Volume No:1, Issue No:10 (March-2016)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Detecting Fraudulent Activities to Bump Rankings of Mobile Applications

Mrs.K.Sangeeta
Assistant Professor,
Department of CSE,
Pydah College of Engineering & Technology, Visakhpatnam, Andhra Pradesh, India.

Mr.Dr.Ramesh Challagundla
Professor & principal,
Pydah College of Engineering & Technology, Visakhpatnam, Andhra Pradesh, India.

Srinivasu Gubbala
M.Tech Student,
Department of CSE,
Pydah College of Engineering & Technology, Visakhpatnam, Andhra Pradesh, India.

Abstract:

A mobile app is a computer program designed to run on mobile devices such as smartphones and tablet computers. Usage of mobile apps has become increasingly prevalent across mobile phone users. No matter what store, app discoverability became more difficult now a days. Organic downloads from the app stores were mainly attributed to App Store Optimization. However, given the increasing competition, app publishers must invest in mobile marketing campaigns to build and retain their user base. Many mobile apps include a special Software development kit that will assist them in tracking installs from various ad networks. Ranking fraud in the mobile App market refers to fraudulent or deceptive activities which have a purpose of bumping up the Apps in the popularity list. Indeed, it becomes more and more frequent for App developers to use shady means, such as inflating their Apps' sales or posting phony App ratings, to commit ranking fraud. This paper gives a holistic perspective of positioning misrepresentation and propose a Ranking fraud identification framework for mobile Apps. In particular, it is proposed to precisely find the mining so as to pose extortion the dynamic periods, to be specific driving sessions, of portable Apps. Furthermore, we investigate three types of evidences, i.e., ranking based evidences, rating based evidences and review based evidences, by modeling Apps' ranking, rating and review behaviors through statistical hypotheses tests. In addition, we propose an optimization based aggregation method to integrate all the evidences for fraud detection.

Keywords: Mobile Applications, App Stores, App Rankings, Popularity lists, Ranking fraud.

Introduction:

Mobile native apps stand in contrast to software applications that run on desktop computers, and to web applications which run in mobile web browsers rather than directly on the mobile device. The term "app" is a shortening of the term "application software". Mobile apps were originally offered for general productivity and information retrieval, including email, calendar, contacts, stock market and weather information. However, public demand and the availability of developer tools drove rapid expansion into other categories, such as those handled by desktop application software packages. As with other software, the explosion in number and variety of apps made discovery a challenge, which in turn led to the creation of a wide range of review, recommendation, and curation sources, including blogs, magazines, and dedicated online app-discovery services.Usage of mobile apps has become increasingly prevalent across mobile phone users. A May 2012 comScore study reported that during the previous quarter, more mobile subscribers used apps than browsed the web on their devices: 51.1% vs. 49.8% respectively. Researchers found that usage of mobile apps strongly correlates with user context and depends on user's location and time of the day. Market research firm Gartner predicted that 102 billion apps would be downloaded in 2013 (91% of them free), which would generate $26 billion in the US, up 44.4% on 2012's US$18 billion. By Q2 2015, the Google Play and Apple stores alone generated $5 billion. An analyst report estimates that the app economy creates revenues of more than €10 billion per year within the European Union, while over 529,000 jobs have been created in 28 EU states due to the growth of the app market.

Volume No:1, Issue No:10 (March-2016)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Mobile App Development:

Developing apps for mobile devices requires considering the constraints and features of these devices. Mobile devices run on battery and have less powerful processors than personal computers and also have more features such as location detection and cameras. Developers also have to consider a wide array of screen sizes, hardware specifications and configurations because of intense competition in mobile software and changes within each of the platforms.Mobile application development requires use of specialized integrated development environments. Mobile apps are first tested within the development environment using emulators and later subjected to field testing. Emulators provide an inexpensive way to test applications on mobile phones to which developers may not have physical access.Mobile user interface (UI) Design is also essential. Mobile UI considers constraints and contexts, screen, input and mobility as outlines for design. The user is often the focus of interaction with their device, and the interface entails components of both hardware and software. User input allows for the users to manipulate a system, and device's output allows the system to indicate the effects of the users' manipulation. Mobile UI design constraints include limited attention and form factors, such as a mobile device's screen size for a user's hand. Mobile UI contexts signal cues from user activity, such as location and scheduling that can be shown from user interactions within a mobile application. Overall, mobile UI design's goal is primarily for an understandable, user-friendly interface.Mobile UIs, or front-ends, rely on mobile back-ends to support access to enterprise systems. The mobile back-end facilitates data routing, security, authentication, authorization, working off-line, and service orchestration. This functionality is supported by a mix of middleware components including mobile app servers, Mobile Back-end as a service (MBaaS), and SOA infrastructure.

