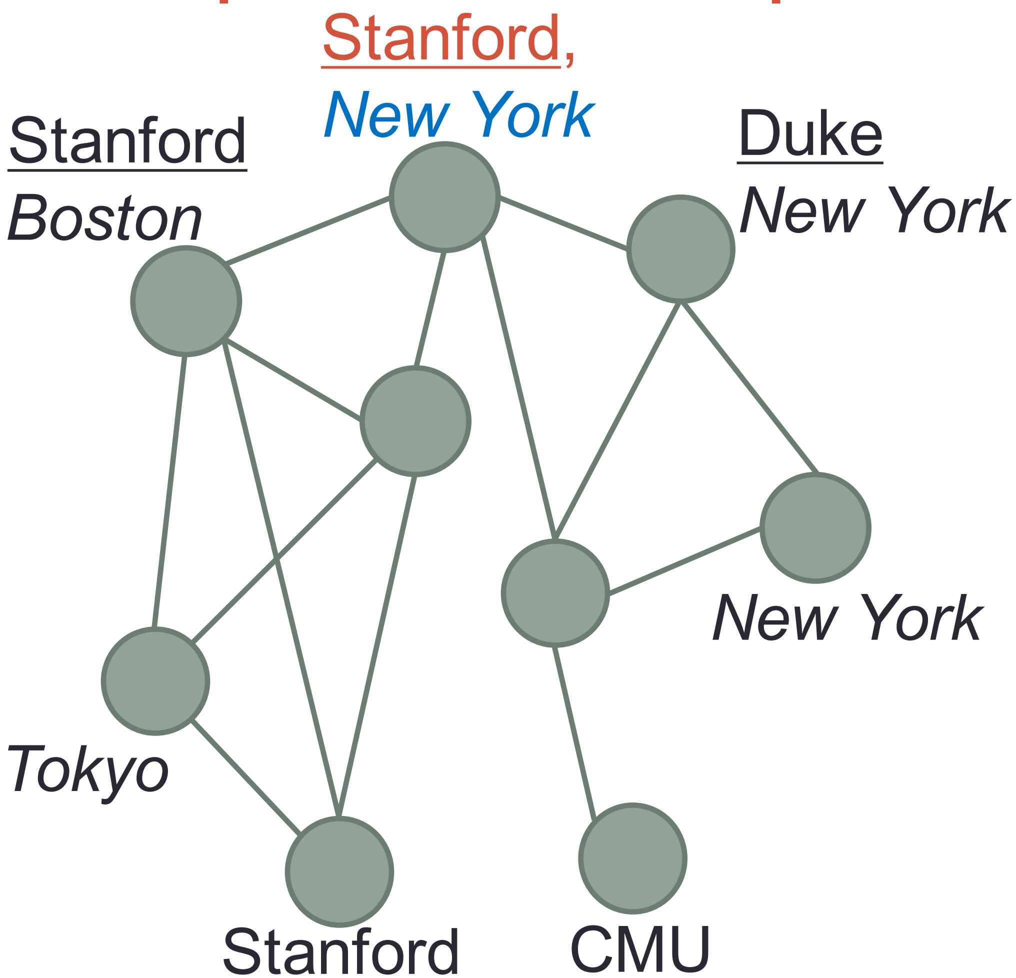


If we have incomplete labels of many types, can we infer the rest of the labels?

Idea:
Adjacent nodes only need to be agree on one label

Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak,
Jonathan Chang, Sofus A. Macskassy
ICML 2014

Explain friendships to infer user properties

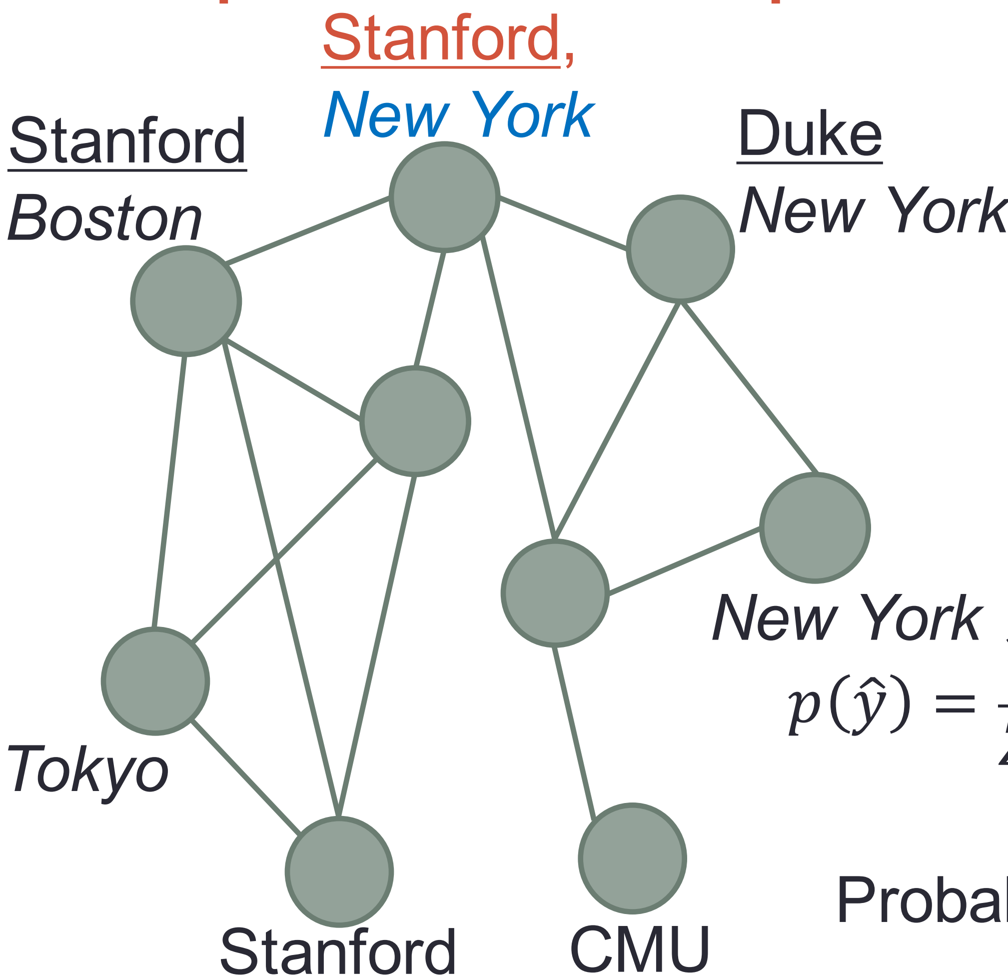


If we have incomplete labels of many types, can we infer the rest of the labels?

Idea:
Adjacent nodes only need to be agree on one label

Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak,
Jonathan Chang, Sofus A. Macskassy
ICML 2014

Explain friendships to infer user properties



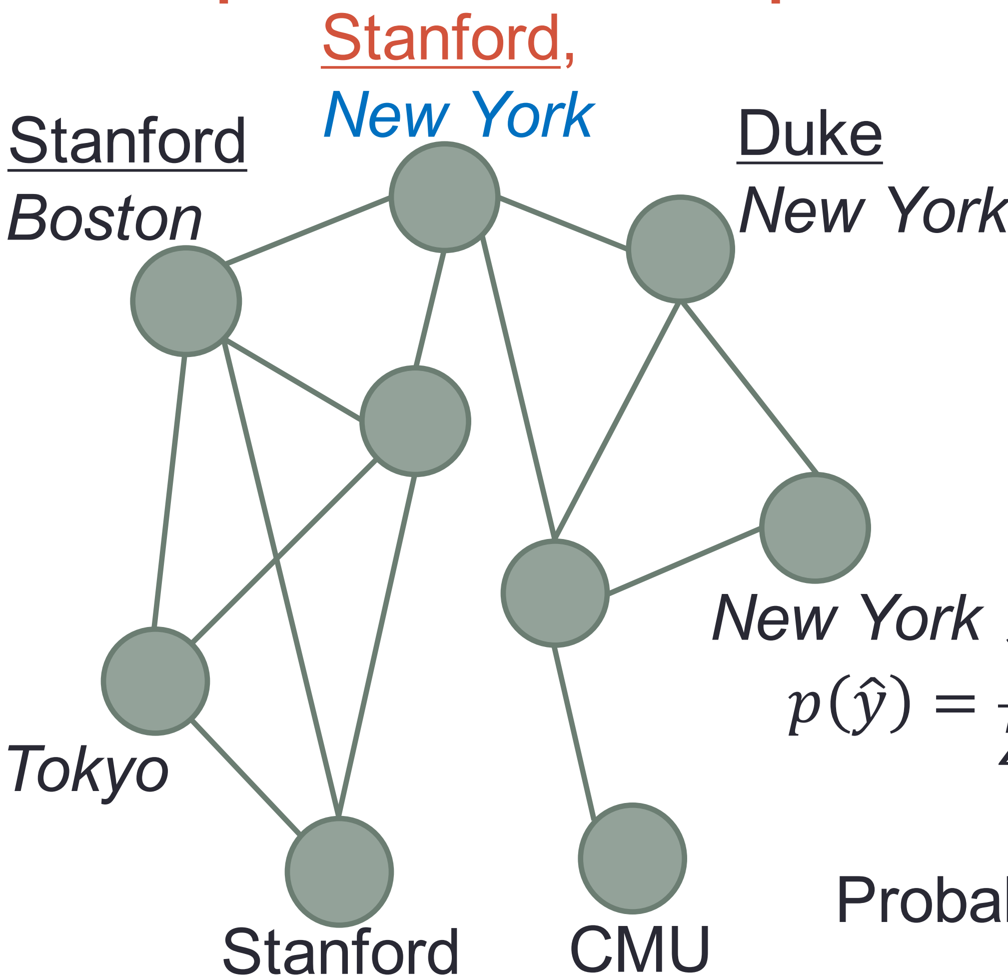
For each user *u* and property type *t*, we can observe 1 label from set *L(t)*, as given by vector $y_{u,t}$

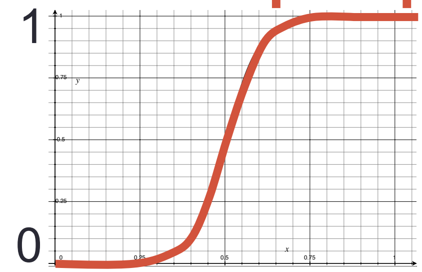
$$p(\hat{y}) = \frac{1}{Z} \prod_{(u,v)\in E} \sigma\left(c + \alpha \sum_{\text{Type } t} \hat{y}_{u,t} \cdot \hat{y}_{v,t}\right)$$

Probability vectors: $\sum_{l\in L(t)} \hat{y}_{u,t,l} = 1$

Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak,
Jonathan Chang, Sofus A. Macskassy
ICML 2014

Explain friendships to infer user properties





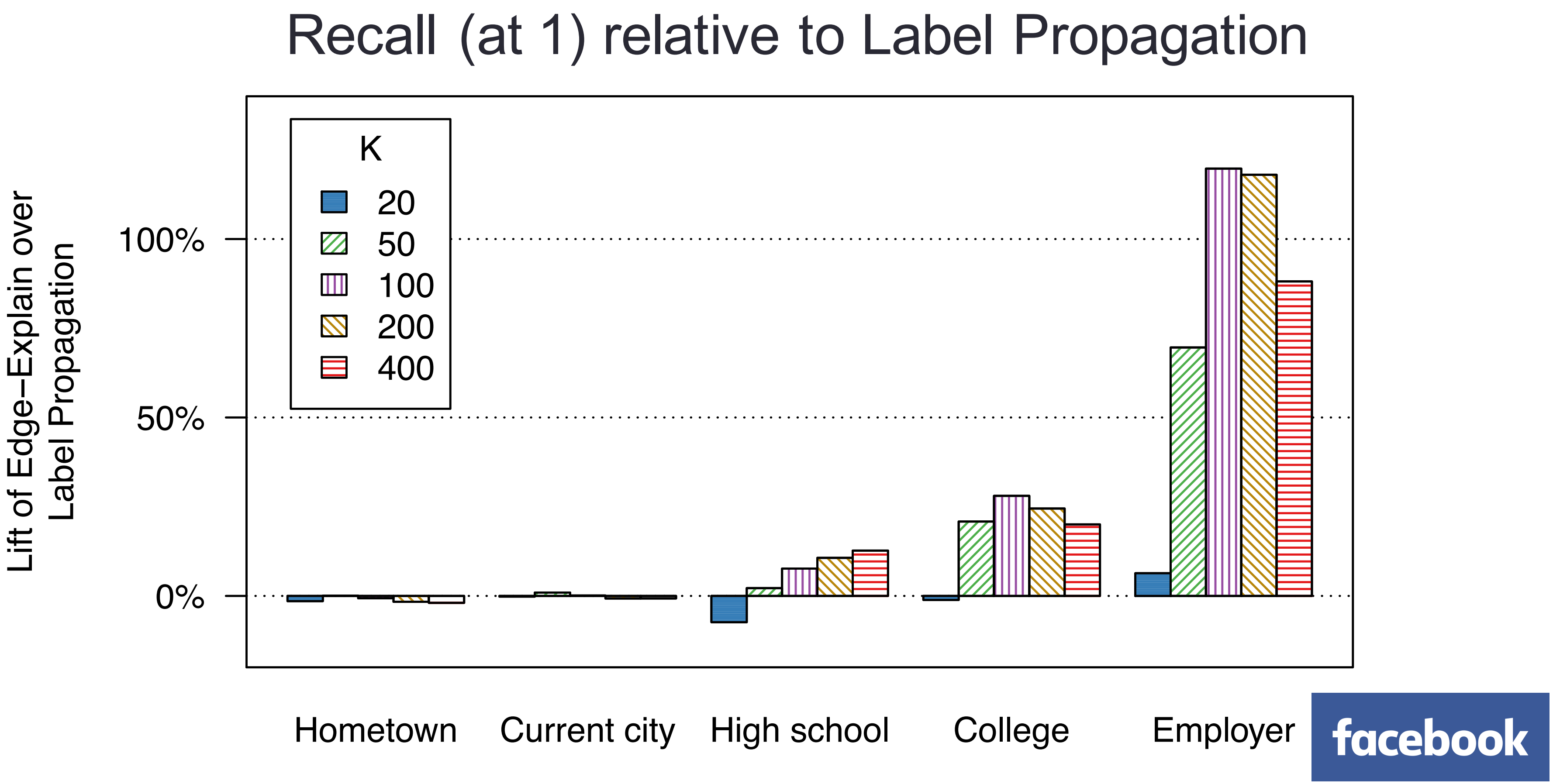
Sigmoid function so no additional benefit beyond being similar on one type

$$p(\hat{y}) = \frac{1}{Z} \prod_{(u,v)\in E} \sigma\left(c + \alpha \sum_{\text{Type } t} \hat{y}_{u,t} \cdot \hat{y}_{v,t}\right)$$

Probability vectors: $\sum_{l \in L(t)} \hat{y}_{u,t,l} = 1$

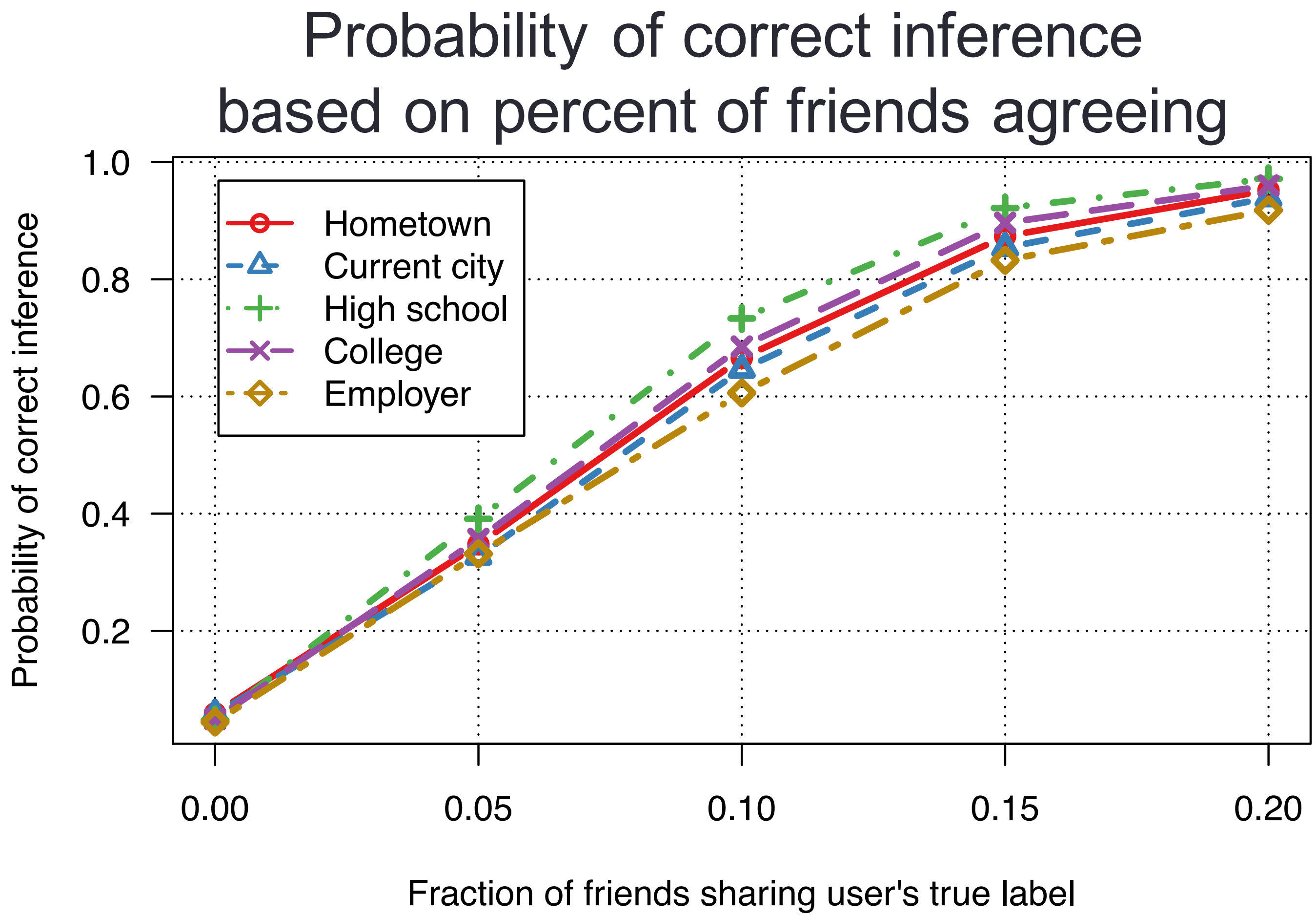
Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak,
Jonathan Chang, Sofus A. Macskassy
ICML 2014

Explain friendships to infer user properties



Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak,
Jonathan Chang, Sofus A. Macskassy
ICML 2014

Explain friendships to infer user properties



Joint Inference of Multiple Label Types in Large Networks
Deepayan Chakrabarti, Stanislav Funiak,
Jonathan Chang, Sofus A. Macskassy
ICML 2014

1. Subgraph Analysis

2. Propagation Methods

a) Background

b) Normal Behavior

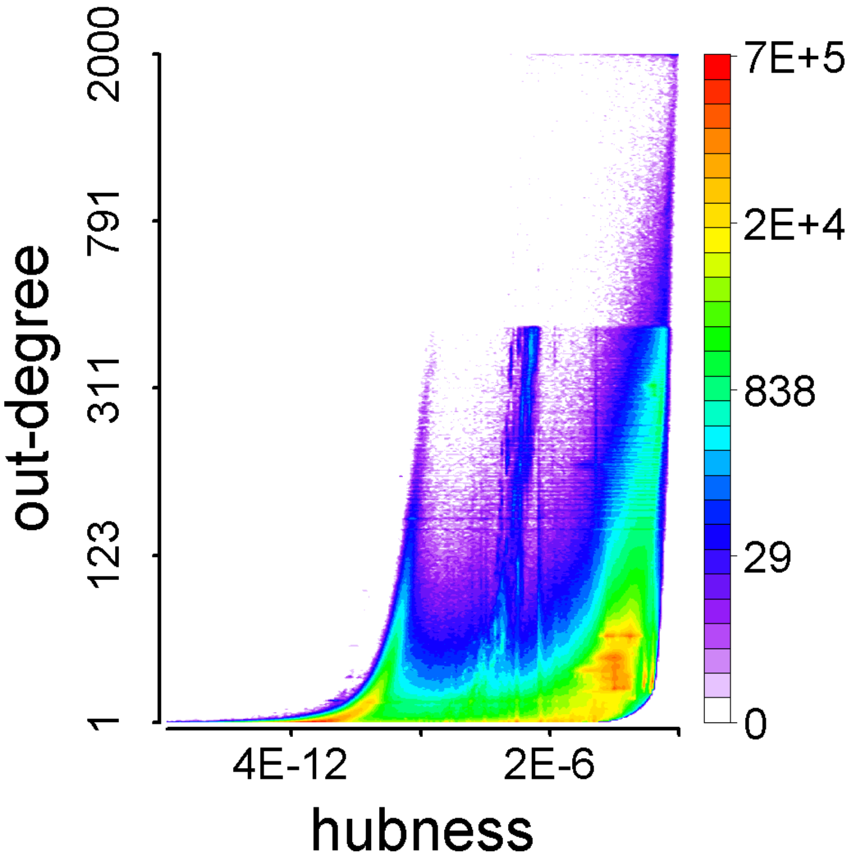
c) Abnormal Behavior

3. Latent Factor Models

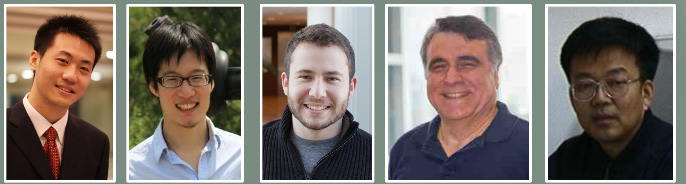
Find Fraud in HITS

Model a user's following behavior:

