Detection and Elimination of Fraud Ranking of Mobile Apps

Junaid Naeem Khan
B. E. Computer Engineering
SSBT's College of Engineering and Technology
Jalgaon, India

Aalip Aayub Pinjari
B. E. Computer Engineering
SSBT's College of Engineering and Technology
Jalgaon, India

Nourin Sadik Deshmukh
B. E. Computer Engineering
SSBT's College of Engineering and Technology
Jalgaon, India

Nikhil Suryabhan Patil
B. E. Computer Engineering
SSBT's College of Engineering and Technology
Jalgaon, India

***Abstract*—Now days, mobile apps are ranked by popularity. Rankings get affected by reviews and ratings. As when the app get more positive and in favor reviews, its popularity increases. The popularity affects its ranking and attracts more users. Indeed, the review manipulation is becoming more common for fraud ranking of mobile apps. The imposter often post fake reviews with a purpose of bumping up the apps in the popularity list. So, the necessity of preventing fraud ranking increases. Such fake reviews must be detected and discarded so that real apps get to their genuine positions in the ranking. For this purpose, the proposed system provide a holistic view of fraud ranking by fake reviews and provide a fake rating and review detection and elimination system for mobile apps.**

Keywords— Mobile apps, ranking fraud detection, rating, review, linear combination

I. INTRODUCTION

The number of mobile Apps has grown up at a breathtaking rate over the past few years. As an example, as of the end of April 2013, there are more than 1.6 million Apps at Apples App store and Google Play. To stimulate the development of mobile Apps, several App stores launched daily App leaderboards, which demonstrate the chart rankings of most popular Apps. Indeed, the App leaderboard is one in all the foremost vital ways for promoting mobile Apps. A better rank on the leaderboard usually results in a large number of downloads and million dollars in revenue. Therefore, App developers tend to explore numerous ways that such as advertising campaigns to promote their Apps so as to possess their Apps graded as high as possible in such App leaderboards.

However, as a recent trend, rather than relying on traditional marketing solutions, shady App developers resort to some dishonest means to deliberately boost their Apps and eventually manipulate the chart rankings on an App store. This is typically implemented by using supposed "bot farms" or "human water armies" to inflate the App downloads, ratings and reviews in a very short time. as an example, an article from Venture Beat [2] reported that, when an App was promoted with the help of ranking manipulation, it could be propelled from no 1,800 to the highest 25 in Apple's top free leaderboard and over 50,000-100,000 new users might be acquired within a couple of days. In fact, such ranking fraud raises great issues to the mobile App business. As an example, Apple has warned of cracking down on App developers who commit ranking fraud [3] in the Apple's App store.

Overview: The remainder of this paper is organized as follows. In section 2, some brief discussion on related work is provided. The existing system is described in Section 3. The Section 4 presents the proposed system and methodology. Some further discussion about the proposed system is shown in Section 5. And finally, Section 6 concludes the paper and some future research direction.

II. RELATED WORK

In the literature, there are some related work, such as web ranking spam detection[4], [5], online review spam detection [6], [7], and mobile App recommendation [8], [9], but the problem of detecting ranking fraud for mobile app is still under-explored.

A. Web Ranking Spam Detection

The web ranking spam refers to any provident actions that bring to selected webpages an unjustifiable admiring relevance or importance. As an example, Ntoulas et al. [4] have studied no of aspects of content-based spam on the web and presented variety of heuristic strategies for detecting content based spam. Zhou et al. [10] have studied the problem of unsupervised web ranking spam detection. Specifically, they proposed an efficient onlinelink spam and term spam detection strategies using spam city. Recently, Spirin and Han [5] have reported a survey on web spam detection, which comprehensively introduces the principles and algorithms in the literature. Indeed, the work of web ranking spam detection is principally based on the analysis of ranking principles of search engines, like PageRank and query term frequency. This is different from ranking fraud detection for mobile Apps.

B. Online Review Spam Detection

Lim et al. [6] have known many representative behaviors of review spammers and model

these behaviors to find the spammers. Wu et al. [11] have studied the problem of detecting hybrid shilling attacks on rating data. The proposed approach relies on the semi supervised learning and might be used for trustworthy product recommendation. Xie et al. [7] have studied the problem of singleton review spam detection. Specifically, they resolved this problem by detecting the co-anomaly patterns in multiple review based time series. Though some of above approaches may be used for anomaly detection from historical rating and review records, they are not capable to extract fraud evidences for a given period (i.e., leading session).