App Distribution:
Google Play:

Google Play (formerly known as the Android Market) is an international online software store developed by Google for Android devices. It opened in October 2008. In August 2014, there were approximately 1.3+ million apps available for Android and the estimated number of applications downloaded from Google Play was 40 billion.

In July 2013, the number of apps downloaded via the Google Play Store surpassed 50 billion, of the over 1 million apps available. As of February 2015, According to Statista.com the number of apps available exceeded 1.4 million.

App Store:

Apple's App Store for iOS was not the first app distribution service, but it ignited the mobile revolution and was opened on July 10, 2008, and as of January 2011, reported over 10 billion downloads. The original AppStore was first demonstrated to Steve Jobs in 1993 by Jesse Tayler at NeXTWorld Expo[16] As of June 6, 2011, there were 425,000 apps available, which had been downloaded by 200 million iOS users.[17][18] During Apple's 2012 Worldwide Developers Conference, Apple CEO Tim Cook announced that the App Store has 650,000 available apps to download as well as 30 billion apps downloaded from the app store until that date.[19] From an alternative perspective, figures seen in July 2013 by the BBC from tracking service Adeven indicate over two-thirds of apps in the store are "zombies", barely ever installed by consumers.

Existing System:

The number of mobile Apps has grown at a breathtaking rate over the past few years. For example, as of the end of April 2013, there are more than 1.6 million Apps at Apple's App store and Google Play. To stimulate the development of mobile Apps, many App stores launched daily App leader boards, which demonstrate the chart rankings of most popular Apps. Indeed, the App leader board is one of the most important ways for promoting mobile Apps. A higher rank on the leader board usually leads to a huge number of downloads and million dollars in revenue. Therefore, App developers tend toexplore various ways such as advertising campaigns to promote their Apps in order to have their Apps ranked as high as possible in such App leader boards.

Disadvantages:

In other words, ranking fraud usually happens in these leading sessions. The main Disadvantage is Missing detecting ranking fraud of mobile Apps is actually to detect ranking fraud within leading sessions of mobile Apps.

Volume No:1, Issue No:10 (March-2016)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Proposed System:

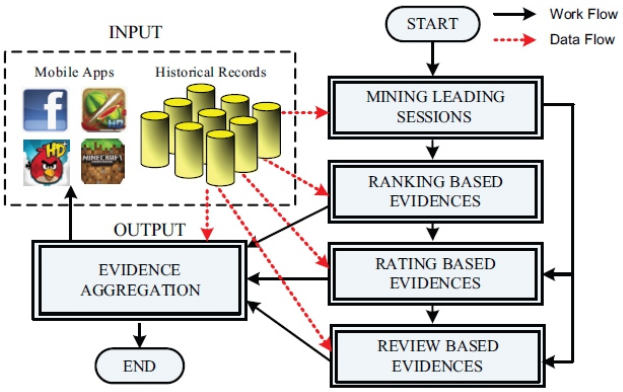
While the importance of preventing ranking fraud has been widely recognized, there is limited understanding and research in this area. To this end, in this paper, we provide a holistic view of ranking fraud and propose a ranking fraud detection system for mobile Apps. Specifically, we first propose to accurately locate the ranking fraud by mining the active periods, namely leading sessions, of mobile Apps. Such leading sessions can be leveraged for detecting the local anomaly instead of global anomaly of App rankings. Furthermore, we investigate three types of evidences, i.e., ranking based evidences, rating based evidences and review based evidences, by modeling Apps' ranking, rating and review behaviors through statistical hypotheses tests. In addition, we propose an optimization based aggregation method to integrate all the evidences 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 regularity of ranking fraud activities. we proposed an optimization based aggregation method to integrate all the evidences for evaluating the credibility of leading sessions from mobile Apps. An unique perspective of this approach is that all the evidences can be modeled by statistical hypothesis tests, thus it is easy to be extended with other evidences from domain knowledge to detect ranking fraud. Finally, we validate the proposed system with extensive experiments on real-world App data collected from the Apple's App store. Experimental results showed the effectiveness of the proposed approach.

