International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 11
July 2016

Detection of Fake Grade in Mobile Application Environment

Mr. K.VINAY KUMAR REDDY
Associate Professor
Department of CSE

Ms .C B SUSHMA
M.Tech in Computer Science
Department of CSE

Malla Reddy Institute of Technology, Maisammaguda, Dhulapally Post, Secunderabad and Telengana State, India.

Abstract: Ranking fraud within the mobile App market refers to deceitful or deceptive activities that have a purpose of bumping up the Apps within the quality list. Indeed, it becomes additional and additional frequent for App developers to use shady means that, like inflating their Apps' sales or posting phony App ratings, to commit ranking fraud. Whereas the importance of preventing ranking fraud has been wide recognized, there's restricted understanding and analysis during this space. To the current finish, during this paper, we offer a holistic read of ranking fraud and propose a ranking fraud detection system for mobile Apps. Specifically, we have a tendency to initial propose to accurately find the ranking fraud by mining the active periods, particularly leading sessions, of mobile Apps. Such leading sessions may be leveraged for police work the native anomaly rather than world anomaly of App rankings. Moreover, we have a tendency to investigate 3 varieties of evidences, i.e., ranking primarily based evidences, rating {based| based mostly| primarily primarily based} evidences and review based evidences, by modeling Apps' ranking, rating and review behaviors through statistical hypotheses tests. Additionally, we have a tendency to propose associate improvement primarily based aggregation technique to integrate all the evidences for fraud detection. Finally, we have a tendency to evaluate the projected system with real-world App information collected from the iOS App Store for an extended fundamental quantity. In the experiments, we have a tendency to validate the effectiveness of the projected system, and show the quantify-ability of the detection rule furthermore as some regularity of ranking fraud activities.

Index Terms: Mobile Apps, ranking fraud detection, proof aggregation, historical ranking records, rating and review.

INTRODUCTION:

The quantity of mobile apps has grown at a breathtaking Price over the last few years. As an instance, as of the give up of april 2013, there are greater than 1.6 million apps at Apple's app save and google play. To stimulate the development of cellular apps, many app stores launched every day App leader-boards, which show the chart scores of Maximum popular apps. Certainly, the app leader-board is one among the most

important ways for selling mobile apps. A Higher rank on the leader-board typically ends in a massive Range of downloads and million bucks in revenue. Therefore, app developers generally tend to explore diverse ways including advertising campaigns to promote their apps in order to have their apps ranked as excessive as possible in such app leader-boards. However, as a latest trend, rather than counting on traditional Advertising and marketing solutions, shady app developers' motel to a few fraudulent methods to intentionally increase their apps And finally manage the chart scores on an app store. This is usually implemented with the aid of the usage of so-referred to as "bot Farms" or "human water armies" to inflate the app downloads, Ratings and reviews in a very quick time. For example, A piece of writing from venture-beat said that, when an app turned into promoted with the help of ranking manipulation, it can be propelled from number one,800 to the top 25 in Apple's pinnacle free leader-board and more than 50,000-100,000 new users will be received inside multiple days. In Truth, such ranking fraud raises extraordinary concerns to the cellular App enterprise. As an example, apple has warned of cracking down on app builders who dedicate rating fraud in the apple's app keep. Within the literature, even as there are few associated works, which include internet ranking junk mail detection online overview junk mail detection and cellular app recommendation the problem of detecting ranking fraud for cell apps is still underneath-explored. To fill this important void, in this paper, we advise to increase a ranking fraud Detection system for cell apps. Alongside this line, we pick out several important challenges. First, ranking fraud does no longer usually manifest inside the entire existence cycle of an app, so we need to stumble on the time when fraud occurs. Such task can be appeared as detecting the nearby anomaly in place of global anomaly of cell apps. 2nd, because of the massive quantity of mobile apps, it miles tough to manually label rating fraud for each app, so it's miles essential to have a scalable way too routinely Come across ranking fraud without using any benchmark Information.

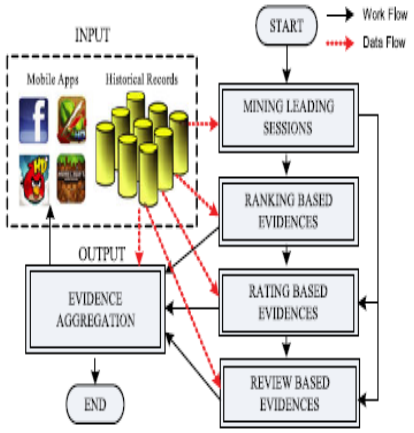


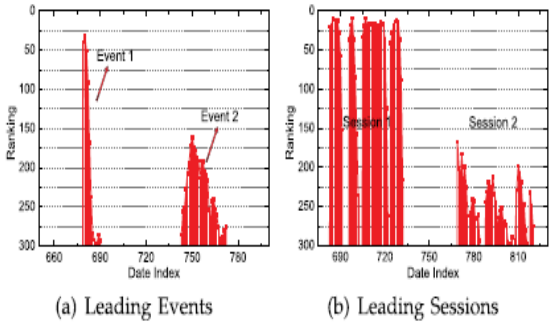
Fig. 1. The framework of our ranking fraud detection system for mobile Apps.

