Detecting Suspicious Following Behavior in Multimillion-Node Social Networks

Meng Jiang[1], Peng Cui[1], Alex Beutel[2], Christos Faloutsos[2], Shiqiang Yang[1]
[1] Department of Computer Science and Technology, Tsinghua University, Beijing, China
[2] Computer Science Department, SCS, Carnegie Mellon University, PA, USA

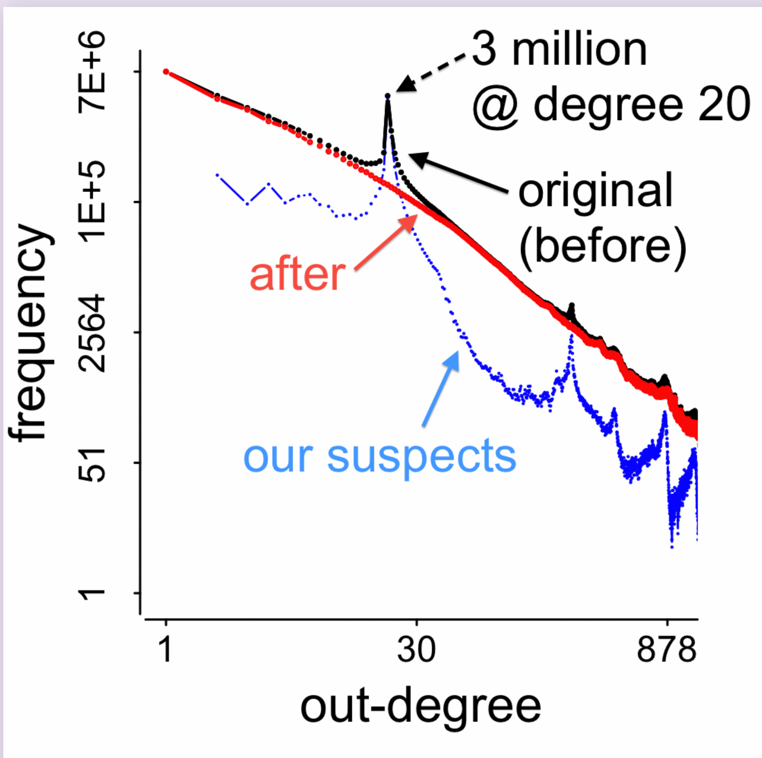
Zombie Followers in "Who-Follows-Whom" Networks

Who are they? What do they do?

- Fraudsters are paid to make certain accounts seem more legitimate or famous through giving them many additional followers.
- They are often required to perform some tasks (e.g., follow the same group of users) together.

Deviations on out-degree distribution

- 3 million Twitter zombie followers create a spike at degree 20.
- **CatchSync** restores normal (power-law-like and smooth) patterns.



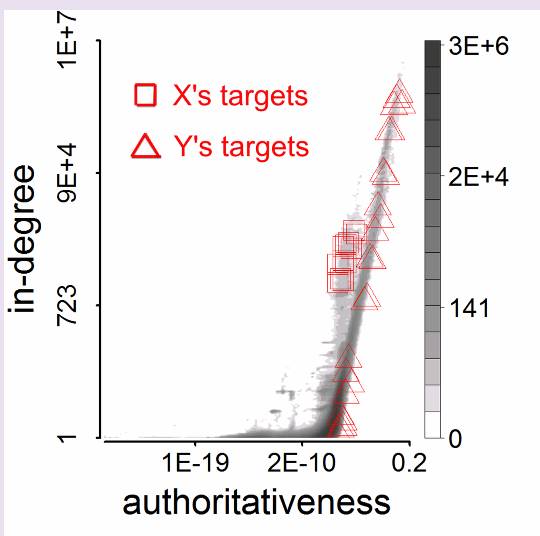
Suspicious Following Behavior Patterns

Synchronized: they have extremely similar behavior pattern.
Abnormal: their behavior patterns are different from the majority.

Feature space [HITS, Kleinberg et al., 1999]

Followee	**In-degree**	**Authority:** high-quality celebrities/idols
Follower	**Out-degree**	**Hub:** containing links to famous idols

Example:
User X: a zombie follower
User Y: an honest account
(1) Out-degree values are the same: 20.
(2) X's hub (4.7×10^{-7}) << Y's hub (1.6×10^{-4})
(3) X's and Y's targets (followees):
- X's targets are too similar and very strange.



