Detection of Ranking Fraud in Mobile Applications

Ms.Manasi Mhatre[1], Ms. Surabhi Mhatre[2], Ms. Dhikshashri Dhemre [3], Prof.Saroja.T.V[4]

[123] B.E. Students, Department of Computer Engineering, Shivajirao S. Jondhale College of Engineering, Dombivli (E)
[4]Associate Professor, Department of Computer Engineering, Shivajirao S. Jondhale College of Engineering, Dombivli (E), Mumbai University, Maharashtra, India

---***---

Abstract - *The Mobile App is an extremely popular and renowned idea as a result of the fast advancement of the mobile technology. As a result of the massive range of mobile Apps, ranking fraud is the key challenge leading of the mobile App market. Ranking fraud refers to fraud or vulnerable activities that have a purpose of bumping up the Apps within the leading list. Whereas the importance and necessity of preventing ranking fraud have been widely known. In this aggregation methodology, we are proposing 2 enhancements. Firstly, by using Approval of scores from the admin to spot the precise reviews and rating scores. Secondly, the faux feedbacks by the same person for pushing up that app on the leaderboard are restricted.*

Key Words**: *Mobile Apps, Ranking Fraud Detection, Evidence Aggregation, Historical Ranking Records, Rating and Review.***

1. INTRODUCTION

Ranking fraud within the mobile app market refers to deceitful or deceptive activities. Nowadays it has become normal for app developers to use any means in order to increase their app rating thus committing ranking fraud. In this paper, we offer a comprehensive view of ranking fraud and propose a ranking fraud detection system for mobile apps. In order to find the ranking fraud, we need to perform extraction on active periods, particularly leading sessions, of mobile Apps. Such leading sessions will be leveraged for investigating the native anomaly and not the international anomaly of app rankings.

Moreover, we tend to investigate 3 kinds of evidence, i.e., ranking based evidence, rating evidence, and review based evidence, by modeling apps' ranking, rating and review behaviors through statistical hypotheses tests. In Rating based Evidence, specifically, when an App has been released, it will be rated by any user who has downloaded it. An App that has higher rating could attract a lot of users. Thus, rating manipulation is a crucial perspective of ranking fraud. In Review based Evidence, most of the App stores allow the users to write down some comments as App reviews.

2. EXISTING SYSTEM

In the existing system, whereas there's some interrelated work, like internet ranking spam detection, online review spam detection, and mobile app recommendation. The issue of detection ranking fraud for mobile apps remain under explored. Usually speaking, the related works of this study are often classified into 3 classes. The first class is regarding internet ranking spam detection. The second class is targeted on detection online review spam. Finally, the third class includes the studies on mobile app recommendation.

3. PROPOSED SYSTEM

In this project, we use effective algorithm to spot the leading sessions of every App depend on its historical ranking records. Then, with the analysis of Apps' ranking behaviors, we discover that the fraud Apps usually have completely different ranking patterns in every leading session compared with normal Apps.

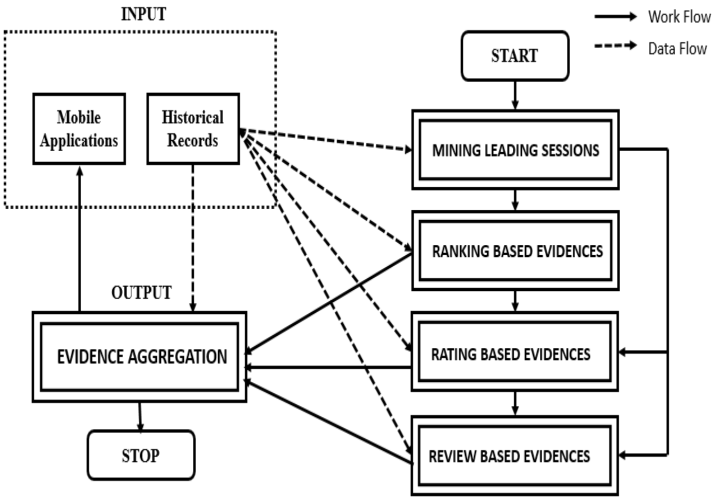


Fig 3.1: The Framework of ranking fraud detection system for mobile Apps

3.1 Identifying Leading Sessions

We need to observe ranking fraud inside leading sessions of mobile Apps then propose an easy but effective algorithm to spot the leading sessions of every App depend on its historical ranking records .After this we discover that the deceitful Apps usually have completely different ranking patterns in every leading session compared with traditional Apps.

