International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016 [ICSSCCET 2016]

ISBN	978-81-929866-6-1	**VOL**	02
Website	icssccet.org	**eMail**	icssccet@asdf.res.in
Received	25 – February – 2016	**Accepted**	10 - March – 2016
Article ID	ICSSCCET052	**eAID**	ICSSCCET.2016.052

Ranking Reliable Mobile Application using Evidence Aggregation Technique

V Dhayanithi[1], S Surendhar[2], B Vanitha[3], P Vanitha[4], J K Kiruthika[5]
[1,2,3,4] Final Year B.E (CSE), [5]Faculty (CSE), Karpagam Institute of Technology, Coimbatore, India

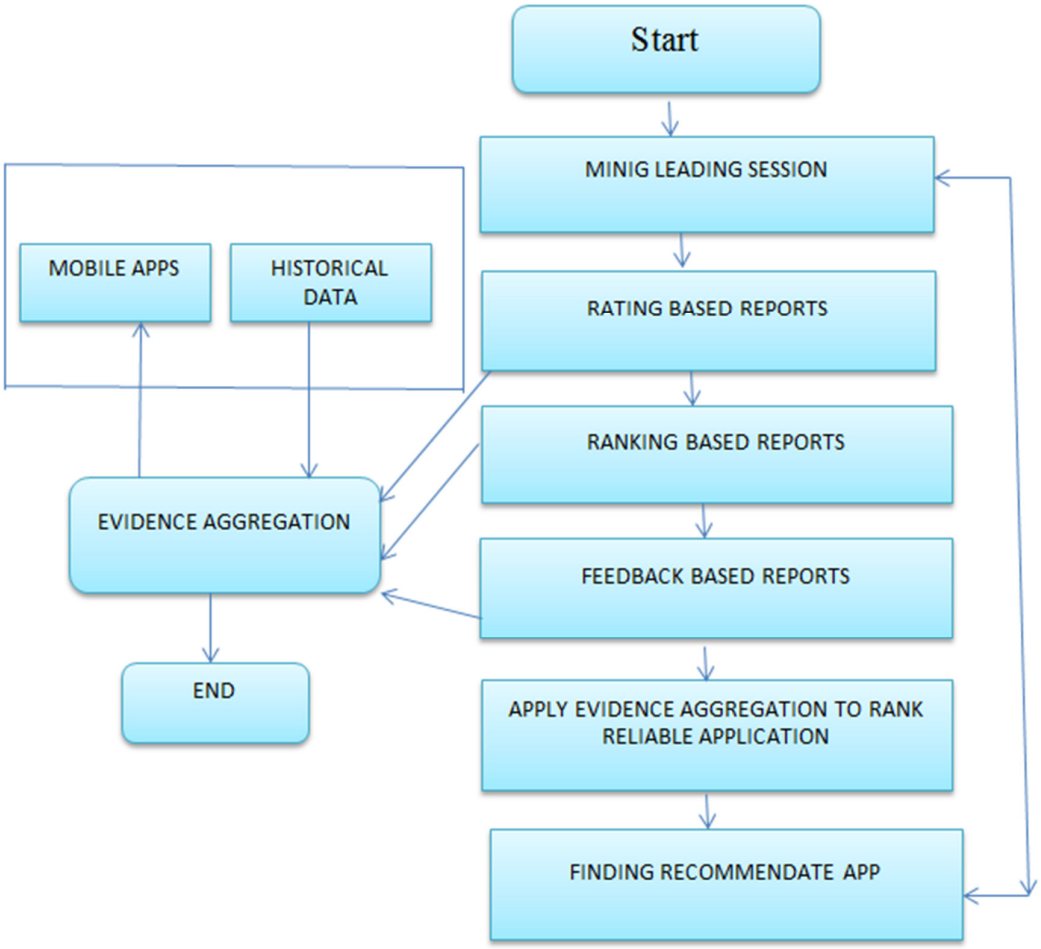
Abstract: *Segregation of reliable and unreliable mobile applications and ranking the reliable Apps based on Ranking, rating and review based techniques. This leads to the importance of preventing ranking reliable application which has been widely recognized, and there is a limited understanding and research in this area. To this end, we propose a system for ranking reliable mobile Apps. To start with locate the ranking reliable Apps by mining the active periods, namely leading sessions, of mobile Apps. Such leading sessions can be control for detecting the local abnormality instead of global abnormality of App rankings. By a proof, we investigate this by three types of evidences, i.e., ranking, rating and feedback based reports, by investigating Apps' ranking, rating and review behaviors through various experimental, proved and statistical tests. In addition, we introduced an optimization based aggregation method to integrate all the evidences for detecting reliable Apps. Finally, we validate the effectiveness of the proposed system, and show the scalability of the detection algorithm.*