Synchronicity and Normality

Synchronicity: similarity between X's targets.

$$sync(u) = \frac{\sum_{(v,v')\in\mathcal{F}(u)\times\mathcal{F}(u)} \mathbf{p}(v)\cdot\mathbf{p}(v')}{d(u)\times d(u)}$$

Normality: similarity between X's targets and other nodes.

$$norm(u) = \frac{\sum_{(v,v')\in\mathcal{F}(u)\times\mathcal{U}} \mathbf{p}(v)\cdot\mathbf{p}(v')}{d(u)\times N}$$

* Features in vector **p**: (1) each source/target node; (2) structural features like centrality, eigenvectors; (3) side information like dates of birth, names.

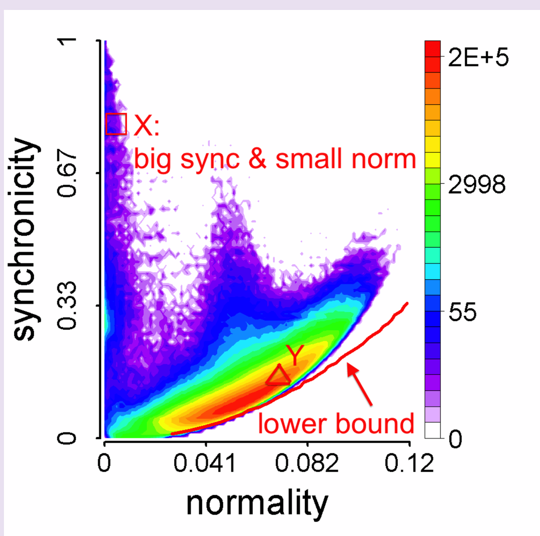
Theorem *For any distribution, there is a* <u>*parabolic lower limit*</u> *in the synchronicity-normality plot.*

$$\boxed{s_{min} = (-Mn^2 + 2n - s_b)/(1 - Ms_b)}$$

Proof. It is based on Lagrange multipliers.

- X has much **bigger synchronicity** and **smaller normality** than Y (from majority).

CatchSync Algorithm: (1) feature space; (2) sync and norm; (3) outlier detection.
Complexity analysis: O(E) – **scalable**.

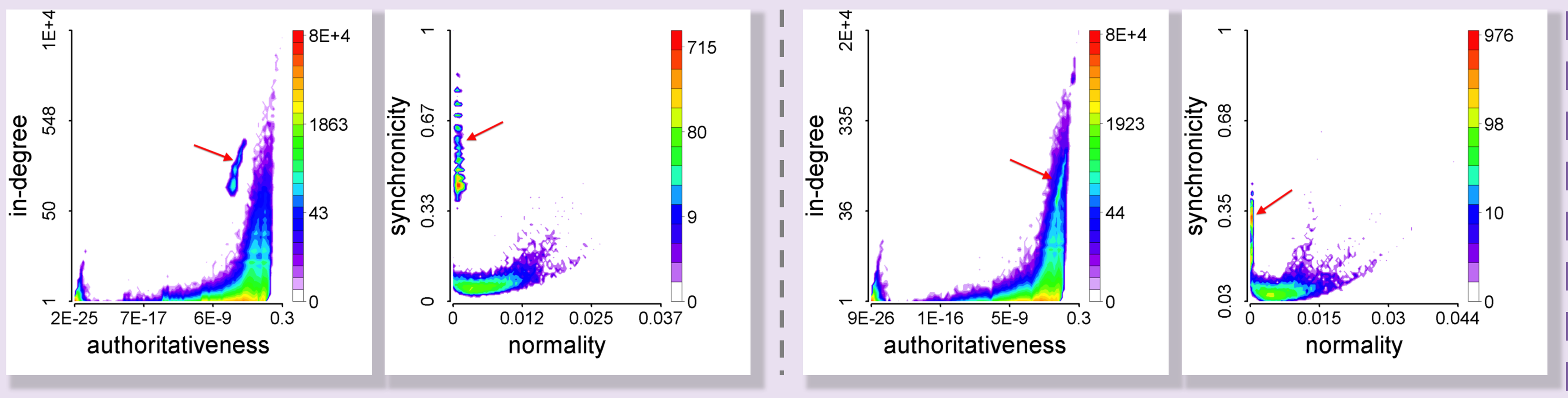


Experimental Results on Synthetic Data

Injection on random power law graph

- $3M^2$ + 5 groups (16K sources× 1.6K targets, 8K × 800, …, 1K × 100).
- **Camouflage:** 10% more **RAND**om users; 50% top **POP**ular idols.

