

# Detection of Fraud App

<sup>1</sup>Sonali Patel , <sup>2</sup>Rushikesh Belekar, <sup>3</sup>Heramb Matwad , <sup>4</sup>Anuja Shelar, <sup>5</sup>Dr. K. V. Metre

*Department of Information Technology MET's Institute Of Engineering, Nashik, India*  
 Email : *patelsonali0908@gmail.com*<sup>1</sup>, *2belekarrushikesh6@gmail.com*<sup>2</sup>,  
*herambmatwad@gmail.com*<sup>3</sup>, *anujashelar30@gmail.com*<sup>4</sup>, *kvmetre@gmail.com*<sup>5</sup>

## ABSTRACT

*With the mobile application development, the number of mobile apps grown at very high rate. There are many app markets providing apps with Leader Boards. High ranking on such leader boards accelerate the app downloading. Many app developers are using some fraudulent ways to increase app ranking. To identify such frauds, there is need of fraud detection system. The proposed "Detection of Fraud App" system extracts the three types of evidences- ranking based evidences, rating based evidences, review based evidences and these evidences are aggregated to produce results. This system will evaluate the fraud detection effectively.*

**Keywords**—*Mobile Apps, Evidence Aggregation, Historical Ranking-Review-Rating Records, Rating Evidence, Ranking Evidence, Review Evidence.*

## 1. INTRODUCTION

Today, most of the people depend heavily on mobile devices. There are millions of mobile apps are available. Each app is used for specific purpose. There are number of apps for the same purpose has been developed by number of developers. There are number of mobile app markets are providing mobile apps. They are also launches the daily leader boards to show top rated apps. The app leader board is one of the most important way to promote app and increase downloads of the app as well as million dollars in revenue. High rating helps to accelerate number of downloads of the apps. The app has high ranking as the app users can give rating as per their uses of app, personal experience, and performance of the app. Users can also write some textual comment about the app. This reviews can helps the developers to improve the app as well as give brief description to other users about the app. This ranking, rating and reviews can helps to other users to download the app. This can increase the number of downloads of the app. To promote the app the app developers use various marketing strategies such as advertising, limited time discounts etc.

To accelerate such downloads of app some app developers uses the fraud way to show there app ranking as high as possible. Most of the rating and reviews are implemented by 'bot farms', or 'human water armies' due the high cost of human resource. This can increase review spamming. Review spammers often post the multiple reviews about the same app. Also post duplicate reviews.

There is need of such fraud detection mechanism. In this paper a system called "Detection of Ranking Fraud" is proposed to detect the ranking fraud in mobile applications. There are several factors which should be considered for this. There is ranking, rating, reviews are associated with each app. This can be used to find the ranking fraud. This also called as evidences. There are two terms, Leading Event, Leading Session are helpful to extract the ranking based evidence. The ranking of the app mostly depend on the popularity of the app. This can be short as well as can be long time. In this time period fraud can be happened. So we need to find such Leading Event and Leading Session. Users rating also can help to find the apps behavior. Rating can show the users perception toward the app. So there is need to extract the ratings. Reviews are nothing but the uses experiences while using app. Users put their experiences in the form of reviews. They are very useful to find the app is useful for other users and can save the data. These three evidences are very useful to find fraud. To find the evidences need the apps historical data about their ranking, rating, review from popular app stores or app markets. In this proposed system the data is fetched from the popular Google Play Store.

