Enhancement and Efficient Prediction of Fraud Application Based on Ranking and Review

J.Kavitha[1],D.Vinotha[2]

[1]M.Tech, Dept.of Computer Science & Engineering, PRIST University,Vallam, Thanjavur.
[2]Assistant Professor, Dept.of Computer Science & Engineering, PRIST University,Vallam, Thanjavur.

***Abstract*—** Ranking fraud in the mobile App business arcade refers to fraudulent or false activities which have a purpose of hitting up the Apps in the popularity list. Indeed, it becomes more and more frequent for App creators to use shady means, such as inflating their App's sales or deflating other apps through posting of phony ratings, to commit ranking fraud. While the significance of preventing fraud has been widely acknowledged there is limited recognition and research in this area.we first propose to accurately locate the ranking fraud by mining the lively periods, namely leading sessions, of mobile Apps. Such leading sessions can be leveraged for detecting the local irregularity instead of global irregularity of App rankings. Furthermore, we explore three types of facts, i.e., ranking, rating and review based facts by molding Apps' ranking, rating and review behaviors through numerical hypotheses tests. In addition, we propose an optimization based accumulation method to integrate all the facts for fraud detection. Finally, we evaluate the proposed system with real-world App data collected from the iOS App Store for a long time period. In the experiments, we validate the effectiveness of the proposed system, and show the scalability of the detection algorithm as well as some predictability of ranking fraud activities.

***Index Terms*—facts accumulation,historical ranking records,Mobile Apps, ranking fraud detection.**

I. INTRODUCTION

The amount of mobile Apps has grown at an enormous rate over the past few years.

Manuscript received March, 2016.
***J.Kavitha**,M.Tech,Dept. ofComputer Science and Engineering,PRIST University Thanjavur,India.*

***D.Vinotha,** M.S. Dept. of Computer Science and Engineering, Assistant Professor, PRIST University Thanjavur,India.*

For a sample, there are more than 1.6 million Apps at Apple's App store and Google Play services.To encourage the development of mobile Apps, many App stores tossed daily App leaderboards, which display the chartrankings of most popular Apps. Indeed, the Appleaderboard is one of the most important ways for sponsoring mobile Apps. A higher rank on the leaderboardusually leads to enormous number of downloads and million dollarsin revenue.However, as a recent trend, instead of depending on customary marketing solutions, shady App developers route to some fraudulent means to deliberately increase their Apps and eventually position the chart rankings on an App store.This is usually implemented by using so-called "bot farms" or "human water armies" to inflate and deflate the App downloads, ratings and reviews in a very short time[7],[8],[9].For example, an article from Venture Beat reported that, when an App was promoted with the help of ranking influence, it could be pushed from number 1,800 to the top 25 in Apple's top free leaderboard and more than 50,000-100,000 new users could beacquired within a couple of days.In truth, such ranking fraud advances fears to the mobile App industry. For example, Apple has cautioned of cracking down on App developers who commit ranking fraud in the Apple's App store.

II. RELATED WORK

The related work of this paper narrates web ranking spam detection, online review spam detection and mobile App recommendation, the problem of detecting ranking fraud for mobile Apps is still under-explored.To fill this vital void, in this paper, we propose to develop a ranking fraud detection system for mobile Apps[5].

First, ranking fraud does not always happen in the whole life cycle of an App, so we need to discover the time when fraud happens. Such experiment can be regarded as identifying the local irregularity instead of global irregularity of mobile Apps.Second, due to the huge number of mobile Apps, it is difficult to manually label ranking fraud for each App, so it is important to have a scalable way to automatically detect ranking fraud without using any standard information. Finally, due to the vibrant nature of chart rankings, it is not easy to find and confirm the facts linked to ranking fraud, which stimulates us to discover some hidden fraud patterns of mobile Apps as facts.Fig 1. Shows the ranking fraud detection framework of our system.The detection of ranking fraud involves identifying the leading session of each app based on its historical ranking records.This detection will find the ranking fraud happened in the leading sessions of mobile Apps.

