ENGINEERING IN ADVANCED RESEARCH SCIENCE AND TECHNOLOGY

ISSN 2278-2566
Vol.02, Issue.03
October -2018
Pages: -137-142

DETECTING RANKING FRAUD FOR MOBILE APPS AND MALWARE DETECTION

R.Ashok[1], Sambasiva Rao.P[2],

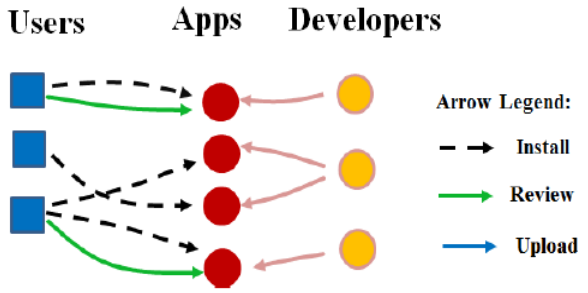
*[1] M. Tech., Dept of CSE, Sri Sunflower College of Engineering and Technology,*ashokr36012@gmail.com
[2] M. Tech. Asst Prof, Dept of CSE. Sri Sunflower College of Engineering and Technology,
sambasiva.phd@gmail.com

ABSTRACT
The most famous Android application advertise, fuel look rank maltreatment and malware multiplication. To recognize malware, past work has concentrated on application executable and authorization investigation. In this paper, we give an all encompassing perspective of positioning misrepresentation and propose a positioning extortion location framework for versatile Apps. In particular, we initially propose to precisely find the positioning extortion by mining the dynamic time frames, to be specific driving sessions, of portable Apps. Such driving sessions can be utilized for identifying the nearby oddity rather than worldwide irregularity of App rankings. Besides, we examine three sorts of confirmations, i.e., positioning based confirmations, rating based confirmations and audit based confirmations, by demonstrating Apps' positioning, rating and survey practices through factual speculations tests. Moreover, we propose an improvement based conglomeration strategy to incorporate every one of the confirmations for misrepresentation location. At last, we assess the proposed framework with true App information gathered from the iOS App Store for quite a while period. In the examinations, we approve the adequacy of the proposed framework, and demonstrate the adaptability of the discovery calculation and in addition some consistency of positioning misrepresentation exercises
Key-Words :-TwitterAndroid market, search rank fraud, malware detection

----

I.INTRODUCTION:-

The commercial success of Android app markets such as Google Play [1] and the incentive model they offer to popular apps, make them appealing targets for fraudulent and malicious behaviors. Some fraudulent developers deceptively boost the search rank and popularity of their apps (e.g., through fake reviews and bogus installation counts) [2], while malicious developers use app markets as a launch pad for their malware [3]–[6]. The motivation for such behaviors is impact: app popularity surges translate into financial benefits and expedited malware proliferation. Fraudulent developers frequently exploit crowd sourcing sites (e.g., Freelancer [7], Fiver [8], Best App Promotion [9]) to hire teams of willing workers to commit fraud collectively, emulating realistic, spontaneous activities from unrelated people (i.e., "crowd turfing" [10]), see Figure 1 for an example. We call this behavior "search rank fraud". In addition, the efforts of Android markets to identify and remove malware are not always successful. For instance, Google Play uses the Bouncer system [11] to remove malware. However, out of the 7, 756 Google Play apps we analyzed using VirusTotal [12], 12% (948) were flagged by
at least one anti-virus tool and 2% (150) were identified as malware by at least 10 tools (see Figure 6). Previous mobile malware detection work has focused on dynamic analysis of app executables [13]–[15] as well as static analysis of code and permissions [16]–[18]. However, recent Android malware analysis revealed that malware evolves quickly to bypass anti-virus tools [19]. In this paper, we seek to identify both malware and search rank fraud subjects in Google Play. This combination is not arbitrary: we posit that malicious developers resort to search rank fraud to boost the impact of their malware. Unlike existing solutions, we build this work on the observation that fraudulent and malicious behaviors leave behind telltale signs on app markets. We uncover these nefarious acts by picking out such trails. For instance, the high cost of setting up valid Google Play accounts forces fraudsters to reuse their accounts across review writing jobs, making them likely to review more apps in common than regular users. Resource constraints can compel fraudsters to post reviews within short time intervals. Legitimate users affected by malware may report unpleasant experiences in their reviews. Increases in the number of requested permissions from one version to the next, which we will call "permission ramps", may indicate benign to malware (Jekyll-Hyde) transitions.