- Out-degree
- Hubness



CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

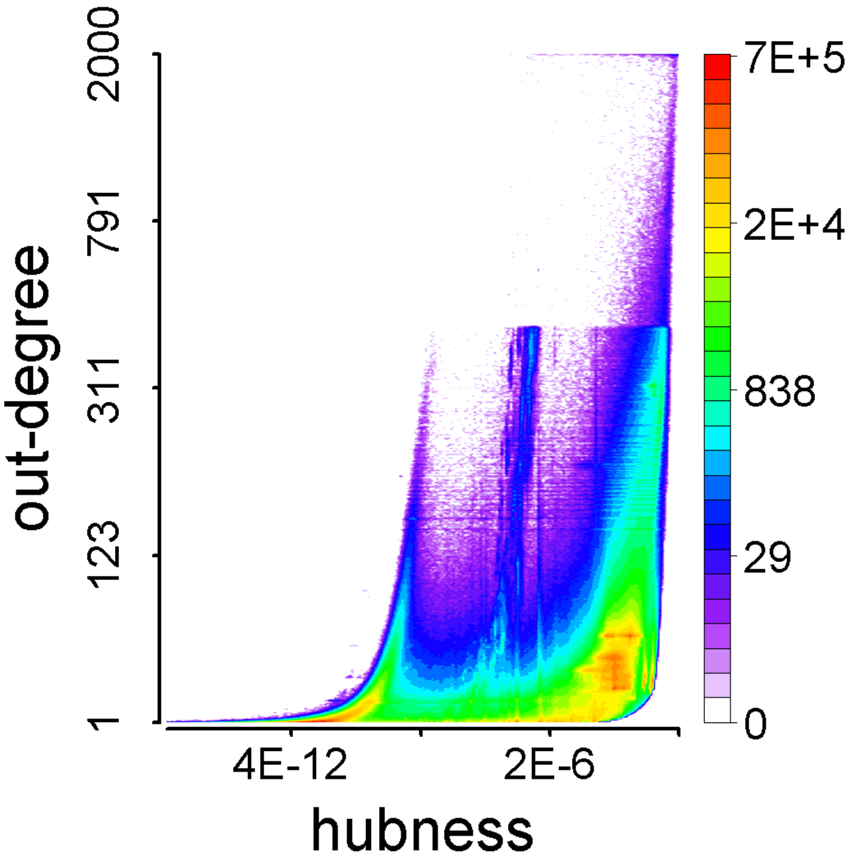


Find Fraud in HITS

Model a user's following behavior:

- Out-degree
- Hubness

Surprising to have high out-degree but low hubness



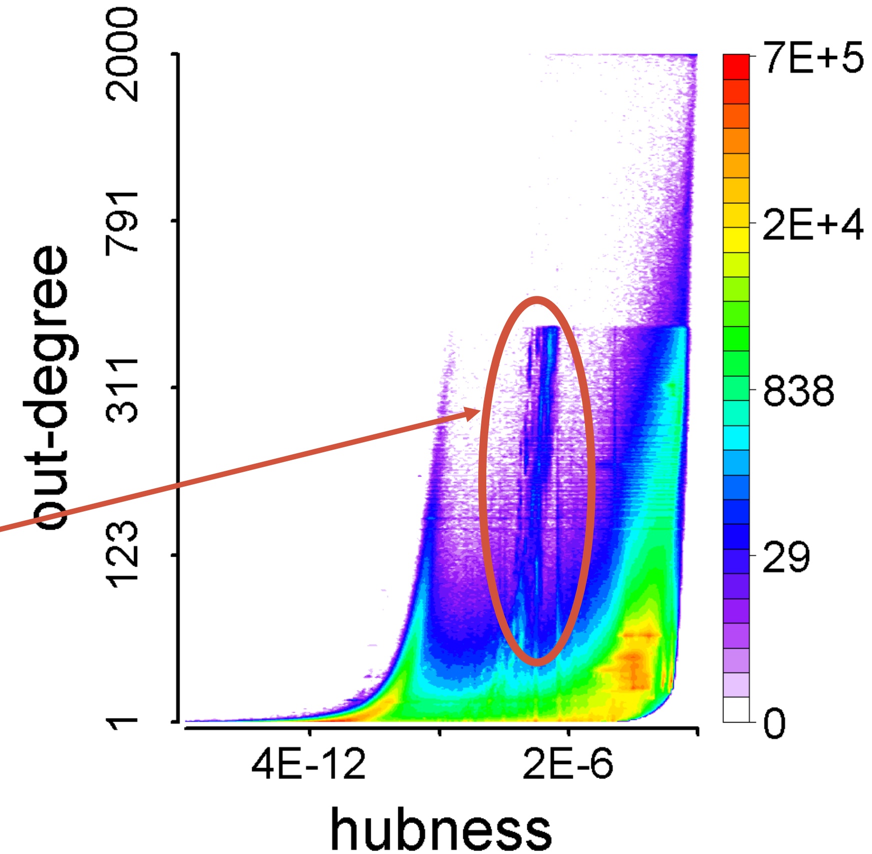
CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

Find Fraud in HITS

Model a user's following behavior:

- Out-degree
- Hubness

Surprising to have high out-degree but low hubness

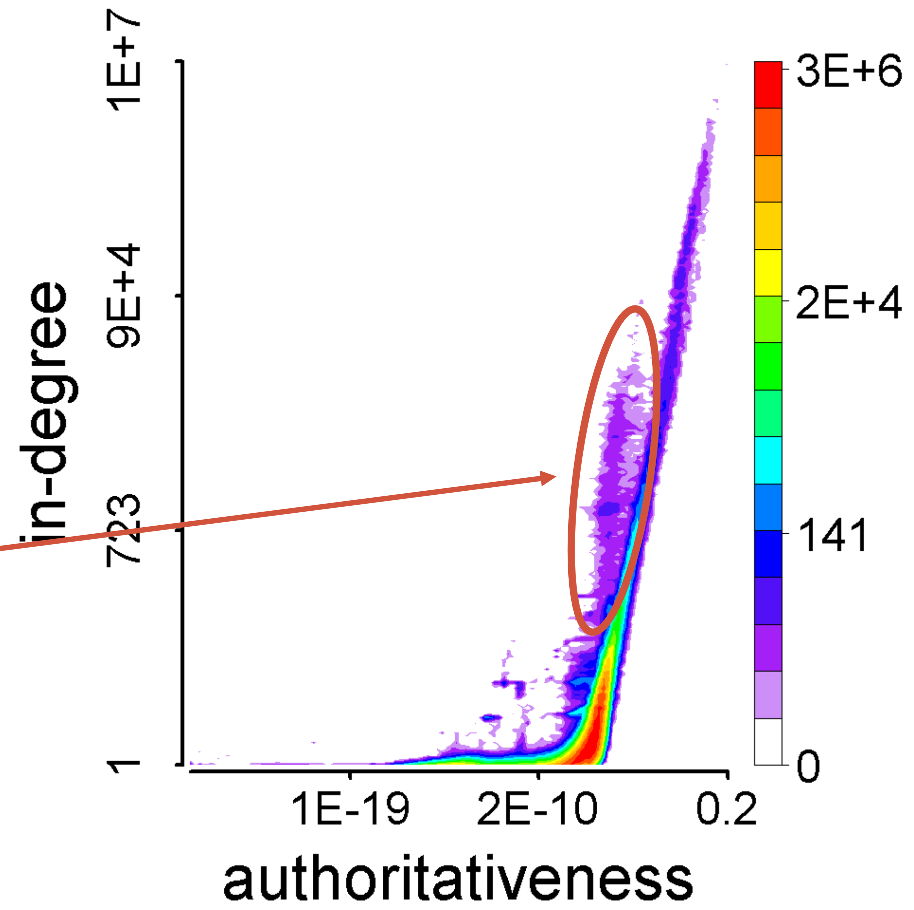


CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

Find Fraud in HITS

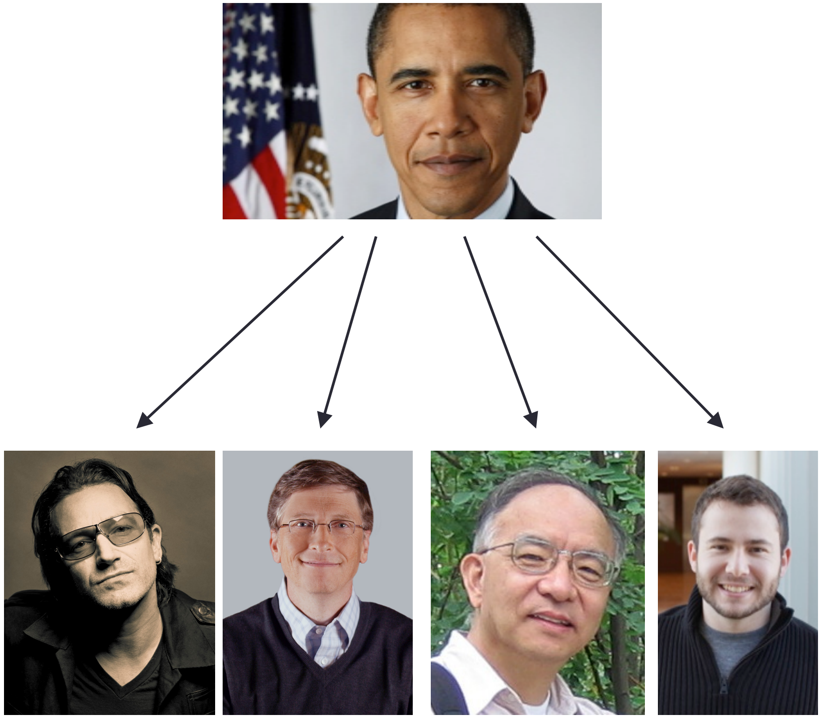
Model a user's followee behavior:
- In-degree
- Authoritativeness

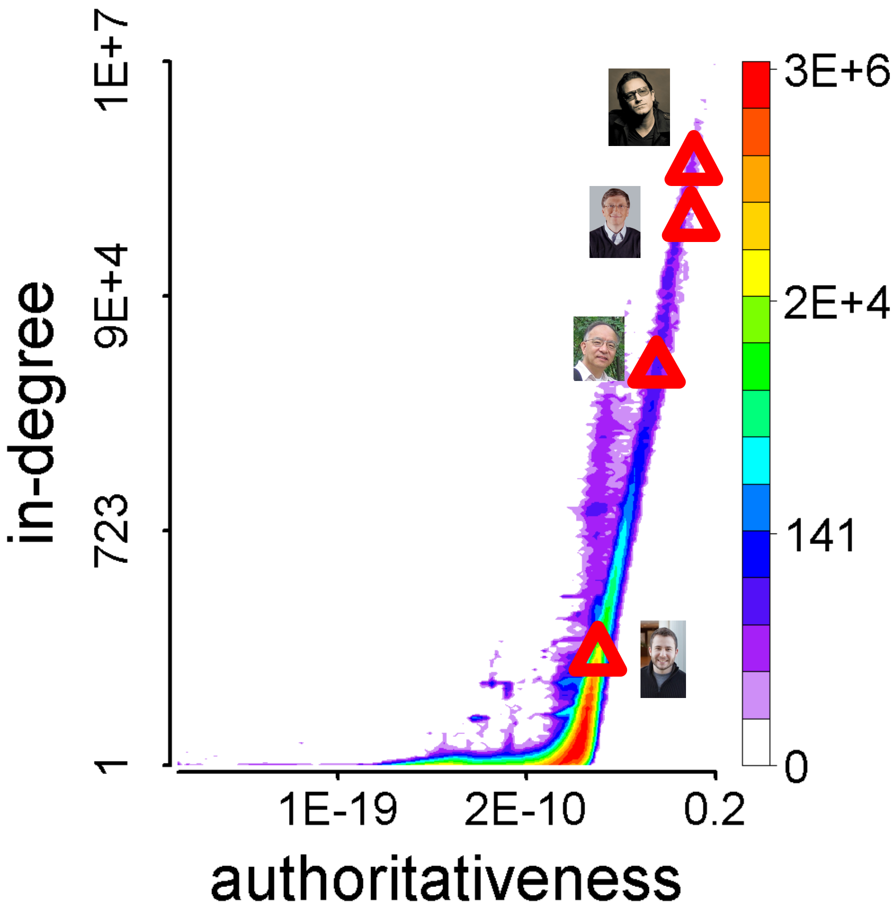
Surprising to have high in-degree but low authoritativeness



CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

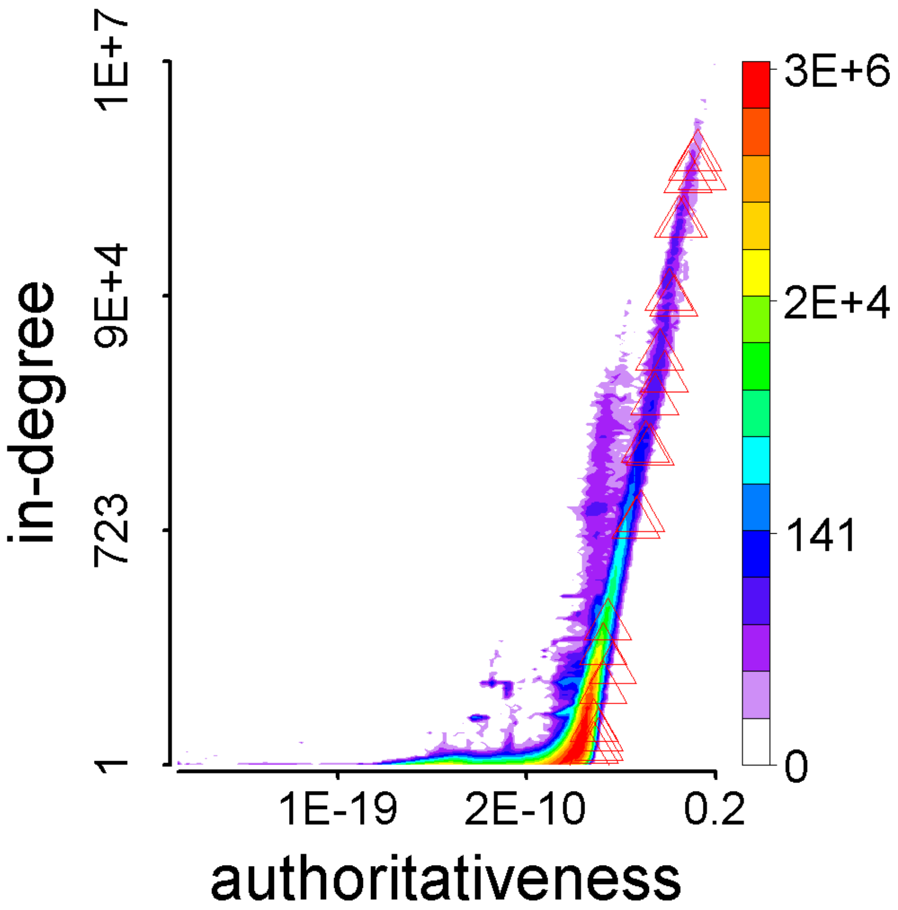
Find Fraud in HITS





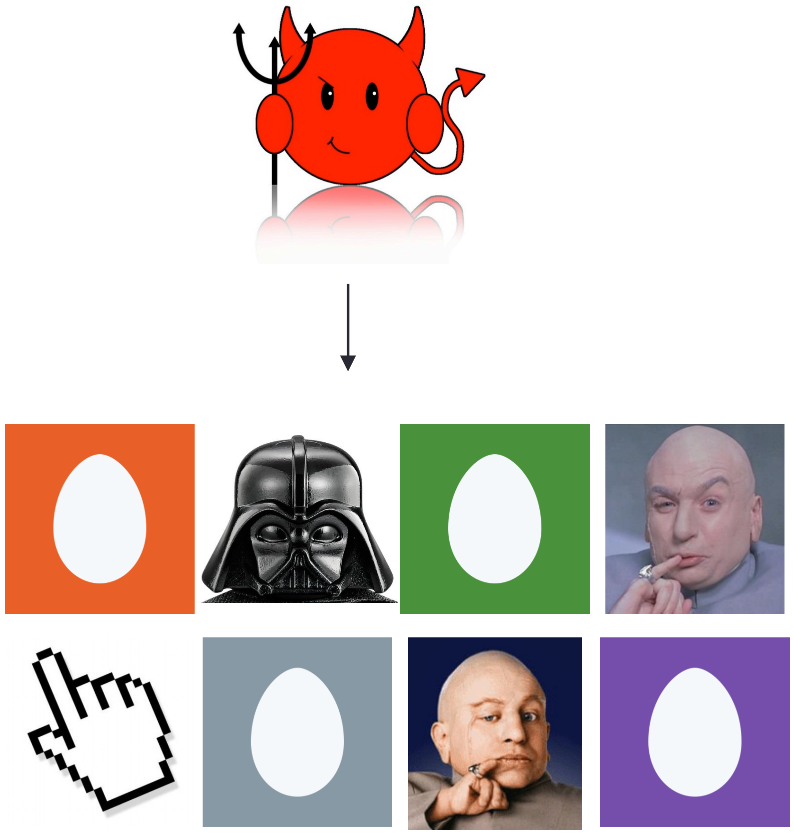
CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