Finally, due to the dynamic nature of chart ratings, it isn't easy to pick out and verify the evidences linked to ranking fraud, which motivates us to find out a few implicit fraud patterns of cellular apps as evidences. Indeed, our cautious remark well-known shows that mobile apps aren't constantly ranked high within the leader-board, however most effective in a few main occasions, which shape exceptional leading periods.

Note that we are able to introduce both main activities and main Periods in element later. In different phrases, ranking fraud normally happens in those leading classes. Consequently, detecting rating fraud of cellular apps is certainly to come across ranking Fraud within main classes of cell apps. Especially, we first advocate a simple yet powerful algorithm to pick out the main sessions of every app based on its historic ranking records. Then, with the analysis of Apps' ranking behaviors, we discover that the fraudulent Apps often have one-of-a-kind rating styles in every main session compared with normal Apps. Therefore, we symbolize a few fraud evidences from Apps' ancient ranking statistics, and expand three features to extract such ranking based fraud evidences. Nevertheless, the ranking based evidences may be suffering from App builders' reputation and a few valid advertising campaigns, inclusive of "restrained-time bargain". As an end result, it isn't always enough to most effective use ranking based totally evidences. Consequently, we similarly suggest two sorts of fraud evidences primarily based on Apps' score and evaluate history, which replicate a few anomaly patterns from Apps' ancient score and review records. Further, we develop an unmonitored proof-aggregation technique to integrate these 3 varieties of evidences for comparing the credibility of leading classes from cellular Apps. Fig. 1 indicates the framework of our ranking fraud detection device for cell Apps. It's far worth noting that everyone the evidences are extracted by way of modeling Apps' rating, rating and evaluate behaviors thru statistical hypotheses assessments. The proposed framework is scalable and can be extended with other domain generated evidences for ranking fraud detection. In the end, we evaluate the proposed machine with actual-world App data accrued from the Apple's App keep for long time duration, i.e., greater than years. Experimental consequences display the effectiveness of the proposed system, the scalability of the detection algorithm in addition to a few regularity of ranking fraud sports evaluation. The remainder of this paper is organized as follows. In segment 2, we introduce some preliminaries and the way to mine leading periods for mobile Apps. Segment three affords how to extract rating; rating and overview based evidences and integrate them for rating fraud detection. In section 4 we make a few in addition discussion approximately the proposed method. In segment 5, we file the experimental consequences on two long-term actual-world statistics units. Section 6 provides a short review of related works. Subsequently, in segment 7, we finish the paper and advocate a few future studies guidelines.

IDENTIFYING LEADING SESSIONS FOR MOBILE APPS:

In this section, we first introduce some preliminaries, and then show how to mine leading sessions for mobile Apps from their historical ranking records.



(a) Leading Events (b) Leading Sessions

Fig. 2. (a) Example of leading events; (b) Example of leading sessions of mobile Apps.

Preliminaries: The App leader-board demonstrates top K popular Apps with respect to different categories, such as "Top Free Apps" and "Top Paid Apps". Moreover, the leader-board is usually updated periodically (e.g., daily).

Definition 1 (Leading Event). *Given a ranking threshold $K^* \in [1, K]$, a leading event* e *of App* a *contains a time range $T_e = [t^e_{start}, t^e_{end}]$ and corresponding rankings of* a*, which satisfies $r^a_{start} \leq K^* < r^a_{start-1}$, and $r^a_{end} \leq K^* < r^a_{end+1}$. Moreover, $\forall t_k \in (t^e_{start}, t^e_{end})$, we have $r^a_k \leq K^*$.*

Notice that we observe a ranking threshold k that is generally Smaller than k right here due to the fact ok can be very massive (e.G., extra than 1,000), and the ranking data beyond k (e.g 300) are not very useful for detecting the ranking manipulations. Moreover, we additionally discover that a few apps have several adjoining main activities which are near every different and Shape a main session. As an instance, fig. 2b suggests an instance of adjoining main events of a given cellular app, which shape main sessions. Specifically, a leading event which does not produce other close by buddies also can be dealt with as a special main session. The formal definition of main session is as follows:

Definition 2 (Leading Session). *A leading session* s *of App* a *contains a time range $T_s = [t^s_{start}, t^s_{end}]$ and n adjacent leading events $\{e_1, \ldots, e_n\}$, which satisfies $t^s_{start} = t^{e_1}_{start}$, $t^s_{end} = t^{e_n}_{end}$ and there is no other leading session s^* that makes $T_s \subseteq T_{s^*}$. Meanwhile, $\forall i \in [1, n)$, we have $(t^{e_{i+1}}_{start} - t^{e_i}_{end}) < \phi$, where ϕ is a predefined time threshold for merging leading events.*

Mining Leading Sessions: There are two important steps for mining main classes. First, we need to discover main occasions from the App's historic ranking records. 2d, we need to merge adjacent main activities for constructing leading classes. Specially, set of rules 1 demonstrates the pseudo code of mining leading classes 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 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

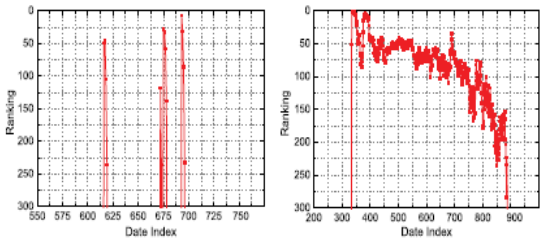
3. EXTRACTING THE EVIDENCES FOR RANKING FRAUD DETECTION:

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 11
July 2016

In this section, we study how to extract and combine fraud evidences for ranking fraud detection.