Index Terms—*Mobile Apps, ranking reliable mobile Apps, evidence aggregation, historical ranking records, rating and review*

I. INTRODUCTION

The number of mobile Apps has developed at a faster rate over the fewer years. The amount of mobile Apps has developed at an increased rate in the course of recent years. For example, the growth of apps were increased by 1.6 million at Google Play. To indicate the improvement of portable Apps, numerous App stores dispatched for each and every day App leaderboards, which exhibit the rankings of the Apps. Certainly,the App leaderboard is a out standing among the most essential courses for advancing mobile Apps. The highest priority on the leaderboard more often than not satisfies countless and more than million dollars in income. In this way, App designers have a capability to investigate various routes, for instance, promoting battles to advance their Apps to achieve the target to have their Apps valued as high as could be encouraged under certain conditions in such App leaderboards. However, as a reason pattern, instead of depending on various customary showcasing arrangements made, shrouded App engineers resort to some deception intends to help their Apps at the end control of diagram rankings on an App store. This is typically actualized by using the specialized method to overcome the disadvantages of the App downloads, report in a briefly manner. From the recent article of Venture Beat reported that, when an App was developed with the additional feature of positioning various control measures. The real fact such as positioning offense raises amazing worries to the compact App industry. For example, Apple has warning of getting serious about App inventor who consult a positioning exaction in the most recently developed Apple's App store.

Cite this article as: V Dhayanithi, S Surendhar, B Vanitha, P Vanitha, J K Kiruthika. "Ranking Reliable Mobile Application using Evidence Aggregation Technique". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 280-285. Print.



System Architecture

II. Related Work

"An Unsupervised Learning Algorithm for Rank Aggregation", Alexandre Klementiev, Dan Roth, and Kevin Small.

Many applications in information restore, natural language processing, data mining, and related fields require a ranking of occurances with respect to a specified criteria as disaggreeing to a classification. Furthermore, for many such problems, numerous established and well promoted ranking based proposed models have been well described and it is desirable to merge their outcomes into a joint based ranking, a which is denoted as rank aggregation.

"A Review Paper on Mobile App Recommendation & Ranking Fraud Detection on Relationship among Rating, Review & Ranking" by Ms. Neha Het, Prof. Dhananjay Sable

Ranking exaction in the convinient App business sector alludes to fake or misleading exercises which have a motivation behind beating up the Apps in the recognition list. For assurance, it turns out to be more successive for App designers to utilize cloudy means, for example, fashionable their Apps' business or posting fake App appraisals, to submit positioning exaction. While the significance of preventing positioning exaction has been broadly perceived, there is restricted comprehension and examination here. Finally, in this system, we propose an all –encompassing perspective of positioning misrepresentation and intoduce a positioning exaction acceptance framework for portable Apps.