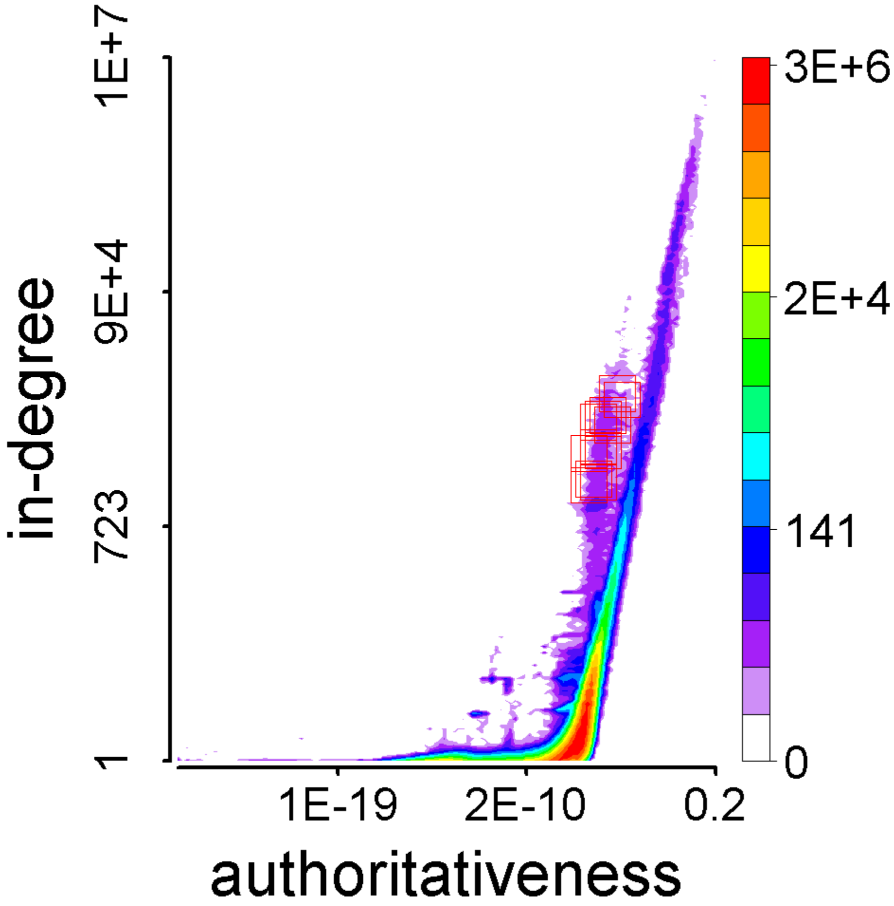
Find Fraud in HITS



CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

Find Fraud in HITS

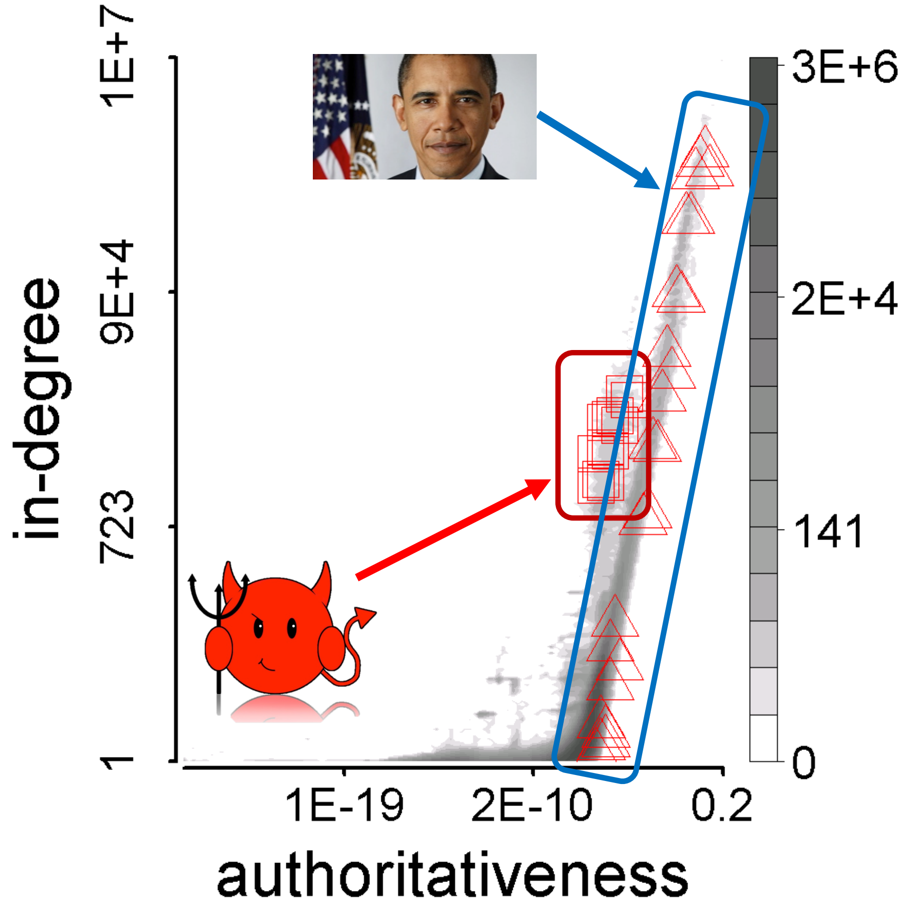




CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

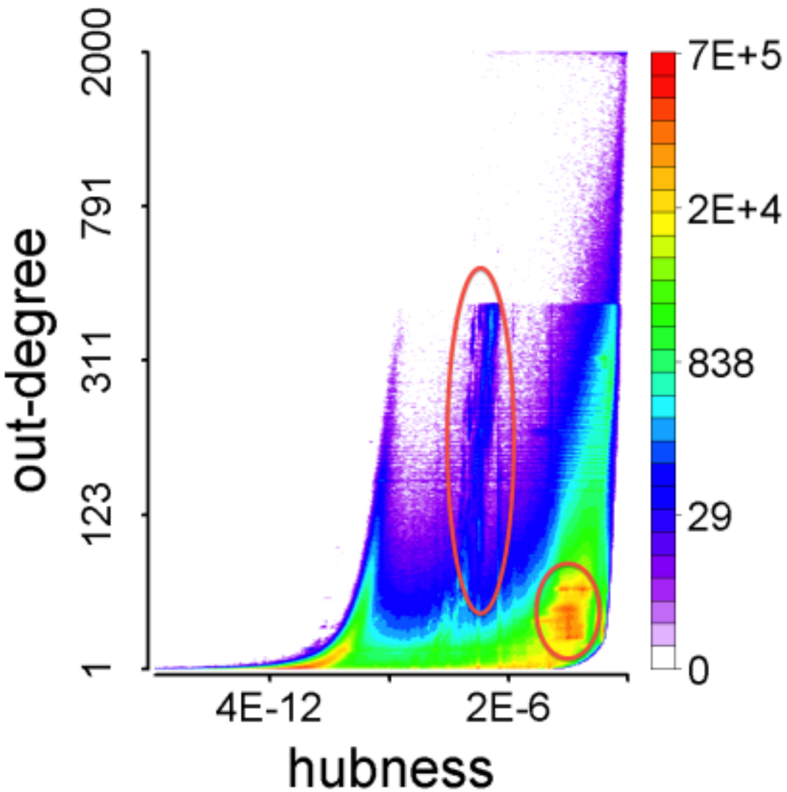
Find Fraud in HITS

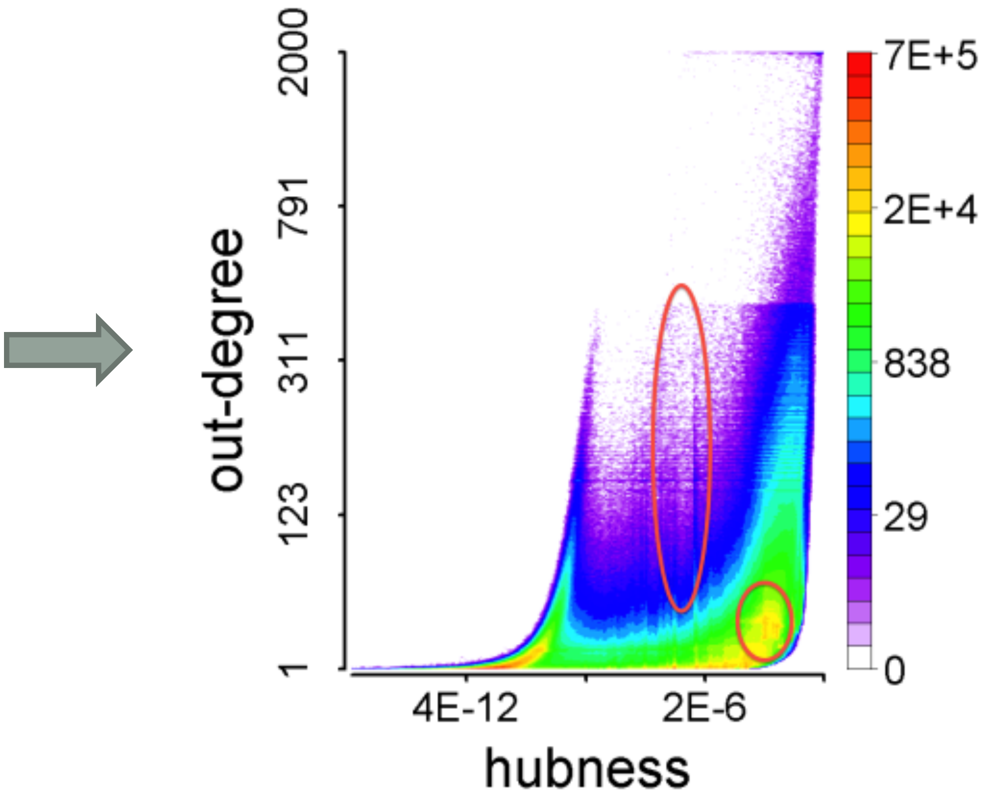
Suspicious Behavior:
- Synchronized
- Abnormal



CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

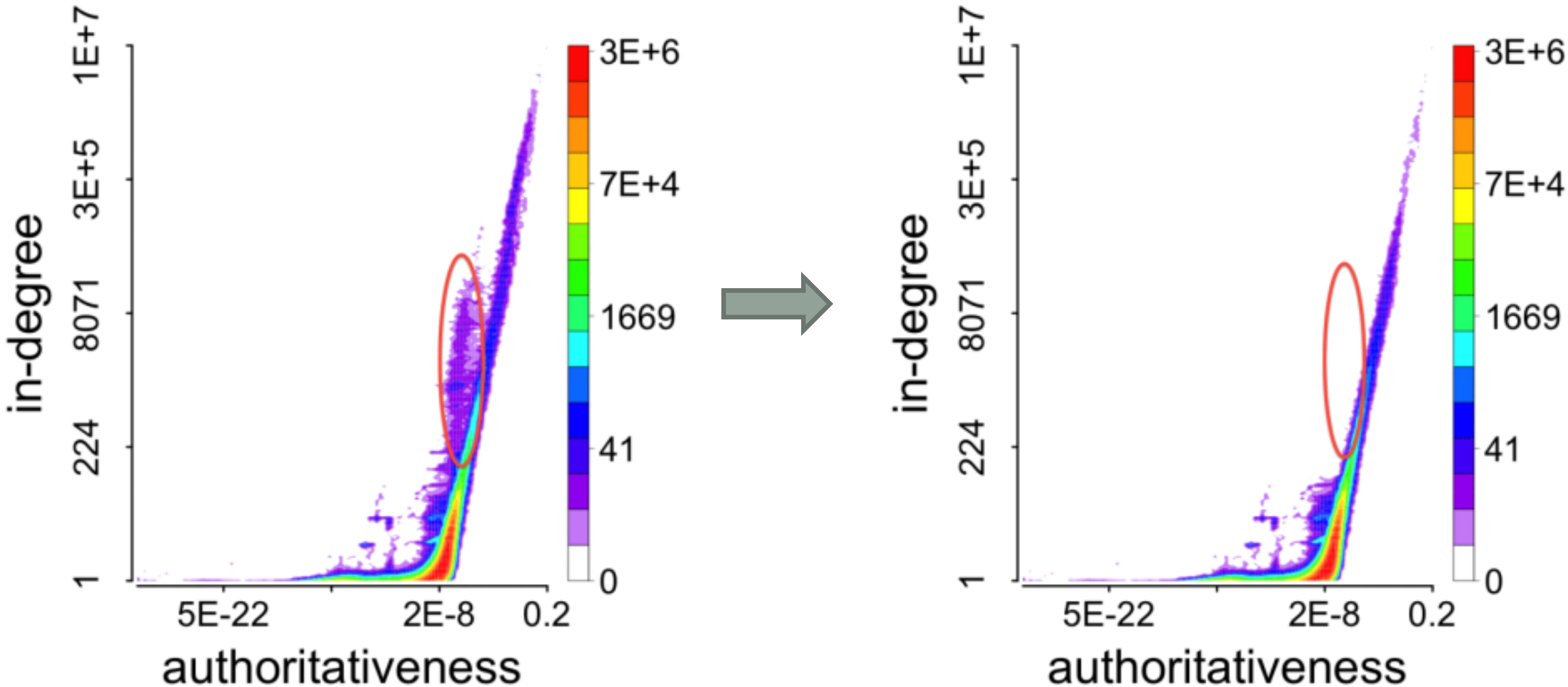
Find Fraud in HITS





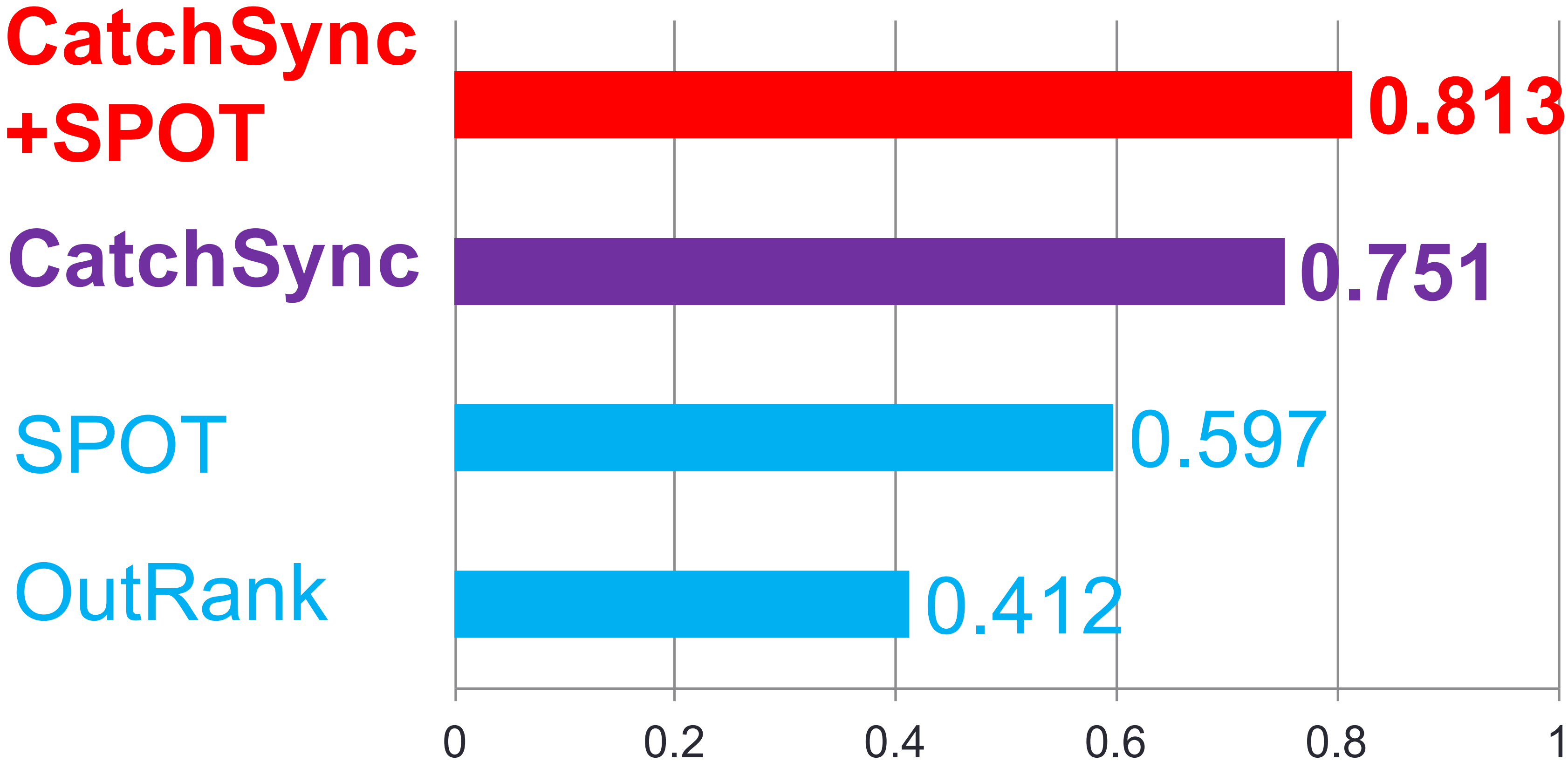
CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

Find Fraud in HITS



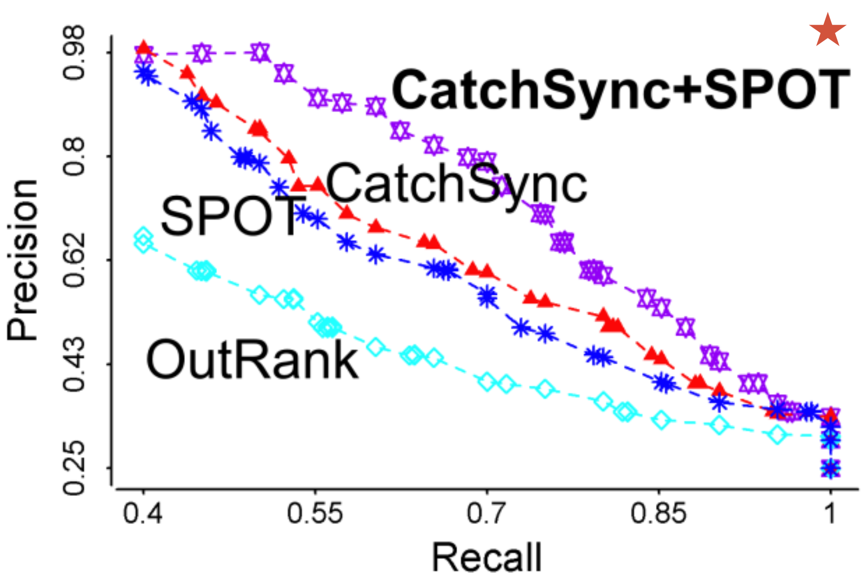
CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

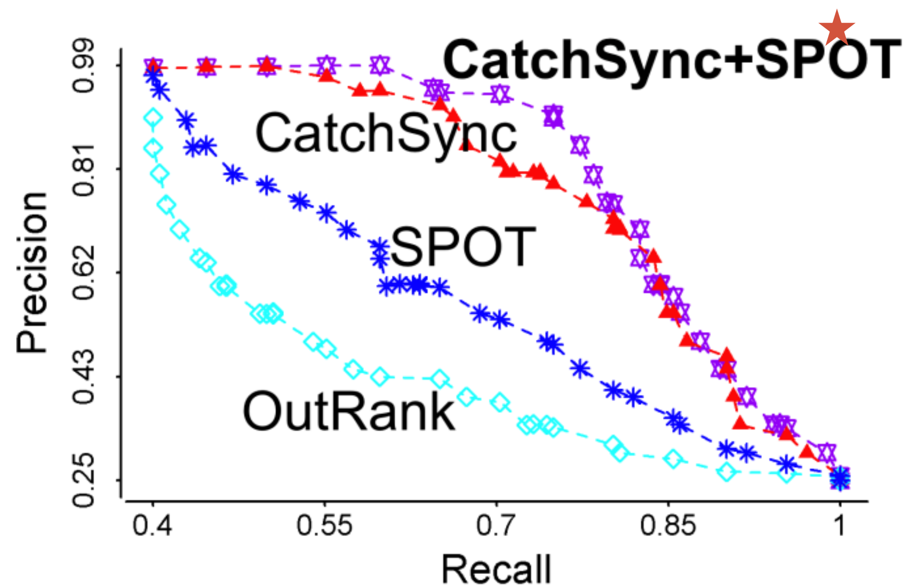
Find Fraud in HITS



CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

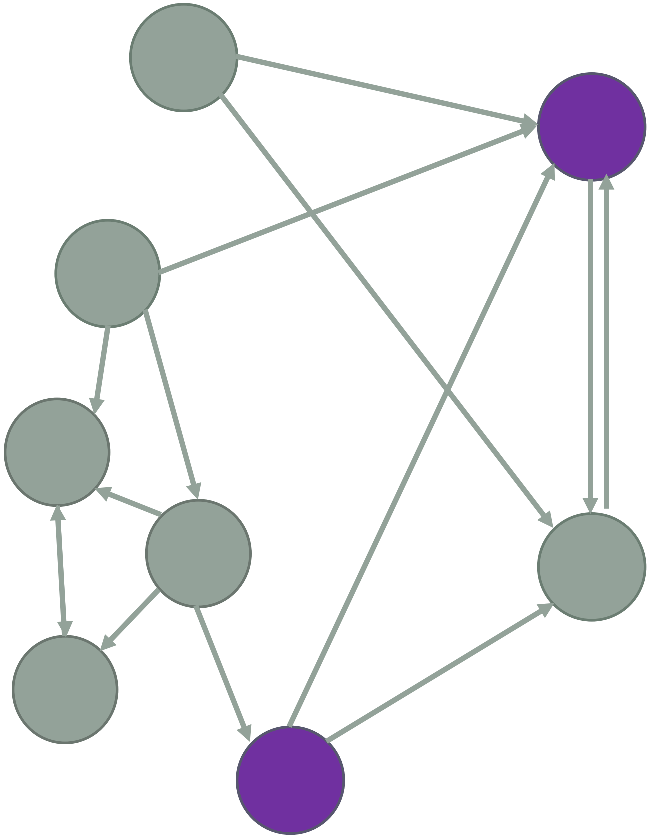
Find Fraud in HITS





CatchSync: Catching Synchronized Behavior in Large Directed Graphs
Meng Jiang, Peng Cui, Alex Beutel,
Christos Faloutsos, Shiqiang Yang
KDD, 2014

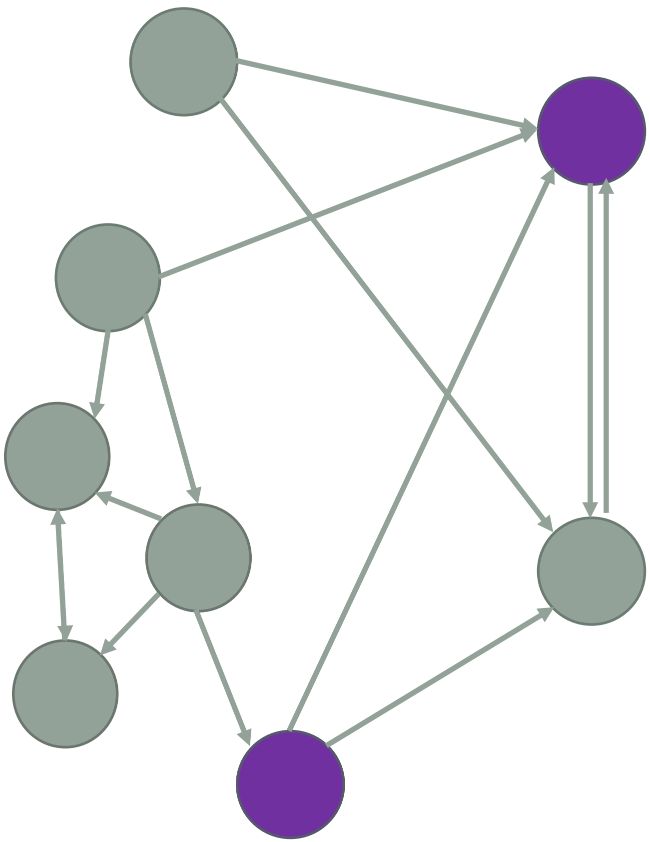
TrustRank



Given a small seed set
of trustworthy pages,
can we score how trustworthy
all pages are?

Combating Web Spam with TrustRank
Zoltán Gyöngyi, Hector Garcia-Molina, Jan Pedersen
VLDB 2004

TrustRank

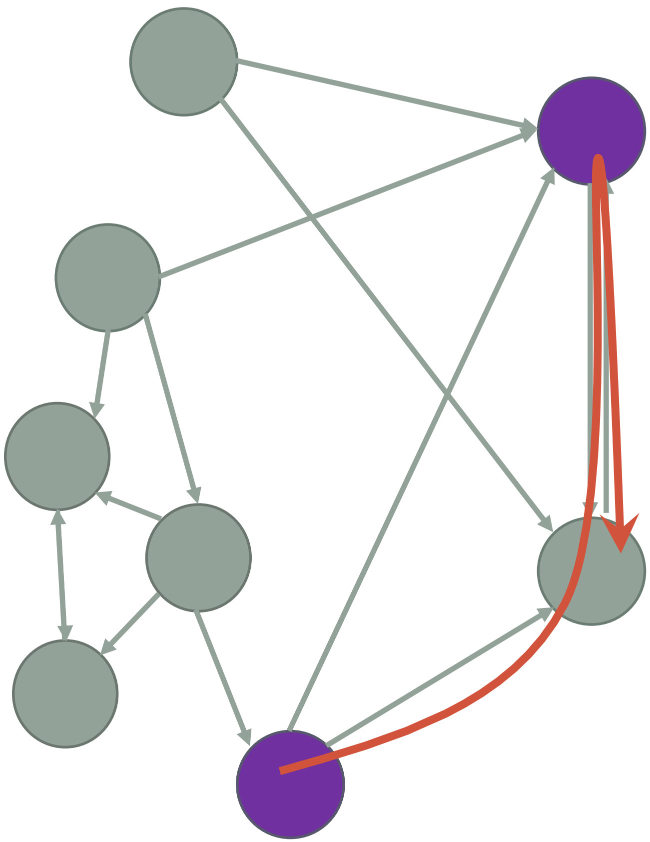


Given a small seed set
of trustworthy pages,
can we score how trustworthy
all pages are?

