Comprencive Study on Fraud Malware Detection in Google Play Store

Singampalli Sankeerthi[1], Pogolu Nagjyothi[2]
[1]Assistent Professor, Dept of MCA
[2]Dept of MCA
[1, 2] St. Mary's Group of Institutions, Guntur, Andhra Pradesh, India

Abstract- *The use of mobile devices as well as Tablets, Smart watch, and notebooks are increasing day by day. Automaton has the key share within the mobile application market. Automaton mobile applications become a straightforward target for the attackers due to its open supply environment. Also, the user's ignorance the process of installing and usage of the apps. to spot pretend and malware applications, all the previous ways targeted on obtaining permission from the user and execution that individual mobile application. A malware detection framework that discovers and break traces left behind by dishonorable developers, to sight search rank fraud moreover as malware in Google Play. The fraud app is detected by aggregating the 3 items of proof like ranking primarily based, co-review {based based mostly primarily based} and rating based proof. Finally aggregating all the activities of front-running apps, it can do bound accuracy in classifying benign normal datasets of malware, fraudulent and legitimate apps. In addition, we tend to apply progressive learning approach to characterize an outsized variety of datasets. It combined effectively with all the proof for fraud detection. To accurately find the ranking fraud, there's a necessity to mining the active period's particularly leading sessions, of mobile Apps.*

Keywords- Mobile applications, Malware, Ranking, Rating, Google Play.

I. INTRODUCTION

Google play first releases its app in 2008.Since that it distributes applications to all the Android users. In Google Play Store, it provides services that user can discover the particular application, purchase those applications and install it on their mobile devices. Since Android is open source environment all the detail about the application users can be easily accessed by the application developers through Google play. In Google play 1.8 Million mobile applications are available and that is downloaded by over 25 billion users across the world. This leads to greater chance of installing malware to the applications that could affect user's mobile devices. Google play store uses its own security system known as Bouncer system [6] to remove the malicious apps from its store. However, this method is not effective as testing some apps using virus tools many apps are found as malicious which are not detected by Bouncer system [6]. Fraudulent developers use search ranking algorithm to promote their apps to the top while searching. After downloading mobile applications from Google play users are asked to give the ratings and reviews about that particular downloaded applications. However fraudulent developers give fake ratings and reviews about their application promote their application to the top. There are two typical approaches used for detecting malware in Google Play. Thus are Static and Dynamic. The dynamic approach needs apps to be run in a secure environment to detect its benign. The static approach is not used as the need to give all types of attack in early stage itself but that is impossible as everyday attackers find the new way to inject malware on applications.

II. LITERATURE SURVEY

As we know before us many great peoples worked on this android app ranking fraud detection through ads so we just go through their study work and take inspiration from their work and build our improved system. [1] In this paper, they give an extensively comprehensive point of view of situating trickery and propose a situating compulsion exposure structure for flexible Apps. Particularly, they first proposed to extremely situate the situating blackmail by mining the dynamic time periods, particularly motivating sessions, of flexible Apps. They scrutinize three sorts of pronouncements, i.e., situating based confirmations, rating based verifications and study based verification, by showing Apps' situating, rating and review hones through experimental hypotheses tests. Besides, propose a progression based accretion system to join each one of the pronouncement for compulsion characteristic proof.

Ranjitha.R, Mathumitha.K, Meena.S, S.Hariharan [2] had proposed system additionally, they are proposing two enhancements using appreciation of keep a tally by the admin to recognize the exact reviews and rating scores. Secondly, the fake response as a feedback by a same person for pushing up that app on the leader board is restricted. Two different

limitations are taking into account for accommodating the feedback given to an application as a part of their response toward the app whether it is good or bad. The first constraint is that an app can be rated only once from a one particular user login and the second are put into action with the id of IP address that limits the number of user login logged per day. Finally, the proposed system will be estimated with real-world App data which is to be composed from the App Store for a long-time period.

R.Vinodharasi, P.Ramadoss[3] proposed to precisely situate the ranking fraud by mining the dynamic periods, namely leading sessions, of mobile Apps. Additionally, we examine 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 arithmetical mining based hypotheses tests. In addition, in this project and optimization based application used to incorporate all the evidences for fraud recognition based on EIRQ (efficient information retrieval forranked query) algorithm. Finally, estimate the projected system with real-world App data collected from the IOS App Store for a long-time period. Experimentation was need to be done for authenticate the efficiency of the proposed system, and show the scalability of the recognition algorithm as well as some reliability of ranking fraud activities.

