International Journal of Research
Available at https://edupediapublications.org/journal
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 17
November 2016

A Generative Model For Evidence Aggregation Based Ranking Fraud Detection(EA-RFD)

[1] K.Sudha Kumari [2] K. Ramesh

[1]M.Tech Student, Dept. of CSE, GATES Institute of Technology, Gooty, Anantapuramu Dist. AP-515401
[2]Associate Professor & HOD, Dept. of CSE, , GATES Institute of Technology, Gooty, Anantapuramu Dist. AP-515401

Abstract: The Mobile App is a very popular and well known concept due to the rapid advancement in the mobile technology. Due to the large number of mobile Apps, ranking fraud is the key challenge in front of the mobile App market. Ranking fraud refers to fraudulent or vulnerable activities which have a purpose of bumping up the Apps in the popularity list. While the importance and necessity of preventing ranking fraud has been widely recognized. In the existing system the leading event and leading session of an app is identified from the collected historical records. Then three different types of evidences are collected from the user feedbacks namely ranking based evidence, rating based evidence and review based evidence. These three evidences are aggregated by using evidence aggregation method. In the proposed system additionally, we are proposing two enhancements. Firstly, we are using Approval of scores by the admin to identify the exact reviews and rating scores. Secondly, the fake feedbacks by a same person for pushing up that app on the leader board are restricted. Two different constraints are considered for accepting the feedback given to an application. The first constraint is that an app can be rated only once from a user login and the second is implemented with the aid of IP address that limits the number of user login logged per day. Finally, the proposed system will be evaluated with real-world App data which is to be collected from the App Store for a long time period.

Keywords
Mobile Apps, ranking fraud detection, evidence aggregation, historical ranking records, rating and review.

I. INTRODUCTION

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 Rating Based Evidences, specifically, after an App has been published, it can be rated by any user who downloaded it. Indeed, user rating is one of the most important features of App advertisement. An App which has higher rating 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. In Review Based Evidences, besides ratings, most of the App stores also allow users to write some textual comments as App reviews. Especially, this paper proposes a simple and effective algorithm to recognize the leading sessions of each mobile App based on its historical ranking records. This is one of the fraud evidence. Also, rating and review history, which gives some anomaly patterns from apps historical rating and reviews records

2. LITERATURE SURVEY

Leif Azzopardi et al. studied an Investigating the Relationship between Language Model Perplexity and IR Precision Recall Measures the perplexity of the language model has a systematic relationship with the achievable precision recall performance though it is not statistically significant. A latent variable unigram based LM, which has been successful when applied to IR, is the so called probabilistic latent semantic indexing (PLSI).

Ee-Peng Lim et al. presented a number of detecting Product Review Spammers using Rating Behaviors to detect users generating spam reviews or review spammers. We identify several characteristic behaviors of review spammers and model these behaviors so as to detect the spammers.

David F. Gleich et al. has done a survey on Rank Aggregation via Nuclear Norm Minimization the process of rank aggregation is intimately intertwined with the structure of skew-symmetric matrices. To produces a new method for ranking a set of items. The essence of our idea is that a rank aggregation describes a partially filled skew-symmetric matrix. We extend an algorithm for matrix completion to handle skew-symmetric data and use that to extract ranks for each item.

Alexandre Klementiev, Dan Roth et al. studied an Unsupervised Learning Algorithm for Rank Aggregation, (ULARA) which returns a linear combination of the individual ranking functions based on the principle of rewarding ordering agreement between the rankers.

3. METHOD

Detection of ranking fraud for mobile Apps is still under a subject to research. To fill this crucial lack, we propose to develop a ranking fraud detection system for mobile Apps. We also determine several important challenges. First challenge, in the whole life cycle of an App, the ranking fraud does not always happen, so we need to detect the time when fraud happens. This challenge can be considered as detecting the local anomaly in place of global anomaly of mobile Apps. Second challenge, it is important to have a scalable way to positively detect ranking fraud without using any basis information, as there are huge number of mobile Apps, it is very difficult to manually label ranking fraud for each App. Finally, due to the dynamic nature of chart rankings, it is difficult to find and verify the evidences associated with ranking fraud, which motivates us to discover some implicit fraud patterns of mobile Apps as evidences.