In PageRank random walk,
jump to seed trustworthy nodes

Combating Web Spam with TrustRank
Gyöngyi, Zoltán, Hector Garcia-Molina, Jan Pedersen
VLDB 2004

TrustRank

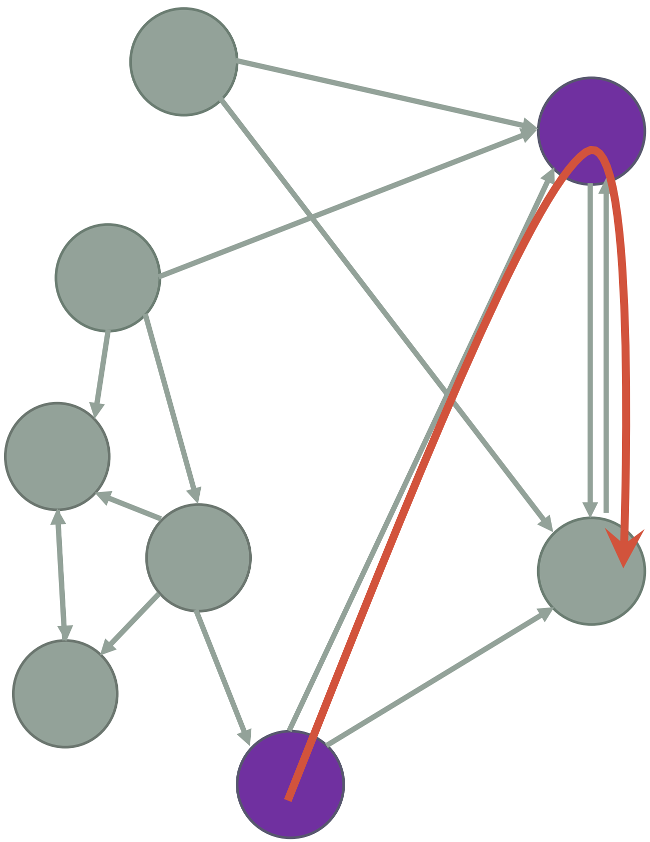


Given a small seed set of trustworthy pages, can we score how trustworthy all pages are?

In PageRank random walk, jump to seed trustworthy nodes

Combating Web Spam with TrustRank
Gyöngyi, Zoltán, Hector Garcia-Molina, Jan Pedersen
VLDB 2004

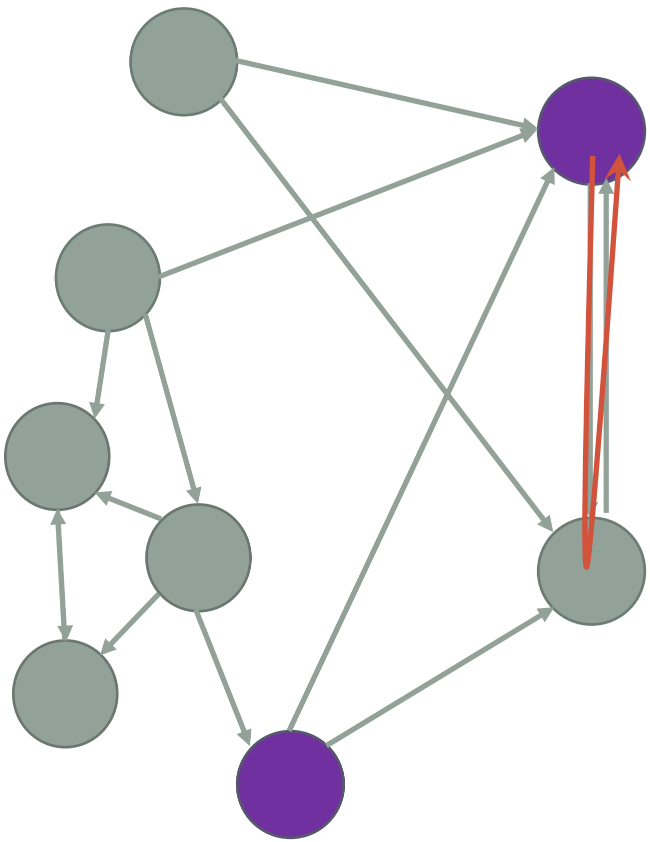
TrustRank



Given a small seed set
of trustworthy pages,
can we score how trustworthy
all pages are?

In PageRank random walk,
jump to seed trustworthy nodes

TrustRank

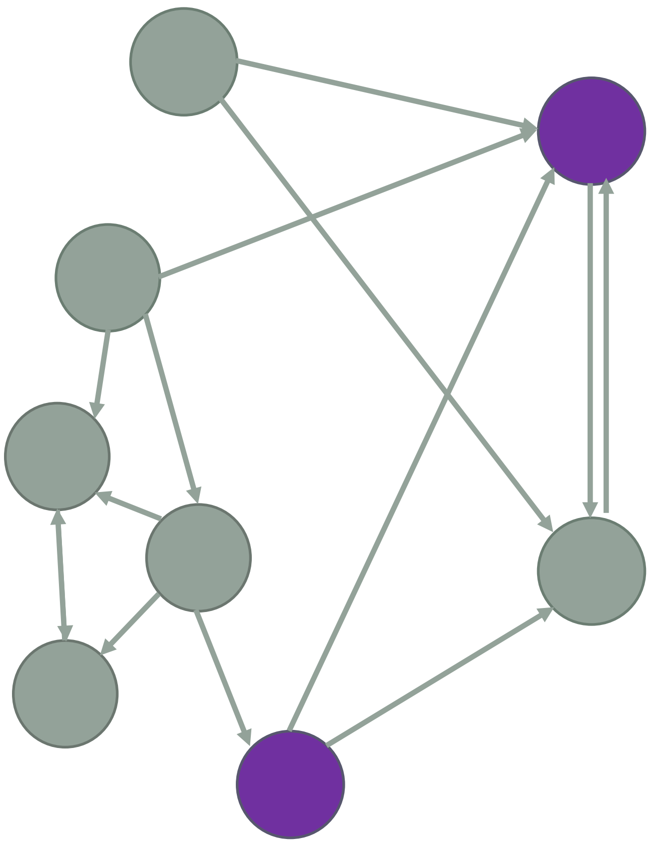


Given a small seed set
of trustworthy pages,
can we score how trustworthy
all pages are?

In PageRank random walk,
jump to seed trustworthy nodes

Combating Web Spam with TrustRank
Zoltán Gyöngyi, Hector Garcia-Molina, Jan Pedersen
VLDB 2004

TrustRank



Given a small seed set
of trustworthy pages,
can we score how trustworthy
all pages are?

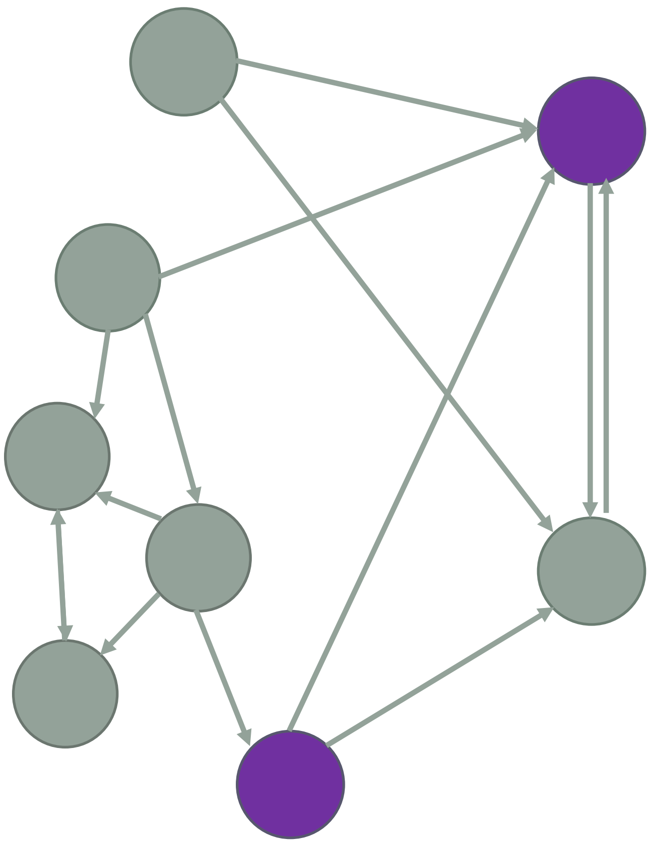
In PageRank random walk,
jump to seed trustworthy nodes

Seed Trustworthy vector $\vec{t}$

$$\overrightarrow{\mathrm{TrustRank}} = cA'\overrightarrow{\mathrm{TrustRank}} + \frac{1-c}{|\vec{t}|}\vec{t}$$

Combating Web Spam with TrustRank
Gyöngyi, Zoltán, Hector Garcia-Molina, Jan Pedersen
VLDB 2004

TrustRank



Given a small seed set of trustworthy pages, can we score how trustworthy all pages are?

In PageRank random walk, jump to seed trustworthy nodes

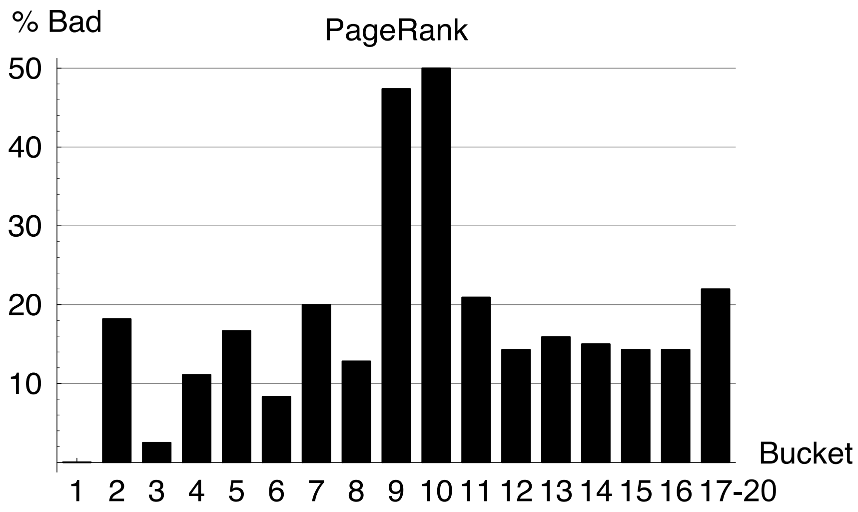
Seed Trustworthy vector $\vec{t}$

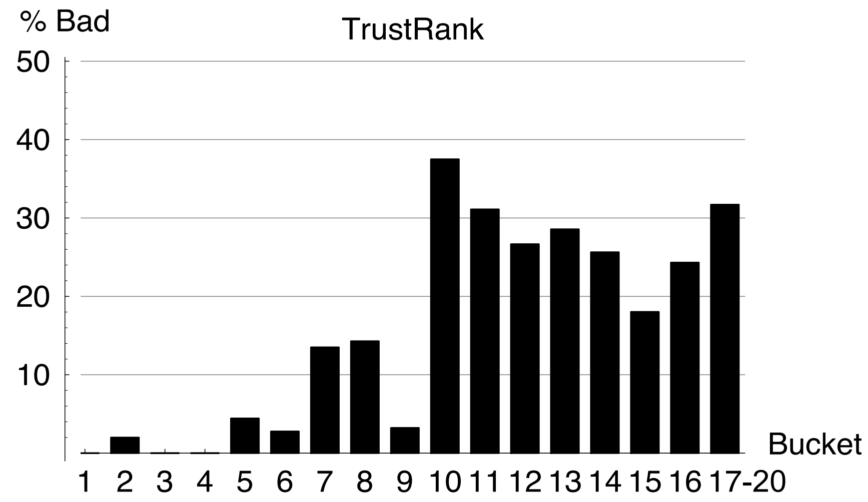
$$\overrightarrow{\text{TrustRank}} = cA'\overrightarrow{\text{TrustRank}} + \frac{1-c}{|\vec{t}|}\vec{t}$$

"Guilt" (Trust) by Association

Combating Web Spam with TrustRank
Zoltán Gyöngyi, Hector Garcia-Molina, Jan Pedersen
VLDB 2004

TrustRank

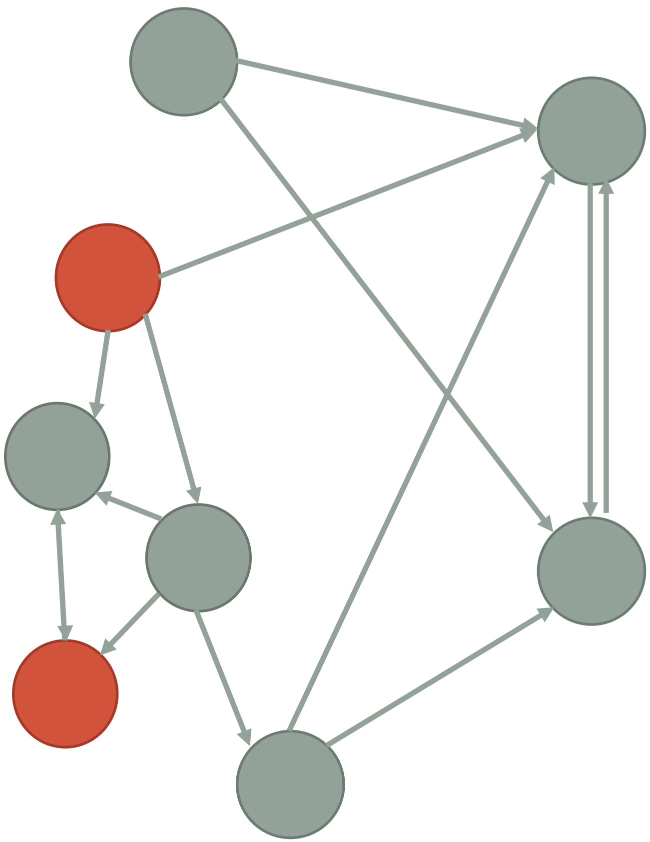




Bad pages by “Bucket”

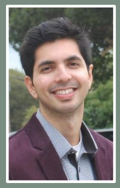
Combating Web Spam with TrustRank
Zoltán Gyöngyi, Hector Garcia-Molina, Jan Pedersen
VLDB 2004

Distrust Rank

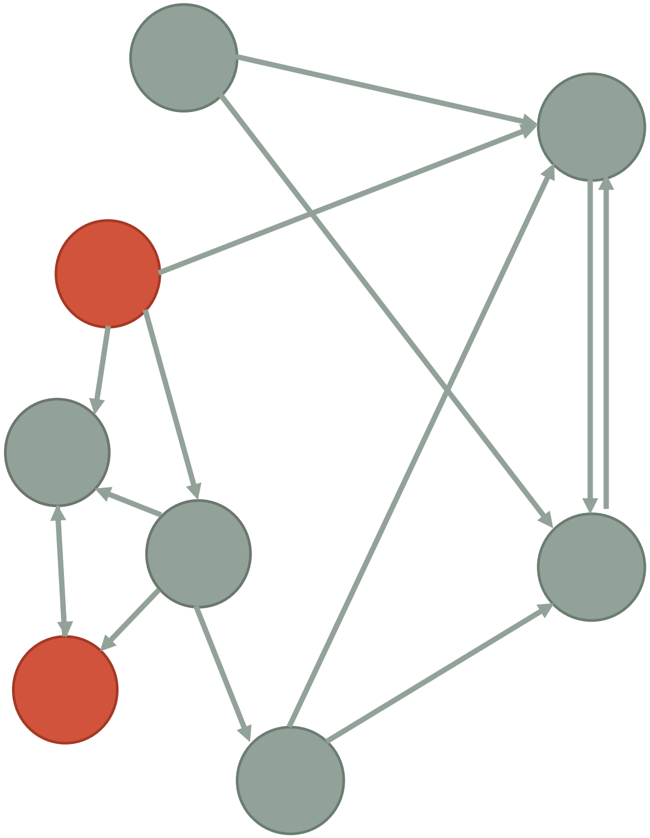


Given a small seed set
of spam pages,
can we find other spammy pages?

Propagating Trust and Distrust to Demote Web Spam
Baoning Wu, Vinay Goel, Brian D. Davison
WWW 2006

Distrust Rank

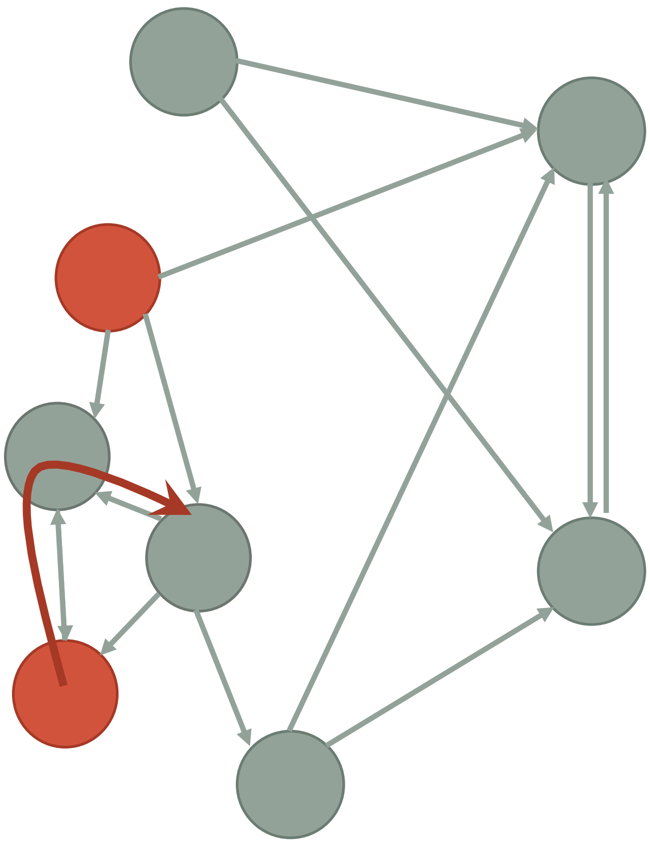


Given a small seed set
of spam pages,
can we find other spammy pages?

Spam pages are pointed to by
spam pages

Propagating Trust and Distrust to Demote Web Spam
Baoning Wu, Vinay Goel, Brian D. Davison
WWW 2006

Distrust Rank



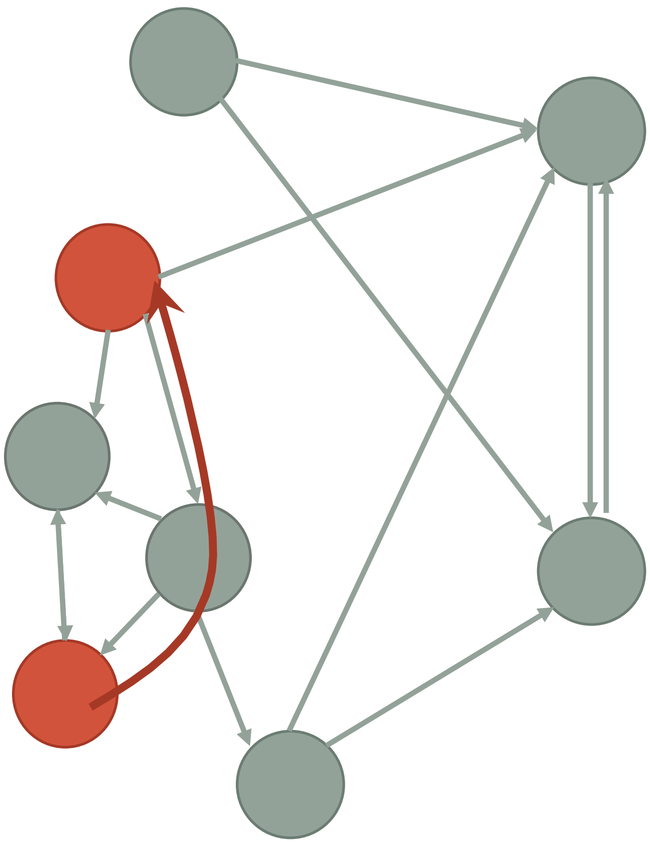
Given a small seed set
of spam pages,
can we find other spammy pages?

