Detecting Suspicious Following Behavior in Multimillion-Node Social Networks

Meng Jiang
Dept. of Computer Science
Tsinghua University
jm06@mails.tsinghua.edu.cn

Peng Cui
Dept. of Computer Science
Tsinghua University
cuip@tsinghua.edu.cn

Alex Beutel
Dept. of Computer Science
Carnegie Mellon University
abeutel@cs.cmu.edu

Christos Faloutsos
Dept. of Computer Science
Carnegie Mellon University
christos@cs.cmu.edu

Shiqiang Yang
Dept. of Computer Science
Tsinghua University
yangshq@tsinghua.edu.cn

ABSTRACT

In a multimillion-node network of who-follows-whom like Twitter, since a high count of followers leads to higher profits, users have the incentive to boost their in-degree. Can we spot the suspicious following behavior, which may indicate zombie followers and suspicious followees? To answer the above question, we propose CatchSync, which exploits two tell-tale signs of the suspicious behavior: (a) synchronized behavior: the zombie followers have extremely similar following behavior pattern, because, say, they are generated by a script; and (b) abnormal behavior: their behavior pattern is very different from the majority. Our CatchSync introduces novel measures to quantify both concepts and catches the suspicious behavior. Moreover, we show it is effective in a real-world social network.

Categories and Subject Descriptors

J.4 [**Computer Application**]: Social and Behavioral Sciences

Keywords

social graph, zombie follower, synchronized behavior

1. INTRODUCTION

In social networks, the trustworthiness of marketers and the popularity of celebrities is measured by how many followers they have. Branding accounts, celebrities, and even grass-root users need to attract as many followers as they can. Many companies abuse social networks by creating and selling fake accounts that act as "zombie followers", i.e. accounts who mindlessly follow others. This phenomenon creates distorted images of popularity and trustworthiness, with unpleasant or even dangerous effects to honest users.

To address this problem, instead of trying to understand the behavior that the zombie followers *appear to have* (e.g., duplicate tweets [2], malicious urls [7]), we propose to study and detect the behavior that they *must have* (e.g., follow and get followed) to achieve their monetary goals. In more detail, the zombie followers exhibit behavior that is (a) synchronized: they often follow the very same accounts and (b) abnormal: their behavior pattern is very different from the majority. It is exactly these two properties that we propose to exploit, to spot the zombie followers.

In our proposed CatchSync, we give a novel way to measure the synchronicity and the normality of a group of users and spot the outliers. We extensively evaluate the proposed method on Tencent Weibo (one of the largest microblogs in China). The experimental results demonstrated that the proposed CatchSync can spot the suspicious nodes.

WWW'14 Companion, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2745-9/14/04.
http://dx.doi.org/10.1145/2567948.2577306.

2. OUR APPROACH

Here we first give the problem definition of suspicious following behavior detection: given a directed graph of N nodes in the node set $\mathcal{U}$, find a set of zombie followers $\mathcal{U}_{sync}$ that have *synchronized* and *abnormal* behavior. The word "synchronized" means the followers connect to the very similar users, and "abnormal" means their behavior pattern is very different from the majority of follower nodes.

We define $\mathbf{p}(u)$ as the feature vector of node u. In this paper, we choose the degree values (out-degree and in-degree) and HITS scores (hubness and authoritativeness) [3] for two reasons: these features are fast to compute, as well as easy to plot. Figure 1(a) shows a 2d feature space (out-degree vs hubness) of all the follower nodes in Tencent Weibo. As our experiments show, these features work well in pin-pointing suspicious nodes. We denote by $\mathcal{F}(u)$ the set of node u's followees and by $d(u) = |\mathcal{F}(u)|$ u's out-degree. We define the synchronicity of node u as the average similarity of the feature vectors between each pair of u's followees (v, v'):

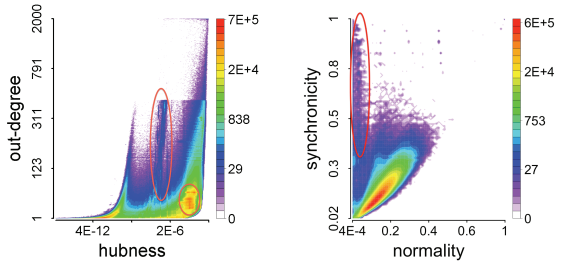
$$sync(u) = \frac{\sum_{(v,v')\in\mathcal{F}(u)\times\mathcal{F}(u)} \mathbf{p}(v)\cdot\mathbf{p}(v')}{d(u)\times d(u)} \quad (1)$$

Also, we define the normality of u as the average similarity of the feature vectors between each pair of u's followees and other nodes (v, v'):

$$norm(u) = \frac{\sum_{(v,v')\in\mathcal{F}(u)\times\mathcal{U}} \mathbf{p}(v)\cdot\mathbf{p}(v')}{d(u)\times N} \quad (2)$$

Figure 1(b) shows the synchronicity-normality plot, in which we can spot outliers over the majority. The outliers have the largest synchronicity scores and smallest normality scores,

indicating that their behavior patterns are synchronized and abnormal. In CatchSync we use a distance-based anomaly detection method to catch the suspicious nodes.