Proposed Advantages:

We propose to develop a ranking fraud detection system for mobile Apps.

1. ranking fraud does not always happen in the whole life cycle of an App, so we need to detect the time when fraud happens. Such challenge can be regarded as detecting the local anomaly instead of
global anomaly of mobile Apps. Second, due to the huge number of mobile Apps, it is difficult.
2. 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 benchmark information.

Architecture:



Module 1
Leading events:

Definition 1 (Leading Event). Given a ranking threshold K e¼½testart;teend_ and corresponding rankings of a, which satisfies Ra start _ K _ a start_1 <r , and r a end _ K _ <r a endþ1 . Moreover, 8t k 2ðt e start ;t e end Þ, we have r a k . Note that we apply a ranking threshold K _ K _ _ which is usually smaller than K here because K may be very big (e.g., more than 1,000), and the ranking records beyond K _ (e.g., 300) are not very useful for detecting the ranking manipulations. Furthermore, we also find that some Apps have several adjacent leading events which are close to each other and form a leading session. For example, Fig. 2b shows an example of adjacent leading events of a given mobile App, which form two leading sessions. Particularly, a leading event which does not have other nearby neighbors can also be treated as a special leading session.

Module 2:
Leading Sessions:

A leading session s of App a contains a time range T s ¼½t s start ;t s end _ and n adjacent leading events fe 1 ; ...;e n g, which satisfies t and there is no other leading session s s start _ ¼ t e 1 start , t that makes T _ . Meanwhile, 8i ½1;nÞ, we have ðt e iþ1 start _ t e I end s end s I ¼ t _ _ T Þ < f, where f is a predefined time threshold for merging leading events. Intuitively, the leading sessions of a mobile App represent its periods of popularity, so the ranking manipulation will only take place in these leading sessions. Therefore, the problem of detecting ranking fraud is to detect fraudulent leading sessions. Along this line, the first task is how to mine the leading sessions of a mobile App from its historical ranking records.

Volume No:1, Issue No:10 (March-2016)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Module 3

Identifying the Leading Sessions for Mobile APPs:

There are two main steps for mining leading sessions. First, we need to discover leading events from the App's historical ranking records. Second, we need to merge adjacent leading events for constructing leading sessions. Specifically, Algorithm demonstrates the pseudo code of mining leading sessions for a given App In Algorithm , we denote each leading event e and session s as tuples <t e start ;t e end > and <t s start > respectively, where E is the set of leading events in session s. Specifically, we first extract individual leading event e for the given App a (i.e., Step 2 to 7) from the beginning time. For each extracted 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 based on Definition 2. Particularly, if ðt s e start ;t _ t Þ < f, will be considered as a new leading session (i.e., Step 8 to 16). Thus, this algorithm can identify leading events and ses- sions by scanning a's historical ranking records only once. S end s end ;E s

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 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: $E_s \cup = e$; $t^s_{start} = t^e_{start}$; $t^s_{end} = t^e_{end}$;
10: **else if** $(t^e_{start} - t^s_{end}) < \phi$ **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

In Algorithm , we denote each leading event e and session s as tuples <te start, te end > and <ts start, tsend,Es> respectively, where Es is the set of leading events in session s. Specifically, we first extract individual leading event e for the given App a (i.e., Step 2 to 7) from the beginning time. For each extracted 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 based on Definition 2.

Particularly, if (te start −ts end) < ϕ, e will 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.

CONCLUDING REMARKS:

In this paper, we developed a ranking fraud detection system for mobile Apps. Specifically, we first showed that ranking fraud happened in leading sessions and provided a method for mining leading sessions for each App from itshistorical ranking records. Then, we identified ranking based evidences, rating based evidences and review based evidences for detecting ranking fraud. Moreover, we proposed an optimization based aggregation method to integrate all the evidences for evaluating the credibility of leading sessions from mobile Apps. An unique perspective of this approach is that all the evidences can be modeled by statistical hypothesis tests, thus it is easy to be extended with other evidences from domain knowledge to detect ranking fraud. Finally, we validate the proposed system with extensive experiments on real-world App data collected from the Apple's App store. Experimental results showed the effectiveness of the proposed approach. In the future, we plan to study more effective fraud evidences and analyze the latent relationship among rating, review and rankings. Moreover, we will extend our ranking fraud detection approach with other mobile App related services, such as mobile Apps recommendation, for enhancing user experience.