Spam pages are pointed to by
spam pages

Random walk backward in graph

Propagating Trust and Distrust to Demote Web Spam
Baoning Wu, Vinay Goel, Brian D. Davison
WWW 2006

Distrust Rank



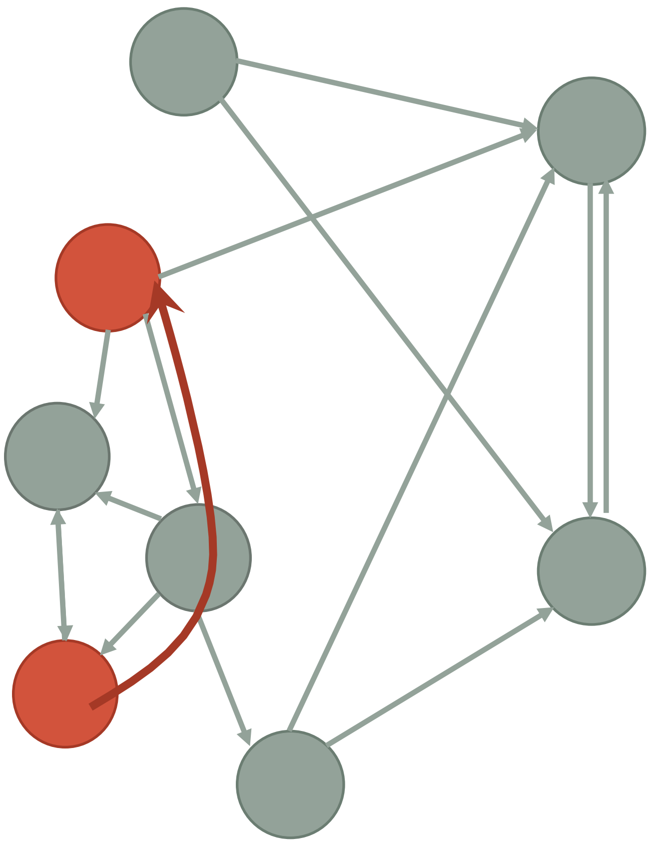
Given a small seed set of spam pages, can we find other spammy pages?

Spam pages are pointed to by spam pages

Random walk backward in graph

Propagating Trust and Distrust to Demote Web Spam
Baoning Wu, Vinay Goel, Brian D. Davison
WWW 2006

Distrust Rank



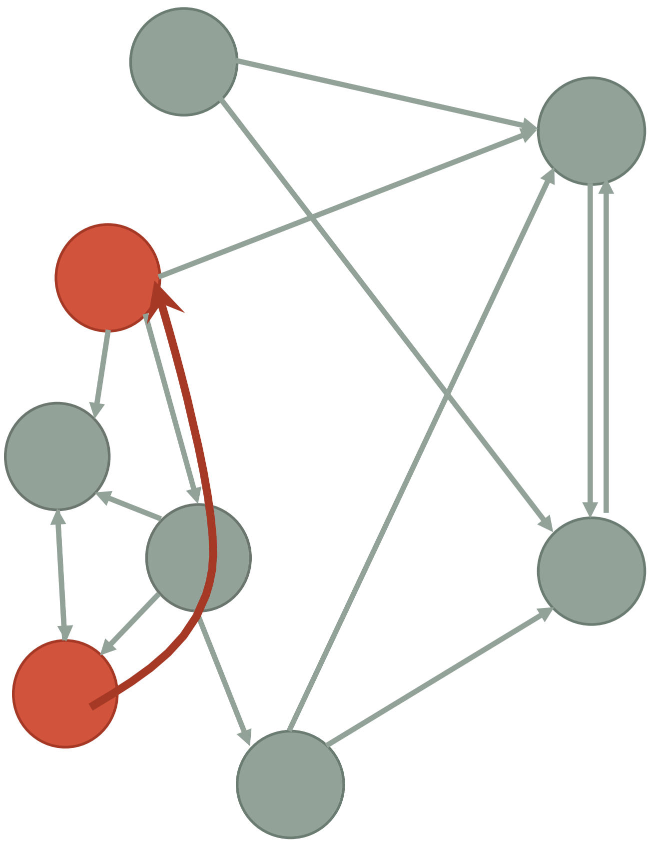
Random walk backward in graph

Reverse Normalized Adjacency Matrix *M*

$$M_{p,q} \begin{cases} \dfrac{1}{\mathrm{InDegree}(q)} & \text{if } (p,q) \in E \\ 0 & \text{if } (p,q) \notin E \end{cases}$$

Propagating Trust and Distrust to Demote Web Spam
Baoning Wu, Vinay Goel, Brian D. Davison
WWW 2006

Distrust Rank



Random walk backward in graph

Reverse Normalized Adjacency Matrix *M*

$$M_{p,q} \begin{cases} \dfrac{1}{\mathrm{InDegree}(q)} & \text{if } (p,q) \in E \\ 0 & \text{if } (p,q) \notin E \end{cases}$$

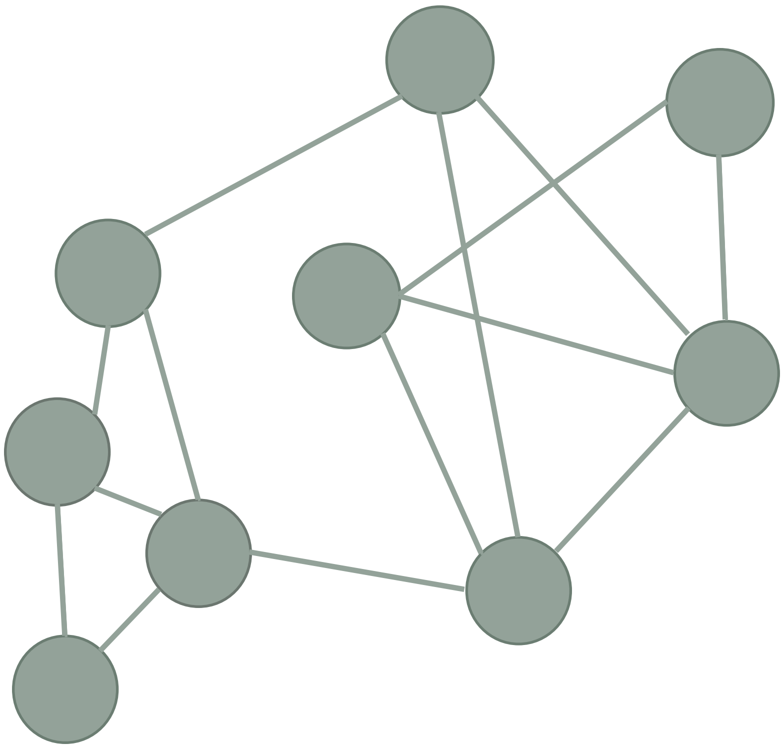
Seed spam pages in vector $\vec{s}$

$$\overrightarrow{\mathrm{DistrustRank}} = cM\overrightarrow{\mathrm{DistrustRank}} + \frac{1-c}{|\vec{s}|}\vec{s}$$

Propagating Trust and Distrust to Demote Web Spam
Baoning Wu, Vinay Goel, Brian D. Davison
WWW 2006

SibylRank

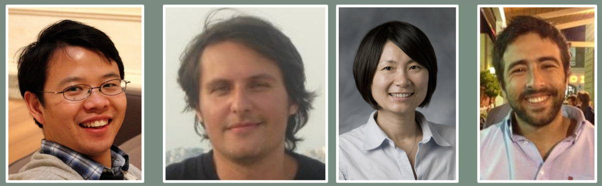
Given a small seed set of normal *social network accounts*, can we find fake accounts?



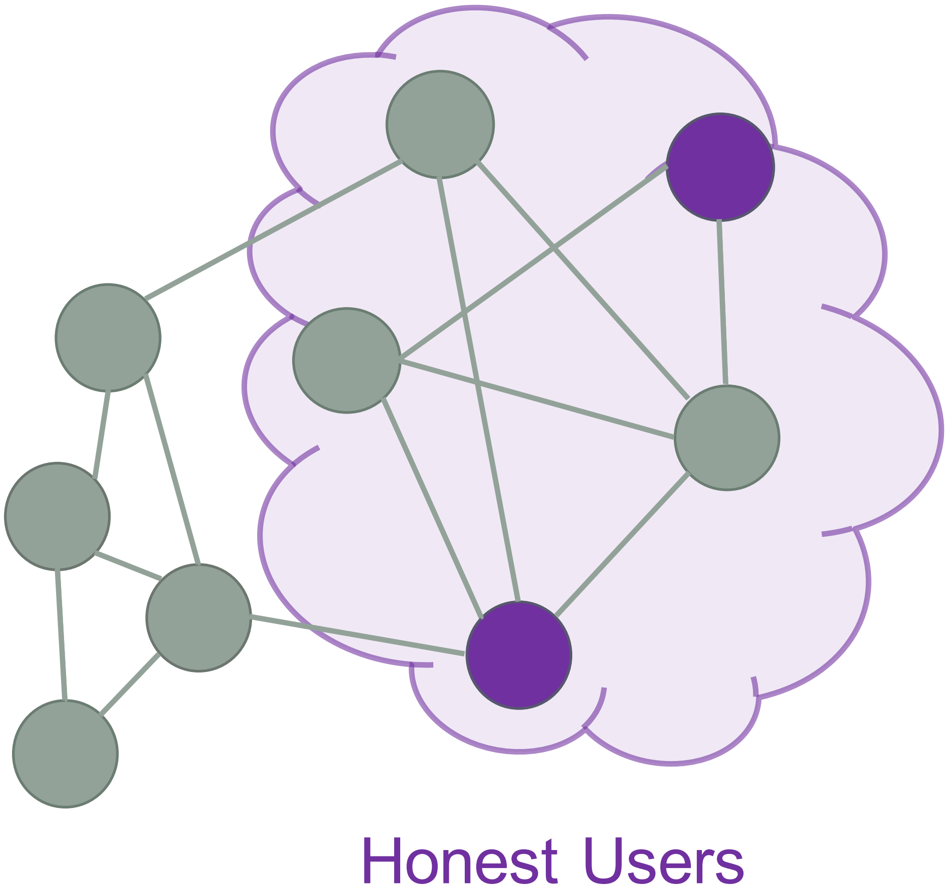
Aiding the Detection of Fake Accounts in Large Scale Social Online Services

Qiang Cao, Michael Sirivianos,
Xiaowei Yang, Tiago Pregueiro

NSDI 2012



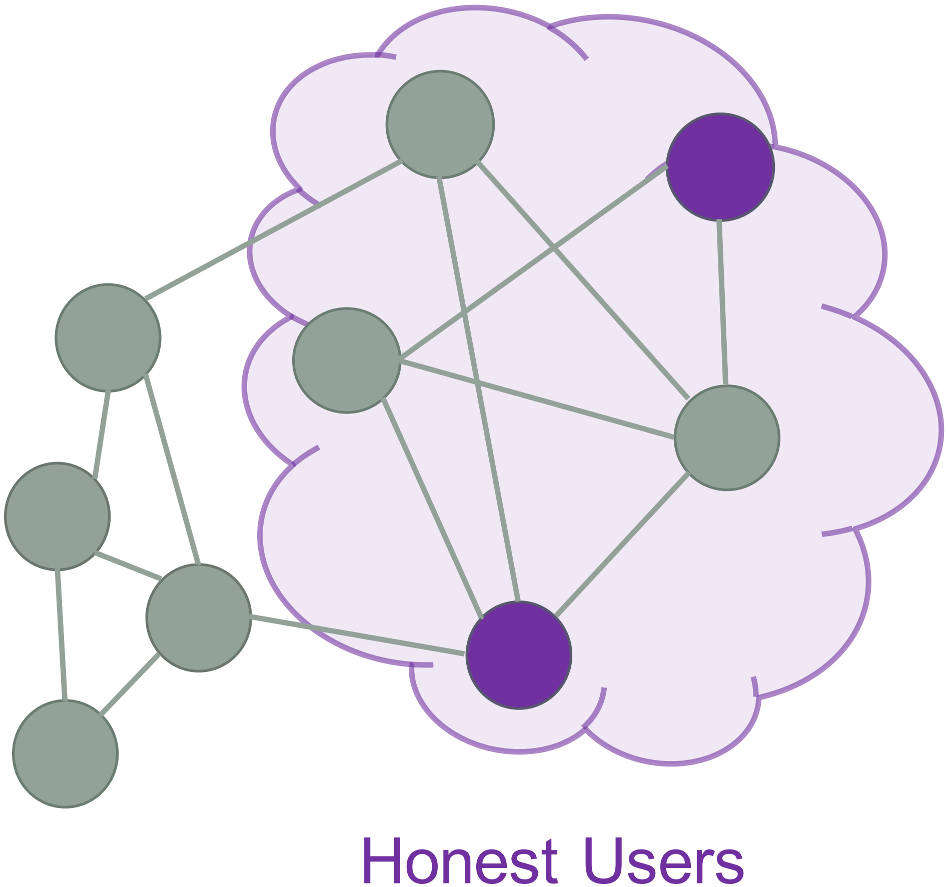
SibylRank



Given a small seed set of normal *social network accounts*, can we find fake accounts?

Aiding the Detection of Fake Accounts in Large Scale Social Online Services
Qiang Cao, Michael Sirivianos,
Xiaowei Yang, Tiago Pregueiro
NSDI 2012

SibylRank

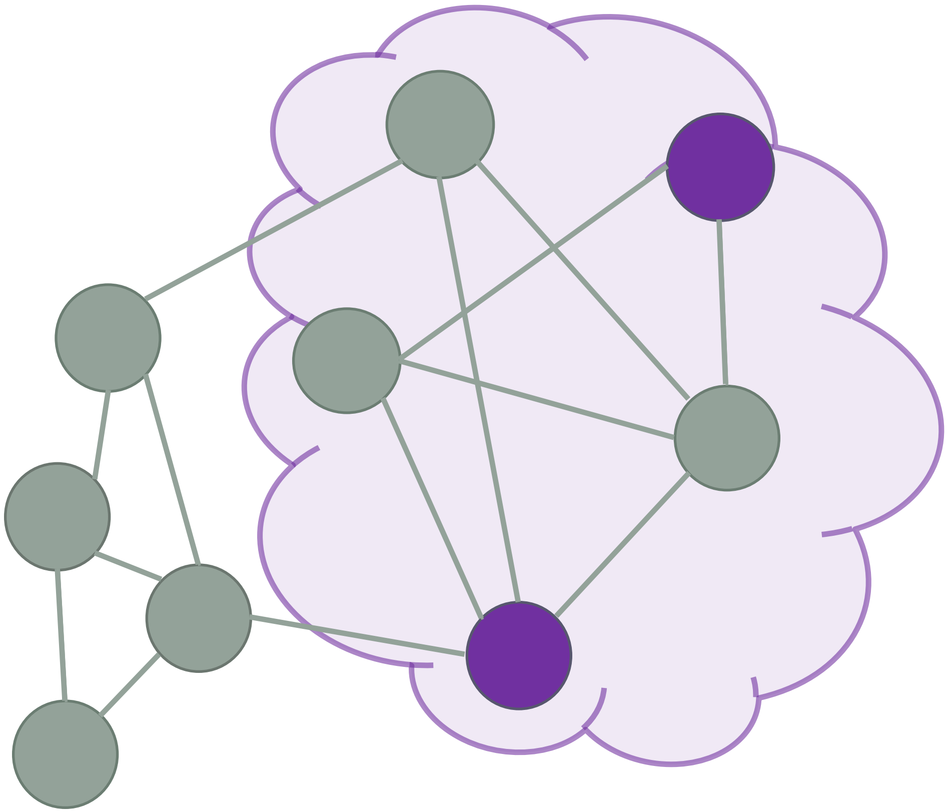


Given a small seed set of normal *social network accounts*, can we find fake accounts?

Observation: It is difficult for fake users to connect to honest users.

Aiding the Detection of Fake Accounts in Large Scale Social Online Services
Qiang Cao, Michael Sirivianos,
Xiaowei Yang, Tiago Pregueiro
NSDI 2012

SibylRank



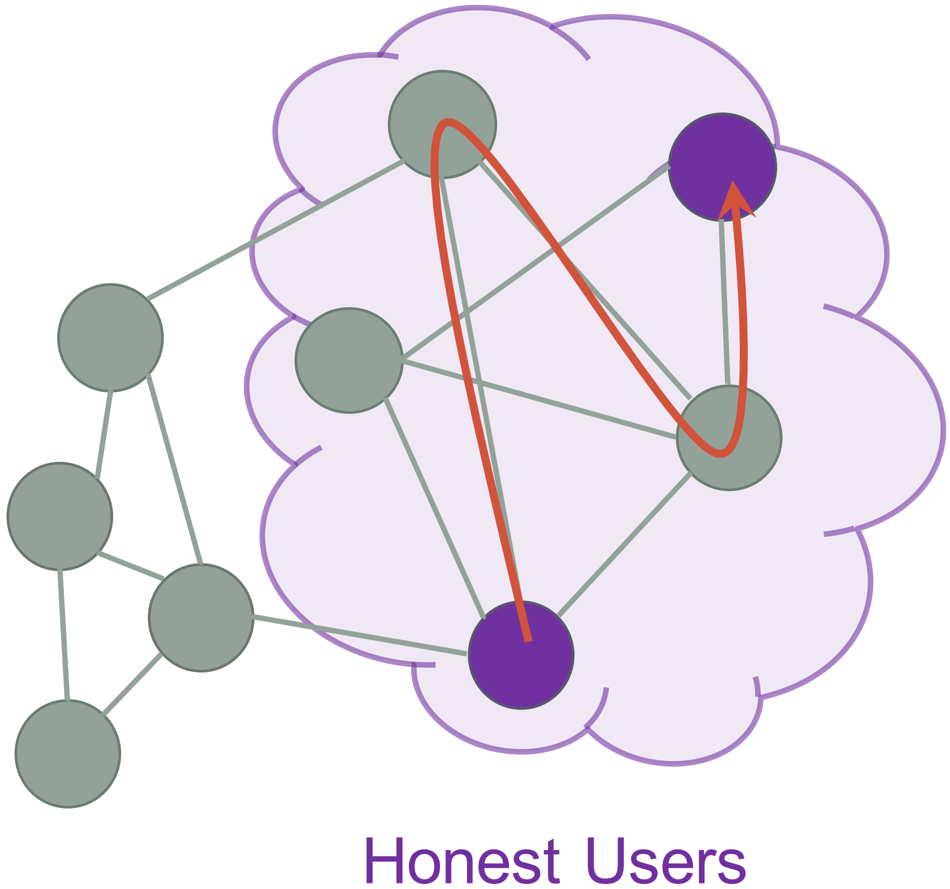
Given a small seed set of normal *social network accounts*, can we find fake accounts?

Observation: It is difficult for fake users to connect to honest users.

Solution: Use Early-Termination Random Walks

Aiding the Detection of Fake Accounts in Large Scale Social Online Services
Qiang Cao, Michael Sirivianos,
Xiaowei Yang, Tiago Pregueiro
NSDI 2012

SibylRank



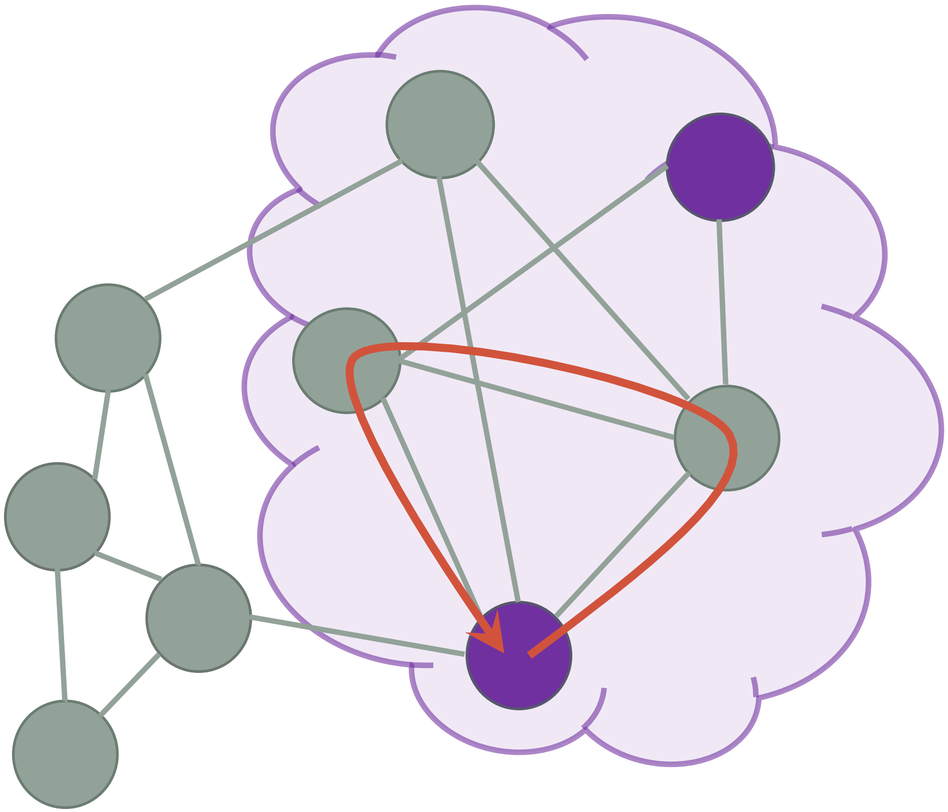
Given a small seed set of normal *social network accounts*, can we find fake accounts?

Observation: It is difficult for fake users to connect to honest users.