Algorithm 1 Mining Leading Sessions

Input 1: a's historical ranking records R_a;
Input 2: the ranking threshold K^*;
Input 2: the merging threshold ϕ;
Output: the set of a's leading sessions S_a;
Initialization: $S_a = \emptyset$;

1: $E_s = \emptyset$; $e = \emptyset$; $s = \emptyset$; $t^e_{start} = 0$;
2: **for each** $i \in [1, |R_a|]$ **do**
3: **if** $r^a_i \leq K^*$ **and** $t^e_{start} == 0$ **then**
4: $t^e_{start} = t_i$;
5: **else if** $r^a_i > K^*$ **and** $t^e_{start} \neq 0$ **then**
6: //found one event;
7: $t^e_{end} = t_{i-1}$; $e =< t^e_{start}, t^e_{end} >$;
8: **if** $E_s == \emptyset$ **then**
9: $E_s \cup = e$; $t^s_{start} = t^e_{start}$; $t^s_{end} = t^e_{end}$;
10: **else if** $(t^e_{start} - t^s_{end}) < \phi$ **then**
11: $E_s \cup = e$; $t^s_{end} = t^e_{end}$;
12: **else then**
13: //found one session;
14: $s =< t^s_{start}, t^s_{end}, E_s >$;
15: $S_a \cup = s$; $s = \emptyset$ is a new session;
16: $E_s = \{e\}$; $t^s_{start} = t^e_{start}$; $t^s_{end} = t^e_{end}$;
17: $t^e_{start} = 0$; $e = \emptyset$ is a new leading event;
18: **return** S_a

II. Literature Survey:-

1) A flexible generative model for preference aggregation

AUTHORS: M. N. Volkovs and R. S. Zemel

Many areas of study, such as information retrieval, collaborative filtering, and social choice face the preference aggregation problem, in which multiple preferences over objects must be combined into a consensus ranking. Preferences over items can be expressed in a variety of forms, which makes the aggregation problem difficult. In this work we formulate a flexible probabilistic model over pairwise comparisons that can accommodate all these forms. Inference in the model is very fast, making it applicable to problems with hundreds of thousands of preferences. Experiments on benchmark datasets demonstrate superior performance to existing methods .

2) Getjar mobile application recommendations with very sparse datasets

AUTHORS: K. Shi and K. Ali

The Netflix competition of 2006 [2] has spurred significant activity in the commendations field, particularly in approaches using latent factor models [3,5,8,12] However, the near ubiquity of the Netflix and the similar MovieLens datasets1 may be narrowing the generality of lessons learned in this field. At GetJar, our goal is to make appealing recommendations of mobile applications (apps). For app usage, we observe a distribution that has higher kurtosis (heavier head and longer tail) than that for the aforementioned movie datasets. This happens primarily because of the large disparity in resources available to app developers and the low cost of app publication relative to movies.

In this paper we compare a latent factor (PureSVD) and a memory-based model with our novel PCA-based model, which we call Eigenapp. We use both accuracy and variety as evaluation metrics. PureSVD did not perform well due to its reliance on explicit feedback such as ratings, which we do not have. Memory-based approaches that perform vector operations in the original high dimensional space over-predict popular apps because they fail to capture the neighborhood of less popular apps. They have high accuracy due to the concentration of mass in the head, but did poorly in terms of variety of apps exposed. Eigenapp, which exploits neighborhood information in low dimensional spaces, did well both on precision and variety, underscoring the importance of dimensionality reduction to form quality neighborhoods in high kurtosis distributions.