Ranking Based Evidences: In keeping with the definitions added in section 2, a leading Consultation consists of numerous main occasions. Therefore, we have to first examine the primary characteristics of leading activities for extracting fraud evidences via analyzing the apps' ancient ranking records, we look at that apps' ranking behaviors in a main event continually satisfy a selected ranking pattern, which includes 3 distinct ranking levels, particularly, rising section, preserving Phase and recession segment. Mainly, in every main Occasion, an app's ranking first will increase to a height function in the leader-board (i.e., rising segment), then maintains such top function for a length (i.e., preserving section), and subsequently decreases until the stop of the occasion (i.e., recession segment). Fig. 3 suggests an example of various rating stages of a main occasion. Indeed, any such ranking pattern shows a critical Knowledge of leading occasion. Inside the following, we officially define the 3 ranking levels of a leading event.

Definition 3 (Ranking Phases of a Leading Event). *Given a leading event e of App a with time range* $[t^e_{start}, t^e_{end}]$, *where the highest ranking position of a is* r^a_{peak}, *which belongs to* ΔR. *The* ***rising phase*** *of e is a time range* $[t^e_a, t^e_b]$, *where* $t^e_a = t^e_{start}$, $r^a_b \in \Delta R$ *and* $\forall t_i \in [t^e_a, t^e_b)$ *satisfies* $r^a_i \notin \Delta R$. *The* ***maintaining phase*** *of e is a time range* $[t^e_b, t^e_c]$, *where* $r^a_c \in \Delta R$ *and* $\forall t_i \in (t^e_c, t^e_{end}]$ *satisfies* $r^a_i \notin \Delta R$. *The* ***recession phase*** *is a time range* $[t^e_c, t^e_d]$, *where* $t^e_d = t^e_{end}$.



(a) Example 1 (b) Example 2

Fig. 4. Two real-world examples of leading events.

- EVIDENCE 1. As shown in Fig. 3, we use two shape parameters u1 and u2 to quantify the ranking patterns of the rising phase and the recession phase of App a's leading event e, which can be computed by

$$\theta^e_1 = \arctan\left(\frac{K^* - r^a_b}{t^e_b - t^e_a}\right), \theta^e_2 = \arctan\left(\frac{K^* - r^a_c}{t^e_d - t^e_c}\right), \quad (1)$$

Therefore, a leading session, which has more leading events with large u1 and u2 values, has higher probability of having ranking fraud. Here, we define a fraud signature us for a leading session as follows:

$$\bar{\theta}_s = \frac{1}{|E_s|}\sum_{e \in s}(\theta^e_1 + \theta^e_2), \quad (2)$$

▷ HYPOTHESIS 0: *The signature* $\bar{\theta}_s$ *of leading session s is not useful for detecting ranking fraud.*

▷ HYPOTHESIS 1: *The signature* $\bar{\theta}_s$ *of leading session s is significantly greater than expectation.*

Then, we can calculate the p-value by

$$\mathbb{P}(\mathcal{N}(\mu_{\bar{\theta}}, \sigma_{\bar{\theta}}) \geq \bar{\theta}_s) = 1 - \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{\bar{\theta}_s - \mu_{\bar{\theta}}}{\sigma_{\bar{\theta}}\sqrt{2}}\right)\right), \quad (3)$$

where erf(x) is the Gaussian Error Function as follows:

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 11
July 2016

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt. \quad (4)$$

Intuitively, a leading session with a smaller p-value P has more chance to reject Hypothesis 0 and accept Hypothesis 1. This means it has more chance of committing ranking fraud. Thus, we define the evidence as

$$\Psi_1(s) = 1 - \mathbb{P}\left(\mathcal{N}\left(\mu_{\bar{\theta}}, \sigma_{\bar{\theta}}\right) \geq \bar{\theta}_s\right). \quad (5)$$

- EVIDENCE 2: As discussed above, the Apps with ranking fraud often have a short maintaining phase with high ranking positions in each leading event.

$$\chi_s = \frac{1}{|E_s|} \sum_{e \in s} \frac{K^* - \bar{r}_m^e}{\Delta t_m^e}, \quad (6)$$

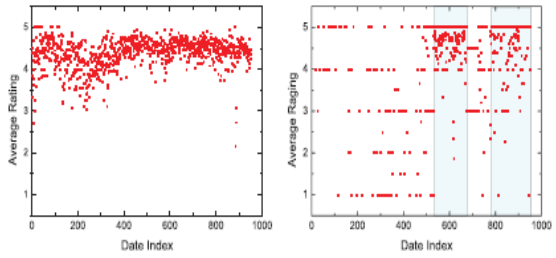
where K^* is the ranking threshold in Definition 1.