C. Mobile App Recommendation

Yan and Chen [9] developed a mobile App recommender system, named Appjoy, which relies on records of user's App usage to create a preference matrix rather than using specific user ratings. Also, to solve the sparseness drawback of App usage records, Shi and Ali [8] studied many recommendation models and proposed content based cooperative filtering model, named Eigenapp, for recommending Apps in their web site Getjar. In addition, some researchers studied the problem of exploiting enriched discourse information for mobile App recommendation. For example, Zhu et al. [12] proposed a consistent framework for personalized context-aware recommendation, which can integrate both context independence and dependency assumptions. However, as per our knowledge, no one has studied the problem of ranking fraud detection for mobile Apps.

III. Existing System

Hengshu Zhu et al. [1] provides ranking fraud detection system for mobile Apps that accurately locate the ranking fraud by mining the active periods, such as leading sessions, of mobile Apps. Then three types of evidences are found in these leading sessions i.e. ranking based evidences, rating based evidences and review based evidences for recognizing ranking fraud. Hengshu Zhu also proposes an optimization based aggregation methodology for integrating all the evidences for fraud detection. In this validation is performed on the system and a few regularities of ranking fraud activities has been displayed. In this all evidences are modeled by hypothesis tests so that, it is simple to extend with alternative evidences to discover ranking fraud.

IV. Proposed System

As careful observation reveals that mobile Apps don't seem to be always graded high in the leaderboard, however only in some leading events, that form different leading sessions. In other words, ranking fraud typically happens in these leading sessions. Therefore, detecting ranking fraud of mobile Apps is truly to notice ranking fraud inside leading sessions of mobile Apps. Specifically, we tend to first propose a straightforward however effective algorithm to identify the leading sessions of every App based on its historical ranking records. Then, with the analysis of Apps' ranking behaviors, we discover that the deceitful Apps usually have completely different ranking patterns in every leading session compared with traditional Apps. Thus, we have a tendency to characterize some fraud evidences from Apps' historical ranking records, and develop three functions to extract such ranking based fraud evidences. Nonetheless, the ranking based evidences can be affected by App developers' reputation and some legitimate marketing campaigns, like "limited-time discount". As a result, it's not enough to only use ranking based evidences. Fig. 1 shows the system architecture of the proposed system.

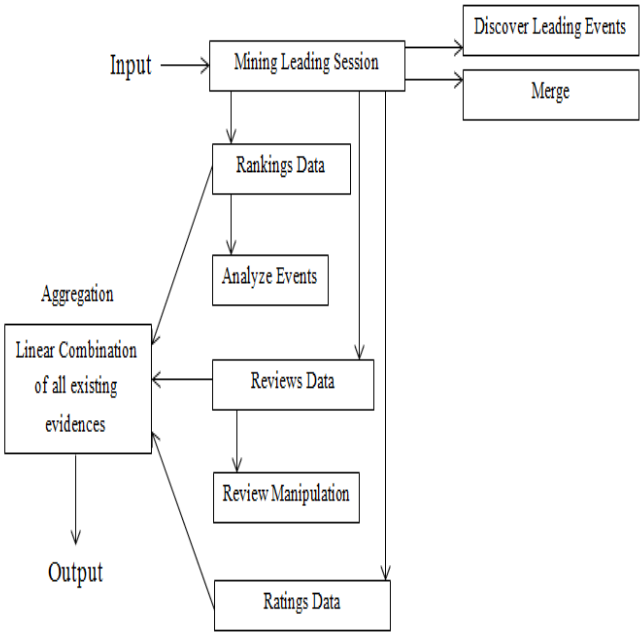

Fig. 1. System Architecture.

A. Mining Leading Sessions

Hengshu Zhu et al., in "Discovery of Ranking Fraud for Mobile Apps" [1] say that the App leaderboard demonstrates top K popular Apps with respect to different categories, such as "Top Free Apps" and "Top Paid Apps". Moreover, the leaderboard is usually updated periodically (e.g., daily). Therefore, each mobile App a has many historical ranking records which can be denoted as a time series, $R_a=\{r^a_1,\ldots,r^a_i,\ldots,r^a_n\}$, where $r^a_i \in \{1,\ldots,K,+\infty\}$ is the ranking of a at time stamp t_i; $+\infty$ means a is not ranked in the top K list; n denotes the no of all ranking records. Note that, the smaller value r^a_i has, the higher ranking position the App obtains.