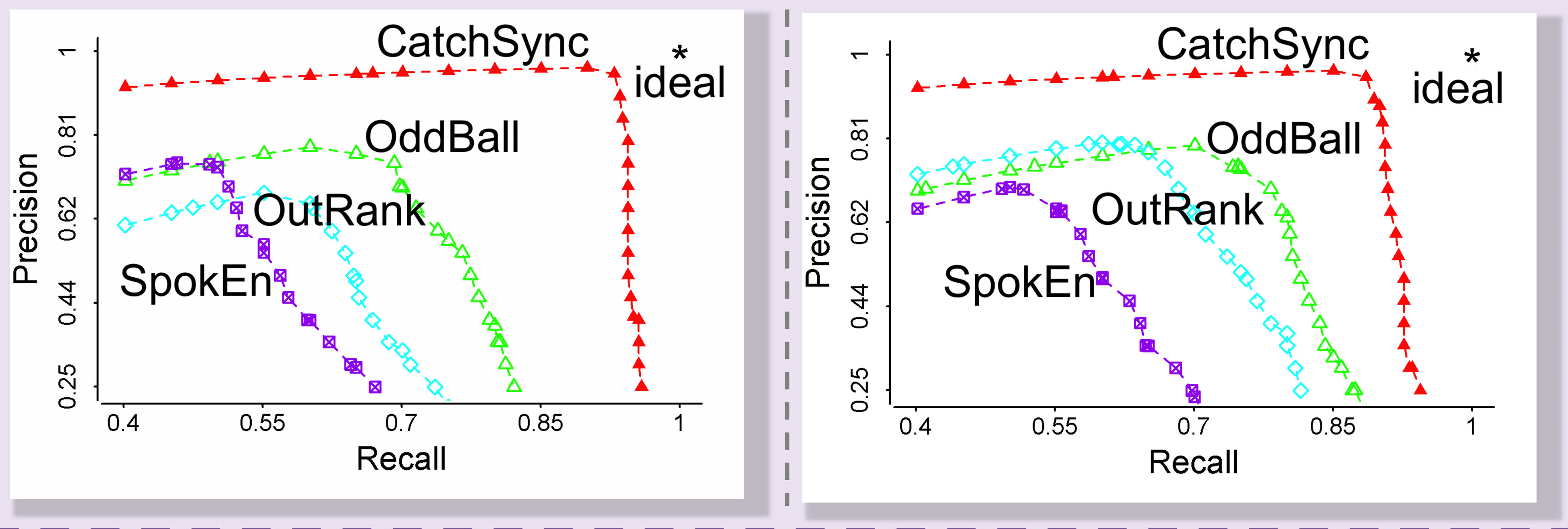
Synchronicity-normality plots



Accuracy on 3M, 3M-RAND and 3M-POP

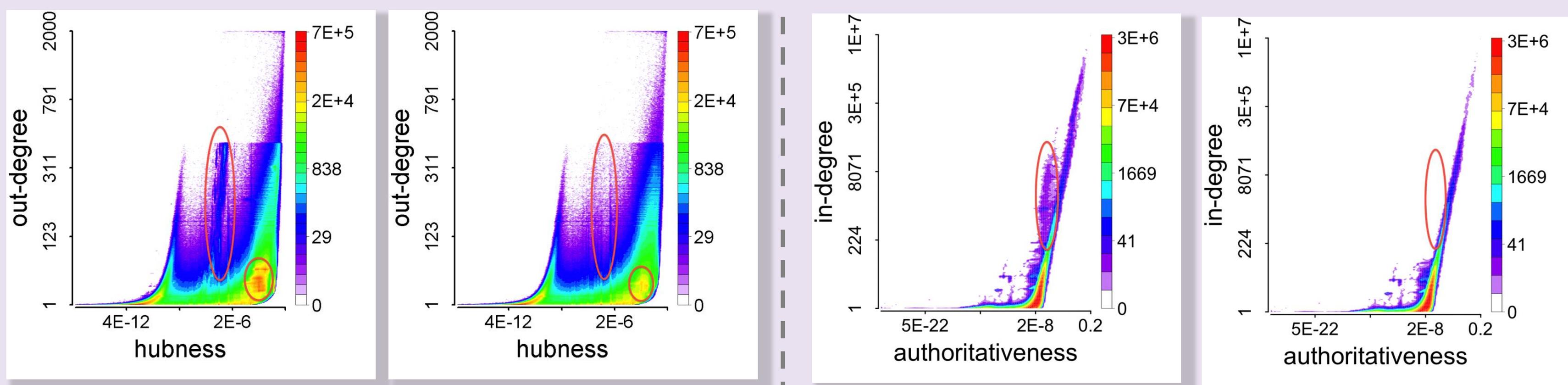
Synthetic graph	SYNTH-3M	SYNTH-3M-RAND		SYNTH-3M-POP	
Camouflage (d_{camou})	None (0)	10%	50%	10%	50%
CATCHSYNC	**0.956**	**0.910**	**0.764**	**0.885**	**0.792**
ODDBALL	0.755	0.702	0.525	0.657	0.433
OUTRANK	0.725	0.678	0.516	0.694	0.392
SPOKEN	0.677	0.586	0.470	0.553	0.351

