Ranking Detection and Avoidance Frauds in Mobile Apps Store

ABHIILASH T P, L DINESHA
PG Scholar, Assistant Professor
Deptartment of Computer Science & Engineering,SSIT,Tumakuru,Karnataka, India
abhilash.panchu@gmail.com , ldinesha.ssit@gmail.com

ABSTRACT- There are millions of apps are available in market for the application of mobile users. However, all the mobile users first prefer high ranked apps when downloading it. But we cannot guarantee the reliability for the downloaded application since there is increasing number of ranking frauds. 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. 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 OS 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. There are in huge number of official and unofficial markets are available for mobile users to get variety of application. However, we cannot guarantee that the applications available in the market are trust worthy. Therefore, the application needs to be validated. In this paper we are introducing new protocol for detecting malicious apps.

Keywords— Rank Based evidence, Ranking Fraud, Mobile Applications.

I. INTRODUCTION

Mobile telephone fraud is the unauthorized use of the telecommunications network accomplished via deception. Mobile telephone fraud is a tremendous difficulty for network vendors and their customers: in some local areas it is estimated that more than half the use is fraudulent.

To stimulate the progress of cell Apps, many App outlets launched daily App leader boards, which demonstrate the chart rankings of most trendy Apps. Certainly, the App leader board is likely one of the primary ways for promoting cellular Apps. A bigger rank on the leader board normally leads to a tremendous quantity of downloads and million bucks in earnings. Consequently, App builders are likely to explore various approaches corresponding to promoting campaigns to promote their Apps in an effort to have their Apps ranked as high as possible in such App leader boards.

In the literature, even as there are some associated work, similar to web ranking junk mail detection [1], [2] online overview junk mail detection and mobile App recommendation the main issue of detecting ranking fraud for cellular Apps is still underexplored.

We propose to advance a ranking fraud detection process for mobile Apps. Along this line, we establish a couple of main challenges. First, ranking fraud does now not perpetually happen in the whole life cycle of an App, so we need to discover the time when fraud occurs. Such challenge can be regarded as detecting the local anomaly instead of global anomaly of cell Apps. 2nd, as a result of the big quantity of mobile Apps, it's intricate to manually label ranking fraud for each and every App, so it's fundamental to have a scalable method to mechanically detect ranking fraud without making use of any benchmark information. In the end, due to the dynamic nature of chart rankings, it's not easy to establish and confirm the evidences linked to ranking fraud, which motivates us to observe some implicit fraud patterns of cellular Apps as evidences.

Indeed, our careful commentary displays that cell Apps aren't constantly ranked high within the leader board, however handiest in some leading events, which type distinct leading Sessions.

Ranking fraud most of the time happens in these leading sessions. Therefore, detecting rating fraud of cellular Apps is truly to detect ranking fraud inside leading sessions of cellular Apps. Specifically, we first recommend a easy but powerful algorithm to identify the leading sessions of each and every App established on its historical rating files. Then, with the analysis of Apps‘ ranking behaviors, we discover that the fraudulent Apps quite often have exclusive ranking patterns in each and every main session when compared with normal Apps. Therefore, we symbolize some fraud evidences from Apps‘ historical ranking documents, and enhance three services to extract such ranking based fraud evidences. Nonetheless, the rating based evidences can be affected through App builders‘ fame and some reliable advertising campaigns, reminiscent of ―constrained-time discount‖. As a result, it is not ample to only use ranking established evidences. Consequently, we further suggest two forms of fraud evidences headquartered on Apps‘ ranking and assessment historical past, which reflect some anomaly patterns from Apps‘ ancient rating and evaluation documents. Furthermore, we boost an unsupervised evidence-aggregation system to integrate these three types of evidences for evaluating the credibility of main periods from mobile Apps detection system for mobile Apps.

II. Proposed System

Figure 1 indicates the block diagram of proposed architecture. For locating the leading session we'd like the ancient data. Old files accrued from quite a lot of sources. Then discovering the leading session shall be finished. This can be completed with the aid of discovering leading events from the App‘s historic rating documents. Second, we need to merge adjoining main pursuits for setting up leading periods.