There are two steps for mining leading sessions. First, the leading events are located from apps historical ranking records. Second, these leading events are merged together to form leading sessions. Algorithm 1 shows the pseudo code of mining leading sessions for a given App a.

Algorithm 1 Mining Leading Sessions:

Input 1: a's historical ranking records R_a;

Input 2: the ranking threshold K^*;

Input 3: the merging threshold ϕ;

Output: the set of a's leading sessions S_a;

Initialization: $S_a = \emptyset$;

1. $E_s = \emptyset$; $e = \emptyset$; $s = \emptyset$; $t^e_{start} = 0$;
2. **for each** $i \in [1,|R_a|]$ **do**
3. **if** $r^a_i \leq K^*$ **and** $t^e_{start} == 0$ **then**
4. $t^e_{start} = t_i$;
5. **else if** $r^a_i > K^*$ **and** $t^e_{start} \neq 0$ **then**

6. //Found one event
7. $t^e_{end} = t_{i-1}$; $e = < t^e_{start}, t^e_{end} >$;
8. **if** $E_s == \emptyset$ **then**
9. $E_s \cup e$; $t^e_{start} = t^e_{start}$; $t^e_{end} = t^e_{end}$;
10. **else if** $(t^e_{start} - t^e_{end}) < \emptyset$ **then**
11. $E_s \cup e$; $t^e_{end} = t^e_{end}$;
12. **else then**
13. //Found one session;
14. $s = < t^e_{start}, t^e_{end}, E_s >$;
15. $S_a \cup s$; $s = \emptyset$ is a new session;
16. $E_s = \{e\}$; $t^e_{start} = t^e_{start}$; $t^e_{end} = t^e_{end}$;
17. $t^e_{start} = 0$; $e = \emptyset$ is a new leading event;
18. **return** S_a

In Algorithm 1, we denote each leading event e and session s as tuples < testart, teend> and < testart, teend, Es > respectively, where Es is the set of leading events in session s. Now, with the analysis of App's historical records, it is discovered that the deceitful apps usually have completely different ranking patterns in every leading sessions compared with traditional Apps. Thus, some fraud evidences are obtained from Apps' historical ranking records.

B. Ranking Based Evidences

By inspecting the Apps' historical ranking records, it is found that Apps' ranking behaviors during a leading event usually satisfy a particular ranking pattern, that consists of three different ranking phases, namely, rising phase, maintaining phase and recession phase. Specifically, in every leading event, an App's ranking first increases to a peak position in the leaderboard (i.e., rising phase), then keeps such peak position for some duration (i.e., maintaining phase), and at last decreases till the end of the event (i.e., recession phase).

If ranking manipulation is carry out in the leading session s of App a, then a's ranking behavior should be different in these three ranking phases from those in a normal leading session. Actually, it is observed that every app with fraud ranking usually has an expected ranking target e.g. top 20 in leaderboard for one week. More even, after reaching and maintaining the expected ranking for a required period, the manipulation will be stopped and the ranking of the malicious App will decrease dramatically. As a result, the suspicious leading events might contain very short rising and recession phases. Also, the cost of ranking manipulation with high ranking expectations is quite expensive due to the unclear ranking principles of App stores and the tough competition between App developers. Therefore, the leading event of fraudulent Apps usually has very short maintaining phase with high ranking positions.

C. Rating Based Evidences

The ranking based evidences are effective for ranking fraud detection. But sometimes, it is not sufficient to only use ranking based evidences. For example, some Apps created by the famous developers, such as Game loft, may have some leading events with short rising phase due to the developers' credibility and the "word-of-mouth" advertising effect. More even, some of the legal marketing services, such as "limited-time discount", may also result in significant ranking based evidences. To resolve this problem, fraud evidences from Apps' historical rating records are also extracted.

Specifically, after An App has been published, the rating can be given to it by different users who downloaded it. As really, user rating is also an App advertisements' one of important feature. More users are attracted towards the app which has higher rating and it is ranked higher in the leaderboard. Thus, rating manipulation is also a crucial perspective of ranking fraud. Probably, the fraud ranked Apps in a leading session s may have anomaly patterns of rating during the period of s compared with its historical ratings, which might be used for constructing rating based evidences. It is noticed that a normal App always receives similar average rating each day, whereas a dishonest App may receive comparatively higher average ratings in some time periods (e.g., leading sessions) than other times.