3.2 Ranking based Evidences

We need to analyze the fundamental characteristics of leading events for extracting fraud evidences. By analyzing the Apps' historical ranking records, we observe that Apps' ranking behaviors in an exceedingly leading event continuously satisfy a particular ranking pattern, that consist of different ranking phases, namely, rising phase, maintaining phase and recession phase. Specifically, in every leading event, an App's ranking first will increase to a peak position within the leader board, then keeps such peak position for a time being, and finally decreases until the end of the event.

3.3 Rating Based Evidences

User rating is one among the utmost vital features of App promotion. An App that has higher rating could attract a lot of users to download and can even become higher on the leaderboard. Thus, rating manipulation is additionally a crucial perspective of ranking fraud.

3.4 Review Based Evidences

The App stores permit users to write some comments as reviews. Such reviews will project the personal perceptions and usage experiences of existing users for specific mobile Apps. Before downloading or buying a new mobile App, users read its historical reviews & based on lot of positive can download it. Thus, imposters usually post faux reviews within the leading sessions of a particular App so as to inflate the App downloads.

3.5 Evidence Aggregation

The study describes a ranking fraud detection process where there is some evidence considered and integrated to get an aggregated result which is most reliable in finding a fraudulent application in a mobile market[1]. Most generally the ranking fraud is happening in some particular phase of many leading events [1]. A leading event may occur due to an advertisement campaign or etc. This study can be extended to get a recommender system to enhance user experience

4. LITERATURE SURVEY

Agarwal ET. al. [2] examine sentiment analysis on Twitter information. The authors experimented with 3 kinds of models: the unigram model, a feature based model, and a tree kernel-based model. They assumed the unigram model as a baseline. They investigated 2 types of models: tree kernel and feature-based models and demonstrated that each these models outperform the unigram baseline.

David F. Gleich et al. [3] has done a survey on Rank Aggregation via Nuclear Norm minimization process of rank aggregation is intimately tangled with the structure of skew-symmetric matrices. To provide a replacement methodology for ranking a collection of things. The essence of our plan is that a rank aggregation describes a partly stuffed skew-symmetric matrix.

Leif Azzopardi et al. [6] studied an investigation the link between Language Model perplexity and IR precision Recall Measures the perplexity of the language model includes a systematic relationship with the accomplishable precision-recall performance although it's not statistically important. A latent variable unigram based mostly lm that has been successful once applied to IR, is that the so-called probabilistic latent semantic indexing (PLSI).

N. Jindal and B. Liu [7] conferred variety of police work Product Review Spammers mistreatment Rating Behaviors to discover users generating spam reviews or review spammers. We tend to find many characteristic behaviors of review spammers and model these behaviors therefore on discover the spammers.

5. SENTIMENT ANALYSIS ALGORITHM

Sociologists have studied human sentiment for half a century. In the pattern of interaction between people, the meaning of vocabularies has the central role to show people's reaction to each other and also works and other actions meant to evoke a sentimental response from between vocabularies.

The increase of social media like blogs and social networks has fueled interest in sentiment analysis. So as to find the new opportunities and to manage the reputations, business folks typically read the reviews/ ratings/ recommendations and different types of online opinion. this enables to not solely realize the words that are indicative of sentiment however additionally to seek out the relationships between words in order that each word that modifies the sentiment and what the sentiment is regarding will be accurately identified.

Scaling system is employed to work out the sentiment for the words having a positive, negative and neutral sentiment. It also analyzes the following concepts to know the words and the way they relate to the conception.