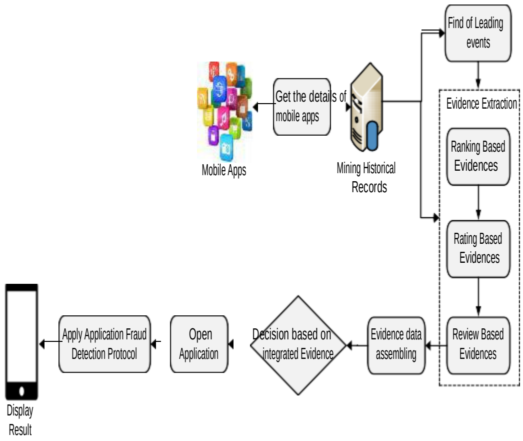


Figure 1: Proposed architectureIn the next step we ought to collect the evidences from the historic documents. Evidences will likely be collected headquartered on prior ranking, rating and experiences respectively.

A. Identifying the Leading Sessions of App’s:

a. Preliminaries

The App leader board demonstrates top k popular Apps with admire to specific classes, corresponding to ―top Free Apps‖ and ―prime Paid Apps‖. Additionally, the leader board is mainly up to date periodically. So each and every cellular app ‗a‘ has number of historic files which is denoted with the aid of R (a).

$$R(a) = \{r_1^a, \dots r_i^a, \dots r_n^a\} \tag{1}$$

Where n denotes the quantity of all rating files notice that, the smaller value ―ri power a‖ has the larger ranking function the App obtains.

By means of inspecting the old rating documents of cell Apps, we become aware of that Apps customarily aren't without end ranked excessive within the leader board, however simplest in some leading events. Furthermore, we find that some Apps have several adjoining main hobbies which might be just about each other and kind a leading session. The leading classes of a mobile App characterize its intervals of fame, so the rating manipulation will best take place in these leading sessions. As a result, the trouble of detecting ranking fraud is to detect fraudulent leading sessions.

Consequently, the main issue of detecting rating fraud is to realize fraudulent leading sessions. Along this line, the first task is the way to mine the leading sessions of a mobile App from its historic ranking records.

b. Mining Leading Sessions

There are two fundamental steps for mining leading sessions. First, we have got to observe leading events from the App‘s historic ranking files. 2nd, we ought to merge adjacent primary event for setting up leading sessions.

c. Extraction of Evidences

In determining ranking frauds historical evidences plays vital role. Here also we collect the useful evidences based on rating, ranking and experiences.

C1.1. Ranking Based Evidences

By way of inspecting the Apps‘ ancient ranking files, we realize that Apps‘ rating behaviors in a leading event continuously fulfill a targeted rating pattern, which contains three specific ranking phases, namely, rising phase, maintaining phase and recession phase. Especially, in every leading event, an App‘s ranking first increases to a top position in the leader board (i.e., rising phase), then keeps such top position for a period (i.e., maintaining phase), and ultimately decreases till the tip of the event (i.e., recession phase).

As soon as a common App is ranked high within the leader board, it obviously owns lots of honest enthusiasts and would appeal to more and more purchasers to down load. Consequently, this App is usually ranked high inside the leader board for a long time. From the above discussion, we propose some ranking established evidences of major sessions to assemble fraud evidences for ranking fraud detection.

Proof 1:

We use two shape parameters θ1 and θ2 to quantify the score patterns of the rising phase and the recession phase of App a‘s main event e, which may also be computed by using

$$\theta 1^e = arctan \frac{n(K - r_b^u)x^c}{t_b^e - t_a^e} \tag{2}$$

$$\theta 2^e = arctan \frac{(K - r_c^u)}{t_d^e - t_c^e} \tag{3}$$

Where, k is the rating threshold Intuitively, a higher θ1 may just point out that the App has been bumped to a high rank within a short interval, and a tremendous θ2 could factor out that the App has dropped from a excessive rank to the bottom within a brief interval. Hence, a leading session, which has more leading movements with significant θ1 and θ2 values, has larger chance of getting rating fraud. Correct now, we outline a fraud signature θs for a predominant session as follows

$$\theta_s = \frac{1}{|E_s|} \Sigma_{e \in s} \theta_1^e + \theta_2^e \tag{4}$$

Where $E_s|$ quantity of primary routine in session s. Intuitively, if a leading session s includes significantly higher θs in evaluation with exceptional leading durations of Apps within the leader board, it has high hazard of having rating fraud. To grab this, we propose to apply statistical speculation scan for computing the value of θs for each major session. In designated, we define two statistical hypotheses as follows and compute the p-price of every leading session.