3) Detecting spam web pages through content analysis

AUTHORS: A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly

In this paper, we continue our investigations of "web spam": the injection of artificially-created pages into the web in order to influence the results from search engines, to drive traffic to certain pages for fun or profit. This paper considers some previously-undescribed techniques for automatically detecting spam pages, examines the effectiveness of these techniques in isolation and when aggregated using classification algorithms. When combined, our heuristics correctly identify 2,037 (86.2%) of the 2,364 spam pages (13.8%) in our judged collection of 17,168 pages, while misidentifying 526 spam and non-spam pages (3.1%).

4) Spotting opinion spammers using behavioral footprints

AUTHORS: A. Mukherjee, A. Kumar, B. Liu, J. Wang, M. Hsu, M. Castellanos,
and R. Ghosh

Opinionated social media such as product reviews are now widely used by individuals and organizations for their decision making. However, due to the reason of profit or fame, people try to game the system by opinion spamming (e.g., writing fake reviews) to promote or to demote some target products. In recent years, fake review detection has attracted significant attention from both the business and research communities. However, due to the difficulty of human labeling needed for supervised learning and evaluation, the problem remains to be highly challenging. This work proposes a novel angle to the problem by modeling spamicity as latent. An unsupervised model, called Author Spamicity Model (ASM), is

proposed. It works in the Bayesian setting, which facilitates modeling spamicity of authors as latent and allows us to exploit various observed behavioral footprints of reviewers. The intuition is that opinion spammers have different behavioral distributions than non-spammers. This creates a distributional divergence between the latent population distributions of two clusters: spammers and non-spammers. Model inference results in learning the population distributions of the two clusters. Several extensions of ASM are also considered leveraging from different priors. Experiments on a real-life Amazon review dataset demonstrate the effectiveness of the proposed models which significantly outperform the state-of-the-art competitors.

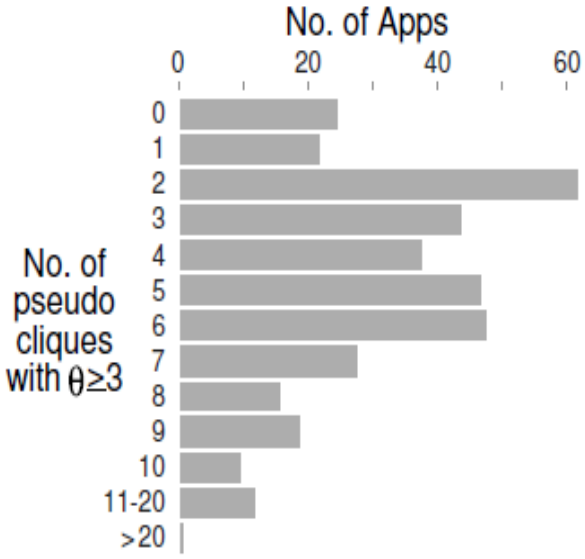
5) Unsupervised rank aggregation with domain-specific expertise

AUTHORS: A. Klementiev, D. Roth, K. Small, and I. Titov

Consider the setting where a panel of judges is repeatedly asked to (partially) rank sets of objects according to given criteria, and assume that the judges' expertise depends on the objects' domain. Learning to aggregate their rankings with the goal of producing a better joint ranking is a fundamental problem in many areas of Information Retrieval and Natural Language Processing, amongst others. However, supervised ranking data is generally difficult to obtain, especially if coming from multiple domains. Therefore, we propose a framework for learning to aggregate votes of constituent rankers with domain specific expertise without supervision. We apply the learning framework to the settings of aggregating full rankings and aggregating top-k lists, demonstrating significant improvements over a domain-agnostic baseline in both cases.