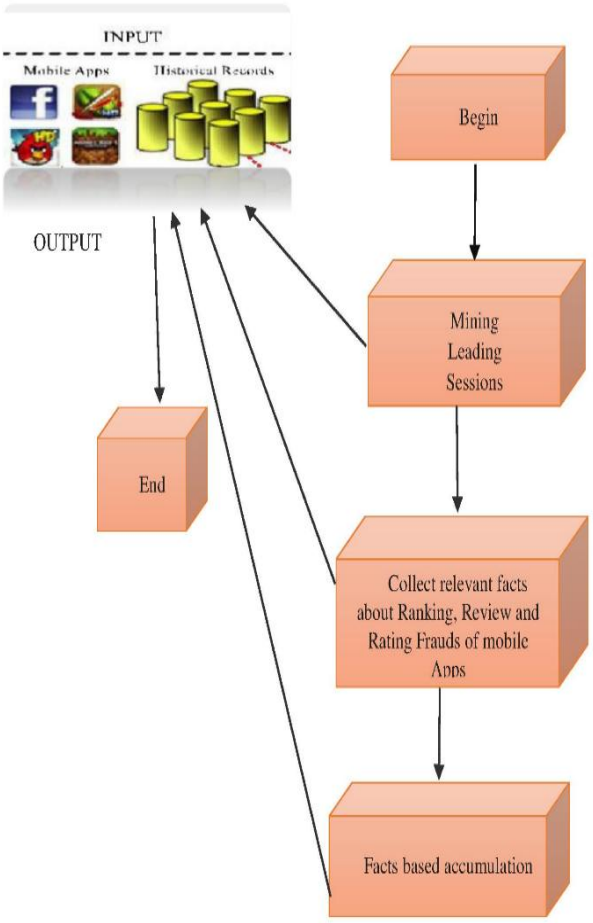


Fig 1.The framework of our ranking fraud detection system.

III. MINING THE LEADING SESSION

There are two steps in mining the leading sessions. Discovering the leading events and merging the leading events for building the leading sessions of an app.

Algorithm 1 Mining Leading Sessions

Input 1: a's historical ranking records R_a;
Input 2: the ranking threshold K^*;
Input 2: the merging threshold $\dot{\omega}$;
Output: the set of a's leading sessions
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: $E_s \cup = e$; $t^s_{start} = t^e_{start}$; $t^s_{end} = t^e_{end}$;
10: **else if** $(t^e_{start} - t^s_{end}) < \dot{\omega}$ **then**
11: $E_s \cup = e$; $t^s_{end} = t^e_{end}$;
12: **else then**
13: //found one session;
14: $s =< t^s_{start}, t^s_{end}, E_s >$;
15: $S_a \cup = s$; $s = \emptyset$ is a new session;
16: $E_s = \{e\}$; $t^s_{start} = t^e_{start}$; $t^s_{end} = t^e_{end}$;
17: $t^e_{start} = 0$; $e = \emptyset$ is a new leading event;
18: **return** S_a

The tuples for leading event, session is <t^e_{start}; t^e_{end}>, <t^s_{start}; t^s_{end}; where E_s is the sets of leading events in session s. we first, obtain the individual leading event e for the given App a (i.e., Step 2 to 7) from the beginning of the algorithm.

For each individual leading event e, we check the time span between e and the current leading session s to decide whether they belong to the same leading session. ewill be considered as a new leading session (i.e., Step 8 to 16). Thus,this algorithm can identify leading events and sessions by scanning a's historical ranking records only once.

IV. COLLECTING RELEVANT FACTS FOR RANKING, RATING AND REVIEW

A leading session is composed of several leading events. Therefore, we should first analyze the basic characteristics of leading events.By evaluating the Apps' historical ranking records, we spot that Apps' ranking behaviors in a leading event always satisfya specific ranking pattern.It is shown in Fig 2,which consists of three different ranking phases, namely, rising phase, maintaining phaseand recession phase.