• The signature $\theta s$of leading session s is no longer useful for detecting ranking fraud.

• The signature θs of main session s is enormously better than expectation.

Proof 2:

The Apps with rating fraud most as a rule have a short retaining part with excessive rating positions in each leading occasion. Hence, if we denote the retaining segment of a important party e as $\Delta t_m^e = t_c^e - t_b^e + 1$ and the typical rank on this retaining phase as $\bar{r}_m^e$we can outline a fraud signature $x_s$for every main session as follows

$$x_s = \frac{1}{|E_s|}\sum \frac{k^a - \bar{r}_m^e}{\Delta t_m^e} \qquad (5)$$

If a leading session includes greatly higher xs when put next with other leading sessions of Apps inside the leader board, it has high threat of having score fraud. To detect such signatures, we outline two statistical hypotheses as follows to compute the importance of xs for every leading session.

- The signature $\chi s$of main session s is not useful for detecting ranking fraud.
- The signature χs of main session s is significantly larger than expectation.

C1.2 Experience based evidences:

Experiences are very predominant proof in deciding whether the data is riskless or no longer. However most likely it‘s difficult to gauge established on simplest experiences.

Experiences can replicate the individual perceptions and utilization experiences of existing users for targeted cellular Apps. Most likely, comparison manipulation is no doubt probably the most primary perspective of App rating fraud. Mainly, earlier than downloading or purchasing a brand new cell App, users more often than not first of all learns its prior experiences to ease their resolution making, and a cell App entails more positive reports may just attract further customers to download. For that reason, imposters often put up false reviews inside the leading classes of a targeted App so that you could inflate the App downloads, and thus propel the App‘s ranking positions within the leader board.

Proper here we endorse two fraud evidences based on App‘s stories behaviors in main sessions for detecting rating fraud.

Proof 1:

Many of the experience manipulations are applied by way of boot farms when you consider that of the immoderate expense of human useful resource. As a consequence, review spammers most of the time submits a couple of duplicate or near-reproduction studies on the equal App to inflate download. In difference, the traditional App perpetually has various studies given that customers have exceptional individual perceptions and utilization. From the above observations, right here we define a fraud signature $Sim(s)$, which denotes the natural mutual similarity between the reports inside main session s. In particular; this fraud signature can be computed with the help of following steps.

- For each review c in leading session s, we remove all stop words (e.g., ―of‖, ―the‖) and normalize verbs and adjectives (e.g., ―plays → play‖, ―better → good‖).
- We build a normalized words vector $\underset{Wc}{\rightarrow}= dim(n)$ for each review c, where n indicates the number of normalized words in all reviews of s.

$$\mathrm{dim}[i] = \frac{\mathrm{freq}_{i,c}}{\sum_i \mathrm{freq}_{i,c}} \quad (1 \le i \le n) \qquad (6)$$

Freq is the frequency of ithword in c

Finally, we can calculate the similarity between two reviews ci and cj by the Cosine $\cos(Wci, Wcj)$.

Thus the fraud signature $Sim(s)$is

$$Sim(s) = 2 * \sum_{1\le i<j<Nb} cos(Wci, Wcj) \backslash Ns * (Ns - 1) \qquad (7)$$

Where Ns is the number of stories for the period of principal session s. Intuitively, the better valued at of sim(s) suggests further reproduction/close-reproduction reviews in s. For that reason, if a leading session has significantly bigger worth of Sim(s) in comparison with one of kind fundamental courses of Apps in the chief board, it has excessive likelihood of having ranking fraud.

To compute this, we define statistical hypotheses to compute the value of Sim(s) for each main session as follows.

- The signature Sim(s) of leading session s is not useful for detecting ranking fraud.
- The signature Sim(s) of leading session s is significantly higher than expectation.