III. Related Work:-

Generally speaking, the related works of this study can be grouped into three categories. The first category is about web ranking spam detection. Specifically, the web ranking spam refers to any deliberate actions which bring to selected webpages an unjustifiable favorable relevance or importance [30]. For example, Ntoulas et al. [22] have studied various aspects of content-based spam on the web and presented a number of heuristic methods for detecting content based spam. Zhou et al. [30] have studied the problem of unsupervised web ranking spam detection. Specifically, they proposed an efficient online link spam and term spam detection methods using spam city.Recently, Spirin and Han [25] have reported a survey on web spam detection, which comprehensively introduces the principles and algorithms in the literature. Indeed, the work of web ranking spam detection is mainly based on the analysis of ranking principles of search engines, such as Page Rank and query term frequency. This is different from ranking fraud detection for mobile Apps. The second category is focused on detecting online review spam. For example, Lim et al. [19] have identified several representative behaviors of review spammers and model these behaviors to detect the spammers. Wu et al. [27] have studied the problem of detecting hybrid shilling attacks on rating data. The proposed approach is based on the semi supervised learning and can be used for trustworthy product recommendation. Xie et al. [28] have studied the problem of singleton review spam detection. Specifically, they solved this problem by detecting the co-anomaly patterns in multiple review based time series. Although some of above approaches can be used for anomaly detection from historical rating and review records, they are not able to extract fraud evidences for a given time period (i.e., leading session).Finally, the third category includes the studies on mobile App recommendation. For example, Yan and Chen [29] developed a mobile App recommender system, named App joy, which is based on user's App usage records to build a preference matrix instead of using explicit user ratings. Also, to solve the sparsely problem of App usage records, Shi and Ali [24] studied several recommendation models and proposed a content based collaborative filtering model, named Eigenapp, for recommending Apps in their website Getjar.In addition, some researchers studied the problem of exploiting enriched contextual information for mobile App recommendation. For example, Zhu et al. [32] proposed a uniform framework for personalized context-aware recommendation, which can integrate both context independency and dependency assumptions. However, to the best of our knowledge, none of previous works has studied the problem of ranking fraud detection for mobile Apps.



IV: Conclusion:-

We have introduced FairPlay, a system to detect both fraudulent and malware Google Play apps. Our experiments on a newly contributed ongitudinal app dataset, have shownthat a high percentage of malware is involved in search rank

fraud; both are accurately identified by FairPlay. In addition, we showed Fair Play's ability to discover hundred of apps that evade Google Play's detection technology, including a new type of coercive fraud attack.

V. Future Enhancement and Conclusion

In this paper, we built up a positioning extortion identification framework for versatile Apps. In particular, we previously demonstrated that positioning extortion occurred in driving sessions and gave a technique for digging driving sessions for each App from its authentic positioning records. At that point, we recognized positioning based confirmations, rating based confirmations and survey based confirmations for distinguishing positioning misrepresentation. Besides, we proposed an advancement based conglomeration technique to coordinate every one of the confirmations for assessing the believability of driving sessions from portable Apps. A special point of view of this methodology is that every one of the confirmations can be demonstrated by factual speculation tests, along these lines it is anything but difficult to be expanded with different confirmations from area information to recognize positioning extortion. At last, we approve the proposed framework with broad analyses on true App information gathered from the Apple's App store. Test results demonstrated the viability of the proposed methodology. Later on, we intend to consider more compelling extortion confirmations what's more, break down the inert relationship among rating, survey and rankings. Additionally, we will broaden our positioning misrepresentation recognition approach with other versatile App related administrations, for example, versatile Apps suggestion, for improving client encounter.

VI.REFERENCES:-

[1] (2014). [Online]. Available: http://en.wikipedia.org/wiki/cohen's_kappa

[2] (2014). [Online]. Available: http://en.wikipedia.org/wiki/information_retrieval

[3] (2012). [Online]. Available: https://developer.apple.com/news/index.php?id=02062012a

[4] (2012). [Online]. Available: http://venturebeat.com/2012/07/03/apples-crackdown-on-app-ranking-manipulation/

[5] (2012). [Online]. Available: http://www.ibtimes.com/applethreatens-crackdown-biggest-app-store-ranking-fra ud-406764

[6] (2012). [Online]. Available: http://www.lextek.com/manuals/onix/index.html

[7] (2012). [Online]. Available: http://www.ling.gu.se/lager/mogul/porter-stemmer.

