International Journal of Contemporary Research in Computer Science and Technology (IJCRCST) e-ISSN: 2395-5325
Volume 2, Issue 3 (March'2016)

RANKING FRAUD DETECTION SYSTEM FOR MOBILE APPS

Dr. Soundararajan
Vice Principal,
Department of Computer Science And Engineering,
Velammal Institute of Technology, Chennai, India

Janaat.V
Student,
Department of Computer Science And Engineering,
Velammal Institute of Technology, Chennai, India

Pothireddy Akhila
Student,
Department of Computer Science And Engineering,
Velammal Institute of Technology, Chennai, India

Keerthana.M
Student,
Department of Computer Science And Engineering,
Velammal Institute of Technology, Chennai, India

Abstract: Ranking fraud in the mobile app market are fraudulent or deceptive activities that have a purpose of bumping up the apps in the popularity list. The existing app stores pose drawbacks such as inability to provide the number of users for an app. Also viewing and downloading an app by the same user increases its hit rate. This makes the app developers to easily commit ranking fraud. While the importance of preventing ranking fraud has been widely recognized, there is limited understanding and research in this area. To end this, in this paper, we provide a fraud discovery system for mobile apps that can overcome the shortcomings of the existing system. This system aims at reducing the fraud ranking that happens by allowing the users to download the app using a secret key provided to them. It provides a way to track the user who involved in the fraud activity and thus makes the admin to maintain a list of unique users for an app. Thus it can make the admin of the app store to identify and award the original apps owners alone. Also through this system the users can be directed to download the genuinely ranked apps.

Keywords: Ranking Fraud, App Rating, Mobile App, Leaderboards, App Store,Hit rate.

I. INTRODUCTION

The number of mobile Apps has grown at a breathtaking rate over the past few years. For example, at the end of July 2015, there were more than 1.6 million Apps at Google Play and 1.5 million Apps at Apple's app store. To stimulate the development of mobile Apps, many App stores launched daily App Leaderboards, which demonstrate the chart rankings of most popular Apps. Indeed, the App Leaderboard is one of the most important ways for promoting mobile App. A higher rank on the Leaderboard usually leads to a huge number of downloads and million dollars in revenue to the app developers. Therefore, App developers tend to explore various ways such as advertising campaigns to promote their Apps in order to have their Apps ranked as high as possible in such App Leaderboards. However, as a recent trend, instead of relying on traditional marketing solutions, shady App developers resort to some fraudulent means to deliberately boost their Apps and eventually manipulate the chart rankings on an App store. Further the existing app stores can increase the hit rate of an app even when the user views the app and there is no means in them to identify the unique users of an app. This disadvantage is used by the app developers to fraudulently increase the app ranking from the user side. For example, an article from VentureBeat [2] reported that, when an App was promoted with the help of ranking manipulation, it could be propelled from number 1,800 to the top 25 in Apple's top free Leaderboard and more than 50,000-100,000 new users could be acquired within a couple of days. In fact, such ranking fraud raises great concerns to the mobile App industry. For example, Apple has warned of cracking down on App developers who commit ranking fraud [1] in the Apple's

App store. In the literature, while there are some related work, such as web ranking spam detection [4], [6], [11], online review spam detection [3], [7], [8], and mobile App recommendation [5], [9], [10], the problem of detecting ranking fraud for mobile Apps is still under-explored. To fill this crucial void, in this paper, we propose to develop a ranking fraud detection system for mobile. This system allows to increase the hit rate of an app only when the user downloads the app and not when the app is viewed. To avoid the hit rate increased by a single user's download, this system implements the Fraud Ranking Blocker which restricts the maximum number of app downloads by a single user to five. This is achieved by the Random Key Generator which assigns a secrete key to each individual user using the app store. The user after registering into the store can use the secrete key to download the apps. Also this has the Humming Bird algorithm implemented that can fetch the app that exactly matches the user's request in spite of the apps that were ranked by fraudulent means. Fig. 1 shows the architecture of our ranking fraud detection system for mobile Apps. Finally, we evaluate the proposed system with real-world App uploading from developer side and trying to download the app for more than five times from the user side. Experimental results show the effectiveness of the proposed system by blocking the download of an app by a user that exceeds fifth time. The remainder of this paper is organized as follows. In Section 2, we introduce how the admin module has been implemented and how a secrete key is generated for each user. Section 3 presents how the user side is implemented with Fraud Ranking Blocker and Session Tracker. In Section 4 we do discussions about the app developer side implementation of the system. In Section 5 we make some further discussion about the proposed approach. In Section 6, we report the experimental results of real-world application. Section 7 provides a brief review of related works. Finally, in Section 8, we conclude the paper and propose some future research directions.