D. Review Based Evidences

Along with ratings, most of the App stores also permit users to express some textual comments as App reviews. Such reviews are the personal perception and usage experiences of existing users for specific mobile apps. Indeed, review manipulation is one among the most important perspective of App ranking fraud. Specifically, before downloading or purchasing a new mobile App, users usually first read its historical reviews to ease their decision making, and a mobile App contains more positive reviews might attract a lot of users to download. Therefore, imposters usually post phony reviews in the leading sessions of a particular App so as to inflate the App download, and therefore propel the App's ranking position within the leaderboard. though some previous works on review spam detection are reported in recent years [6], [13], the problem of detecting the local anomaly of reviews within the leading sessions and capturing them as evidences for ranking fraud detection are still under-explored. To this end, here we tend to propose two fraud evidences based on Apps' review behaviors in leading sessions for detecting ranking fraud.

Certainly, most of the reviews manipulations are carry out by bot farms because of the high cost of human resource. Therefore, review spammers typically post multiple duplicate or near-duplicate reviews on identical App to inflate downloads [6]. Whereas, the normal App always have diversified reviews since users have distinct personal perceptions and usage experiences.

From the real-world observations, it is found that each review c is often related to a particular latent topic z. for instance, some reviews could be associated with the latent topic "worth to play" whereas some may be associated with the latent topic "very boring". Meanwhile, since various users have distinct personal preferences of mobile Apps, each App a could have diverse topic distributions in their historical review records. Intuitively, the topic distribution of reviews in a normal leading session s of App a, ought to be according to the topic distribution in all historical review records of a.

After the extraction of these fraud evidences, the next challenge is how to merge them for ranking fraud detection.

Indeed, there are several ranking and evidence aggregation methods in the literature, like permutation based models [14], [15], score based models [16], [17] and Dempster-Shafer rules [18], [19]. However, some of these strategies concentrate on learning a global ranking for all candidates. This is not proper for detecting ranking fraud for new Apps. Other method relies on supervised learning techniques, which depend on the labeled training data and are exhausting to be exploited. Additionally, Hengshu Zhu et al. propose an unsupervised approach based on fraud similarity to merge these evidences. Instead, the linear combination of these evidences is formed to discover and detect the ranking fraud properly.

V. Results and Analysis

Here the results obtained by analyzing the historical data of the apps are described i.e. ranking, rating and review data. This is given as an input to the system which is nothing but the ranking, rating and review given by different users on apps. As careful observation reveals that mobile Apps don't seem to be always graded high in the leaderboard, however only in some leading events, that form different leading sessions. In other words, ranking fraud typically happens in these leading sessions.

So, first the leading sessions are identified with the help of mining leading session algorithm. Usually, it is found that fraudulent apps often have different pattern of ranking, rating and reviews than the normal apps. So, accordingly we extract the evidences from the leading session such as ranking, rating and reviews evidences. These evidences are extracted by careful analysis and comparison of the behavior of every app in all leading sessions of it.

Then, all these extracted evidences are merged with the liner combination technique so ensure the ranking fraud. And finally, that apps ranking is recalculated by eliminating the fraud ranking, rating and review.

VI. Discussion

Here we provide some discussion about the proposed detection and elimination of fraud ranking of mobile apps system.

First, the download information is a relevant signature for detecting ranking fraud, since ranking manipulation is to use so-called "bot farms" or "human water armies" to inflate the App downloads and ratings in a very short time. However, the instant download information of each mobile App is usually not accessible for analysis. In fact, Apple and Google don't give proper download information on any App. moreover, the App developers themselves are also reluctant to unleash their download information for numerous reasons. Therefore, in this paper, the primarily concentration is on extraction of evidences from Apps' historical ranking, rating and review records for ranking fraud detection. However, this approach is scalable for combining other evidences if available, like the evidences based on the download information and App developers' fame.

Second, the proposed approach can detect ranking fraud happened in Apps' historical leading sessions. However, sometime, we need to detect such ranking fraud from Apps current ranking observations.

Finally, after detecting ranking fraud for every leading session of a mobile App, the remainder problem is how to estimate the quality of this App. Indeed, our approach can discover the local anomaly rather than the global anomaly of mobile Apps. Thus, we should take concentration of such kind of local characteristics when estimating the quality of Apps.