(a) Feature space (b) $sync(u)$ vs $norm(u)$

Figure 1: Heat map highlighting outliers from CatchSync. The outliers are followers whose behavior pattern has the largest synchronicity and smallest normality scores.

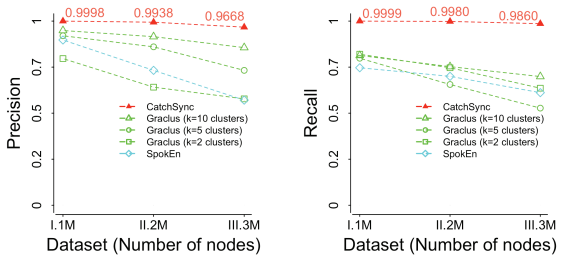
3. EXPERIMENT

Datasets: Table 1 shows the datasets we use for our experiments. We generate three random power-law graphs of $1M$, $2M$ and $3M$ nodes. To each graph we add ten groups of $10,000$ new followers and $1,000$ new followees. Within each group, each follower connects to 20 random followees from the corresponding group. We also use our two real-world datasets Twitter and Weibo. Both of them are huge and complete graphs of popular online social networks.

	Nodes	Edges
Synthetic	1M / 2M / 3M	4M / 8M / 12M
Twitter [4]	41,652,230	1,468,365,182
Weibo	117,288,075	3,134,074,580

Table 1: The real-world social datasets and synthetic data.

Detection Effectiveness: Synthetic data allows us to have clearly labelled regular "not suspicious" and injected "suspicious" nodes, and is easily reproducible. In Figure 2 we compare the *precision* and *recall* of CatchSync against that of Graclus [1] and SpokEn [6]. We tuned the number of clusters for Graclus and the number of communities for SpokEn to achieve the best performance. From this experiment, we can clearly see that CatchSync is extremely effective at detecting this suspicious behavior, while successfully catching very few false positives. Both its precision and recall exceed those of its competitors significantly.



(a) Precision (b) Recall

Figure 2: Detection effectiveness on synthetic data. CatchSync reaches almost 100%. (Higher is better for both.)

Evaluation with Side Information: Much of the related work in the domain of detecting fake accounts focuses on using side information, such as tweet content and profile information. Often, these methods were labelled by users evaluated by hand that very same information. Although our method does *not* use this side information in detecting the suspicious nodes, we use it to label a subset of nodes. The rule to label a node as suspicious was they have Twitter names like @"Buy_XX##", Weibo names like "a#####" (X is a capital letter and # is a digit number) and dates of birth as January 1^{st}. (Note these names were chosen because there were *many* that followed this narrow pattern.) On Twitter and Weibo, we collect $7,738$ and $10,787$ suspicious accounts with the above rule, respectively. As can be seen in Table 2, CatchSync has a very high recall of over 96% and significantly outperforms all other approaches. Again, CatchSync did not use any of the information that we used to label the nodes.

	Twitter	Weibo
CatchSync	**0.992**	**0.967**
Honeypots [5]	0.674	0.752
Graclus [1]	0.336	0.425
SpokEn [6]	0.465	0.279

Table 2: Recall on labelled data of side information.

4. CONCLUSION

In this paper, we presented a study of suspicious following behavior detection. We discover that the zombie followers exhibit behavior that is synchronized and abnormal. We propose CatchSync algorithm to catch the suspicious behavior on large social graphs. The experimental results show that our method is effective on a real-world dataset.

Acknowledgement This work is supported by NSFC, No. 61370022, No. 61003097, No. 60933013, and No. 61210008; International Science and Technology Cooperation Program of China, No. 2013DFG12870; National Program on Key Basic Research Project, No. 2011CB302206; NExT Research Center funded by MDA, Singapore, WBS:R-252-300-001-490. Thanks for the support of NSF, No. CNS-1314632; Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053; U.S. ARO and DARPA, No. W911NF-11-C-0088; and the NSF Graduate Research Fellowship, Grant No. DGE-1252522.

5. REFERENCES

[1] I. S. Dhillon, Y. Guan, and B. Kulis. Weighted graph cuts without eigenvectors a multilevel approach. *TPAMI*, 29(11):1944–1957, 2007.

[2] X. Hu, J. Tang, Y. Zhang, and H. Liu. Social spammer detection in microblogging. In *IJCAI'13*.

[3] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *JACM*, 46(5), 1999.

[4] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *WWW'10*, pages 591–600.

[5] K. Lee, J. Caverlee, and S. Webb. Uncovering social spammers: social honeypots+ machine learning. In *SIGIR'10*, pages 435–442.

[6] B. A. Prakash, A. Sridharan, M. Seshadri, S. Machiraju, and C. Faloutsos. Eigenspokes: Surprising patterns and scalable community chipping in large graphs. In *AKDDM*, pages 435–448. 2010.

[7] K. Thomas, C. Grier, D. Song, and V. Paxson. Suspended accounts in retrospect: an analysis of twitter spam. In *IMC'11*, pages 243–258.