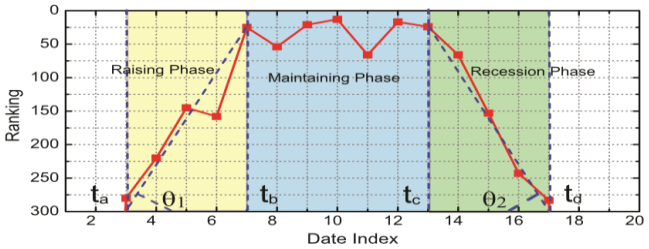


Fig 2. Apps ranking pattern shows different ranking phases ofleading event.

Indeed, this ranking pattern shows an important understanding of leading event.

After an app has been published, it can be rated by any user who downloaded it. The user rating is one of the most important features of App advertisement. An App which has higher rating may draw more users to download and can also be ranked higher in the leaderboard.

Intuitively, if an App has ranking fraud in a leading session s, the ratings during the time period of s may have irregularity patterns compared with its historical ratings, which can be used for making rating based facts.

Besides ratings, most of the App stores also allow users to write some textual remarks as App criticisms. Such reviews can reflect the personal views and usage experiences of existing users for particular mobile Apps. Indeed, review manipulation is one of the most important perspective of App ranking fraud[7]. This review manipulation is implemented by the bot farms. Therefore the review spammersoften post several duplicate reviews on the apps to inflate downloads. In difference to that, the normal apps always have varied reviews since users have different personal observations and usage proficiencies.

V. FACTS BASED ACCUMULATION

Fact based accumulation involves extracting three types of fraud facts and merging them for ranking fraud detection. Indeed, there are many ranking fraud facts aggregation methods in the literature, such as permutation based models, score based models and Dempster-Shafer rules. These, methods focus on learning a global ranking for all candidates. This is not helpful for detecting the ranking fraud for new Apps.It focuses the unsupervised learning techniques that depends on the labeled training data and are hard to be explored. Instead, we propose an unsupervised approach based on fraud similarity to combine these facts.Different facts may have different score values to evaluate the leading sessions.For example,some evidences may always have higher score values than the average fact score,eventhough they can detect abnormal leading sessions and rank them in precise positions.

VI. EXPERIMENTAL RESULTS

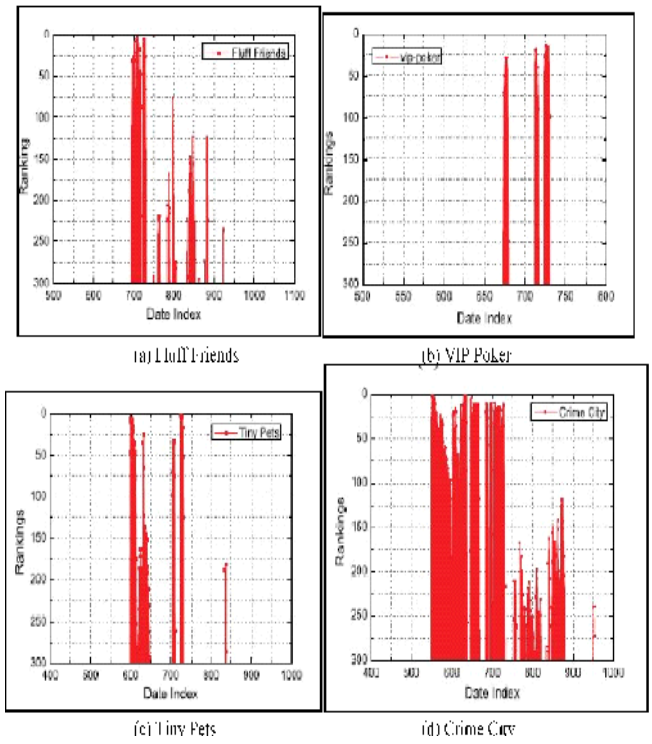
This section, illustrates the performances of ranking fraud detection using real-world App data.The experimental data sets comprises of the “Top Free 300” and “Top Paid 300” leaderboard apps collected from the Apple’s App Store (U.S.) on February 2, 2010 to September 17, 2012. The data sets cover the daily chart rankings1 of top 300 free Apps and top 300 paid Apps, respectively. Moreover, each data set also includesthe user ratings and review evidence. Table 1 shows the detailed characteristic features of our data sets.