III. Identifying Leading Sessions for Mobile Apps

In this section, we initially introduce some guidelines, and then showing users how to manage leading sessions for mobile Apps from their previous records made. Basically, there are two types of steps dealing with mobile unreliable apps. Firstly, from the Apps previous ranking records, discovery of the most popularised events is completed and then combining of nearest leading events is made

Cite this article as: V Dhayanithi, S Surendhar, B Vanitha, P Vanitha, J K Kiruthika. "Ranking Reliable Mobile Application using Evidence Aggregation Technique". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 280-285. Print.

which is used for constructing most leading sessions. Specialized algorithm is shown from the source code of mining sessions made in the given mobile App and that method can now investigate some of the leading events and sessions by monitoring each and every historical records occurded. [4]

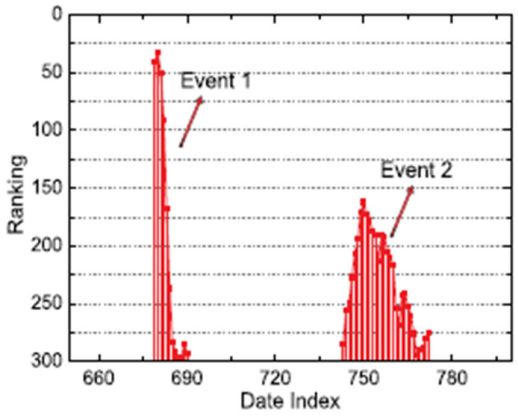
A. Mining Leading Sessions

There are two main steps for mining leading sessions. First, we need to discover leading events from the App's actual, ranking reports. Next step is, we need to merge adjacent leading events for constructing leading sessions.

Definition 1 (Leading Event)

Given a positioning limit a main occasion of App contains a period range. Relating rankings of a, here we apply a locating edge that is normally littler than K here on the grounds that K may be large, and the locating past records are not particularly helpful for recognizing the positioning controls. Moreover, we additionally include the information that is some Apps have a some nearby driving even. Especially, a main occasion which does not have other adjacent neighbours can likewise be handle with as an remarkable driving session.

Note that we apply a ranking inception K_ which is generally lower than K because K may be very big (e.g., more than 1,000), and the ranking records under K_ (e.g., 300) are not very useful for identify the ranking manipulations. Furthermore, we also find that particular Apps have many neighbouring leading events which are close to each other and form a leading session. For example, Fig. 2 shows an example of neighbouring leading events of a given mobile App, which create two leading sessions. Appropriately, a leading event which does not have other nearby neighbours can also be denoted as a peculiar leading session. The proper definition of leading session is as follows:



Leading event

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* $r^a_k \leq K^*$*.*

Note that we apply a ranking threshold K_ which is usually smaller than K here because K may be very big (e.g., more than 1,000), and the ranking records beyond K_ (e.g., 300) are not very useful for detecting the ranking manipulations. Furthermore, we also find that some Apps have several adjacent leading events which are close to each other and form a leading session. For example, Fig. 2b shows an example of adjacent leading events of a given mobile App, which form two leading sessions. Particularly, a leading event which does not have other nearby neighbours can also be treated as a special leading session. The formal definition of leading session is as follows:

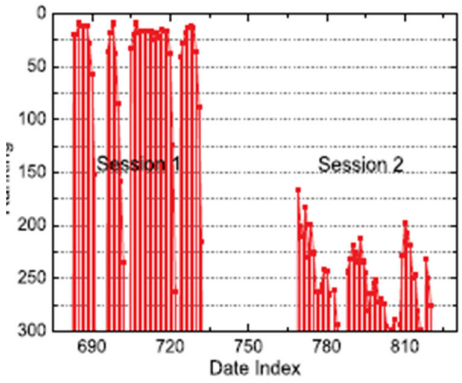
Cite this article as: V Dhayanithi, S Surendhar, B Vanitha, P Vanitha, J K Kiruthika. "Ranking Reliable Mobile Application using Evidence Aggregation Technique". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 280-285. Print.