▷ HYPOTHESIS 0: *The signature χ_s of leading session s is not useful for detecting ranking fraud.*
▷ HYPOTHESIS 1: *The signature χ_s of leading session s is significantly higher than expectation.*

$$\Psi_2(s) = 1 - \mathbb{P}\left(\mathcal{N}(\mu_\chi, \sigma_\chi) \geq \chi_s\right). \quad (7)$$

- EVIDENCE 3. The number of leading events in a leading session, i.e., |Es|, is also a strong signature of ranking fraud. For a normal App, the recession phase indicates the fading of popularity. Therefore, after the end of a leading event, it is unlikely to appear another leading event in a short time unless the App updates its version or carries out some sales promotion. Therefore, if a leading session contains much more leading events compared with other leading sessions of Apps in the leader-board, it has high probability of having ranking fraud. To capture this, we define two statistical hypotheses to compute the significance of |Es| for each leading session as follows:

▷ HYPOTHESIS 0: *The signature $|E_s|$ of leading session s is not useful for detecting ranking fraud.*
▷ HYPOTHESIS 1: *The signature $|E_s|$ of leading session s is significantly larger than expectation.*



(a) Example 1 (b) Example 2

Fig. 5. Two real-world examples of the distribution of Apps' daily average ratings.

Since |Es| always has discrete values, we propose to leverage the Poisson approximation to calculate the p-value with the above hypotheses.

$$\mathbb{P}\left(\mathcal{P}(\lambda_s) \geq |E_s|\right) = 1 - e^{-\lambda_s} \sum_{i=0}^{|E_s|} \frac{(\lambda_s)^i}{i!}. \quad (8)$$

Therefore, we can compute the evidence by

$$\Psi_3(s) = 1 - \mathbb{P}\left(\mathcal{P}(\lambda_s) \geq |E_s|\right). \quad (9)$$

score based totally Evidences: The ranking based totally evidences are useful for

ranking fraud detection. But, once in a while, it isn't sufficient to best use ranking based evidences. as an example, some Apps created by using the famous developers, inclusive of Gameloft, may additionally have some main events with big values of u1 because of the builders' credibility and the "word-of-mouth" advertising effect. Moreover, a number of the legal advertising offerings, together with "constrained-time discount", may additionally result in great ranking primarily based evidences. To solve this issue, we additionally study how to extract fraud evidences from Apps' ancient score information.

- EVIDENCE 4. For a normal App, the average rating in a specific leading session should be consistent with the average value of all historical ratings. In contrast, an App with rating manipulation might have surprisingly high ratings in the fraudulent leading sessions with respect to its historical ratings.

$$\Delta\mathcal{R}_s = \frac{\overline{\mathcal{R}}_s - \overline{\mathcal{R}}_a}{\overline{\mathcal{R}}_a}, \quad (s \in a), \tag{10}$$

▷ HYPOTHESIS 0: *The signature $\Delta\mathcal{R}_s$ of leading session s is not useful for detecting ranking fraud.*
▷ HYPOTHESIS 1: *The signature $\Delta\mathcal{R}_s$ of leading session s is significantly higher than expectation.*

Here, we use the Gaussian approximation to calculate the p-value with the above hypotheses:

$$\Psi_4(s) = 1 - \mathbb{P}\left(\mathcal{N}(\mu_{\mathcal{R}}, \sigma_{\mathcal{R}}) \geq \Delta\mathcal{R}_s\right). \tag{11}$$

- EVIDENCE 5. In the App rating records, each rating can be categorized into |L| discrete rating levels, e.g., 1 to 5, which represent the user preferences of an App.

$$\mathcal{D}(s) = \frac{\sum_{i=1}^{|L|} P(l_i|\mathcal{R}_{s,a}) \times P(l_i|\mathcal{R}_a)}{\sqrt{\sum_{i=1}^{|L|} P(l_i|\mathcal{R}_{s,a})^2} \times \sqrt{\sum_{i=1}^{|L|} P(l_i|\mathcal{R}_a)^2}}. \tag{12}$$

▷ HYPOTHESIS 0: *The signature $\mathcal{D}(s)$ of leading session s is not useful for detecting ranking fraud.*
▷ HYPOTHESIS 1: *The signature $\mathcal{D}(s)$ of leading session s is significantly lower than expectation.*

$$\Psi_5(s) = 1 - \mathbb{P}\left(\mathcal{N}(\mu_{\mathcal{D}}, \sigma_{\mathcal{D}}) \leq \mathcal{D}(s)\right). \tag{13}$$

Review Based Evidences: Besides ratings, most of the App stores also allow users to write some textual comments as App reviews. Such reviews can reflect the personal perceptions and usage experiences of existing users for particular mobile Apps. Indeed, review manipulation is one of the most important perspectives of App ranking fraud. Specifically, before downloading or purchasing a new mobile App, users often first read its historical reviews to ease their decision making, and a mobile App contains more positive reviews may attract more users to download.