Here, we use the Gaussian approximation to compute the p-worth with the above hypotheses. Above all, we count on $Sim(s)$ follows the Gaussian distribution,

$$\varphi_6^{(s)} = p(\mu_{sim}, \sigma_{sim}) \ge sim(s) \qquad (8)$$

Proof 2:

From the real-world observations, we find that each and every overview c is consistently associated with a particular latent discipline z For example; some reviews may just be involving the latent matter ―valued at to play‖ while some may be involving the latent matter ―very boring‖. In the meantime, on account that exceptional purchasers have one-of-a-type private preferences of mobile Apps, every App a would have great subject distributions in their ancient evaluation records. Intuitively, the subject distribution of studies in a average leading session s of App a, i.e., p/zmust be steady with the field distribution in all historic overview files a). It's considering that that the evaluation themes are centered on the customers' private utilization experiences however now not the reputation of cell Apps. In big difference, if the studies of s have been manipulated, the 2 subject distributions possibly markedly particular. For example, there would incorporate extra confident subject matters, similar to –worth to play‖ and –basic‖, inside the leading session.

We advise to leverage discipline modeling to extract the latent issues of experiences. Specially, right here we adopt the mainly used Latent Dirichlet Allocation (LDA) mannequin for learning latent semantic issues. To be more special, the historical stories of a cell App a, i.e. Ca, is believed to be generated as follows.

1st, earlier than generating Ca, ok prior conditional distributions of phrases given latent issues $\phi z$are generated from a previous Dirichlet distribution β.

2nd, a previous latent topic distributioθan is generated from a previous Dirichlet distribution α for each and every cellular App a.

The training procedure of LDA mannequin is to learn right latent variables $\theta = \{P(z|Ca)\}$ and $\phi = \{P(w|z)\}$ for maximizing the posterior distribution of review observations, i.e., $P(Ca|\alpha, \beta, \theta, \phi)$. this paper, we use a Markov chain Monte Carlo process named Gibbs sampling for coaching LDA mannequin. If we denote the experiences in main session s of a asCsa,we are able to use the KL-divergence to estimate the change of topic distributions between Ca and Csa.

$$DKL\ (s||a) = \Sigma\, k\ P(zk|Csa)\ ln\ (\frac{P(zk\,|Csa)}{P(z_k|C_a)} P(zk|Ca))\ (9)$$

Where

$P(zk|Ca))\ and\ P(zk|Cs; a) \propto P(zk)\ \Pi\ w2Csa \quad P(w|zk)$ may also be got by way of the LDA training approach. The larger value of DKL (s||a) suggests the better difference of subject distributions between Ca and Cs; a. For that reason, if a main session has vastly bigger worth of DKL (when put next with other leading sessions of Apps in the chief board, it has high probability of having rating fraud. To seize this, we define statistical hypotheses to compute the significance of DKL (each leading session as follows.

- The signature DKL $(s||a)$ of leading session s is not useful for detecting ranking fraud.
- The signature DKL $(s||a)$ of leading session s is significantly higher than expectation

The Gaussian approximation to compute the p-value with the above hypotheses;

$$\varphi_7(s) = 1 - p\big(N(\mu DL, \sigma DL)\big) \geq DKL(s||a) \qquad (10)$$

The values of two evidences Ψ6(s) and Ψ7(s) are in the range of [0, 1]. Meanwhile, the higher evidence value a leading session has, the more chance this session has ranking fraud activities.

d. Proof Aggregation:

After extracting three forms of fraud evidences, the following assignment is the way to mix them for ranking fraud detection. Certainly, there are various rating and proof aggregation approaches in the literature, comparable to permutation established units [7], rating centered items [1], [6] and Dumpster-Shafer principles [1], [2]. However, some of those approaches focus on finding out a worldwide ranking for all candidates. This isn't right for detecting rating fraud for brand new Apps. Different ways are situated on supervised learning strategies, which depend upon the labeled training data and are tough to be exploited. Instead, we advocate an unmonitored method centered on fraud similarity to combine these evidences.

Certainly, we outline the final proof ranking $\Psi\ (s)$as a linear combination of all of the existing evidences as observe that, right here we advise to use the linear blend since it has been validated to be robust and is largely used in imperative domains, such as rating aggregation [6], [8].