Conclusion

In this paper, a ranking fraud detection and elimination system for mobile Apps is developed. Specifically, first it is exposed that ranking fraud happened in leading sessions and present a method for mining leading sessions for every App from its historical ranking records. Then, ranking based evidences, rating based evidences and review based evidences are identified for detecting ranking fraud. Moreover, linear combination of these evidences is performed to merge all the evidences for evaluating the credibility of leading sessions from mobile Apps. A distinct aspect of this approach is that it is easy to be extended with other evidences from other domain knowledge to detect ranking fraud. Finally, proposed system is validated with experiments on sample data set.

In the future, more effective fraud evidences are planned to be study and analyze the latent relationship among rating, review and rankings. Moreover, ranking fraud detection approach will be extended with other mobile App related services, such as mobile Apps recommendation, for enhancing user experience.

References

[1] Hengshu Zhu, HuiXiong, Senior Member, IEEE, Yong Ge, and Enhong Chen, Senior Member, IEEE, **"***Discovery of Ranking Fraud for Mobile Apps***",** vol.13,n0.1,Jan 2015

[2] (2012). [online]. Available: http//venturebeat.com/2012/07/03/apples-crackdown-on-app-ranking-manipulation.

[3] (2012). [Online]. Available: https://developer.apple.com/news/index.php?id=02062012a

[4] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly, "Detecting spam web pages through content analysis," in Proc. 15th Int. Conf. World Wide Web, 2006, pp. 83–92.

[5] N. Spirin and J. Han, "Survey on web spam detection: Principles and algorithms," SIGKDD Explor. Newslett., vol. 13, no. 2, pp. 50– 64, May 2012.

[6] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers using rating behaviors," in Proc. 19thACMInt. Conf. Inform. Knowl. Manage., 2010, pp. 939–948.

[7] S. Xie, G. Wang, S. Lin, and P. S. Yu, "Review spam detection via temporal pattern discovery," in Proc. 18th ACM SIGKDD Int. Conf.Knowl. Discovery Data Mining, 2012, pp. 823–831.

[8] K. Shi and K. Ali, "Getjar mobile application recommendations with very sparse datasets," in Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2012, pp. 204–212.

[9] B. Yan and G. Chen, "AppJoy: Personalized mobile application discovery," in Proc. 9th Int. Conf. Mobile Syst., Appl., Serv., 2011, pp. 113–126.

[10] B. Zhou, J. Pei, and Z. Tang, "A spamicity approach to web spam detection," in Proc. SIAM Int. Conf. Data Mining, 2008, pp. 277–288.

[11] Z. Wu, J. Wu, J. Cao, and D. Tao, "HySAD: A semi-supervised hybrid shilling attack detector for trustworthy product

recommendation," in Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2012, pp. 985–993.

[12] H. Zhu, E. Chen, K. Yu, H. Cao, H. Xiong, and J. Tian, "Mining personal context-aware preferences for mobile users," in Proc. IEEE 12th Int. Conf. Data Mining, 2012, pp. 1212–1217.

[13] N. Jindal and B. Liu, "Opinion spam and analysis," in Proc. Int. Conf. Web Search Data Mining, 2008, pp. 219–230.

[14] A. Klementiev, D. Roth, and K. Small, "Unsupervised rank aggregation with distance-based models," in Proc. 25th Int. Conf. Mach. Learn., 2008, pp. 472–479.

[15] A. Klementiev, D. Roth, K. Small, and I. Titov, "Unsupervised rank aggregation with domain-specific expertise," in Proc. 21st Int. Joint Conf. Artif. Intell., 2009, pp. 1101–1106.

[16] D. F. Gleich and L.-h. Lim, "Rank aggregation via nuclear norm minimization," in Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2011, pp. 60–68.

[17] M. N. Volkovs and R. S. Zemel, "A flexible generative model for preference aggregation," in Proc. 21st Int. Conf. World Wide Web, 2012, pp. 479–488.

[18] Y. Ge, H. Xiong, C. Liu, and Z.-H. Zhou, "A taxi driving fraud detection system," in Proc. IEEE 11th Int. Conf. Data Mining, 2011, pp. 181–190.

[19] G. Shafer, A Mathematical Theory of Evidence. Princeton, NJ, USA: Princeton Univ. Press, 1976.