- EVIDENCE 6. Indeed, most of the review manipulations are implemented by bot farms due to the high cost of human resource. Therefore, review spammers often post

multiple duplicate or near-duplicate reviews on the same App to inflate downloads. In contrast, the normal App always have diversified reviews since users have different personal perceptions and usage experiences.

$$Sim(s) = \frac{2 \times \sum_{1 \le i < j \le N_s} Cos(\overrightarrow{w_{c_i}}, \overrightarrow{w_{c_j}})}{N_s \times (N_s - 1)},$$

Where Ns is the number of reviews during leading sessions.

▷ HYPOTHESIS 0: *The signature $Sim(s)$ of leading session s is not useful for detecting ranking fraud.*
▷ HYPOTHESIS 1: *The signature $Sim(s)$ of leading session s is significantly higher than expectation.*

$$\Psi_6(s) = 1 - \mathbb{P}\left(\mathcal{N}(\mu_{Sim}, \sigma_{Sim}) \ge Sim(s)\right). \quad (15)$$

 EVIDENCE 7. From the real-world observations, we find that each review c is always associated with a specific latent topic z. For example, some reviews may be related to the latent topic "worth to play" while some may be related to the latent topic "very boring". Meanwhile, since different users have different personal preferences of mobile Apps, each App a may have different topic distributions in their historical review records.

$$\mathcal{D}_{KL}(s\|a) = \sum_k P(z_k|\mathcal{C}_{s,a}) ln \frac{P(z_k|\mathcal{C}_{s,a})}{P(z_k|\mathcal{C}_a)}, \quad (16)$$

▷ HYPOTHESIS 0: *The signature $\mathcal{D}_{KL}(s\|a)$ of leading session s is not useful for detecting ranking fraud.*
▷ HYPOTHESIS 1: *The signature $\mathcal{D}_{KL}(s\|a)$ of leading session s is significantly higher than expectation.*

$$\Psi_7(s) = 1 - \mathbb{P}\left(\mathcal{N}(\mu_{KL}, \sigma_{KL}) \ge \mathcal{D}_{KL}(s\|a)\right). \quad (17)$$

Evidence Aggregation:

After extracting 3 sorts of fraud evidences, the subsequent mission is the way to combine them for rating fraud detection. Certainly, there are many ranking and evidence aggregation methods inside the literature, including permutation based fashions rating based fashions and Dempster-Shafer policies. But, a number of those techniques focus on learning an international ranking for all candidates. This isn't right for detecting rating fraud for brand new Apps. Different methods are based on supervised studying techniques, which rely upon the classified schooling information and are difficult to be exploited. Rather, we suggest an unmonitored technique primarily based on fraud similarity to mix these evidences.

$$\Psi^*(s) = \sum_{i=1}^{N_\Psi} w_i \times \Psi_i(s), \quad s.t. \sum_{i=1}^{N_\Psi} w_i = 1, \quad (18)$$

$$\sigma_i(s) = \left(\Psi_i(s) - \overline{\Psi}(s)\right)^2, \quad (19)$$

$$\mathcal{F}(a) = \sum_{s \in a} [\Psi^*(s) > \tau] \times \Psi^*(s) \times \Delta t^s, \quad (26)$$

$$s.t. \sum_{i=1}^{N_\Psi} w_i = 1; \forall w_i \geq 0. \quad (21)$$

We exploit the gradient based approach with exponentiated updating to solve this problem.

$$\nabla_i = \frac{\partial w_i \cdot \sigma_i(s)}{\partial w_i} = \sigma_i(s). \quad (22)$$

$$w_i = \frac{w_i^* \times \exp(-\lambda \nabla_i)}{\sum_{j=1}^{N_\Psi} w_j^* \times \exp(-\lambda \nabla_j)}, \quad (23)$$

However, sometimes only using evidence scores for evidence aggregation is not appropriate. It is because those different evidences may have different score range to evaluate leading sessions.

TABLE 1
Statistics of the Experimental Data

	Top Free 300	Top Paid 300
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

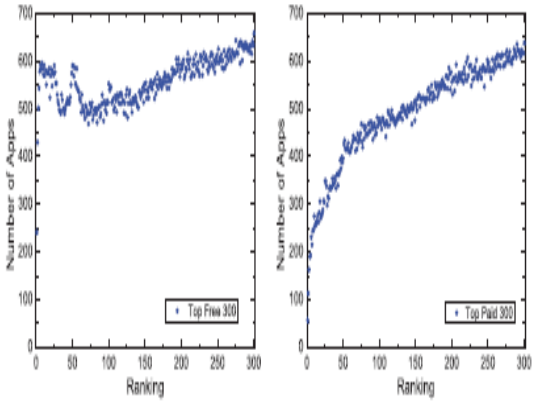
$$\overline{\pi}(s) = \frac{1}{N_\Psi} \sum_{i=1}^{N_\Psi} \pi_i(s). \quad (24)$$

$$\sigma_i^*(s) = \left(\pi_i(s) - \overline{\pi}(s)\right)^2. \quad (25)$$

$$\mathcal{F}(a) = \sum_{s \in a} [\Psi^*(s) > \tau] \times \Psi^*(s) \times \Delta t^s, \quad (26)$$