Phopse P.E, Jondhale S.D[4] had provide a holistic view of ranking fraud and propose a ranking fraud appreciation system for mobile Apps. Additionally, to first propose to precisely 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 irregularity instead of global irregularity of App rankings. Furthermore, we consider three types of evidences, i.e., ranking based evidences, rating based evidences and review based evidences, by modeling Apps' ranking, rating and review nature through statistical hypotheses tests.In addition, to project an optimization based aggregation method to consolidate all the evidences for fraud recognition.

Xiong and Zhu[5] had projected a ranking fraud detection system for android mobile apps. In this paper principally, they both demonstrated that ranking fraud take place in most important sessions for each app from its previous ranking accounts. Then, they recognized ranking based, rating based and review based confirmation for discovering ranking fraud. Moreover, they proposed an optimization based aggregation system to merge all the evidences for estimate the consistency of most important sessions from mobile apps.

Priyanjai and Pankaj[6] planned techniques for assessment of investigation and invent pattern of android apps based on cloud computing and data mining. They developed system ASEF and SAAF for android apps to achieve protection. They also explain a tactic that performs apps security and provide user friendly interface on a mobile phone.

Anuja A. Kadam, Pushpanjali M. Chouragade [7] make available a disciplined study on the different procedures of malicious application recognition in android mobiles. The examination of authorization induces possibility in Android apps on a large-scale in three stages. First upon position all the entity permissions with respect to their feasible risk with different processes. Secondly, classify subsets of risk permissions. Then using several algorithms identifies the suspected apps based on the recognized subsets of risky permissions.

Jakub Zilincan, Michal Gregus [8] had given the dedicated work on Search engine optimization techniques, often summarized to SEO, should lead to first situation in unprocessed search results. Some optimization techniques or procedures do not modify over time, yet still form the foundation of SEO. However, as the Internet and web design develop enthusiastically, new optimization procedures come in to account and sometime does not work. Thus, they have focused on most important features that can help to get better a pose in search outcome. It is important to accentuate, that none of the procedure can make sure it because search engines have complicated algorithms, which measure the superiority of Web pages and obtain their position in search results from.
Xiang Wang, Yan Jia, Ruhua Chen, Bin Zhou [9] in that they had told users can interpret themselves using free tags in microblogging website such as Sina Weibo. The tags of a user exhibit. The description of the user and are normally in a unsystematic direct without any significance or importance information. It restricts the usefulness of user tags in system suggestion and other applications. They also proposed a user tag ranking representation which is based on interactive and attractive dealings between users. Manipulate power between users is measured in our user tag ranking method. Significance scores between tags and users are also utilized to rank user tags.

III. RELATED WORK

IDENTIFYING EVIDENCES FOR RANKING FRAUD DETECTION:

1. **Identifying Leading Sessions:** Leading sessions are the base for detecting fraud in mobile App as ranking fraud usually happens in leading sessions. And hence detecting

ranking fraud is actually detecting ranking fraud within leading session of mobile Apps which we mine from mobile Apps historical ranking records. There are two main steps for mining primary sessions. First, we need to determine leading measures from the

2. App's previous ranking records. Second, we need to collaborate neighboring leading events for developing leading sessions. Specifically, we first propose a simple yet effective algorithm to identify the leading events of each App based on its historical ranking records. Then, we merge adjacent leading events for constructing leading sessions. As per the observation the mobile apps do not always ranked high in the leader boards, in fact in some leading events only. With the analysis of Apps' ranking behaviors, the fraudulent Apps often have different ranking patterns in each leading session compared with normal Apps. Therefore, the problem of identifying ranking fraud is to find out vulnerable leading sessions.

Ranking based evidences: A leading session is composed of several leading events. Therefore, we should first analyze the basic characteristics of leading events for extracting fraud evidences. By analyzing the Apps' historical ranking accounts,Apps' ranking behaviors in a leading incident always assure aspecific ranking pattern, which consists of three different ranking segments, expanding phase, maintaining phase and collapse phase. Mainly, in each leading event, an App's ranking first improve to a peak or extent position in the leaderboard (i.e., rising phase), then maintain such peak position for a phase (i.e., maintaining phase), and at last declines till the end of the event (i.e., recession phase). Definitely, such a ranking pattern confirms a significant consideration of leading event. In next section, we formally describe the three ranking phases of a leading event.