Definition 2 (Leading Session)

The leading sessions of mobile app denote the popularized period and that consists of ranking evaluation alone. So, the task of identifying ranking unreliable is done by identifying deceptive leading sessions. The main focus is to find the leading sessions of a mobile App from its previously identified ranking records made. [4]

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* $t^s_{start} = t^{e1}_{start}$, $t^s_{end} = t^{en}_{end}$ *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.*

where f is a predefined time threshold for merging leading events. 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.



Leading session

IV. Extracting Evidences for Ranking Reliable Application

In this section, we study how to extract and combine deception evidences for ranking reliable mobile application.

A. Rating Based Evidences

The ranking based evidences strengthen ranking based reliable application. Unluckily, it is not enough to utilize ranking based evidences reports and survey made. For example, some of the apps are created by the famous engineers with large values of u1 due to the high credible value. Even some of the legal market based services may cause important ranking based proofs. To overcome this, we analyze how to seggregate fault evidences from Apps' history of rating records made. Once an App has been proposed rating is done by any user who identified it. user based rating is one of the significant features of App advertised media An App which has more rating may inspire users to download and ranking is higher in the leaderboard. Thus, rating evaluation is also an focussed task of ranking reliable. If an App has most prioritized ranking in one of the session s, the ratings in some time period o may have uneven patterns with its past ratings, for constructing rating based proof. For example, the average rating of hike, line, respectively. We can found that a normal App always receives almost same average rating per day, while abnormal App gets higher average ratings in some of the time periods made

B. Ranking Based Evidences

The leading session consists of more leading events. We should first understand the basic features of leading events for detecting insane evidences. By extracting the Apps' past ranking records, we found that behaviors in a leading event ensure a completely specificied ranking pattern, which constitutes of more than two ranking phases ie., rising and recession phase. As specified , for every leading

Cite this article as: V Dhayanithi, S Surendhar, B Vanitha, P Vanitha, J K Kiruthika. "Ranking Reliable Mobile Application using Evidence Aggregation Technique". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 280-285. Print.

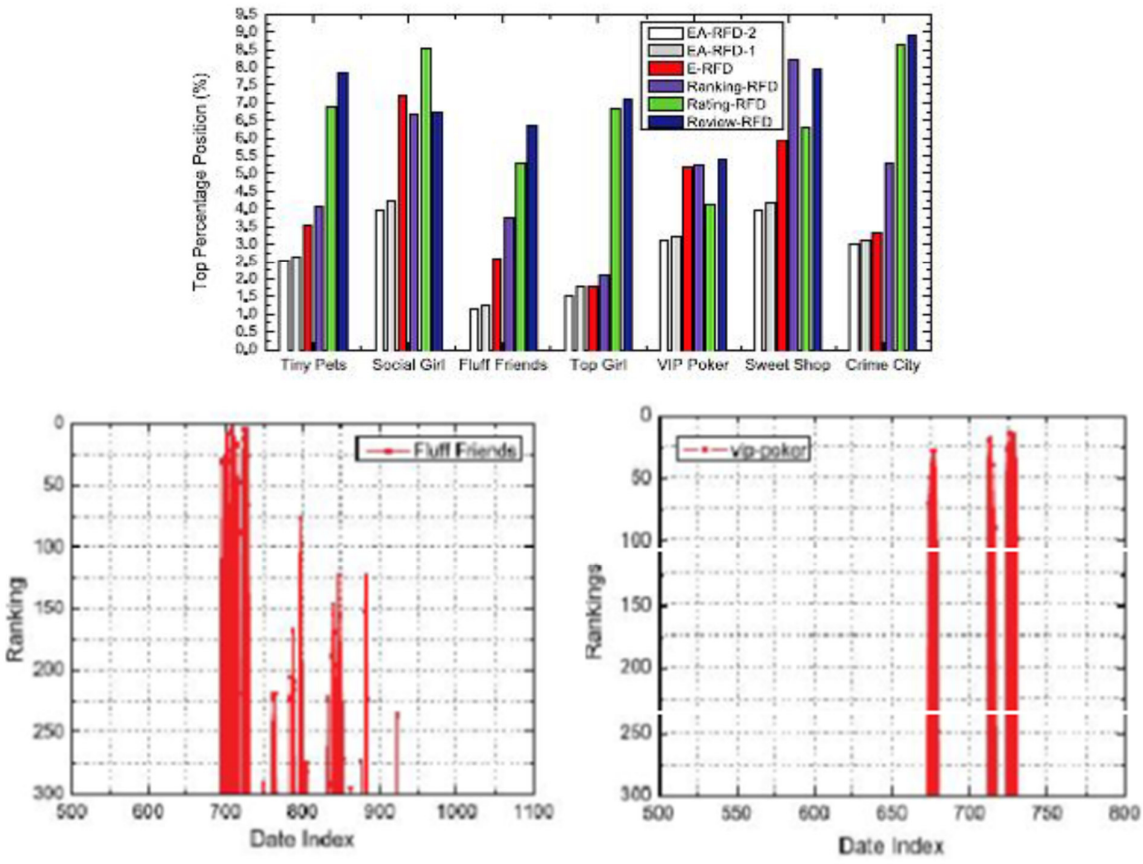
event, an App's ranking initially rises to a peak place in the leaderboard, keeping such peak position for a certain amount of time period (i.e., maintaining phase), and finally reduces till the end of the event session . However ranking pattern highlights the important leading event. Defining the most three ranking phases made in the leading event.