4. DISCUSSION:

Here, we provide a few dialogues approximately the proposed ranking Fraud detection device for cell apps. First, the down load records is an vital signature for detecting ranking fraud, considering that ranking manipulation is to use so-known as "bot farms" or "human water armies" to inflate the app downloads and scores in a totally brief time. However, the immediate down load data of each mobile app is regularly now not available for analysis. In reality, Apple and Google do now not provide correct download information on any app. Furthermore, the app developers themselves are also reluctant to launch their down load statistics for various motives. Therefore, in this paper, we specially attention on extracting evidences from apps' historic ranking, rating and evaluate facts for rating fraud detection. But, our approach is scalable for integrating different evidences if to be had, along with the evidences based on the down load facts and app developers' reputation.



(a) Top Free 300 data set (b) Top Paid 300 data set

Fig. 6. The distribution of the number of Apps w.r.t different rankings.

5. EXPERIMENTAL RESULTS:

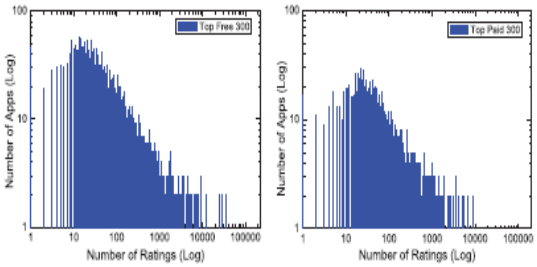
In this section, we evaluate the performances of ranking fraud detection using real-world App data.

The Experimental facts:

The experimental facts units have been gathered from the "pinnacle free three hundred" and "top Paid 300" leader-boards of Apple's App store (U.S.) from February 2, 2010 to September 17, 2012.

The information sets include the each day chart rankings1 of pinnacle three hundred loose Apps and top 300 paid Apps, respectively. Furthermore, every statistics set additionally contains the person scores and evaluation statistics. Table 1 indicates the particular records characteristics of our information sets. Figs. 6a and 6b display the distributions of the variety of Apps with admire to different rankings in these statistics sets.

Mining Leading Sessions:



(a) Top Free 300 data set (b) Top Paid 300 data set

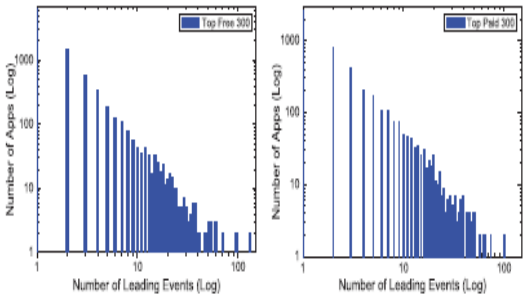
Fig. 7. The distribution of the number of Apps w.r.t different numbers of ratings.

Figs. 8 and 9 show the distributions of the number of Apps with respect to different numbers of contained leading events and leading sessions in both data sets. In these figures, we can see that only a few Apps have many leading events and leading sessions. The average numbers of leading events and leading sessions are 2:69 and 1:57 for free Apps, and 4:20 and 1:86 for paid Apps. Moreover, Figs. 10a and 10b show the distribution of the number of leading sessions with respect to different numbers of contained leading events in both data sets. In these figures, we can find only a few leading sessions contain many leading events.

Human Judgment Based Evaluation: To the best of our knowledge, there is no existing benchmark to decide which leading sessions or Apps really contain ranking fraud. Thus, we develop four intuitive baselines and invite five human evaluators to validate the effectiveness of our approach Evidence Aggregation based Ranking Fraud Detection (EA-RFD). Particularly, we denote our approach with score based aggregation (i.e., Principle 1) as EA-RFD-1, and our approach with rank based aggregation (i.e., Principle 2) as EA-RFD-2, respectively.

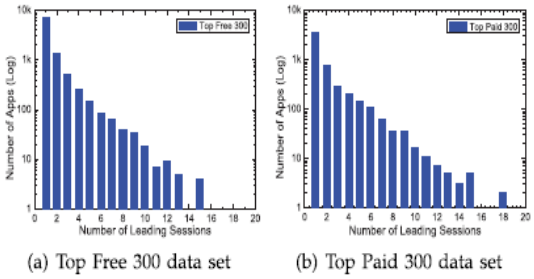
Baselines:

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 11
July 2016



(a) Top Free 300 data set (b) Top Paid 300 data set

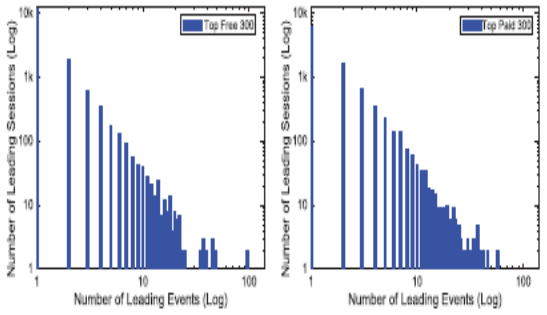
Fig. 8. The distribution of the number of Apps w.r.t different numbers of leading events.