- **Rating based evidences:** The ranking based evidences are first step towards ranking fraud recognition. However, sometimes, it is not satisfactory to only use ranking based evidences. Take an example, some Apps formed by the legendary developers, such as Gameloft, may have some leading events due to the developers' trustworthiness and the "word-of-mouth" advertising effect. Moreover, some of the permissible marketing services, such as "limited-time discount", may also consequence in significant ranking based evidences. To solve this matter, we also study how to extort fraud evidences from Apps' historical previous rating records. Indeed, user rating is one of the most important features of App advertisement. A higher rated App may attract more users to download and can also be ranked higher in the leader board. Thus, rating manipulation is also an important perspective of ranking fraud. Intuitively, if an App has ranking fraud in a leading session, the ratings during the time period of that leading session may have drastically changed patterns if seen from its previous historical ratings, which can be used for constructing rating based evidences. Rating to app is given by the user who downloaded it. Hence rating is one of the main evidence in ranking fraud of apps. In this module it performs preprocessing of ratings that is it removes ratings that are less than or equal to two in number given as star to that App that is if 5 star given to the App is one in number among 100 users given other rating but not 5 star then it should be deleted and thus calculates rating score by summing all the ratings class collected and decision is taken on the basis of rating which scores high amongst all.

- **Review based evidences:** Including ratings, most of the App stores also allow users to write some textual comments as App reviews to submit to the developer. Such reviews can reflect the personal observations and usage understanding of breathing users for particular mobile Apps. Indeed, review management is one of the most important base of finding App ranking fraud.
    - Specifically, before downloading or purchasing a new mobile App, users often first read its previous historical reviews to simplify their conclusion making and a mobile App includes more encouraging reviews may attract more users to download. Therefore, imposters often place counterfeit reviews in the leading sessions of a specific App in order to inflate the App downloads, and thus boost the App's ranking position in the leaderboard. Therefore, manipulation and detection of reviews is one way used over shady app developers to expertise the app. Hence reviews are used to detect the ranking fraud in Mobile App industry is the foremost viewpoint to find ranking fraud.

On semantic analysis level review rechecking can be done to show the concluded review to user of app to make them easy to judge that app. As the Sentiment Analysis is a natural language processing task that deals with finding orientation of opinion in a piece of text with respect to a topic. To determine the semantic orientation of the sentences a dictionary based technique of the unsupervised approach is adopted. To determine the opinion words and their synonyms and antonyms WordNet is used as a dictionary. This module performs pre-processing of reviews and then performs sentiment analysis on pre-processed reviews. As the growing market of internet brought to the conclusion of product reviews as it made easy our decision about that product and as Internet is used by everyone the numbers of reviews that a product receives grow rapidly. To read all of comments is very time taking task for a potential customer and make a decision on whether to buy that product or not. Thus, mining this data about reviews, preprocessing that data, and classify them is an important task to make the reviews result corrected as shown below on stepwise proposing of such work: Gathering data for reviews from app store, and other sources: To determine the polarity of the sentences, based on aspects, large numbers of reviews are collected from the Web. There are lots of websites on the Internet where the large numbers of customer reviews are available. Amazon website (www.amazon.com) and also play stores like google play are used to collect the reviews. Pre-processing data to remove any missing entries (using filtering technique): To determine the semantic orientation of the sentences a dictionary based technique of the unsupervised approach is adopted. To determine the opinion words and their synonyms and antonyms WordNet is used as a dictionary; also, it plays a vital role in detecting any missing entries using filtering technique. Semantic matching for finding quality of review (Positive, Negative or Neutral): A large amount of reviews of users are collected on the Web that needs to be explored, analyze and organized for better decision making. Opinion Mining or Sentiment Analysis is a Natural Language Processing and Information Extraction task that identifies the user's views or opinions explained in the form of positive, negative or neutral comments and quotes underlying the text. Aspect based opinion mining is one of the level of Opinion mining that determines the aspect of the given reviews and classify the review for each feature. Semantic Matching: Algorithms and Implementation - Semantic Scholar

IV. THE SYSTEM PERFORMS THIS TASK IN SEVERAL STEPS AS FOLLOWS

4.1 Data Collection: To determine the polarity of the sentences, based on aspects, large numbers of reviews are collected from the Web. There are lots of websites on the Internet where the large numbers of customer reviews are available. Amazon website (www.amazon.com) is used to collect the reviews.