There are many sentiment analysis algorithms available for developers. Implementing sentiment analysis in your apps is an easy job. There are not any servers to set up, or settings to configure. Sentiment Analysis analyzes the text of news articles, social media posts like Tweets, Facebook, and more. Social Sentiment Analysis is an algorithm that's tuned to research the sentiment of social media content, like tweets and status updates. The algorithm takes a string, and returns the sentiment rating for the "positive," "negative," and "neutral." Additionally, this algorithm provides a compound result that is an overall sentiment of the string.

For this purpose we tend to use Classifiers is Sentiment Analysis is to see the subjective value of a text-document, i.e. however positive or negative is that the content of a text document. Regrettably, for this purpose, these Classifiers fail to determine a similar accuracy. This can be because of the subtleties of human language, irony, context interpretation, use of slang, cultural variations and also the other ways during which opinion will be expressed. In this paper, we are using Naive Bayes classifiers.

5.1 Naive Bayes

Naive Bayes classifiers are studying the classification task from a Statistical point of view. The starting point is that the probability of a class C is given by the posterior probability P (C|D) given a training document D. Here D refers to all of the text in the entire training set. It is given by D= (d_1, d2, .d_n), where d_i is the attribute (word) of document D .Using Bayes' rule, this posterior probability can be rewritten as:

$$P(C = c_i|D) = \frac{P(D|C=c_i) \cdot P(C=c_i)}{P(D)}$$

Since the marginal probability P (D) is equal for all classes, it can be disregarded and the equation becomes:

$$P(C = c_i|D) = P(D|C = c_i) \cdot P(C = c_i)$$

The document D belongs to the class C which maximizes this probability, so:

$$C_{NB} = argmaxP(D|C) \cdot P(C)$$

$$C_{NB} = argmaxP(d_1, d_2, .., d_n|C) \cdot P(C)$$

Assuming conditional independence of the words $d_{i,}$ this equation simplifies to:

$$C_{NB} = argmaxP(d_1|C) \cdot P(d_2|C) \cdot \cdot cdotP(d_n|C) \cdot P(C)$$

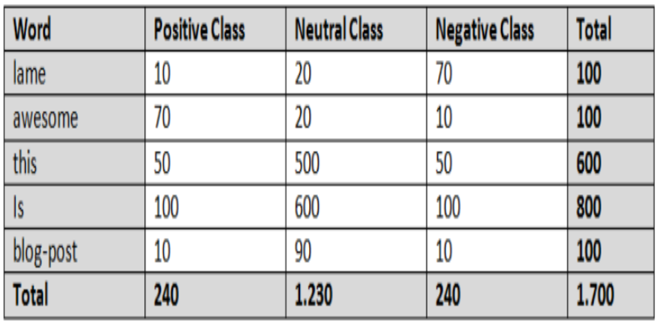
$$C_{NB} = argmaxP(C) \cdot \prod_i P(d_i|C)$$

Here P (d_i | C) is the conditional probability that word i belongs to class C. For the purpose of text classification, this probability can simply be calculated by calculating the frequency of word i in class C relative to the total number of words in class C.

$$P(d_i|C) = \frac{count(d_i,C)}{\sum_i count(d_i,C)}$$

We need to multiply the class probability with all of the prior-probabilities of the individual words belonging to that class. In supervised machine learning algorithm: we can estimate the prior-probabilities with a training set with documents that are already labeled with their classes. With this training set we can train the model and obtain values for the prior probabilities. This trained model can then be used for classifying unlabeled documents.

This is relatively easy to understand with an example. Let's say we have counted the number of words in a set of labeled training documents. In this set each text document has been labeled as either Positive, Neutral or as Negative. The result will then look like:



Word	Positive Class	Neutral Class	Negative Class	Total
lame	10	20	70	**100**
awesome	70	20	10	**100**
this	50	500	50	**600**
Is	100	600	100	**800**
blog-post	10	90	10	**100**
Total	**240**	**1.230**	**240**	**1.700**

From this table we can already deduce each of the class probabilities:

$$P(C_{pos}) = 0.141,$$

$$P(C_{neu}) = 0.723\,,$$

$$P(C_{neg}) = 0.141.$$

If we look at the sentence "This blog-post is awesome.", then the probabilities for this sentence belonging to a specific class are:

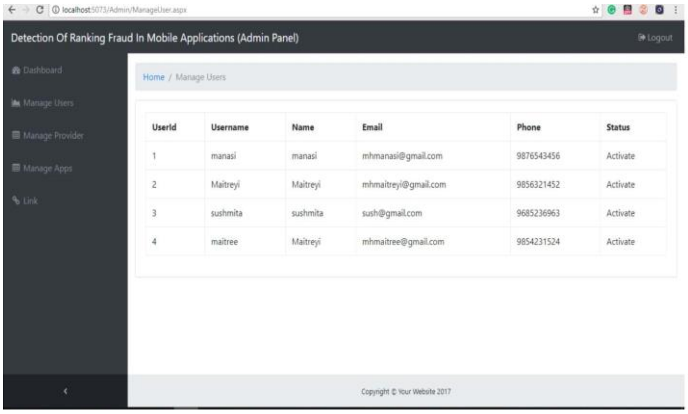
$$P(C_{pos}) = 0.141 \cdot 50/240 \cdot 10/240 \cdot 100/240 \cdot 70/240 = 1.49 \cdot 10^{-3}$$

$$P(C_{neu}) = 0.723 \cdot 500/1230 \cdot 90/1230 \cdot 600/1230 \cdot 20/1230 = 1.71 \cdot 10^{-4}$$

$$P(C_{neg}) = 0.141 \cdot 50/240 \cdot 10/240 \cdot 100/240 \cdot 10/240 = 2.12 \cdot 10^{-4}$$

This sentence can thus be classified in the positive category.

6. SCREENSHOTS

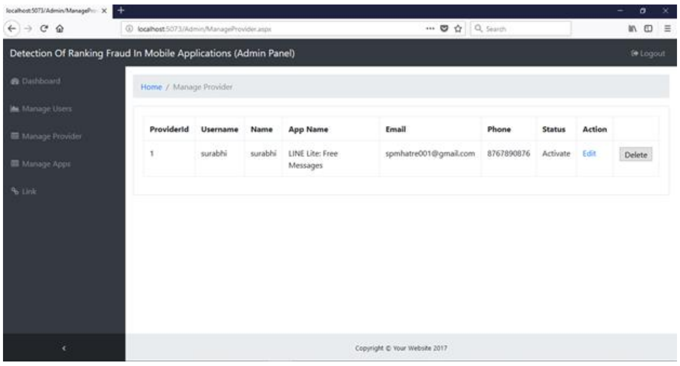


Detection Of Ranking Fraud In Mobile Applications (Admin Panel)

UserId	Username	Name	Email	Phone	Status
1	manasi	manasi	mhmanasi@gmail.com	9876543456	Activate
2	Maitreyi	Maitreyi	mhmaitreyi@gmail.com	9856321452	Activate
3	sushmita	sushmita	sush@gmail.com	9685236963	Activate
4	maitree	Maitreyi	mhmaitree@gmail.com	9854231524	Activate

Copyright © Your Website 2017

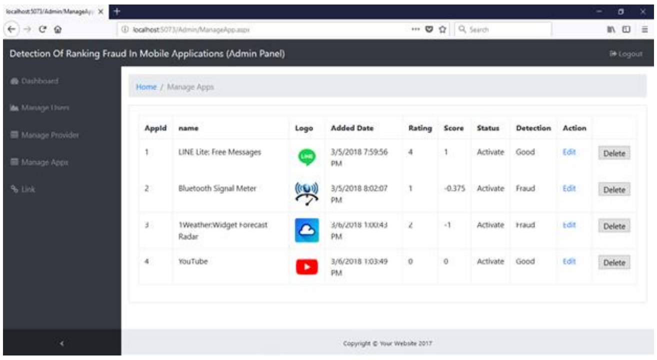
Fig 6.1: Admin Panel (Manage user)



Detection Of Ranking Fraud In Mobile Applications (Admin Panel)

ProviderId	Username	Name	App Name	Email	Phone	Status	Action	
1	surabhi	surabhi	LINE Lite: Free Messages	spmhatre001@gmail.com	8767890876	Activate	Edit	Delete