## 2. LITERATURE SURVEY

The user downloads and usages of the apps are observed and According to download and the usage ranking fraud can be found. Some users just do the number of downloads of the same app and they are not using the app. This downloads are increasing the ranking of the app and this is fraud ranking. To avoid this, system provides the limitation of using the app. And if any user trying to misuse the app by downloading it number of time then this user is blocked by system [1]. Introduced the three types of evidences as Ranking based evidences, Rating based evidences, and Review based evidences. This paper proposed to work on Leading Events and Leading Sessions, Also the optimization based aggregation method [2]. In this system only two types of evidences are used for fraud detection. The two evidences are ranking based evidence and Rating based evidence. The concept of Leading Event and Leading Session is used. This paper stated the algorithm for extracting Leading Events and Leading Sessions. All the work is done on the apples app stores historical data [3]. Here explained the porter streamer algorithm for processing textual comments. This algorithm processes all the user reviews and find out the terms showing positivity and negativity towards the app. This can help to extract the Review Evidence to find out the ranking Fraud [4]. This paper implemented the algorithm to extract Leading Event and Leading Sessions as the probability of fraud in Leading Event and Leading Session is more. There are seven types of evidences are calculated. At last they implemented the linear combination to aggregate all the evidences. They also used the database from the iOS App Store [5].

## 3. PROPOSED SYSTEM

The Proposed system will identify the leading events and leading sessions, the ranking fraud happened in those leading event and leading session. The system will detect that the App ranking is Fraud. The system requires access to the database. The Google API in java gives access to the Google play store database. This Database fetched from Google Play Store will be stored on the cloud server. We need 15GB storage space on cloud. Rest API is used to access the database stored on the cloud database. Further processing done on database contains, Ranking Evidence Calculation, Rating Evidence Calculation, and Review Evidence Calculation. In Ranking Evidence Calculation, mining leading events and leading sessions are calculated. In Review Evidence Calculation, Garbage Value Removal, Duplicate Statement Removal, Sentimental Analysis is done on reviews in database. Finally all the evidences are aggregated and the final result of fraud app detection is returned to the user. All the communication between the user and the server is done through the internet connectivity.

In fraud app detection, the system provides the android app for the android Smartphone, tablets etc. The user will search and enter the app ID into the app. This app ID will send to the Cloud server through internet connectivity. All the background processing will take place on the cloud server. The server calculates all the evidences and final result of fraud detection. This result will return to the user and displayed on the Android app.