4.2 POS Tagging: After collecting the reviews, they are sent to the POS tagging module where POS tagger tag all the words of the sentences to their appropriate part of speech tag. POS tagging is an important phase of opinion mining, it is necessary to determine the features and opinion words from the reviews.POS tagging can be done manually or with the help of POS tagger. Manual POS tagging of the reviews take lots of time. Here, POS tagger is used to tag all the words of reviews. 4.3 Feature Extraction: All the features are extracted from the reviews and stored in a database then its corresponding opinion words are extracted from these reviews. It will find out whether the comment is positive, negative or neutral. If word is positive then it will add plus one to score; if word is negative it will minus one from score. Sometimes it is unable to find sentiment of some reviews, that time it makes the use of Naive Bayes classifier. In this way it will find final score by analyzing sentiment of each review and determine whether app is fraud or not on the basis of review evidences.

Algorithm:

- Read all feedback information
- Divide the information into sessions

- For each session find the feedback obtained, to get the list

  S1 F1
  S2 F2

S3 F3 Sn Fn

Where Si is the session, and Fi is the feedback from that session

- **Check if the feedbacks have a common trait,**

if(F1 = F2 and F2 = F3 and .... Fn-1=Fn)

Then it means the review is genuine else

if there is a abrupt shift in the pattern, then the feedback might be non-genuine

For NLP based technique,

1. Read all feedback information

2. For each feedback, find action words using POS Tagging and Chunking process

3. Evaluate the sentiment from the feedback and mark the feedback as Good or Bad

4. Divide the feedback into sessions
5. For each session find the feedback obtained, to get the list

S1 F1
S2 F2
S3 F3

.
.
Sn Fn

Where Si is the session, and Fi is the feedback from that session

6. Check if the feedbacks have a common trait, if(F1 = F2 and F2 = F3 and .... Fn-1=Fn)

Then it means the review is genuine else

if there is a abrupt shift in the pattern, then the feedback might be non-genuine Combine results from both the algorithms to conclude if the given feedback is genuine or not.

Pattern analysis using machine learning:

There are two main steps for mining leading sessions. First, we need to discover leading events from the App's historical ranking records. Second, we need to merge adjacent leading events for constructing leading sessions. By analyzing the Apps' historical ranking records, we observe that Apps' ranking behaviors 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 leaderboard (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). Indeed, such a ranking pattern shows an important understanding of leading event. An App has several impulsive leading events with high ranking positions. In contrast, the ranking behaviors of a normal App' sleading event may be completely different. For example, ranking records from a popular App "Angry Birds: Space", which contains a leading event with a long-time range (i.e., more than one year), especially for the recession phase. In fact, once a normal App is ranked high in the leaderboard, it often owns lots of honest fans and may attract more and more users to download. Therefore, this App will be ranked high in the leaderboard for a long time. Based on the above discussion, we propose here some ranking based signatures of leading sessions to construct fraud evidences for ranking fraud detection.

6. Result analysis based on the matching: After extorting three types of fraud evidences, the next dare is how to merge them for ranking fraud detection. Indeed, there are many ranking and evidence association techniques in the literature that we have studied before, such as transformation based models, achieve based models, and Dumpster-Shafer rules. However, some of these methods spotlight on learning a worldwide ranking for all contenders.

V. PROPOSED SYSTEM

It proposes malware detection framework system that effectively detects Google Play fraud and malware. To detect fraud and malware, we propose the incremental learning approach to characterize the dataset. We formulate the notion of review modeling by applying Porter stemmer algorithm. We use temporal session of review post times to identify suspicious review spikes received by apps; the application evidence such as rating, ranking and review evidence will be integrated by an unsupervised evidence-aggregation method for evaluating the credibility of leading sessions from mobile Apps. The malware detection framework is scalable and can be extended with other domain generated evidence for ranking fraud detection. When compared to other existing systems this method finds the better mobile app for the end user. Incremental learning approaches effectively characterize all category of app in Google Play. Also based on the review,

rating and rank given by the user is also checked. User can review after they download that particular application using their account from app store.