SYSTEM ARCHITECTURE

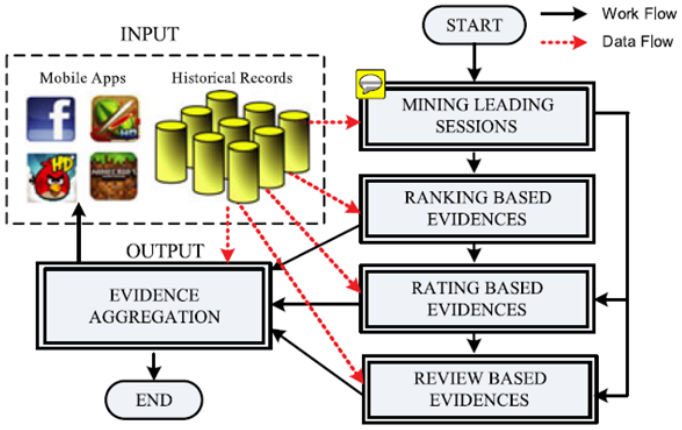


Figure 1.Architecture diagram

Mobile app stores launched many apps daily in the leader boards which show the chart ranking of popular apps. The leader board is the important for promoting apps. Original application grade level decreases due to the arrival of fake apps. The users who are newly logging to the app stores, they decide based on the existing ranking, rating, reviews for the individual apps. In recent activities duplicate version of an application not burned or blocked. This is the major defect. Higher rank leads huge number of downloads and the app developer will get more profit. In this they allow Fake Application also. User not understanding the Fake Apps then the user also give the reviews in the fake application. Exact Review or Ratings or Ranking Percentage are not correctly Calculated. In this paper we introduce admin to manage the ranking evidence to minimize the arrival of fake apps, then the rating and reviews are correctly calculated.

4. IMPLEMENTATION

Mining Leading events and Sessions

Generally apps are classified into two types such as top free apps and top paid apps. Apps in the leader board are updated regularly. Each app has many historical ranking records with respect to time. By observing the historical ranking records of mobile apps only in some leading events apps are ranked higher in the leader board

Definition of leading event:

For a given ranking threshold A belongs to [1,A] and leading event LE of app consists time series denoted with T and its range is [start, end] which satisfies the following condition , threshold rank of an app A must lies between the starting and ending time range.

Usually event refers to occurrence of an action. Manipulating the chart rankings, ratings and reviews of an app in the leader board is considered an action at particular time period. Sequence of actions will lead to leading sessions where anomaly inflate the app rankings in the leader board.

Definition of leading session:

If an app has x adjacent leading events, consists a time range and its leading session LS satisfies a condition, starting time of the LS is equal to the starting time of the LE, ending time of LS is equal to the ending time of LE. Time range must be less than phi, where phi is a predefined time threshold for merging leading events.

Algorithm for mining leading sessions

Procedure:

Input data: historical ranking records, ranking threshold A, phi as merging threshold
Generated output: leading sessions LS
Step 1: initialize leading session LS to null;
Step 2: initialize leading event LE to null;
Step 3: initialize starting time to zero;
Step 4: for every historical record do
Step 5: if ranking threshold is greater than app rank and starting time is equal to zero then
Step 6: update starting time
Step 7: else if ranking threshold is less than app rank and starting time is not equal to zero then
Step 8: note ending time and that event has occurred between starting and ending time.
Step 9: repeat step 4 to 8 for every successive action and merge the successive leading events by using phi.
Step 10: return LS

In the first module, we develop our system environment with the details of App like an app store. Intuitively, the leading sessions of a mobile App represent its periods of popularity, so the ranking manipulation will only take place in these leading sessions. Therefore, the problem of detecting ranking fraud is to detect fraudulent leading sessions. Along this line, the first task is how to mine the leading sessions of a mobile App from its historical ranking records. 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.

Ranking Based Evidences

After mining the leading sessions from the historical ranking records we develop Ranking based Evidences system.By analyzing the Apps' historical ranking records, web serve 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).