TABLE 1

Statistical characteristics of the experimental data

	Top Free 300 Apps	Top Paid 300 Apps
App Num.	9,784	5,261
Ranking Num.	285,900	285,900
Avg. Ranking Num.	29.22	54.34
Rating Num.	14,912,459	4,561,943
Avg. Rating Num.	1,524.17	867.12

To identify the ranking fraud from several leading sessions, an intuitive approach is developed termed as Evidence Aggregation based Ranking Fraud Detection (EA-RFD). Predominantly, this approach is indicated with score based aggregation (i.e., Principle 1) as EA-RFD-1, and approach with rank based aggregation (i.e., Principle 2) as EARFD-2, respectively .In the Fig.3 mentioned below, there exist a seven free Apps which might involve in ranking fraud namely as Tiny Pets, Social Girl, Fluff Friends, Crime City, VIP Poker, Sweet Shop, Top Girl. So each approach like EA-RFD1 and EA-RFD2 is applied on these Apps to find the suspicious Apps with high ranking. In this figure, we find all these Apps have clear ranking based fraud facts. For example, some Apps have very short leading sessions with high rankings and some Apps have leading session with many leading events. These explanations clearly validate the effectiveness of our approach.



(a) Hull Friends (b) VIP Poker (c) Tiny Pets (d) Crime City

Fig 3. Demonstration of ranking records of the four reported Suspicious apps.

VII. CONCLUSION

This paper presents a fraud system which is built up and it is actually a locating extortion discovery framework for mobile Apps. The ranking fraud happened in leading sessions and provided a process for mining leading sessions for each App from its historical ranking records.Then,we identified ranking, rating and review based facts for detecting ranking fraud. Furthermore, we proposed an optimization based accumulation method to incorporate all the facts for evaluating the credibility of leading sessions from mobile Apps. An unique outlook of this approach is that all the evidences can be modeled by numerical hypothesis tests, thus it is easy to be extended with other facts from field knowledge to detect ranking fraud. Finally, we validate the proposed system with wide experiments on real-world App data collected from the Apple's App store. Experimental results showed the efficiency of the proposed approach.

REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," J. Mach. Learn. Res., pp. 993–1022, 2003.

[2] 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.

[3] 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.

[4] T. L. Griffiths and M. Steyvers, "Finding scientific topics," Proc. Nat. Acad. Sci. USA, vol. 101, pp. 5228–5235, 2004.

[5] G. Heinrich, Parameter estimation for text analysis," Univ. Leipzig, Leipzig, Germany, Tech. Rep., http://faculty.cs.byu.edu/~ringger/ CS601R/papers/Heinrich-GibbsLDA.pdf, 2008.

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

[7] A. Klementiev, D. Roth, and K. Small, "An unsupervised learning algorithm for rank aggregation," in Proc. 18th Eur. Conf. Mach. Learn., 2007, pp. 616–623.

[8] 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.

[9] 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.

AUTHOR PROFILE

J.Kavitha,received B.E degree in Computer Science from Parisutham Institute of Technology and Science(Anna University) in 2013.She is currently doing her

M.Tech Computer Science and Engineering in PRIST University,Thanjavur. Her research interests are

Data mining, Big Data and Artificial intelligence.

D.Vinotha, completed her M.S Computer Science and Engineering from Wright State University,USA.She is currently working as an Assistant Professor in PRIST University,Thanjavur.Her research interests are Data mining, Database management systemandsoftware engineering.