Solution: Use Early-Termination Random Walks

Aiding the Detection of Fake Accounts in Large Scale Social Online Services
Qiang Cao, Michael Sirivianos,
Xiaowei Yang, Tiago Pregueiro
NSDI 2012

SibylRank



Honest Users

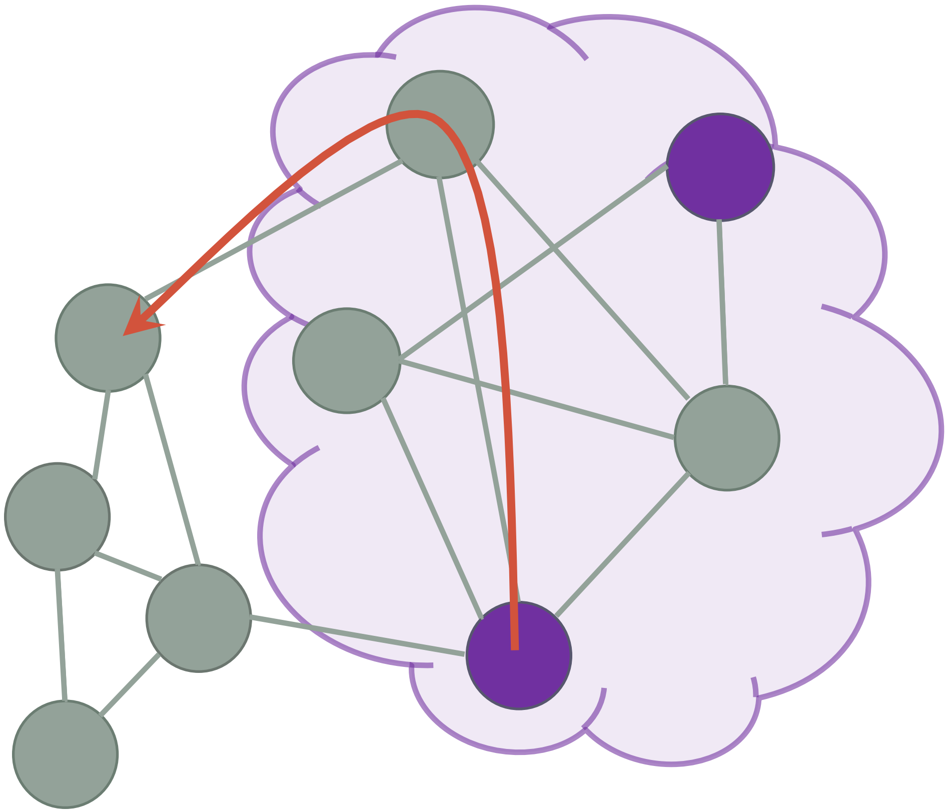
Given a small seed set of normal *social network accounts*, can we find fake accounts?

Observation: It is difficult for fake users to connect to honest users.

Solution: Use Early-Termination Random Walks

Aiding the Detection of Fake Accounts in Large Scale Social Online Services
Qiang Cao, Michael Sirivianos,
Xiaowei Yang, Tiago Pregueiro
NSDI 2012

SibylRank



Honest Users

Given a small seed set of normal *social network accounts*, can we find fake accounts?

Observation: It is difficult for fake users to connect to honest users.

Solution: Use Early-Termination Random Walks

Rarely make it to sybil accounts

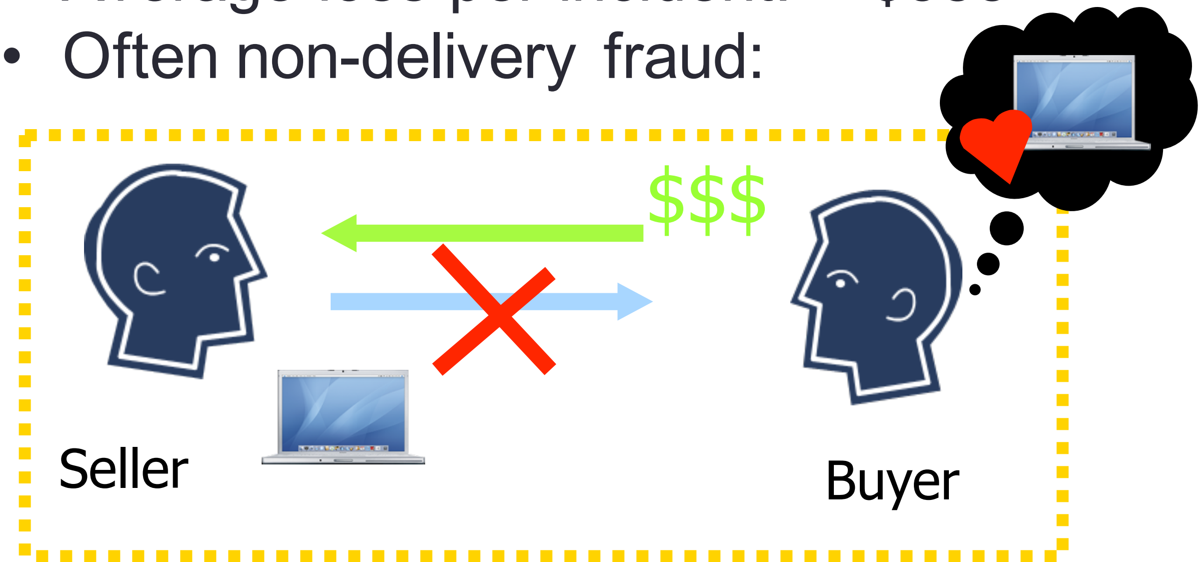
Aiding the Detection of Fake Accounts in Large Scale Social Online Services
Qiang Cao, Michael Sirivianos,
Xiaowei Yang, Tiago Pregueiro
NSDI 2012

Practitioner's Guide

Method	Graph Type	Node Attributes	Edge Attributes	Seed Labels
CatchSync	Directed			
TrustRank	Directed			✓
Distrust Rank	Directed			✓
SibylRank	Directed			✓
NetProbe	Directed			
FraudEagle	Bipartite		✓	
SpEagle	Tripartite	✓		

Fraud in Online Auctions

- Auction sites: Attractive target for fraud
- 63% of complaints to Federal Internet Crime Complaint Center in U.S. in 2006
- Average loss per incident: = **$385**
- Often non-delivery fraud:



NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Individual features, e.g. geography, are too easy to fake!

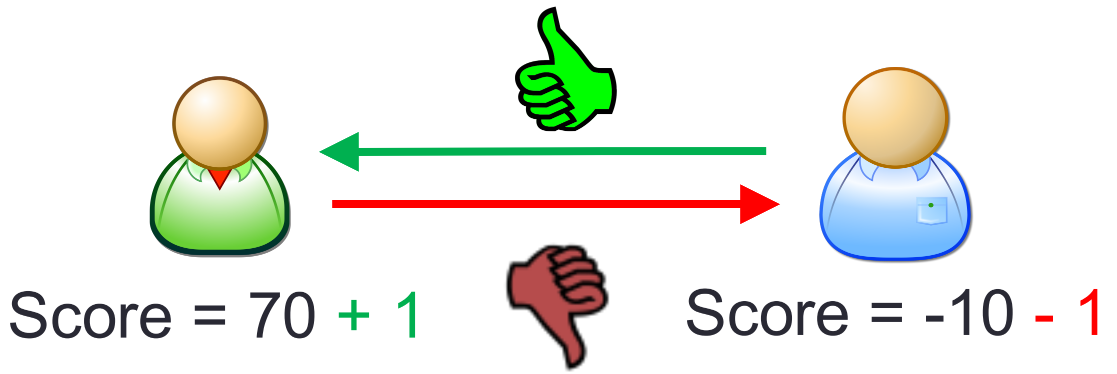
Given a graph of user interactions,
what does fraud look like and how can we catch it?

ebay

NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

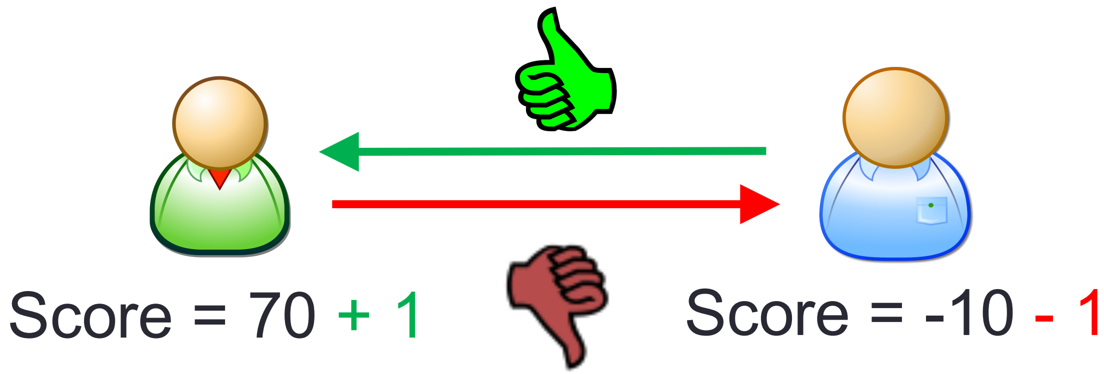
Each user gets a reputation score based on peer feedback:



NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Each user gets a reputation score based on peer feedback:



Fraudsters need to keep a high reputation score

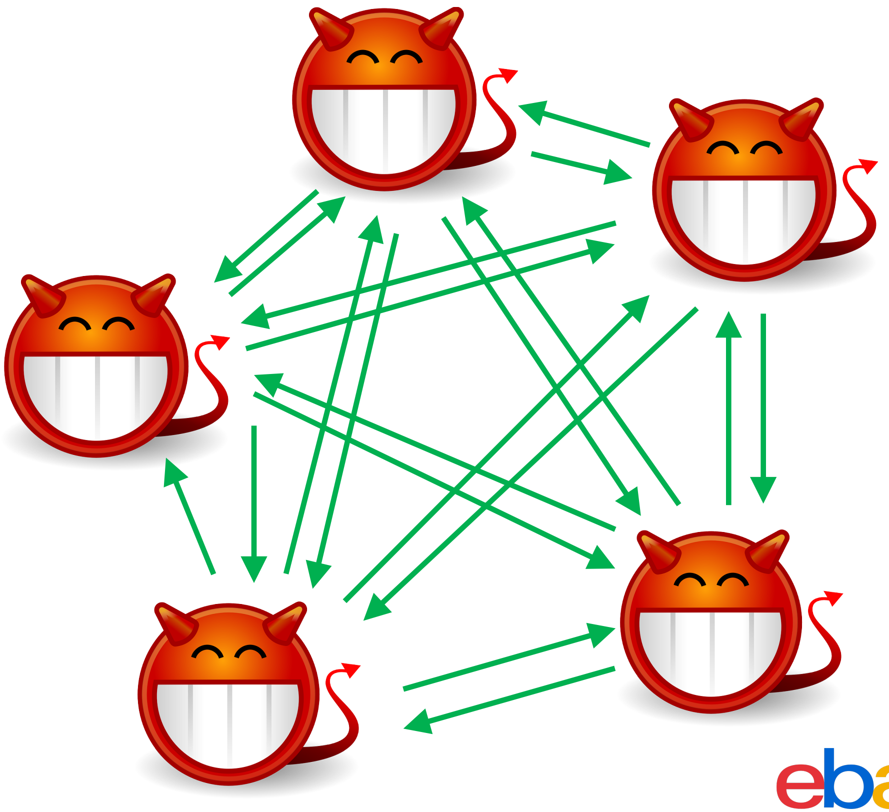
How do they game the system?

ebay

NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Do they all just give each other positive reviews?

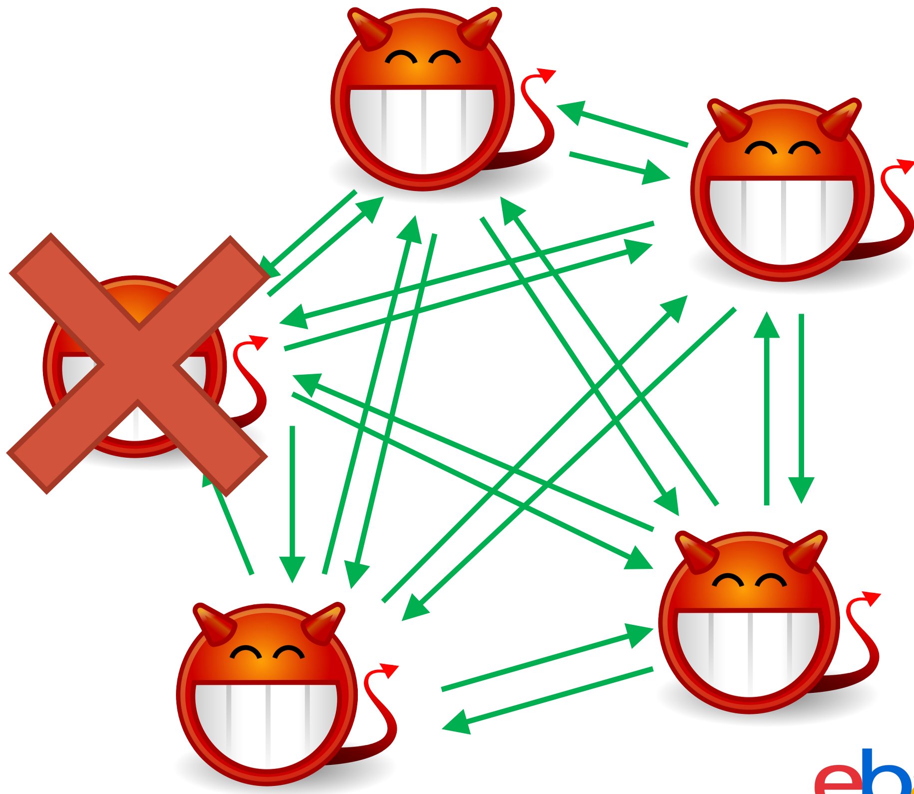


NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Do they all just give each other positive reviews?

No, because if one is caught they are all revealed.

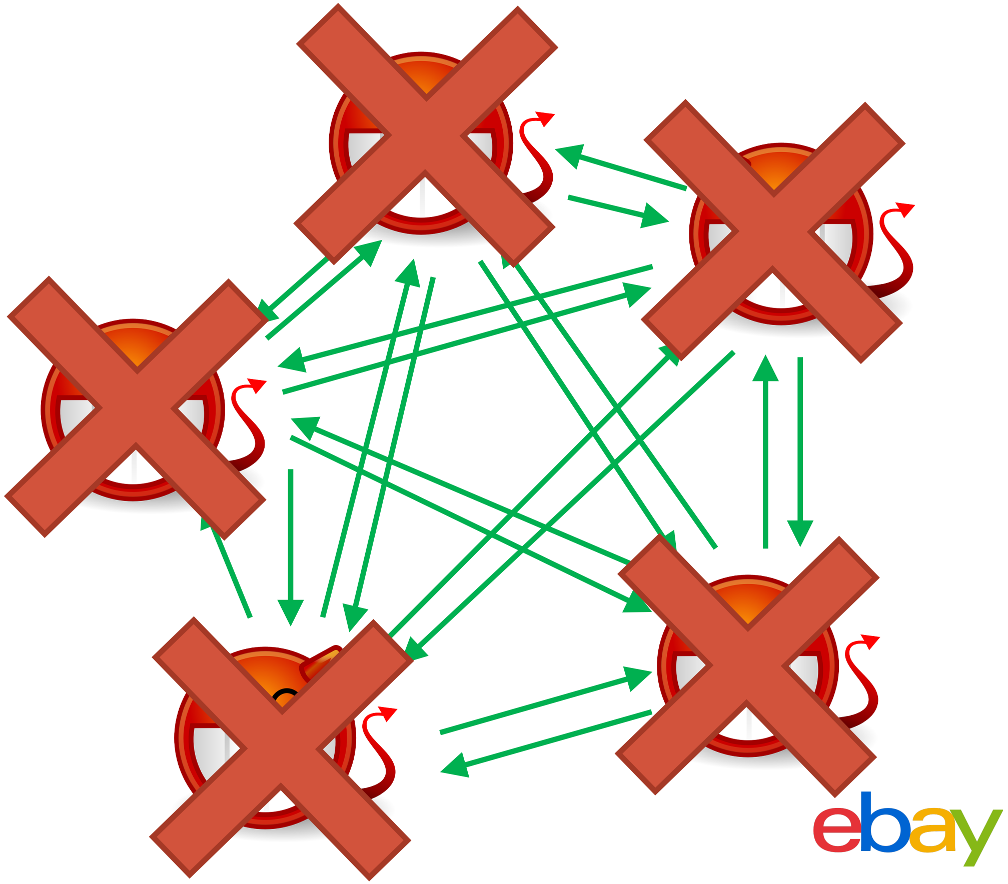


NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Do they all just give each other positive reviews?

No, because if one is caught they are all revealed.

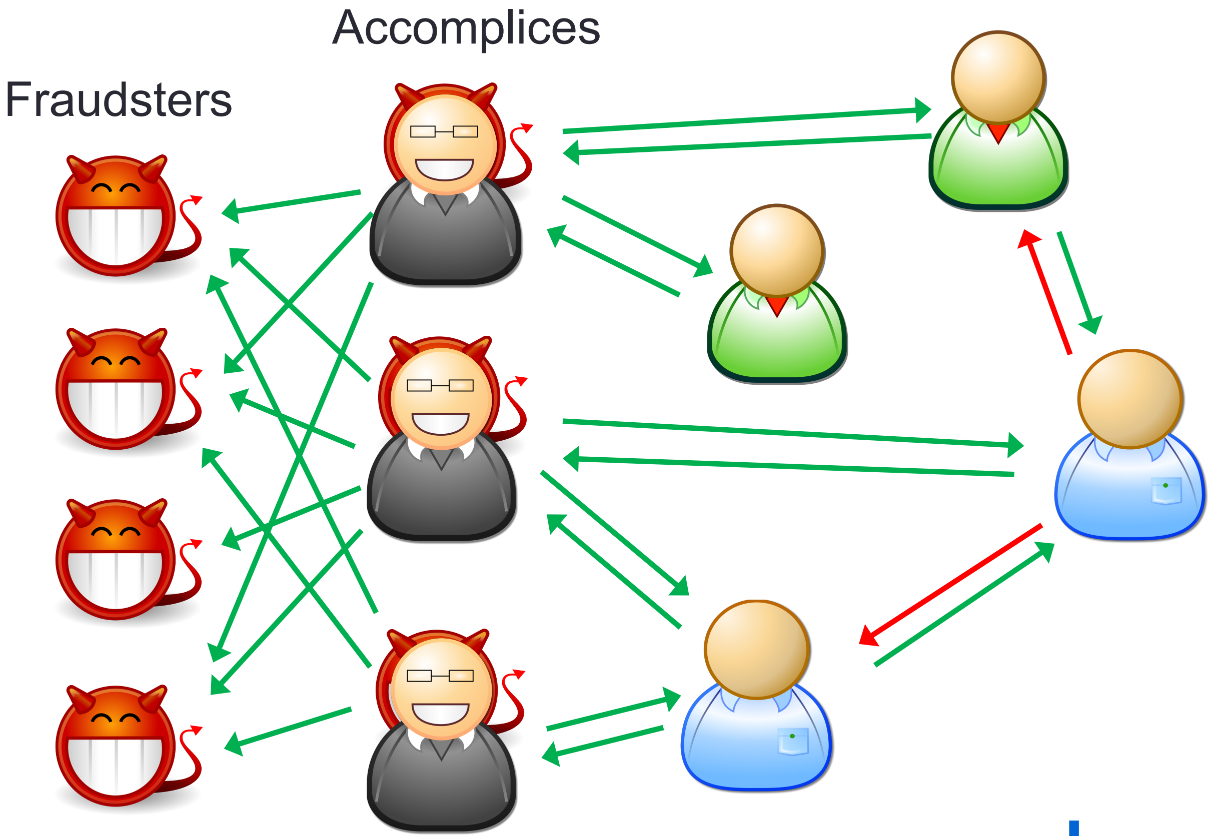


NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Fraudsters form near-bipartite core of 2 roles:

1. Accomplices:
   Trade with honest, looks normal
2. Fraudsters:
   Trade with accomplices
   Fraud with honest



NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Use Belief Propagation!