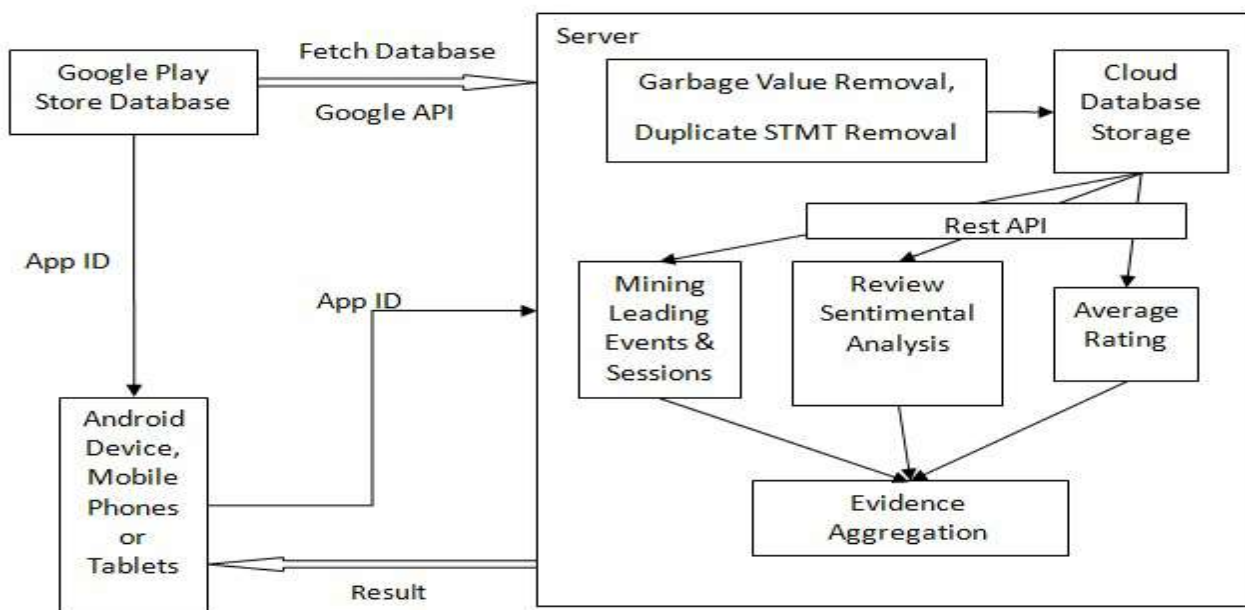


Figure 1: System Architecture

## A. Evidence Extraction For Ranking Fraud Detection

### a) Ranking Based Evidences

1. **Leading event** Given a Ranking Threshold  $K$ , and the app  $A$  has Leading Event  $E$  of time range  $t_E = [t_{ES}, t_{EE}]$  where the app  $A$  satisfies the condition  $r_{AES} \leq K$  and  $r_{AEE} \leq K$ .
2. **Leading Sessions**  
 The app  $A$  has Leading Session  $S$  of time range  $t_S = [t_{SS}, t_{SE}]$ ,  $t_S$  contains  $n$  adjacent Leading events which satisfies  $t_{SS} = t_{E(1)S}$  and  $t_{SE} = t_{E(n)E}$ .  
 There can be another Leading Session  $S^*$  where  $(t_{E(n+1)S} - t_{E(n)E}) > M$ , Where  $M$  is another predefined value called merging threshold, for merging the adjacent Leading Events to a Single Leading Session.

The Normal app has less number of Leading Events or Less Number of Leading Sessions. They have consistent behavior about their ranking. As compared to fraud app there can be too many number of Leading Events and Leading Sessions.

**Algorithm:** Mining Leading Sessions

**Input:**  $A$ 's historical ranking record  $RA$ ,  
 The ranking threshold  $K$ ,  
 The merging threshold  $M$ ,  
 The Set of  $A$ 's Leading Session

**Algorithm Steps:**

- [1] Initialization:  $LS = \text{NULL}$ ;  $LE = E = S = \text{NULL}$ ;  $t_{ES} = 0$ ;
- [2] For each  $RA_i$  belongs to  $RA$
- [3] if  $RA_i \leq K$  and  $t_{ES} == 0$  // New Event started  
 Go to [5].
- [4] Go to [6].
- [5]  $t_{ES} = t_i$ ;      Go to [2].
- [6] if  $RA_i > K$  and  $t_{ES} \neq 0$ ;      // found new event  
 Go to [8].
- [7] Go to [2].
- [8]  $t_{EE} = t_{i-1}$ ;  $E = \{t_{ES}, t_{EE}\}$
- [9] if  $LE == \text{NULL}$       Go to [12].
- [10]      if  $(t_{ES} - t_{SE}) < M$ ;      Go to [13].
- [11]      Go to [14].
- [12]       $LE = LE \cup E$ ;  $t_{SS} = t_{ES}$ ;  $t_{SE} = t_{EE}$ ; Go to [2].
- [13]       $LE = LE \cup E$ ;  $t_{SE} = t_{EE}$ ; Go to [2].
- [14]       $S = \{t_{SS}, t_{SE}, LE\}$
- [15]       $LS = LS \cup S$ ;
- [16]       $S = \text{NULL}$ ;      // This is New Session will Start
- [17]       $LE = \{E\}$ ;  $t_{SS} = t_{ES}$ ;  $t_{SE} = t_{EE}$ ;
- [18]       $t_{ES} = 0$ ;  $E = \text{NULL}$ ;      // This is new Leading Event  
 Go to [2].

**[19]** If no records left  
Return LS;

Where,  $RA_i = i^{\text{th}}$  Ranking Record of A

$t_{ES}$  = Start of Leading Event

$t_{EE}$  = End of Leading Event

$t_{SS}$  = Start of Leading Session

$t_{SE}$  = End of Leading Session

LE = Set of Leading Events

E = Leading Event

S = Leading Session Consists of Leading Events

LS = Set of Leading Sessions

#### b) Rating Based Evidences

Sometimes it is not sufficient to find out the ranking fraud using ranking based evidences. Because some of the apps have Leading Events cause they have created by famous developers like Game loft, EA Sports. Also some of marketing strategies likes providing some offers, limited time discounts, and advertising.