C. Feedback Based Evidences

Under ratings, many of the App stores also allow users to write some textual comments as App feedbacks. Such feedback can reflect the personal approaches and usage experiences of existing users for particular mobile Apps. Certainly, feedback manipulation is one of the most important overview of App ranking fraud. Specifically, before downloading or buying a new mobile App, users repeatedly read its actual reviews to ease their decision making, and a mobile App includes more positive reviews may influenced by more users to download. Therefore, imposters often post fake reviews in the popular sessions of a particular App in order to inflate the App downloads, and thus propel the App's ranking Position in the leader board. Although some previous works on review spam detection have been reported in recent days the problem of identify the local abnormality of comments and feedback in the leading sessions and capturing them as evidences for ranking reliable application are still under-explored. To this end, here we propose two evidences depends on Apps' feedback behaviours in popular sessions for detecting and ranking reliable applications.

V. Evidence Aggregation

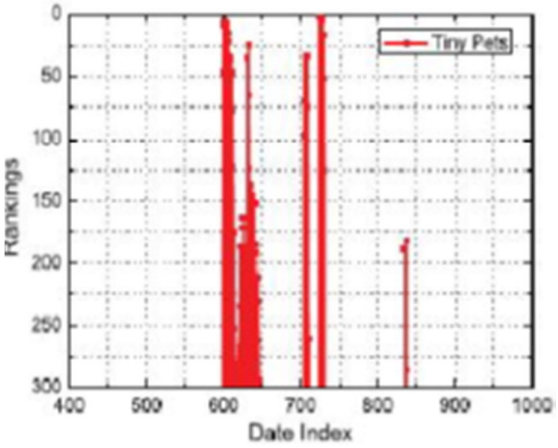
After extracting ranking and rating based fraud evidences, the next challenge is how to combine them for ranking reliable application. Certainly, there are multiple ranking and report aggregation methods in the literature, such as permutation based models, score based models. However, some of these methods focus on learning a global ranking for all candidates. This is not proper for detecting and ranking reliable for new Apps. Other methods are based on supervised learning techniques, which based on the labelled training data and are difficult to be exploited. Instead, we propose an unsupervised approach based on ranking similarity to combine these evidences. Specifically, we define the ultimate report score $\Psi(s)$ as a linear combination of all the existing reports.