ADVANTAGES:

- Detect fraud ranking in daily App leader board.
- Avoid ranking manipulation.
- Finds the better mobile app for the end user.
- Incremental learning approach effectively characterizes the large amount of app evidence details.
- It provides accurate aggregation when compared to our existing approach.

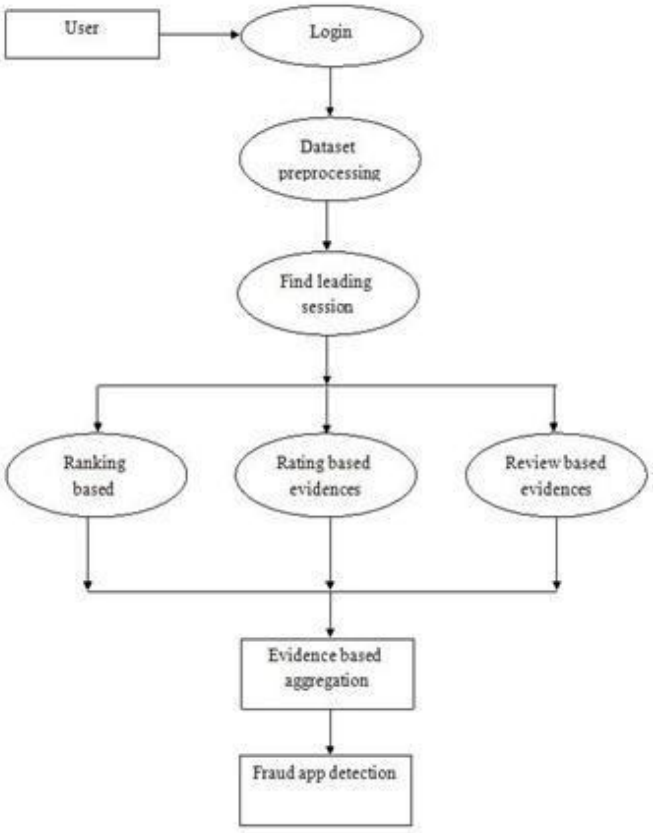

Fig 1: Incremental Learning approach

VI. CONCLUSION

In this project, we developed a fraud detection system for mobile Apps. Specifically, we first showed that fraud happened in leading sessions and provided a method for mining leading sessions for each App from its historical ranking records. We identified that for the detection of the rank ranking, rating, review based evidence are considered. Moreover, we proposed an optimization based aggregation method to integrate all the evidence for evaluating the credibility of leading sessions from mobile Apps. A unique perspective of this approach is that all the evidence can be modeled by statistical hypothesis tests, thus it is easy to be extended with other evidence 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 evidence 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] Alaa Salman Imad H. Elhajj Ali Chehab Ayman Kayss, IEEE Mobile Malware Exposed.International Conference onKnowledge discovery and data mining, KDD'14 pages 978-983.

[2] Alfonso Munoz, Ignacio Mart ˜ ´ın, Antonio Guzman, Jos ´e Alberto Hern ´ andez, IEEE Android malware detectionfrom Google Play meta-data: Selection of important features.2015, pages,245-251.

[3] Chia-Mei Chen, Je-Ming Lin, Gu-Hsin Lai,IEEE Detecting Mobile Application Malicious Behaviors Based on DataFlow of Source Code.2014 International Conference on Trustworthy Systems and their Applications pp 95-109.

[4] D. F. Gleich and L.-h. Lim. Rank aggregation via nuclear norm minimization. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '11, pages 60–68, 2011.Y.T. Yu, M.F.Lau, "A comparison of MC/DC, MUMCUT and several other coverage criteria for logical decisions", Journal of Systems and Software, 2005, in press.

[5] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw.Detecting product review spammers using rating behaviors. In Proceedings of the 19th ACM international conference on Information and knowledge management,CIKM '10, pages 939–948, 2010.

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

[7] J. Oberheide and C. Miller, "Dissecting the Android Bouncer," presented at the SummerCon2012, New York,NY, USA, 2012.

[8] K.Shi and K.Ali. Getjar mobile application recommendations with very

sparse datasets. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '12, pages 204–212, 2012.

[9] 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.

[10] N. Spirin and J. Han. Survey on web spam detection: principles and algorithms. SIGKDD Explor. Newsl.,13 (2):50–64,May2012.