IJCRCST © 2016 | All Rights Reserved

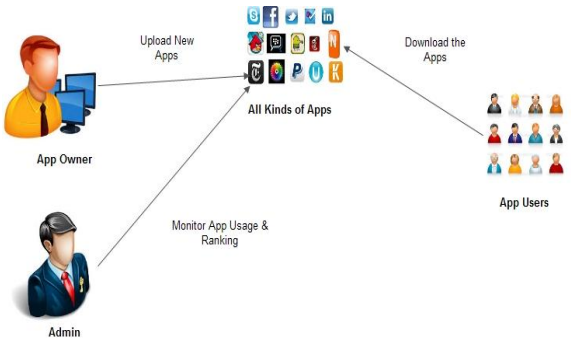

Figure 1: Architecture of our ranking fraud detection system for mobile Apps

II. ADMIN SIDE IMPLEMENTATION OF THE SYSTEM

Admin maintains the storage space status like app ratings, approved users and sending secret key to the users. Every time a new user registers in the store by giving his/her details, the Admin allots a secrete key for that user. The user can login into his/her account using that secrete key alone. This secrete key is unique for each user and is generated using the Random Key Generating algorithm shown below. The Admin also maintains the app ranking and status in the storage space. The admin will receive the user details from the Fake Ranking Blocker and will be able to maintain a unique user list for a particular app. Thus the Admin can provide the genuine app details such as rating and ranking by knowing the number of unique users for an app. This directs the users to opt for an app that has been ranked originally.

Random Algorithm Pseudo Code

A binary sequence (BS) is a sequence $a_0, \ldots, a_{N-1}$ of N bits, i.e.

$$a_j \in \{0,1\} \text{ for } j = 0, 1, \ldots, N-1.$$

A BS consists of $m = \sum a_j$ ones and $N - m$ zeroes.

A BS is a pseudo random binary sequence (PRBS) if its autocorrelation function, given by

$$C(v) = \sum_{j=0}^{N-1} a_j a_{j+v}$$

has only two values:

$$C(v) = \begin{cases} m, & \text{if } v \equiv 0 \pmod{N} \\ mc, & \text{otherwise} \end{cases}$$

where

$$c = \frac{m-1}{N-1}$$

is called the *duty cycle* of the PRBS, similar to the duty cycle of a continuous time signal. For a maximum length sequence, where $N = 2^k - 1$, the duty cycle is 1/2.

III. USER SIDE IMPLEMENTATION OF THE SYSTEM

Users can download apps for Android, ios, Windows and various other kinds of mobile phones. When the user searches for a particular app, the searching is done and the most apt and high ranked app is shown in the result page. This is done by implementing Humming Bird algorithm in the searching function. This algorithm is shown in fig.2. He can view details about an app and can download it. To prevent the fraud ranking it's defined that the user can download an app only for five times. When he tries to download the app for the sixth time our internal architecture will block the user's download and sends the user's system configuration details and the details of the app he is trying to download to the admin. This function is carried out by the Fake Rank Blocker in the system. Now admin can easily know how many users are there for an app. Also, in the existing systems, the hit rate gets changed even when the user views the app. But this system makes sure that the hit rate is affected only when an app is downloaded. The Session Tracking algorithm is implemented to track the user activities in the website.

IV.APP OWNER SIDE

Do not include headers, footers or page numbers other than as already found in this manuscript. Please note that the headers, footers or page numbers are different for the first page, and the rest of the even and odd pages. Actual page numbers and other running heads will be modified when the publications are assembled.

V.FURTHER DISCUSSION ABOUT THE PROPOSED APPROACH

The proposed system overcomes the disadvantages in the existing system by having the following advantages. The first advantage is that the user will be able to increase an app hit rate by downloading the app alone. So there is possibility for the app developer to make a particular user to download their app in order to increase the hit rate of the app. So the system restricts the maximum number of times a user can download an app to five. If the user attempts to download the app for the sixth time the user is suspected to have illegally increased the hit rate of the app. The user details and his/her system configuration are sent to the admin along with the app details he/she is trying to download. This is the second advantage. The users are allowed to login to their account using the secrete key allotted to them by the admin. Therefore with a secrete key only five downloads can be done for an app. The admin can now maintain a list of unique users and knows exact the number of users for an app which is the third advantage. When a user account is suspected for the practice of fraud activity, the system configuration details of that user are sent to the admin thus the user can be tracked by the admin. This is the fourth advantage. The ranking details provided by the admin can guide the users to download a genuine app and this becomes the fifth advantage. The sixth advantage is that this system has been implemented as a website. And so it requires a single team for its maintenance rather than separate teams for each kind of mobile apps. Thus it reduces the cost spent for the maintenance team. This becomes the seventh advantage of the system.Also the presence of storage space in this system eliminates the need for secondary memory in phone.