For normal app has average rating in Leading Session consistent with the average rating in all its historical ratings. There is need of fraud signature to find out such fraud, R,

$$\Delta R = (R_s - R_a) / R_a$$

The higher  $\Delta$  value of  $\Delta$  compared with other leading sessions on Leader Board has high probability of happening Ranking Fraud.

#### c) Review Based Evidences

Most of the app stores provide way to write some reviews for the apps. Such reviews can reflect the personal perceptions and user experiences. So it can be helpful while downloading app.

Indeed, most of the review manipulation is done by bot farms. Therefore review spammers can write some multiple reviews; to increase the number of downloads. Normal app always has diversified review as user experience is different per user. And spams reviews are always have similar.

Based on the above here is defined a fraud signature. Such reviews are processed and the terms in the reviews shows positive meaning or negative meaning are extracted. And after that the frequency of the similar terms are calculated in the Leading sessions. This can helps to find out the fraud is happened in the Leading Session.

Here to process such reviews used the porter Streamers Algorithm which converts the reviews in the stream of terms.

#### Algorithm: Porter Stemming

The purpose of Porter Stemmer algorithm is to remove various suffixes from web log data, to reduce number of words, to have accurately matching stems, to save time and memory space.

**Input:** Text Formats

**Output:** Text format without Stop words.

#### Algorithm Steps:

- 1) Gets rid of plurals and -ed or -ing suffixes  
e.g. interesting > interest, fated > fate
- 2) Turns terminal y to I when there is another vowel in the stem  
e.g. coolly > coolli, grey > grei
- 3) Maps double suffixes to single ones: -ization, -ational, etc  
e.g. optional > option, really > really
- 4) Deals with suffixes, -full, -ness etc  
e.g. largeness > large
- 5) Takes off -ant, -ence, etc  
Precedent > preced
- 6) Removes a final -e

This Terms again processed by N-Gram analysis to find out the correct meaning of the user review.

### **B. Evidence Aggregation**

There is again a challenge to aggregate the three evidences. There are number of methods but in this paper the simple method of Linear Combination is used. This method is simple and widely used. We have  $N = 3$  as number of evidences calculated, and  $W_i = [0/1]$  is the aggregation parameter of evidence.

## **4. CONCLUSION**

An android app can be used to identify app ranking fraud. To detect ranking fraud, historical records of the android apps on the 'Google PlayStore' will be used. Till now, all the efforts are done for the desktop applications. This android app will be very usefull to detect ranking fraud. This app will be very simple and easy to use. This system will evaluate the ranking fraud detection effectively.

## **5. REFERENCES**

- [1] S.Kalaiarasi1, Swetha Ganesh2, Praveena.C.H2,Vaishali.T2, "Ranking Fraud Detection for Mobile Apps using Evidence Aggregation and Humming Bird algorithm", Vol. 4, Issue 4 , April 2016.
- [2] Raghuvveer Dagade, Prof. Lomesh Ahire, "Review: A Ranking Fraud Detection System for Mobile Apps" Vol. 3, Issue 11, November 2015.
- [3] H. Zhu, H. Xiong, Y. Ge, and E. Chen, "Ranking fraud detection for mobile apps: A holistic Review", in Proc. 22<sup>nd</sup> ACM Int. Conf.Inform. Knowl. Manage., pp. 619628. , hkdsak
- [4] Willett, P., "The Porter stemming algorithm: then and now.", Program: electronic library and information systems, 40 (3). pp. 219-223. , 2006.
- [5] Hengshu Zhu, Hui Xiong, Yong Ge and Enhong Chen, "Discovery of Ranking Fraud For Mobile Apps", Vol-27, No1