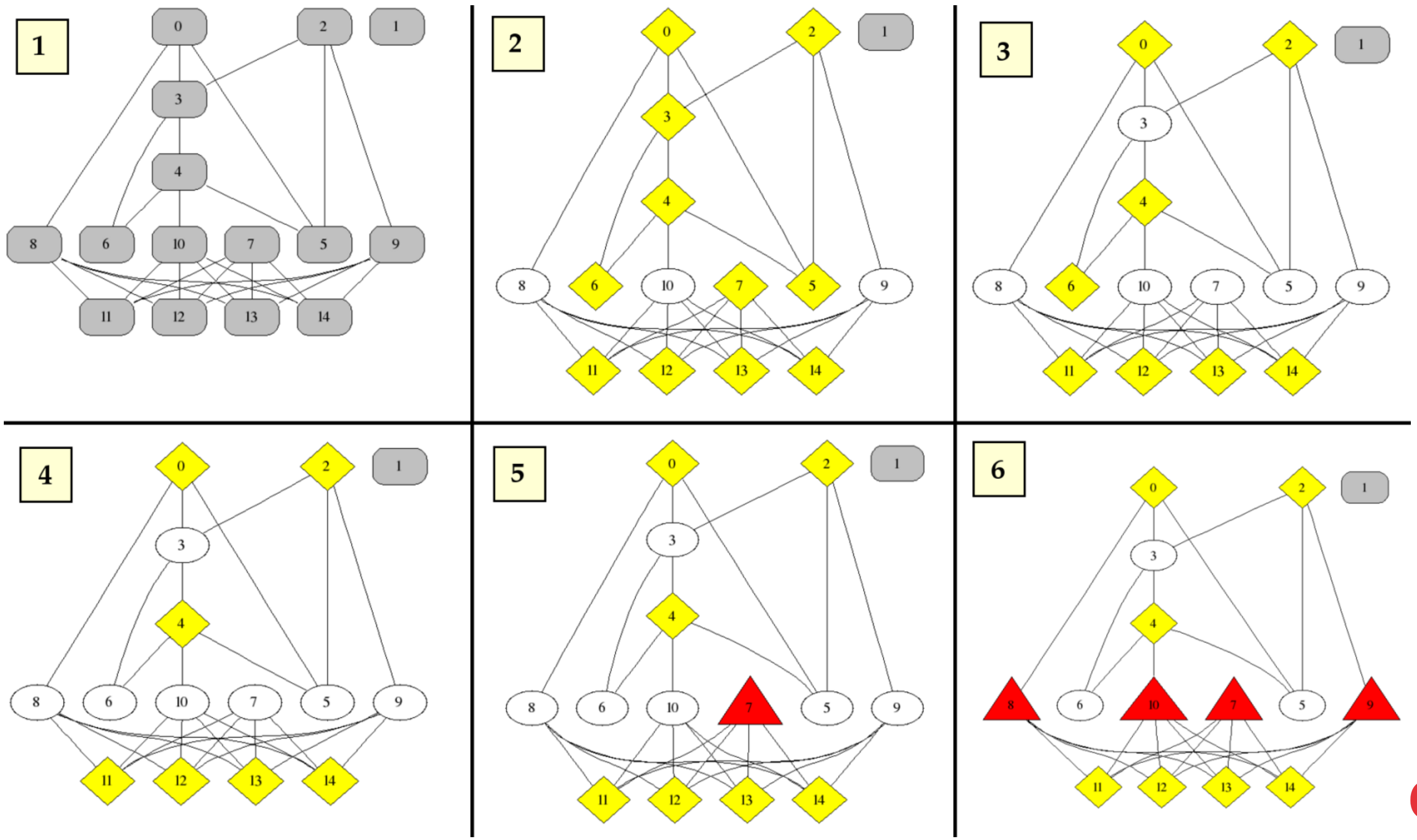
Node State

Neighbor State		Fraud	Accomplice	Honest
	Fraud	ϵ	$1-2\epsilon$	ϵ
	Accomplice	0.5	2ϵ	$0.5-2\epsilon$
	Honest	ϵ	$(1-\epsilon)/2$	$(1-\epsilon)/2$

Captures both homophily and heterophily

NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

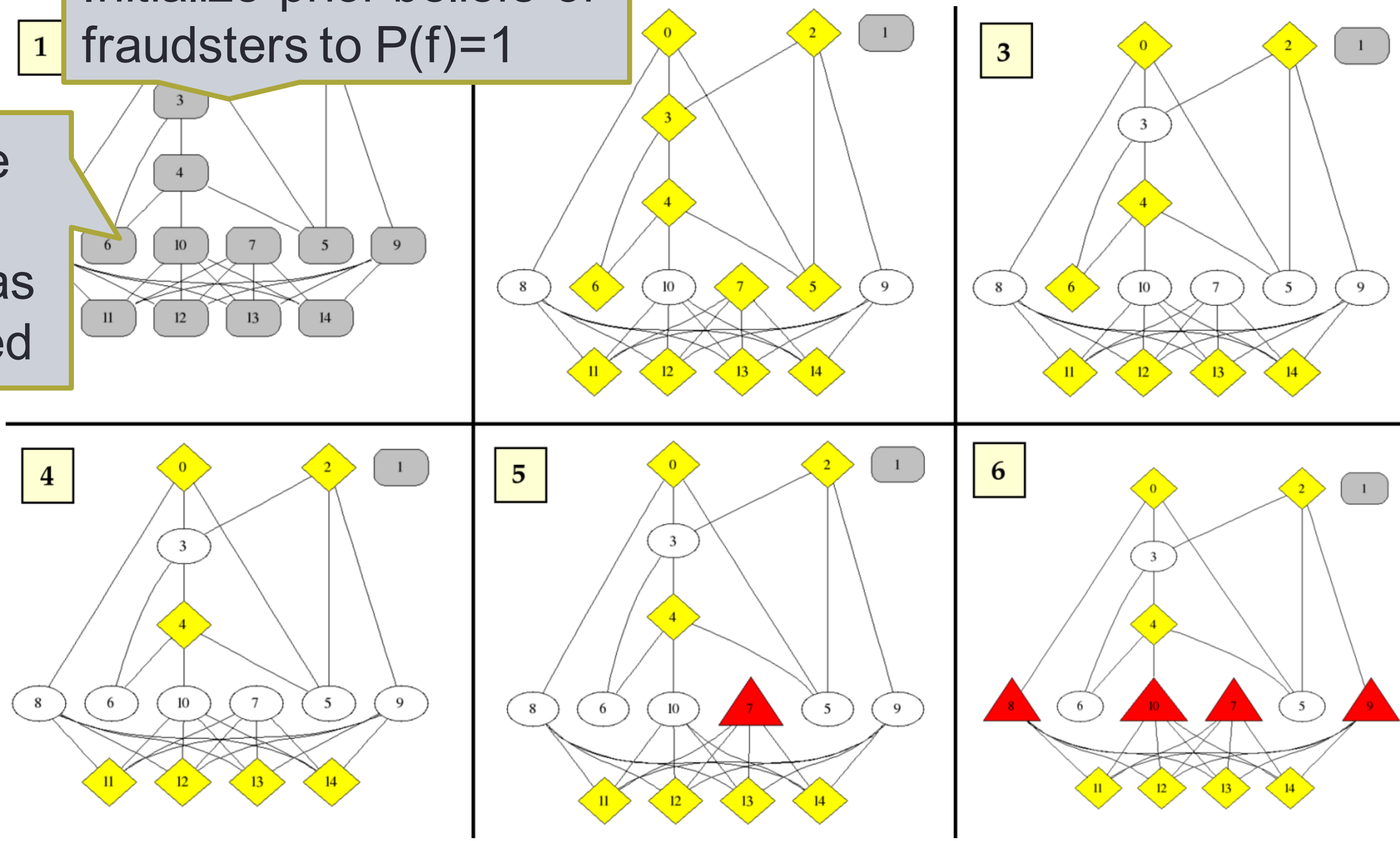


NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Initialize prior beliefs of fraudsters to P(f)=1

Initialize other nodes as unbiased



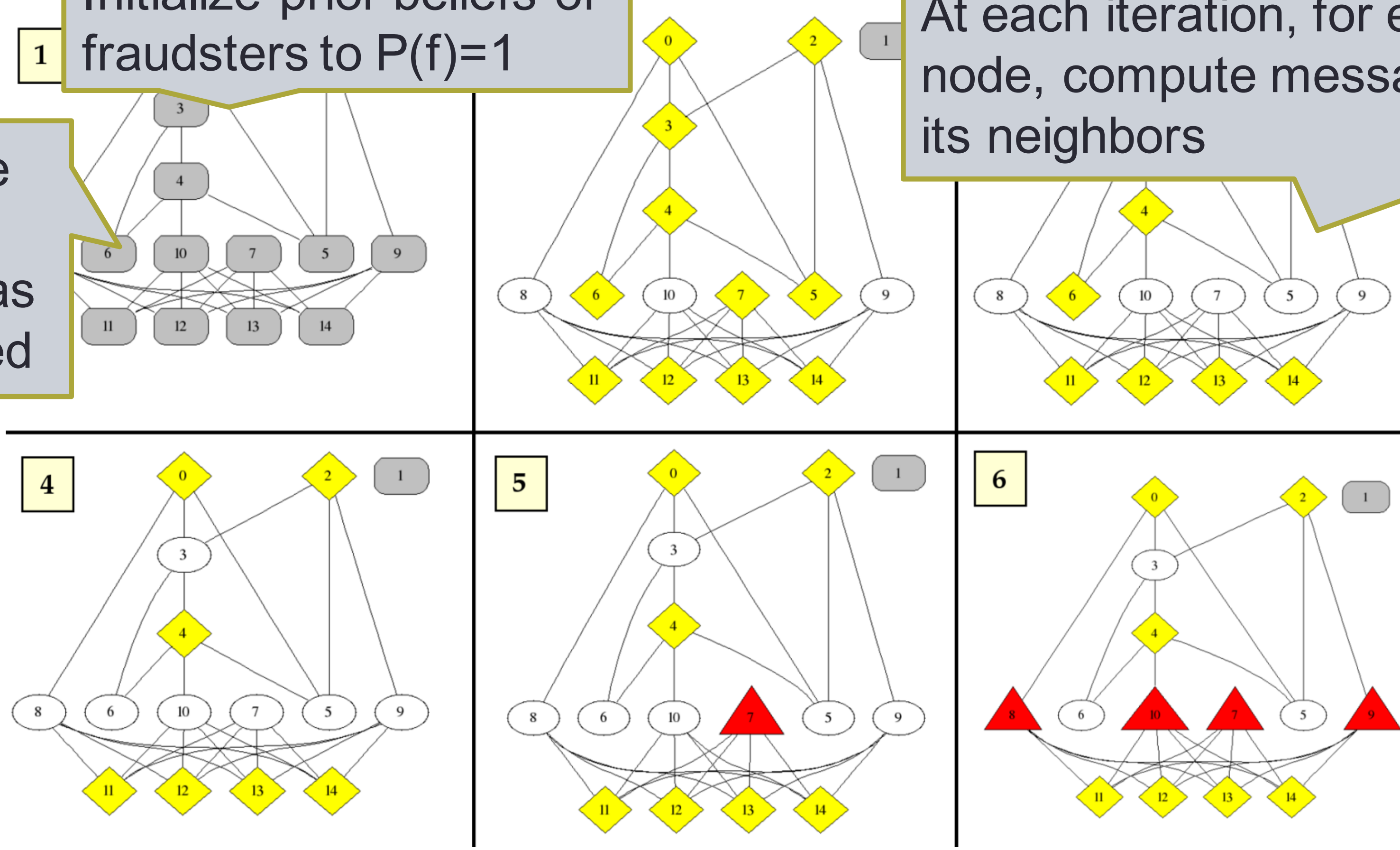
NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Initialize prior beliefs of fraudsters to P(f)=1

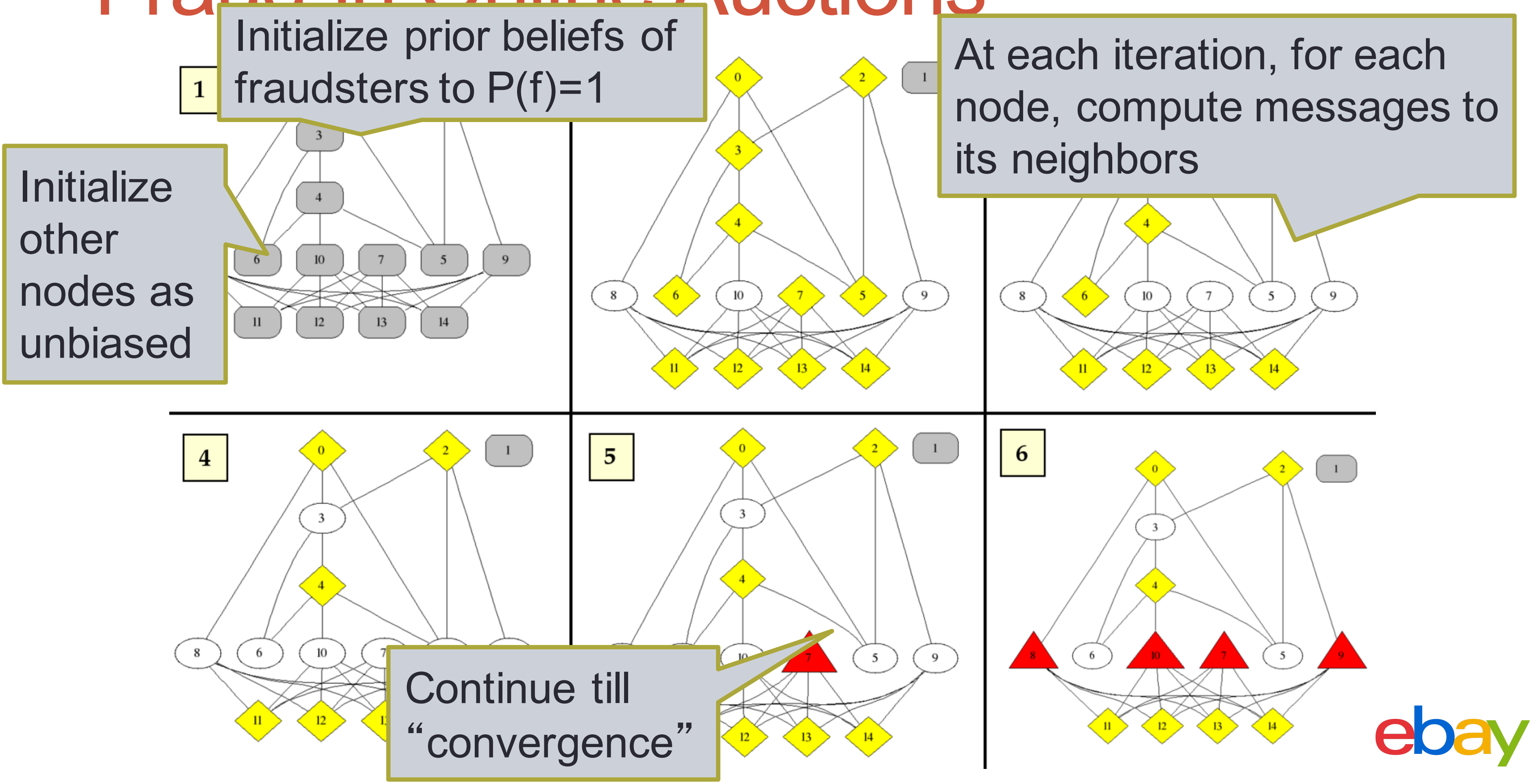
Initialize other nodes as unbiased

At each iteration, for each node, compute messages to its neighbors



NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions



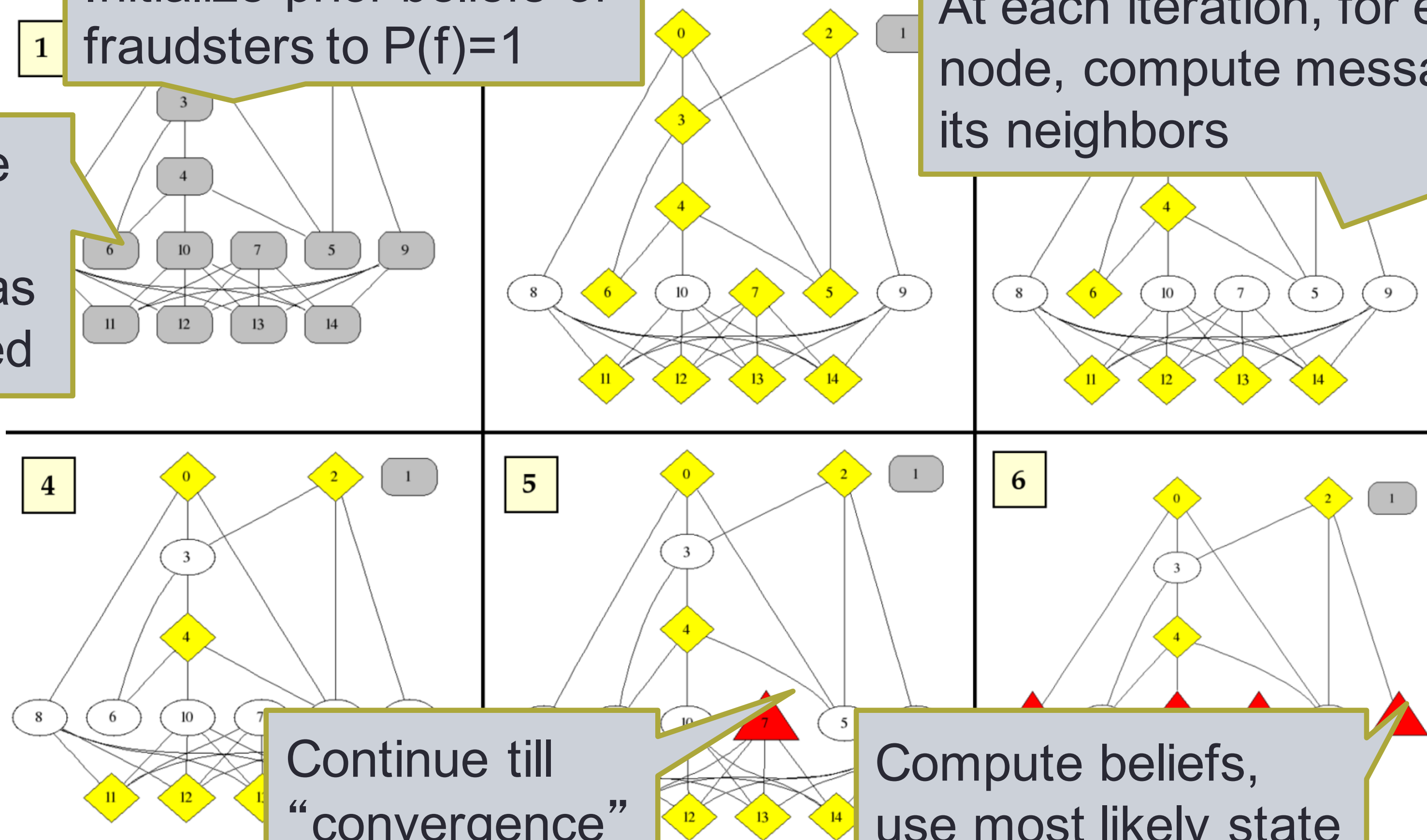
NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

Initialize prior beliefs of fraudsters to P(f)=1

At each iteration, for each node, compute messages to its neighbors

Initialize other nodes as unbiased

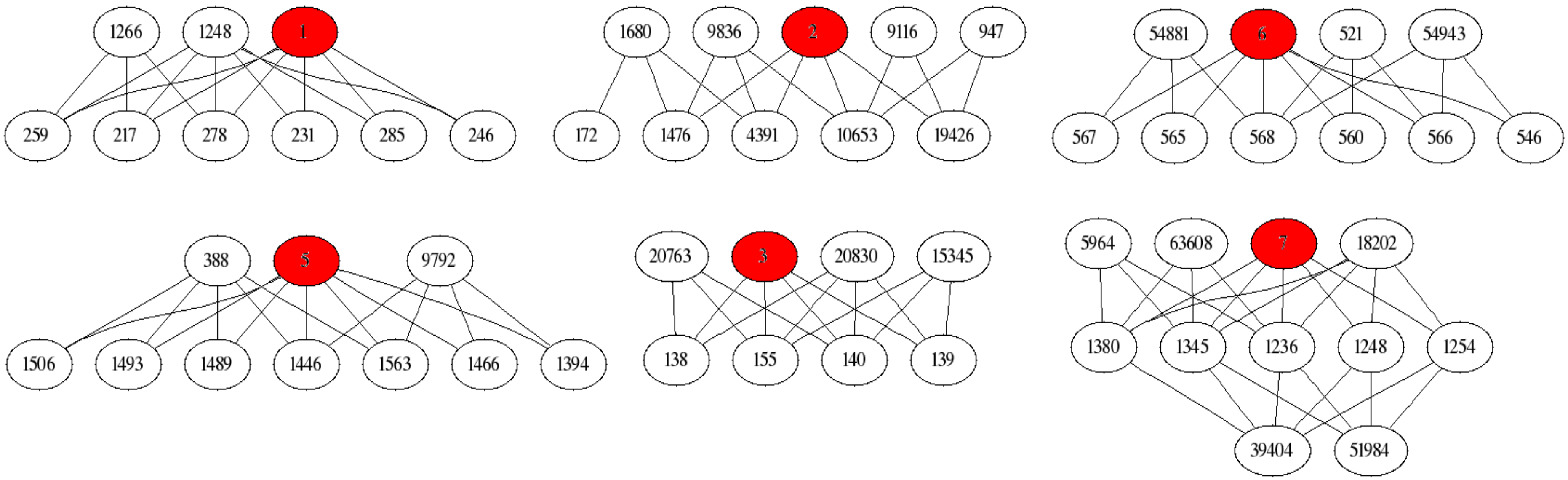


Continue till "convergence"

Compute beliefs, use most likely state

NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

Fraud in Online Auctions

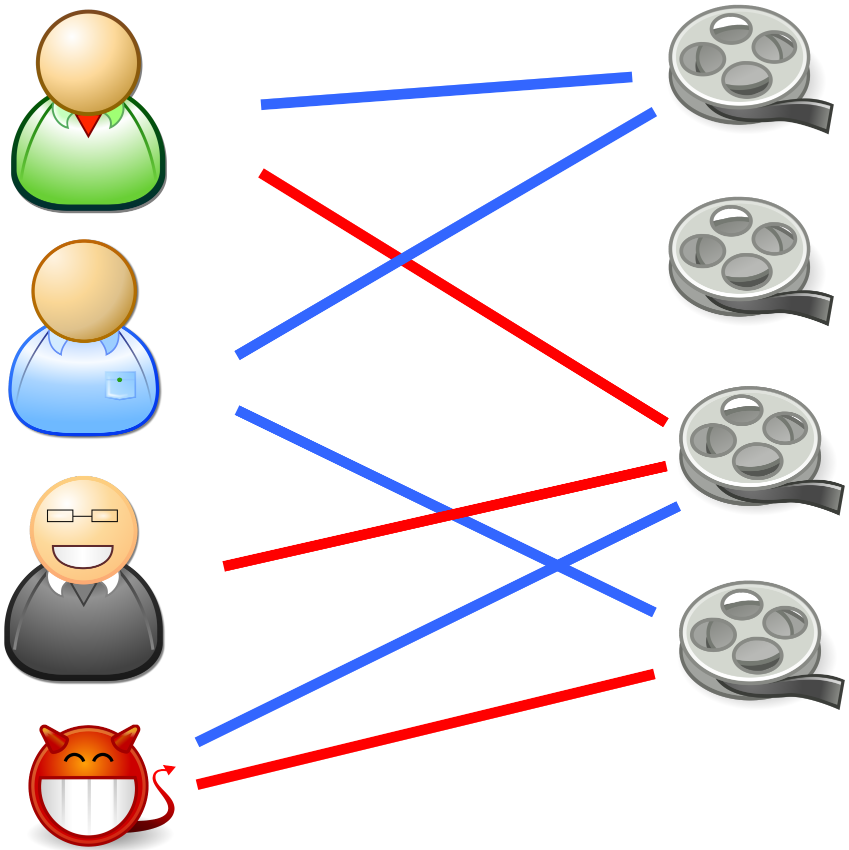


Subgraphs found on eBay
(Red nodes are confirmed fraudsters)

ebay

NetProbe: A Fast and Scalable System for Fraud Detection in Online Auction Networks
Shashank Pandit, Duen Horng Chau,
Samuel Wang, Christos Faloutsos
WWW 2007