IJCRCST © 2016 | All Rights Reserved

```
for i : 1 . . . N do {Initialization}
    R[i] ← (1 − α)/((α^T +1   )N) {Normalization}
    if T≥ 0 then                  {Normal PageRank or Truncated?}
        Score[i] ← 0
    else {Calculate normal PageRank}
        Score[i] ← R[i]
    end if
end for
distance = 1
while not converged do
    Aux ← 0
    for src : 1 . . . N do {Follow links in the graph}
        for all link from src to dest do
            Aux[dest] ← Aux[dest] + R[src]/outdegree(src)
        end for
    end for
    for i : 1 . . . N do {Apply damping factor α}
        R[i] ← Aux[i] ×α
        if distance > T then {Add to ranking value}
            Score[i] ← Score[i] + R[i]
        end if
    end for
    distance = distance +1
end while
return Score
```

Figure 2: Humming Bird algorithm

VI. A BRIEF REVIEW OF THE RELATED WORKS

The first category of the related works is about web ranking spam detection. Specifically, the web ranking spam refers to any deliberate actions which bring to selected web pages an unjustifiable favorable relevance or importance. For example, Ntoulas et al. have studied various aspects of content-based spam on the web and presented a number of heuristic methods for detecting content based spam. Zhou et al. have studied the problem of unsupervised web ranking spam detection. Specifically, they proposed an efficient online link spam and term spam detection methods using spamicity. The second category is focused on detecting online review spam. For example, Lim et al .have identified several representative behaviors of review spammers and model these behaviors to detect the spammers. Wu et al. have studied the problem of detecting hybrid shilling attacks on rating data. Xie et al. have studied the problem of singleton review spam detection. Also, to solve the scarcity problem of App usage records, Shi and Ali studied several recommendation models and proposed content based collaborative filtering model, named Eigenapp, for recommending Apps in their website Get jar. In addition, some researchers studied the problem of exploiting enriched contextual information for mobile App recommendation. For example, Zhu et al. proposed a uniform framework for personalized context-aware recommendation, which can integrate both context independency and dependency assumptions. However, to the best of our knowledge, none of previous works has studied the problem of ranking fraud detection for mobile Apps.

VII.EXPERIMENTAL RESEARCH

The experimental results show the effectiveness of the system. Fig 3 shows the screenshot of the page where a user is blocked while attempting to download Whatsapp app for the fifth time. Fig 4 shows the admin page where the fraud evidences are provided to the admin.

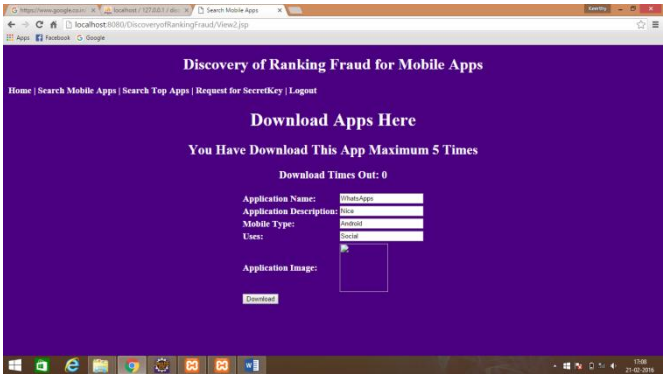

Figure 3: user is blocked

Figure 4: fraud evidences are provided to the admin.

VII.CONCLUSION AND FUTURE RESEARCH DIRECTIONS

In this paper, we developed a ranking fraud detection system for mobile Apps. We developed the system such that it helps in eliminating the ranking fraud that happens in the app stores by taking advantage of the shortcomings of the existing system. It also provides a way to track the user who involved in ranking fraud and makes the admin to know the exact numbers of users for an app. Thus only the genuine app developers get rewarded by the admin and the users can get to know about the real app ranking. Experimental results showed the effectiveness of the proposed approach. Moreover, we will extend our ranking fraud detection approach with other mobile App related services, such as mobile Apps recommendation, for enhancing user experience.

VIII.REFERENCES

[1](2012).[Online].Available:https://developer.apple.com/news/index.php?id=02062012a

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

[3] 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.

[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]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.

[6] 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.

[7] 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.

[8] 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.

[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] H. Zhu, H. Cao, E. Chen, H. Xiong, and J. Tian, "Exploiting enriched contextual information for mobile app classification," in Proc. 21stACMInt. Conf. Inform. Knowl. Manage., 2012, pp. 1617–1621.

IJCRCST
Innovative of current researches