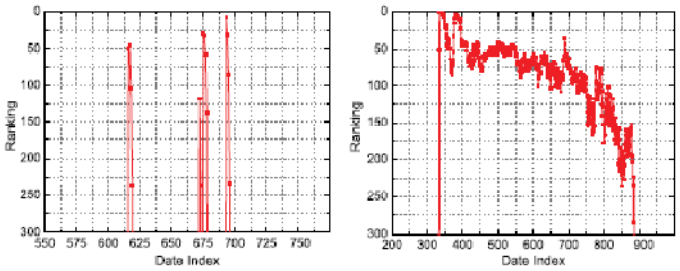
Definition (ranking phases of a leading event): Given a leading event LE of an app with time range [start, end], where the highest ranking position of app is pack which belongs to delta R. the rising phase of LE is a time range [a, b] where a=start, b belongs to delta R and for all input time belongs to [a, b] satisfies input does not belong to delta R. The maintaining phase of LE is a time range [b, c], where input of c belongs to delta R and for all input time belongs to [c, end] satisfies input does not belong to delta R. The recession phase is a time range [c, d], where d=end.

Rating Based Evidences

In the third module, we enhance the system with Rating based evidences module. The ranking based evidences are useful for ranking fraud detection. However, sometimes, it is not sufficient to only use ranking based evidences. For example, some Apps created by the famous developers, such as Gameloft, may have some leading events with large values of u1 due to the developers' credibility and the "word-of-mouth" advertising effect. Moreover, some of the legal marketing services, such as "limited-time discount", may also result in significant ranking based evidences. To solve this issue, we also study how to extract fraud evidences from Apps' historical rating records.

Review Based Evidences

In this module we add the Review based Evidences module in our system. 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. Therefore, imposters an often post fake review in the leading sessions of a specific App in order to inflate the App downloads, and thus propels the App's ranking position in the leader board.



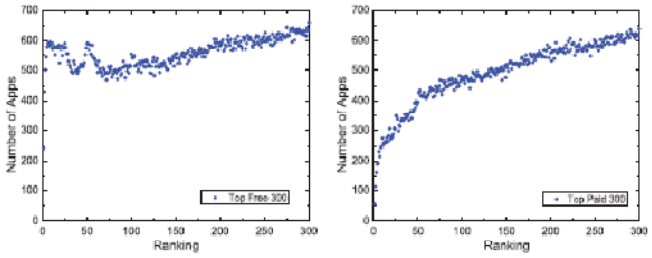
(a) Example 1 (b) Example 2

Fig: Two real world examples of leading events.

Evidence Aggregation

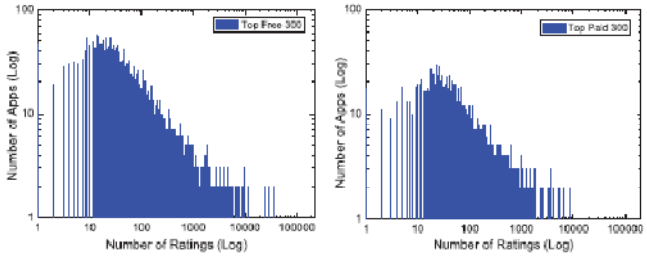
In this module we develop the Evidence Aggregation module to our system. After extracting three types of fraud evidences, the next challenge is how to combine them for ranking fraud detection. Indeed, there are many ranking and evidence aggregation methods in the literature, such as permutation based models score based models and Dempster-Shafer rules . However, some of these methods focus on learning a global ranking for all candidates. This is not proper for detecting ranking fraud for new Apps. Other methods are based on supervised learning techniques, which depend on the labeled training data and are hard to be exploited. Instead, we propose an unsupervised approach based on fraud similarity to combine these evidences. The effective evidences should rank leading sessions from a similar conditional distribution, while poor evidences will lead to a more uniformly random ranking distribution

6. RESULTS:



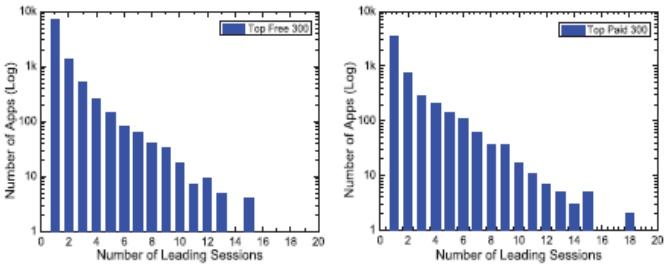
(a) Top Free 300 data set (b) Top Paid 300 data set

Fig: The distribution of the number of apps with respect to different rankings.