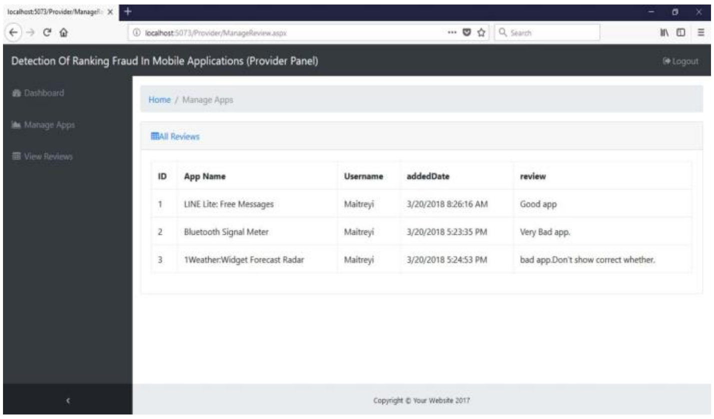
Fig 6.2: Admin Panel (Manage Provider)



Detection Of Ranking Fraud In Mobile Applications (Admin Panel)

AppId	name	Logo	Added Date	Rating	Score	Status	Detection	Action	
1	LINE Lite: Free Messages		3/5/2018 7:59:56 PM	4	1	Activate	Good	Edit	Delete
2	Bluetooth Signal Meter		3/5/2018 8:02:07 PM	1	-0.375	Activate	Fraud	Edit	Delete
3	1Weather:Widget Forecast Radar		3/6/2018 1:00:43 PM	2	-1	Activate	Fraud	Edit	Delete
4	YouTube		3/6/2018 1:03:49 PM	0	0	Activate	Good	Edit	Delete

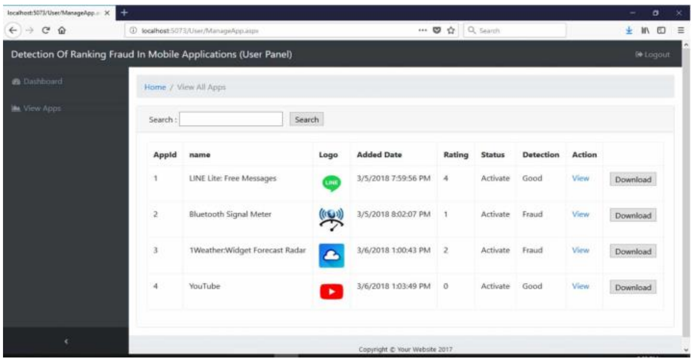
Fig 6.3: Admin Panel (Manage Apps)


localhost:5073/Provider/ManageReview.aspx
Logout
Dashboard
Manage Apps
View Reviews
Home / Manage Apps
All Reviews
Copyright © Your Website 2017

Detection Of Ranking Fraud In Mobile Applications (Provider Panel)

ID	App Name	Username	addedDate	review
1	LINE Lite: Free Messages	Maitreyi	3/20/2018 8:26:16 AM	Good app
2	Bluetooth Signal Meter	Maitreyi	3/20/2018 5:23:35 PM	Very Bad app.
3	1Weather:Widget Forecast Radar	Maitreyi	3/20/2018 5:24:53 PM	bad app.Don't show correct whether.

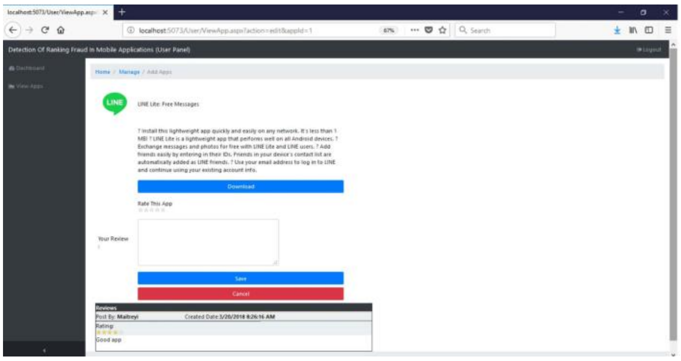
Fig 6.4: Provider Panel


localhost:5073/User/ManageApp.aspx
Logout
Dashboard
View Apps
Home / View All Apps
Search :
Search
Copyright © Your Website 2017