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

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," J. Mach. Learn. Res., pp. 993–1022, 2003.

[10] 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.

[11] 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.

[12] T. L. Griffiths and M. Steyvers, "Finding scientific topics," Proc.Nat. Acad. Sci. USA, vol. 101, pp. 5228–5235, 2004.

[13] 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.

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

[15] 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.

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

[17] A. Klementiev, D. Roth, and K. Small, "Unsupervised rank aggregation with distance-based models," in Proc. 25th Int. Conf. Mach. Learn., 2008, pp. 472–479.

[18] A. Klementiev, D. Roth, K. Small, and I. Titov, "Unsupervised rank aggregation with domain-specific expertise," in Proc. 21st Int. Joint Conf. Artif. Intell., 2009, pp. 1101–1106.

[19] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers using rating behaviors," in Proc. 19thACMInt. Conf. Inform. Knowl. Manage., 2010, pp. 939–948.

[20] Y.-T. Liu, T.-Y. Liu, T. Qin, Z.-M. Ma, and H. Li, "Supervised rank aggregation," in Proc. 16th Int. Conf. World Wide Web, 2007, pp. 481–490.

[21] A. Mukherjee, A. Kumar, B. Liu, J. Wang, M. Hsu, M. Castellanos, and R. Ghosh, "Spotting opinion spammers using behavioral footprints," in Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2013, pp. 632–640.

[22] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly, "Detecting spam web pages through content analysis," in Proc. 15th Int. Conf. World Wide Web, 2006, pp. 83–92.

[23] G. Shafer, A Mathematical Theory of Evidence. Princeton, NJ, USA:Princeton Univ. Press, 1976.

[24] 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.
[25] 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.
[26] 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.
[27] Z. Wu, J. Wu, J. Cao, and D. Tao, "HySAD: A semi-supervised hybrid shilling attack detector for trustworthy product recommendation," in Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2012, pp. 985–993.
[28] S. Xie, G. Wang, S. Lin, and P. S. Yu, "Review spam detection via
temporal pattern discovery," in Proc. 18th ACM SIGKDD Int. Conf.
Knowl. Discovery Data Mining, 2012, pp. 823–831.
[29] B. Yan and G. Chen, "AppJoy: Personalized mobile application discovery," in Proc. 9th Int. Conf. Mobile Syst., Appl., Serv., 2011, pp. 113–126.
[30] B. Zhou, J. Pei, and Z. Tang, "A spamicity approach to web spam detection," in Proc. SIAM Int. Conf. Data Mining, 2008, pp. 277–288.
[31] H. Zhu, H. Cao, E. Chen, H. Xiong, and J. Tian, "Exploiting enriched contextual information for mobile app classification," in Proc. 21st ACMInt. Conf. Inform. Knowl. Manage., 2012, pp. 1617–1621.
[32] H. Zhu, E. Chen, K. Yu, H. Cao, H. Xiong, and J. Tian, "Mining personal context-aware preferences for mobile users," in Proc. IEEE 12th Int. Conf. Data Mining, 2012, pp. 1212–1217.
[33] H. Zhu, H. Xiong, Y. Ge, and E. Chen, "Ranking fraud detection for mobile apps: A holistic view," in Proc. 22nd ACM Int. Conf. Inform. Knowl. Manage., 2013, pp. 619–628.

Rajulapati Ashok is a student of sri sunflower college of Engineering and Techonology, Lankapalli Present he is Pursuing his M.tech[Computer Science & Engineering] from this college and he received B.tech Degree(Bachelor of techonology) from the University of JNTUK,Kakinada.

P.Sambasiva Rao
Associate Professor(Phd) in sunflower college of Engineering and Techonology Lankapalli and Also Received Master Degree from JNTUK University,Having 12 years of Experience in Faculty.