REFERENCES:

[1] Discovery of Ranking fraud for mobile apps. Hengshu Zhu,HuiXiong,Seniormembers,IEEE,YongGe,andEnhongChen,Seniormember,IEEE,IEEE transactions onknowledge and data engineering,vol .27,No.1,January2015.

[2] Detecting product review spammers using ratingbehaviors. E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu,and H. W. Lauw. In Proceedings of the 19th ACMinternational conference on Information and knowledgemanagement.

[3] Supervised rank aggregation. Y.-T. Liu, T.-Y.Liu, T.Qin, Z.-M. Ma, and H. Li In Proceedings of the 16thinternational conference onWorld Wide Web.

Volume No:1, Issue No:10 (March-2016)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

[4] An unsupervised learning algorithm for rankaggregation, A. Klementiev, D. Roth, and K. Small In Proceedings of the 18th Europeanconference onMachine Learning, ECML '07, pages 616–623, 2007.

[5]An unsupervised learning algorithm for rankaggregation, A. Klementiev, D. Roth, and K. SmallIn Proceedings of the 18th Europeanconference onMachine Learning, ECML '07, pages 616–623, 2007.

[6]Getjar mobile application recommendations with verysparse datasets. K. Shi and K. Ali. In Proceedings of the18th ACM SIGKDDinternational conference on-Knowledge discovery and data mining, KDD '12, pages204–212, 2012.

[7]Ranking fraud Mining personal context-awarepreferences for mobile users. H. Zhu, E. Chen, K. Yu, H.Cao, H. Xiong, and J. Tian. In Data Mining (ICDM),2012 IEEE 12th International Conference on,pages1212–1217, 2012. ation and knowledgemanagement, CIKM '10, pages 939–948, 2010.

[8]detection for mobile apps H. Zhu, H. Xiong, Y. Ge, andE. Chen. A holistic view. In Proceedings of the 22ndACMinternational conference on Information and-knowledge management,CIKM '13, 2013.

[9] Exploiting enriched contextual information for mobile-app classification, H. Zhu, H. Cao, E. Chen, H. Xiong,and J. Tian. In Proceedings of the 21st ACMinternational conference on Information and knowledgemanagement, CIKM '12, pages 1617–1621, 2012.

[10] spammers using behavioral Footprints A. Mukherjee, A.Kumar, B. Liu, J. Wang, M. Hsu, M. Castellanos,and R.Ghosh. In Proceedings of the 19th ACM SIGKDDinternational conference on Knowledge discovery anddata mining, KDD '13, 2013.

[11] Detecting product review spammers using ratingbehaviors. E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu,and H. W. Lauw In Proceedings of the 19th ACMinternational conference on Inform

Author's Details:

Mrs.K.Sangeeta, Assistant Professor, Department of CSE, at Pydah College Of Engineering & Technology, Visakhpatnam, Andhra Pradesh, India. Her research interests are in the areas of data mining, network security, artificial intelligence.

Mr.Dr.Ramesh Challagundla, Received his M.S.C degree in phy.electronic from meerut university which is recognized as equivalent to B.E. with specialization in applied/power electronics from Gulbarga university in 1991,was granted A.M.I.E. in 1997 and ph.d from Andhra university college of Visakhapatnam. He joined as service engineer in Hast Alloy castings Ltd in the year 1990. After servicing a year and half, he switched over to teaching and served as lecturer in R.E.C. Affiliated to Gulbarga university during 1992-1993. He joined EEE department ,GITAM,Visakhapatnam and served as lecturer during 1993-96.During 1996-97 he served as lecturer in Bhilai institute of technology, during 1997-98 served as lecturer in birla institute of technology, mesra, ranchi, during 1998-2001 served as assistant professor in GITAM college of engineering Visakhapatnam and from 2001 onwords with ANITS.currently working as professor and principal at Pydah college of engineering and technology, Gambheeram, Visakhapatnam. Ratified as professor by the expert committee of Andhra university in the field of ECE constituted by vice chancellor,who himself was the chairman for the selection committee.

Srinivasu Gubbala pursuing him 2 years m.tech in department of CSE at pydah college of engineering &technology, visakhpatnam,Andhra Pradesh,India .