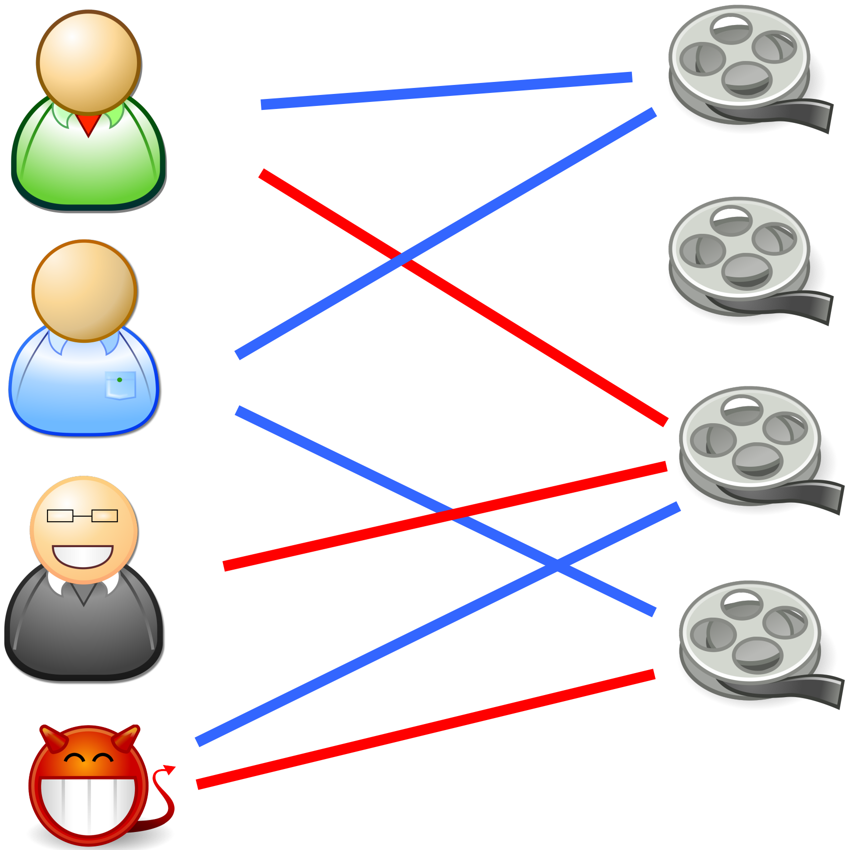
Finding fraudulent reviews



Idea: Fraudsters give
bad reviews to good products
&
good reviews to bad products

Opinion Fraud Detection in Online Reviews by Network Effects
Leman Akoglu, Rishi Chandy, Christos Faloutsos
ICWSM 2013

Finding fraudulent reviews



Idea: Fraudsters give
bad reviews to good products
&
good reviews to bad products

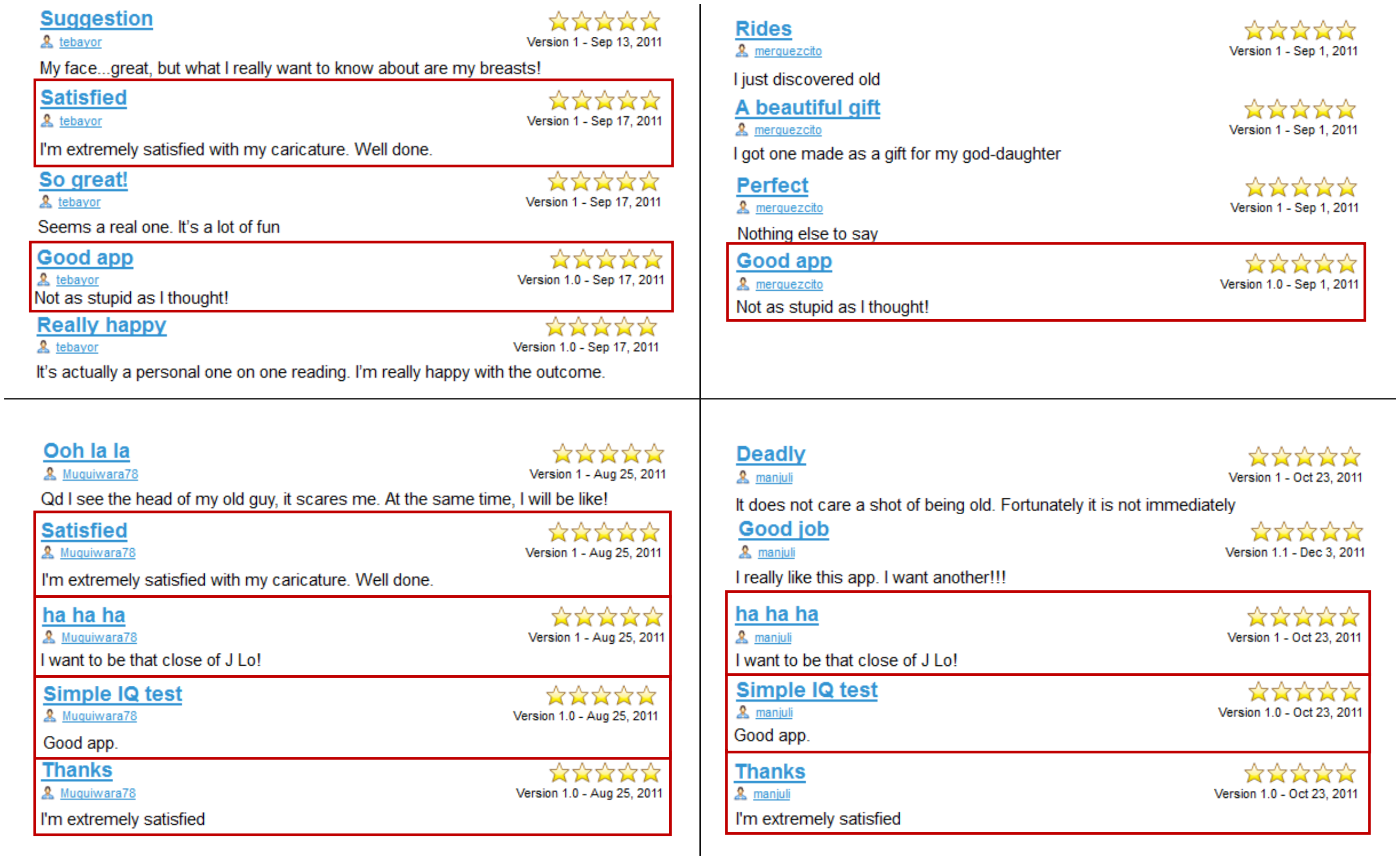
s: +	Products	
Users	Good	Bad
Honest	1-ϵ	ϵ
Fraud	2ϵ	1-2ϵ

Use Belief Propagation

s: -	Products	
Users	Good	Bad
Honest	ϵ	1-ϵ
Fraud	1-2ϵ	2ϵ

Opinion Fraud Detection in Online Reviews by Network Effects
Leman Akoglu, Rishi Chandy, Christos Faloutsos
ICWSM 2013

Found replicated bot reviews

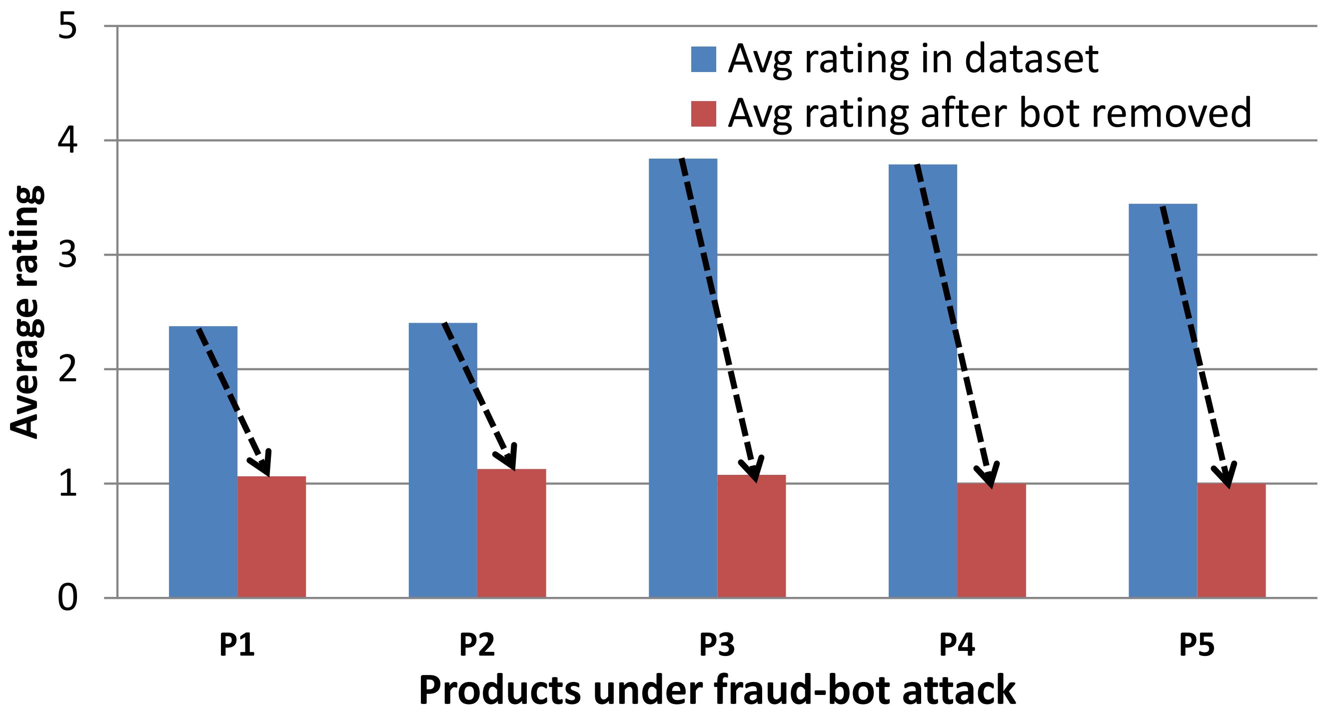


Opinion Fraud Detection in Online Reviews by Network Effects
Leman Akoglu, Rishi Chandy, Christos Faloutsos
ICWSM 2013

Found replicated bot reviews

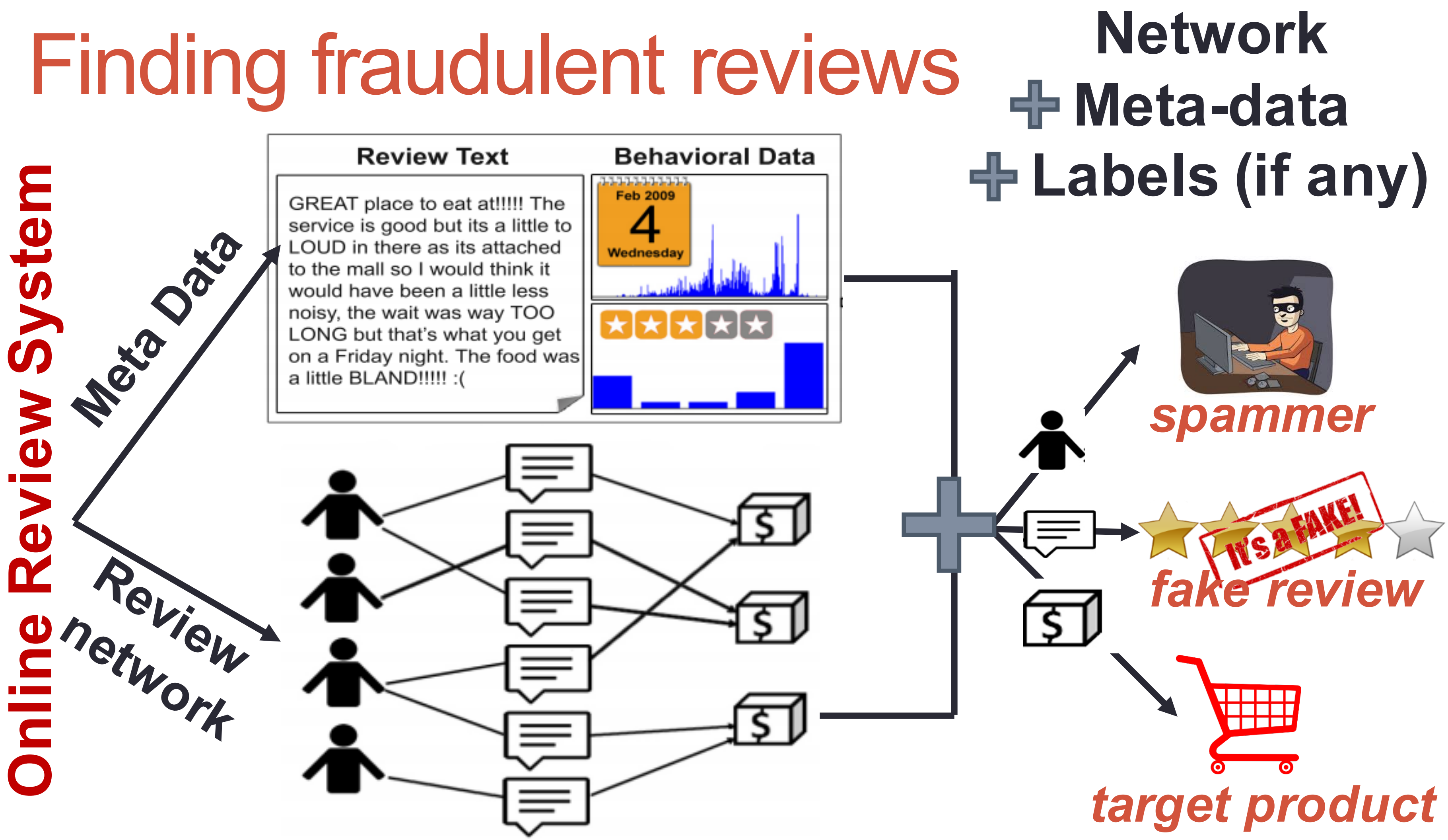
Opinion Fraud Detection in Online Reviews by Network Effects
Leman Akoglu, Rishi Chandy, Christos Faloutsos
ICWSM 2013

Finding fraudulent reviews



Opinion Fraud Detection in Online Reviews by Network Effects
Leman Akoglu, Rishi Chandy, Christos Faloutsos
ICWSM 2013

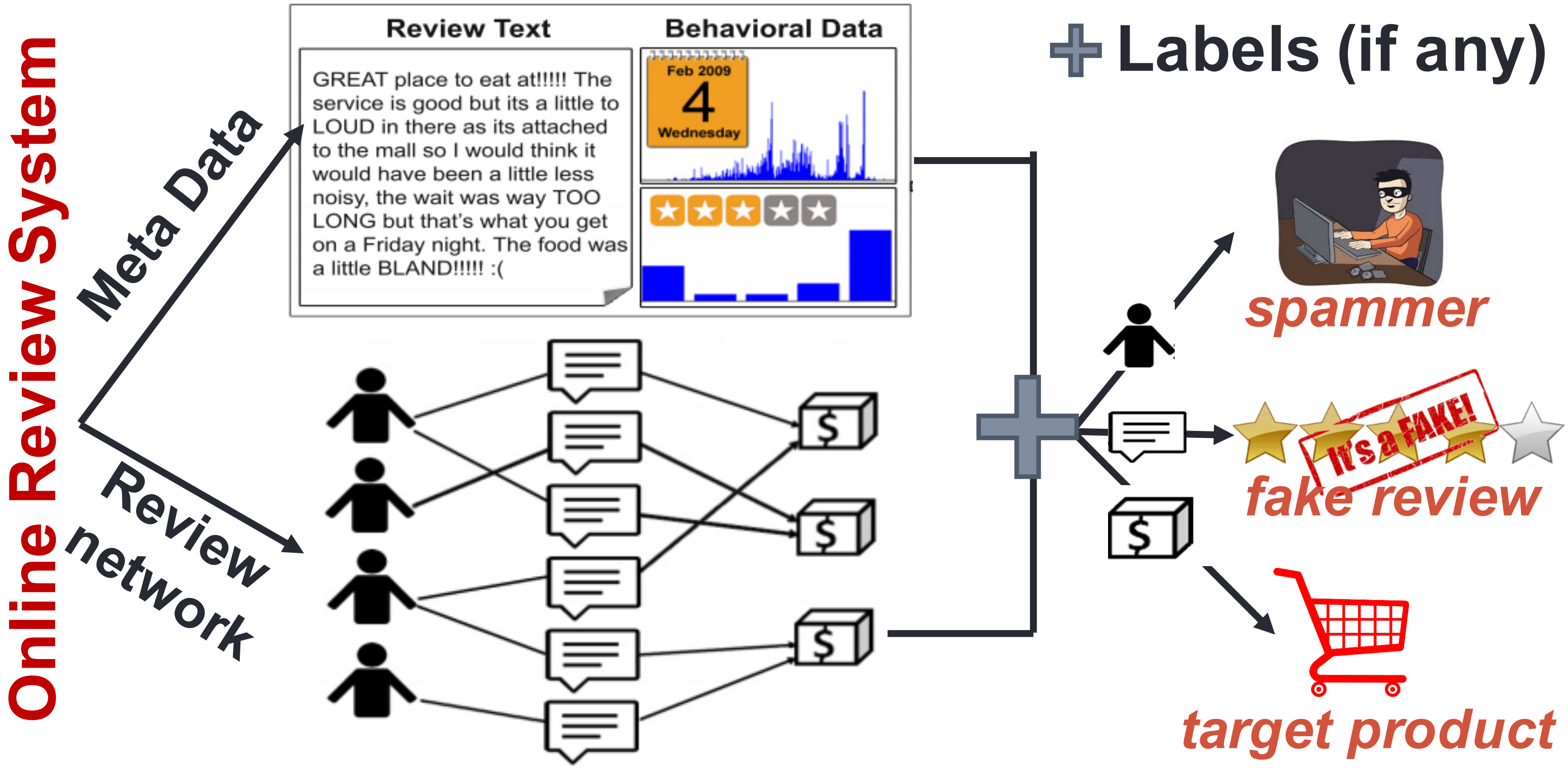
Finding fraudulent reviews



Collective Opinion Spam Detection:
Bridging Review Networks and Metadata
Shebuti Rayana, Leman Akoglu
KDD 2015

Finding fraudulent reviews

Network
+ Meta-data
+ Labels (if any)



Collective Opinion Spam Detection: Bridging Review Networks and Metadata
Shebuti Rayana, Leman Akoglu
KDD 2015

See the talk on Tuesday @ 2:40PM! (Level 2 – Room 2)

Practitioner's Guide

Method	Graph Type	Node Attributes	Edge Attributes	Seed Labels
HITS	Directed			
PageRank	Directed			Optional
Label Prop.	Undirected			✓
pMRF BP	Undirected			Preferred
EdgeExplain	Undirected	✓		✓

Practitioner's Guide

Method	Graph Type	Node Attributes	Edge Attributes	Seed Labels
HITS	Directed			
PageRank	Directed			Optional
Label Prop.	Undirected			✓
pMRF BP	Undirected			Preferred
EdgeExplain	Undirected	✓		✓

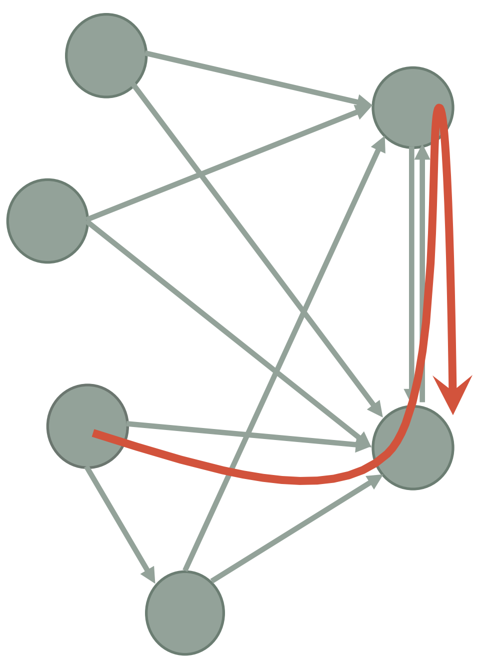
All of the node attributes are the seed labels for semi-supervised learning in EdgeExplain.

Practitioner's Guide

Method	Graph Type	Node Attributes	Edge Attributes	Seed Labels
CatchSync	Directed			
TrustRank	Directed			✓
Distrust Rank	Directed			✓
SibylRank	Directed			✓
NetProbe	Directed			
FraudEagle	Bipartite		✓	
SpEagle	Tripartite	✓		

Key Points

Random walks ~ singular vectors: find important nodes, and communities



"Guilt-by-association" propagate labels (homophily, heterophily, and more)