Detection Of Ranking Fraud In Mobile Applications (User Panel)

AppId	name	Logo	Added Date	Rating	Status	Detection	Action	
1	LINE Lite: Free Messages		3/5/2018 7:59:56 PM	4	Activate	Good	View	Download
2	Bluetooth Signal Meter		3/5/2018 8:02:07 PM	1	Activate	Fraud	View	Download
3	1Weather:Widget Forecast Radar		3/6/2018 1:00:43 PM	2	Activate	Fraud	View	Download
4	YouTube		3/6/2018 1:03:49 PM	0	Activate	Good	View	Download

Fig 6.5: User Panel



Detection Of Ranking Fraud In Mobile Applications (User Panel)

LINE Lite: Free Messages

? install this lightweight app quickly and easily on any network. It's less than 1 MB! ? LINE Lite is a lightweight app that performs well on all Android devices. ? Exchange messages and photos for free with LINE Lite and LINE users. ? Add friends easily by entering in their IDs. Friends in your device's contact list are automatically added as LINE friends. ? Use your email address to log in to LINE and continue using your existing account info.

Your Review

Reviews
Post By: Maitreyi Created Date:3/20/2018 8:26:16 AM
Rating:
★★★★☆
Good app

Fig 6.6: User Panel

7. CONCLUSIONS

A unique perspective of this approach is that each one the evidence is model by statistical hypothesis tests, therefore it's simple to be extended with alternative evidence from domain data to discover ranking fraud. The admin will notice the ranking fraud for mobile application. The Review or Rating or Ranking given by users is accurately calculated. Hence, a new user who needs to download an app for a few purposes will get a clear view of the present applications. Finally we tend to validate the proposed system with in-depth experiments on real-world App information collected from the App Store.

8. FUTURESCOPE

In the future, we will decide to study more practical fraud evidence and analyze the latent relationship among rating, review, and rankings. Moreover, we can add more services in ranking fraud detection approach to enhance user experience.

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]. A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, “Sentiment analysis of twitter data,” in Proceedings of the Workshop on Languages in Social Media. Association for Computational Linguistics, 2011, pp. 30–38.

[3]. 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.

[4]. 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.

[5]. 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.

[6]. 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.

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

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

[9]. 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.

[10]. K. Shi and K. Ali. Getjar mobile application recommendations with very sparse datasets. In Proceedings of the 18th ACMSIGKDD international conference on Knowledge discovery and data mining, KDD '12, pages 204–212, 2012.

[11]. 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.

[12]. B. Yan and G. Chen. Appjoy: personalized mobile application discovery. In Proceedings of the 9th international conference on Mobile systems, applications, and services, MobiSys '11, pages 113–126, 2011.

[13]. S. Xie, G. Wang, S. Lin, and P. S. Yu. Review spam detection via temporal pattern discovery. In Proceedings of the 18th ACMSIGKDD international conference on Knowledge discovery and data mining, KDD '12, pages 823–831, 2012.

[14]. Z. Wu, J. Wu, J. Cao, and D. Tao. Hysad: a semi-supervised hybrid shilling attack detector for trustworthy product recommendation. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '12, pages 985–993, 2012

[15]. M. N. Volkovs and R. S. Zemel. A flexible generative model for preference aggregation. In Proceedings of the 21st international conference on World Wide Web, WWW '12, pages 479–488, 2012.

[16]. 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.

[17]. H. Zhu, H. Cao, E. Chen, H. Xiong, and J. Tian, "Exploiting enriched contextual information for mobile app classification," in Proc.21stACMInt. Conf. Inform. Knowl. Manage. 2012, pp. 1617–1621.

[18]. 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.

[19]. N. Spirin and J. Han, "Survey on web spam detection: Principles and algorithms," SIGKDD Explor. Newslett, vol. 13, no. 2, pp. 50-64, May 201.

[20]. 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.K.