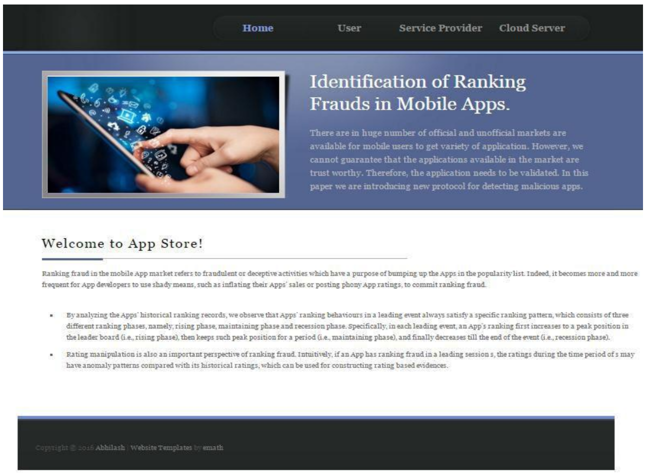
$$\varphi(s) = \Sigma^{i}_{N_\varphi} w_i * {}^{\varphi}_{i}(s)\ , s\ .t\ \Sigma^{N_\varphi}_{i=1} w_i = 1 \qquad (11)$$

d. Algorithm for mining leading sessions

```
Algorithm
Inputs:   1.  a's historical ranking record $R_{ai}$
              2. Ranking threshold k
                  3. Merging threshold $\emptyset$
Outputs:
          a's leading session Sa
Initialization: Sa = $\emptyset$;
        $E_{S=\emptyset}$; e =$\emptyset$;  S = $\emptyset$
        $t^e_{start}$ = 0;
         for each i $\in [1, |R_a|]$ do
       If $r^a_i \leq k$ and $t^e_{start} == 0$ then
                  $t^e_{start} = t_i$

              else if $r^a_i > k\ and\ t^e_{start}! = 0$ then
                    $t^e_{end} = t_{i-1}; e =< t^e_{start}, t^e_{end} >;$
     If $E_S == \emptyset$ then
        $E_s$ u = e; $t^s_{end} = t^e_{end}$;
                 else then
                    S =< $t^s_{start}$ , $t^s_{end}$, $E_s$ >;
                        Sa U= s; S = $\emptyset$ is a new session
                          $E_s = \{e\}$;  $t^s_{start}$  =  $t^e_{start}$  ;
                        $t^s_{start} = t^s_{end}$;
                          $t^s_{start} = 0; e = \emptyset$
                          is a new main event
                return Sa
```

III. RESULTS



Identification of Ranking Frauds in Mobile Apps.

There are in huge number of official and unofficial markets are available for mobile users to get variety of application. However, we cannot guarantee that the applications available in the market are trust worthy. Therefore, the application needs to be validated. In this paper we are introducing new protocol for detecting malicious apps.

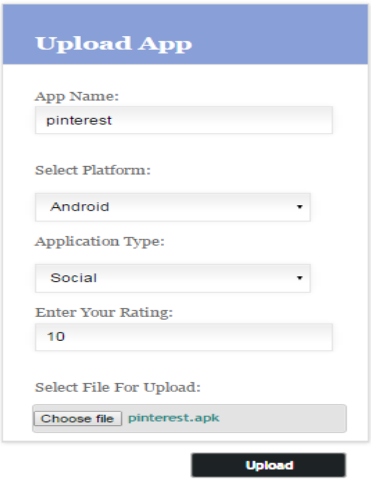
Welcome to App Store!

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.

- By analyzing the Apps' historical ranking records, we observe that Apps' ranking behaviours in a leading event always satisfy a specific ranking pattern, which consists of three different ranking phases, namely, rising phase, maintaining phase and recession phase. Specifically, in each leading event, an App's ranking first increases to a peak position in the leader board (i.e., rising phase), then keeps such peak position for a period (i.e., maintaining phase), and finally decreases till the end of the event (i.e., recession phase).
- Rating manipulation is also an important perspective of ranking fraud. Intuitively, if an App has ranking fraud in a leading session s, the ratings during the time period of s may have anomaly patterns compared with its historical ratings, which can be used for constructing rating based evidences.

Copyright © 2016 Abhilash | Website Templates by emath

Figure 2: Home Page


Upload App

App Name: pinterest

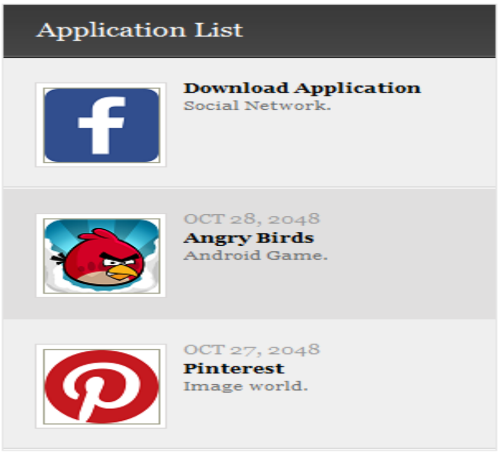
Select Platform: Android

Application Type: Social

Enter Your Rating: 10

Select File For Upload: pinterest.apk

Figure 3: Service Provider Uploading Application


Application List

Download Application
Social Network.