(a) Top Free 300 data set (b) Top Paid 300 data set

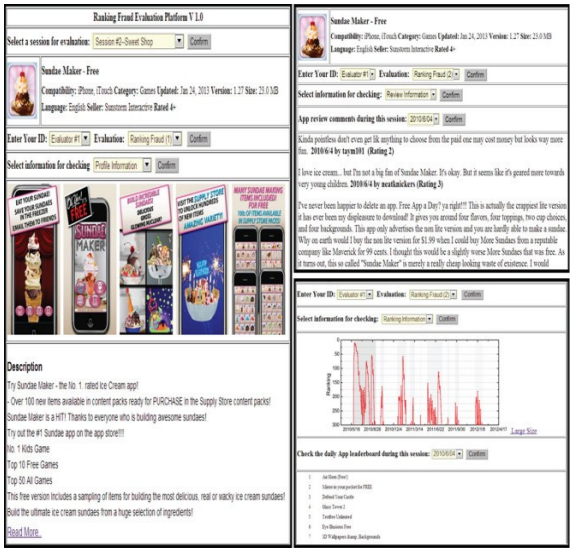
Fig. 9. The distribution of the number of Apps w.r.t different number of leading sessions.

The second baseline Rating-RFD stands for Rating evidence based ranking fraud detection, which estimates the ranking fraud for each leading session by only using rating based evidences (i.e., C4 and C5). These two evidences are integrated by our aggregation approach.



(a) Top Free 300 data set (b) Top Paid 300 data set

Fig. 10. The distribution of the number of leading sessions w.r.t different number of leading events.



The Experimental Setup: To study the performance of ranking fraud detection by each approach, we set up the evaluation as follows. First, for each approach, we selected 50 top ranked leading sessions (i.e., most suspicious sessions), 50 middle ranked leading sessions (i.e., most uncertain sessions), and 50 bottom ranked leading sessions (i.e., most normal sessions) from each data set. Then, we merged all the selected sessions into a pool which consists 587 unique sessions from 281 unique Apps

in "Top Free 300" data set, and 541 unique sessions from 213 unique Apps in "Top Paid 300" data set.

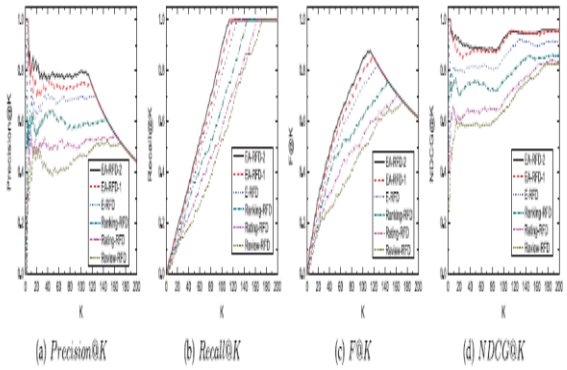


Fig. 12. The overall performance of each detection approach in Top Free 300 data set.

Overall Performances: Figs. 12 and 13 show the evaluation performance of each detection approach in two data sets. From these figures we can observe that the evaluation results in two data sets are consistent. It is because rating and review manipulations are only supplementary to ranking manipulation. Particularly, we observe that Review-RFD may not be able to lead to the good performance in terms of all evaluation metrics on the two data sets.

To further validate the experimental results, we also conduct a series of paired T-test of 0.95 confidence level which show that the improvements of our approach, i.e., EA-RFD- 2/EA-RFD-1, on all evaluation metrics with different K compared to other baselines are all statistically significant.

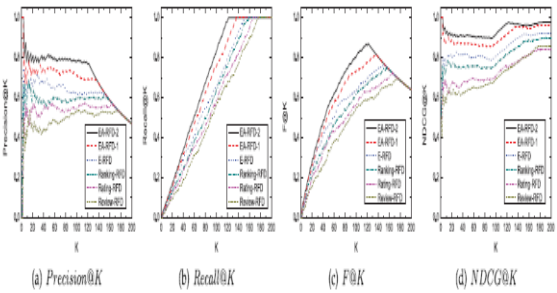


Fig. 13. The overall performance of each detection approach in Top Paid 300 data set.

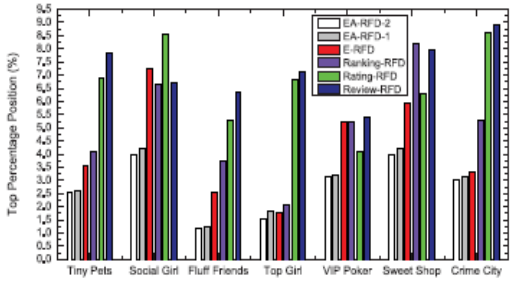


Fig. 14. Case study of reported suspicious mobile Apps.

Fig. 14 shows the top percent function of each App in the ranked list back by every technique. We are able to see that our approach, i.e., EA-RFD-2 and EA-RFD-1, can rank those suspicious Apps into higher positions than different baseline strategies. Further because the effects in segment 5.3.3, best leveraging single form of evidences for fraud detection cannot attain suitable performance, i.e., finding such suspicious Apps in excessive positions.