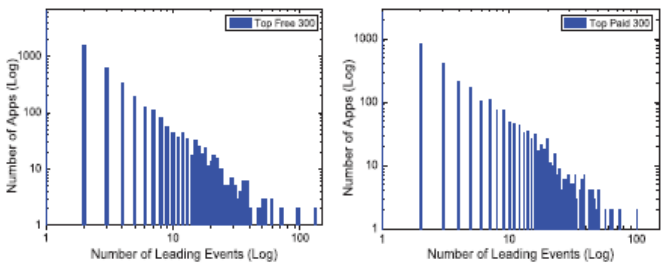
(a) Top Free 300 data set (b) Top Paid 300 data set

Fig: The distribution of the number of apps with respect to different numbers of ratings.



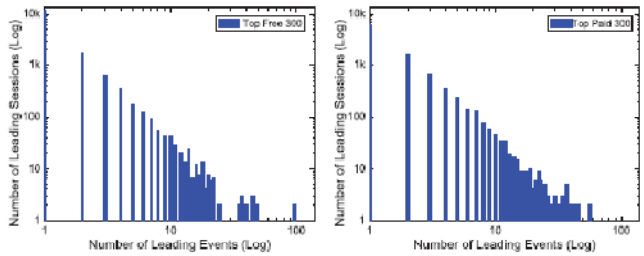
(a) Top Free 300 data set (b) Top Paid 300 data set

Fig: The distribution of the number of apps with respect to different number of leading sessions.



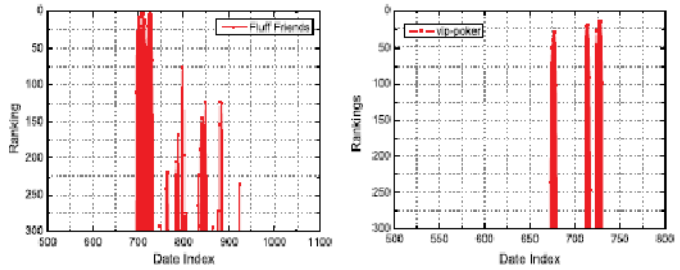
(a) Top Free 300 data set (b) Top Paid 300 data set

Fig: The distribution of the number of apps with respect to different numbers of leading events.



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

Fig: The distribution of the number of leading sessions with respect to different number of leading events



(a) Fluff Friends (b) VIP Poker

(c) Tiny Pets (d) Crime City

Fig: The demonstration of the ranking records of four reported suspicious apps.

	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

Table: Historical data of top free and paid apps

5. CONCLUSION AND FUTURE WORK

In this paper, we analyzed ranking fraud detection model for mobile applications. Currently a large number of mobile application engineers use distinctive fraud frameworks to create their rank. To prevent this, there are distinctive fraud identifying techniques which are introduced in this paper. Such systems are collected into three classes, for instance, web ranking fraud recognition, online review fraud discovery, mobile application recommendation. Each one of these techniques is feasibly handling ranking fraud detection. Besides, it optimized based aggregation technique to integrate all the evidences for assessing the believability of leading sessions from mobile Apps. The recommendation system works for the mobile application recommendation system. The proposed system implements optimization based on admin verification method for evaluating the credibility of leading sessions from mobile Apps. An unique perspective of this approach is that all the evidences can be model by statistical hypothesis tests, thus it is easy to be extended with other evidences from domain knowledge to detect ranking fraud. The admin can detect the ranking fraud for mobile application. The Review or Rating or Ranking given by users is correctly calculated. Hence, a new user who wants to download an app for some purpose can get clear view about the available applications.

REFERENCES

[1] Hengshu Zhu, Hui Xiong,Yong Ge, and Enhong Chen, "Discovery of Ranking Fraud for Mobile Apps"in Proc. IEEE 27th Int. Conf. Transactions on knowledge and data engineering, 2015, pp. 74-87.

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

[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] 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] Klementiev, D. Roth, and K. Small, "An unsupervised learning algorithm for rank aggregation," in Proc. 18th Eur. Conf. Mach. Learn., 2007, pp. 616–623.

[10] M.N Volkovs and R.S. Zemel, "A flexible generative model for preference aggregation," in proc. 21st int. conf. world wide web, 2012, pp. 479-488.

[11] N. Spirin and J. Han, "survey on web spam detection: principles and algorithms," SIGKDD explor. Newslett. vol.13, no. 2, pp. 50-64, may 2012.

[12] K. Shi and K. Ali, "Getjar mobile application recommendations with very sparse datasets," in proc. 18th ACM SIGKDD int. conf. knowl. Discovery data mining, 2012, pp. 204-212.