OCT 28, 2048
Angry Birds
Android Game.

OCT 27, 2048
Pinterest
Image world.

Figure 4: Application List

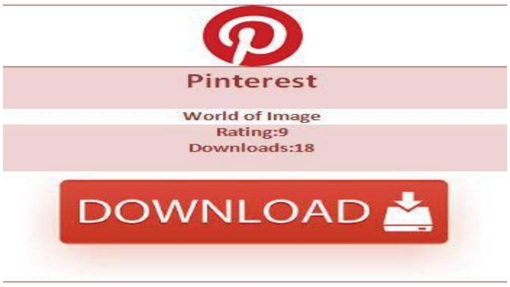
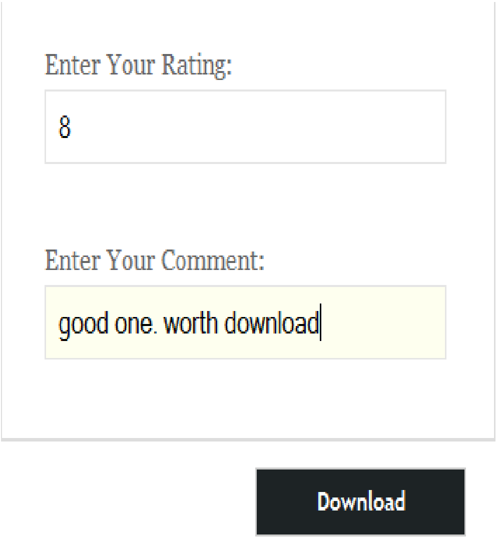

Pinterest
World of Image
Rating:9
Downloads:18

Figure 5: User Rating and number of downloads


Enter Your Rating: 8

Enter Your Comment: good one. worth download

Figure 6: User Comments and Rating when downloading

Application	Pinterest
Fraud Percentage Based on Rating	20%
Fraud Percentage Based on Review	15%
Fraud Percentage Based on Ranking	22%
Result	Good Application

Table 1: Fraud Application Detection Result

IV. Conclusion

Specifically, we first showed that ranking fraud happened in leading periods and provided a process for mining leading periods for each App from its old rating records. Then, identification of ranking based evidences, rating based evidences and experience based evidences for detecting ranking fraud might be accomplished. Additionally we are integrating application fraud detection method to make procedure robust.

REFERENCES

[1]. K. Shi and K. Ali, ―Getjar Mobile Application Recommendations with Very Sparse Datasets‖, International Conference on Knowledge Discovery and Data Mining, 2012.

[2]. N. Spirin and J. Han, ―Survey On Web Spam Detection: Principles and Algorithms‖, SIGKDD Explor, 2012.

[3]. M. N. Volkovs and R. S. Zemel, ―A Flexible Generative Model for Preference Aggregation‖, International Conference on World Wide Web, 2012.

[4]. Clifton Phua, Vincent Lee, Kate Smith and Ross Gayler, ―A Comprehensive Survey of Data Mining-based Fraud Detection Research‖.

[5]. Z.Wu, J.Wu, J. Cao, and D. Tao Hysad, ―A Semi-Supervised Hybrid Shilling Attack Detector for Trustworthy Product Recommendation‖, International Conference on Knowledge Discovery and Data Mining, 2012.

[6]. S. Xie, G. Wang, S. Lin, and P. S. Yu, ―Review Spam Detection via Temporal Pattern Discovery‖, international conference on Knowledge discovery and data mining, 2012.

[7]. B. Yan and G. Chen, ―Appjoy: Personalized Mobile Application Discovery‖, International Conference on Mobile Systems, Applications, and Services, MobiSys, 2011.

[8]. L. Azzopardi, M. Girolami, and K. V. Risjbergen, ―Investigating the relationship between language model perplexity and in precision recall measures‖, In Proceedings of the 26th International Conference on Research and Development in Information Retrieval (SIGIR‗03), pages 369–370, 2003.