Efficiency and Robustness of our Approach: The computational cost of our approach mainly comes from the task of extracting three kinds of fraud evidences for the given leading sessions. Indeed, the main processes of this task can be calculated offline in advance.

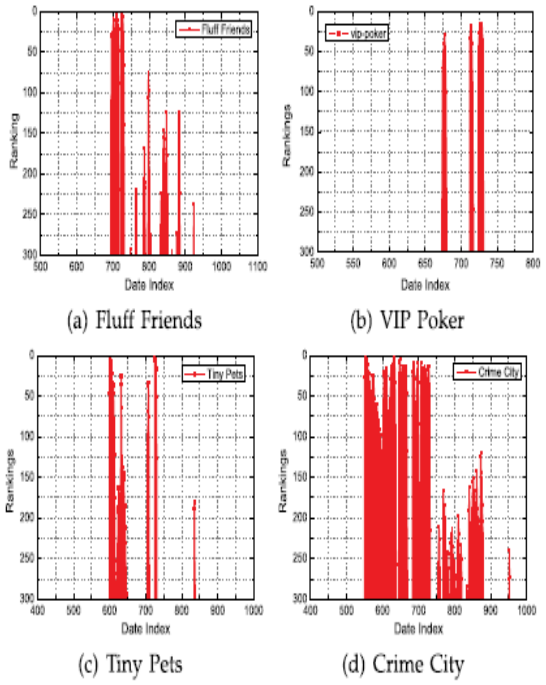


Fig. 15. The demonstration of the ranking records of four reported suspicious Apps.

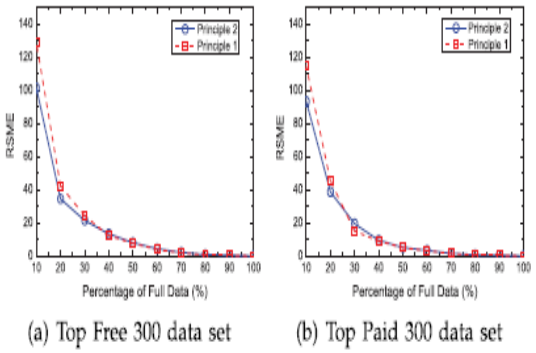


Fig. 16. The robustness test of our aggregation model with two principles.

Meanwhile, a getting to know process is required for proof aggregation. After getting to know the aggregation model on a historic statistics set, every new test App can reuse this version for detecting ranking fraud. However, it is miles still not clear how many gaining knowledge of data are required. To take a look at this problem and validate the robustness of our method, we first rank all main sessions by means of modeling with weight parameters learnt from the entire information set. Then we also rank all leading periods by using modeling with weight parameters learnt from special segmentation of the whole statistics set (i.e., 10; . . . ; 100 percentage).

6 RELATED WORKS:

Generally speaking, the associated works of this take a look at may be grouped into three classes. The primary class is ready net rating spam detection particularly; the internet ranking spam refers to any planned actions which carry to chose web pages an unjustifiable favorable relevance or importance. As an example, Ntoula et al. have studied diverse elements of content-primarily based spam at the net and presented a number of heuristic techniques for detecting content material based junk mail. Zhou et al. have studied the problem of unsupervised web ranking junk mail detection. Specifically, they proposed a green on line hyperlink spam and term unsolicited mail detection strategies the use of spamicity. Recently, Spirin and Han have reported a survey on net spam detection, which comprehensively introduces the ideas and algorithms in the literature. Certainly, the work of web rating junk mail detection is specifically primarily based at the analysis of ranking standards of engines like Google, consisting of Page Rank and query term frequency. that is extraordinary from ranking fraud detection for cell Apps.

7. CONCLUDING REMARKS:

In this project, 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 its historical 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] L. Azzopardi, M. Girolami, and K. V. Risjbergen, "Investigating the relationship between language model perplexity and ir precision- recall measures," in Proc. 26th Int. Conf. Res. Develop. Inform. Retrieval, 2003, pp. 369–370.

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

[3] 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.

[4] 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.

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

[6] 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.

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

[8] J. Kivinen and M. K. Warmuth, "Additive versus exponentiated gradient updates for linear prediction," in Proc. 27th Annu. ACM Symp. Theory Comput., 1995, pp. 209–218.

[9] 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.

Mr. K.VINAY KUMAR REDDY was born in India in the year of 1987. He received B.Tech degree in the year of 2008 & M.Tech PG in the year of 2011from JNTUH. He was expert in COMPUTER NETWORKS, DATABASE MANAGEMENT SYSTEMS, DATA STRUCTURES and C PROGRAMMING subjects. He is currently working as An Asst. Professor in CSE Department in Malla Reddy Institute of Technology, Maisammaguda, Dhulapally Post, Secunderabad and Telengana State, India.

Mail id : keshireddyvinay@gmail.com

Ms .C B SUSHMA was born in India . She pursuing M.Tech degree in Computer Science & Engineering in CSE Department in Malla Reddy Institute of Technology, Maisammaguda, Dhulapally Post, Secunderabad and Telengana State, India.

Mail id: balasushma022@gmail.com