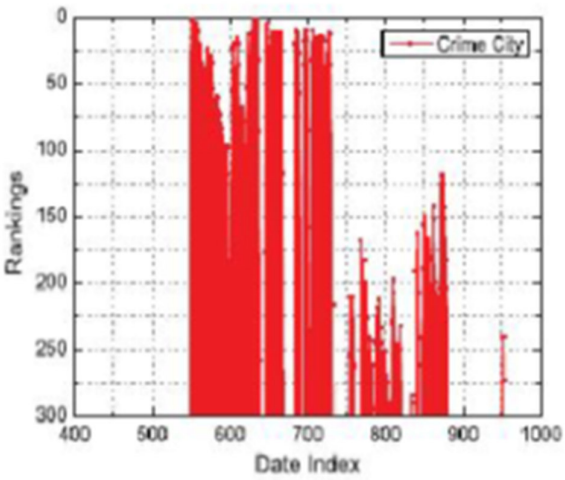
(a) Fluff Friends

(b) VIP Poker

Cite this article as: V Dhayanithi, S Surendhar, B Vanitha, P Vanitha, J K Kiruthika. "Ranking Reliable Mobile Application using Evidence Aggregation Technique". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 280-285. Print.



(c) Tiny Pets



(d) Crime City

VI. Efficiency and Robustness of Our Approach

The proposed framework is scalable and can be extended with other domain generated evidences for ranking reliable application. Experimental results show the effectiveness of the proposed system, the adaptability of the detection algorithm as well as some regularity of ranking fake activities. It can Detect Fake ranking in daily App leader boards and it avoids ranking manipulation. Meanwhile, a learning process is required for evidence aggregation. After revised the aggregation model on a actual data set, each new test App can reuse this model for ranking reliable application. However, it is still not clear how many learning data are required. To study this problem and validate the robustness of our approach, we first rank all leading sessions by modelling with weight parameters learnt from the complete data set. Then we also rank all popular sessions by modelling with weight constants learnt from different subdivisions of the entire data set. We can also test the Root Mean Squared Error (RMSE) of the ranking of popular sessions between different results, shows the results of robust test on two data objects. We can catch that the aggregation framework does not require a lot of revised data, thus the robustness of our approach is reasonable.

VI. Conclusion and Future Work

We proposed a ranking based reliable system for mobile Apps. We first proved that ranking deception occured in most of the leading sessions and specified a method for identifying the leading sessions for each and every App from its past ranking records. Then, we found ranking, rating and review based evidences for identifying and monitoring unreliable application. Moreover, we developed an optimized based aggregation method to evaluate all the evidences for calculating the credible values of leading sessions made from mobile Apps. An unique view of this approach is that all the proof can be verified by statistical and experimental tests, thus it is easy to be combined with other evidences and proofs from expert knowledge to identify ranking deception. Finally, we verify the proposed system with extended experiments on real-time App data gathered from the Apple's store. Proved results enhanced the effectiveness of the proposed approach. In the future decades, our aim is to study more effective fraud proof and analyze the relationship among review, rating and rankings. Moreover, we will extend our approach with other mobile App related services for efficient user experience.

References

1. Enhong Chen, Available:http://www.researchgate.net/publication.
2. Detecting product review spammers using rating behaviors.E.P.Lim, V.A.Nguyen,N.Jindal,B.Liu and H.W.Lauw.
3. Hengshu zhu,Hui Xiong,Senior member,IEEE ,Yong Ge and Enhong Chen ,Senior member IEEE.
4. A.Klementiev, D.Roth and K.Small. "An Unsupervised learning algorithm for rank aggregation".
5. A.Klementiev, D.Roth and K.Small. "An Unsupervised rank aggregation with distance-based models".
6. J.Kivinen and M.K.Warmuth. "Additive versus exponentiated gradient updates for linear prediction".

Cite this article as: V Dhayanithi, S Surendhar, B Vanitha, P Vanitha, J K Kiruthika. "Ranking Reliable Mobile Application using Evidence Aggregation Technique". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 280-285. Print.