Precision-recall on 3M-RAND and 3M-POP



Experimental Results on Twitter and Tencent Weibo Data

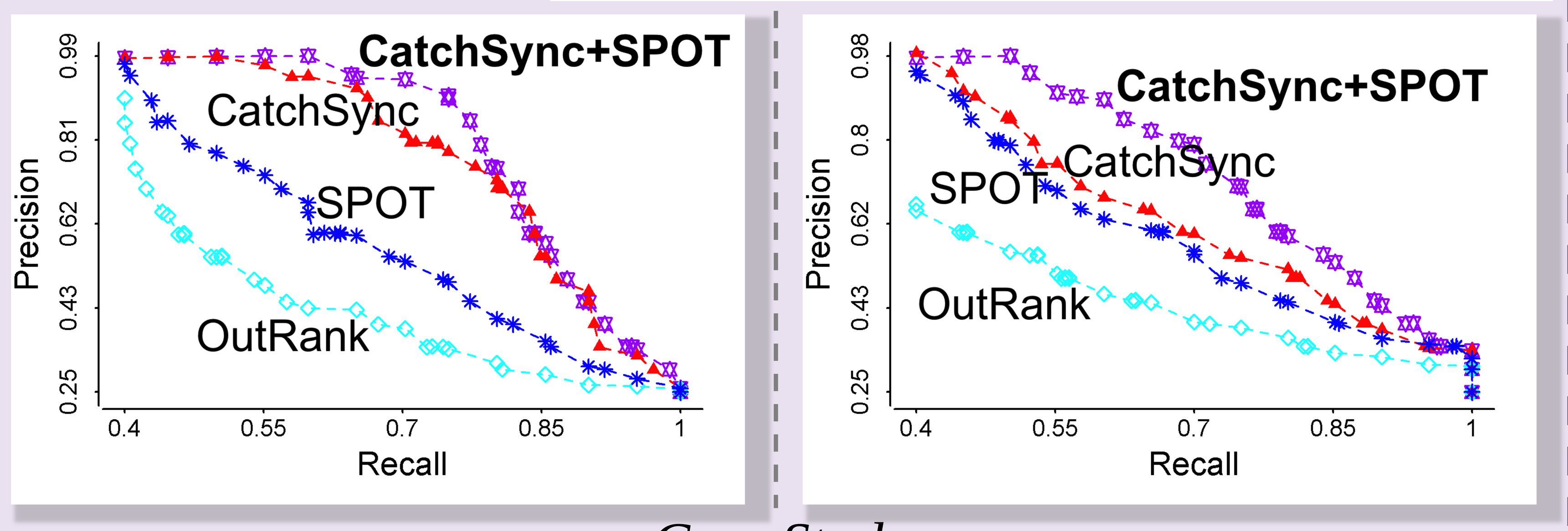
Restore normal patterns in feature space/distributions



Accuracy, precision and recall on fraudsters

SPOT: content-based
CatchSync: graph-based
CatchSync OR(+) SPOT

	TWITTERSG	WEIBOSG
CATCHSYNC	0.751	0.694
OUTRANK	0.412	0.377
SPOT	0.597	0.653
CATCHSYNC+SPOT	**0.813**	**0.785**



Case Study

Acknowledgement

Contact: Meng Jiang
<jm06@mails.tsinghua.edu.cn>

清華大學 Tsinghua University
Carnegie Mellon University