SURVEY OF FRAUD GOOGLE APP DETECTION USING COMMUNICATION CLUSTERING MODEL

S.Saranya[1], Mrs S.Sabitha [2]

M.Phil Research Scholar, Vivekanandha College of Arts and Sciences for Women[1], Assistant Professor, Department of Computer Science and Applications ,Vivekanandha College of Arts and Sciences for Women[2]

ABSTRACT

In this paper overview of FairPlay and a one of a kind arranging finds and use follows gone last by fraudsters, to identify together malware and applications exposed to look rank extortion. FairPlay associates inexact traps and totally joins distinguished examination dealings with semantic and conduct signals gathered from Google Play application information in group to perceive suspicious applications. Enemies can have opportunities to dispatch assaults by get-together injured individual's data constantly. This study portray that a foe can effectively derive an injured individual's vertex personality and network character by the learning of degrees inside a timespan. The study likewise prescribe to another administered bunching calculation to discover gatherings of information group. It straightforwardly fuses the data of test classes into the extortion grouping process.

Keywords— Graph Mining, Co-Review Mining, Clustering, FairPlay, Security, Clique location.

I. INTRODUCTION

While vindictive engineers use application advertises as a start cushion for their malware. Fraud and Malware Detection Approach is to identify fraud and malware. co-review pseudo-cliques—formed by reviewers with significantly overlapping co-review tricks across short time windows. The main objectives of the FairPlay are

1. To automatically detect malicious and fraudulent apps.

The achieve the main goal, the specific objectives required are

- To propose review feedbacks approach which exploits feedback left by genuine reviewers?
- To prepare clique from the Co-Review graph so that most related fraudulent users are found out.

II RELATED WORKS

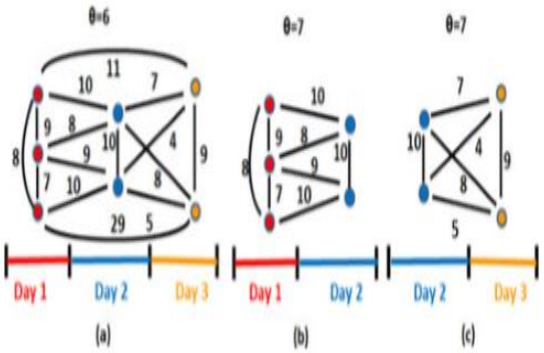


Fig 3.1 Cliques and PCF output

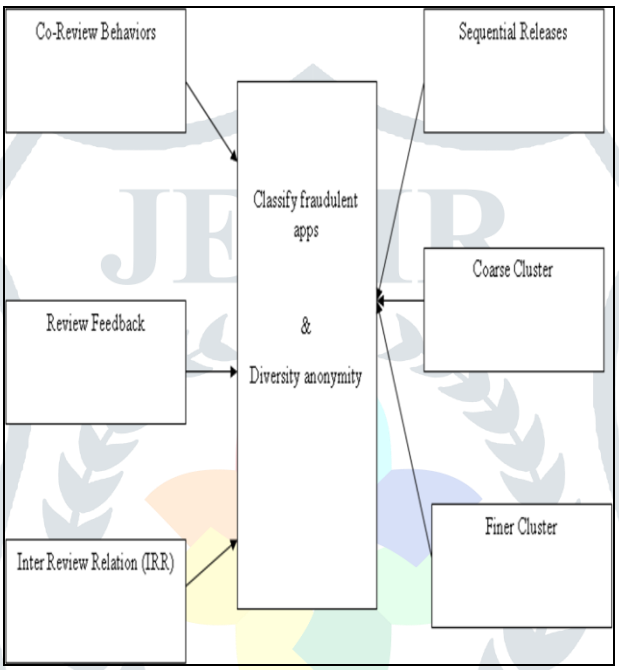


Fig3.2 Architecture Diagram

Here coarse cluster is the generated main graph. Fine cluster is the graph with least connected nodes removed. If a node with all the edge weights below a given threshold, then the edges and that node are removed. The following modules are present in the project.

- Tweets Collection for reviews.
- Co-Review Graph Construction.
- Finding Cliques to get fraud users.
- Remove nodes with edge weights below threshold so normal users are treated as non-fraud users.

Tweets Collection For Reviews

In this module,

- Using twitter package and search twitter function, the tweets are downloaded and preprocessed.
- Stop word removal, punctuation removal, unicode character removal are carried out.
- Key Terms are filtered such that first 50 more occurrence words are taken.
- Then unique users in the tweet are also found out.

Co-Review Graph Construction

In this phase,

- From unique users in the tweet are found out.
- Same Key word present in two topics of two different users are found, then two nodes and one edge is formed in the graph.
- Thus the full graph is constructed. During edge addition, co-occurrence count is also found out and set as edge weight.

Finding Cliques To Get Fraud Users

In this phase,

- From the full graph constructed, cliques are found out with minimum 5 nodes in them.
- These cliques denote the users who are densely connected.
- These users are treated as fraud users.

Remove nodes with edge weights below threshold so normal users are treated as non-fraud users

In this phase,

- One nodes, all edges are taken. If all the edge weights are below the given threshold values, it means the user is giving rating less times only.
- The user is treated as normal user.

IV. EXPERIMENTAL RESULTS

The following **Table 4.1** describes experimental result for Clique and Coarse Cluster analysis. The table contains finding number of Google App usage for attacks in malware social environments are shown.

Table 4.1 Fig 4.1 Clique and Coarse Cluster Performance Analysis

S.NO	Clique Techniques	Coarse Cluster
1	0.16	0.19
2	0.19	0.22
3	0.24	0.29
4	0.31	0.34
5	0.38	0.43
6	0.43	0.49
7	0.50	0.54
8	0.59	0.62
9	0.67	0.69
10	0.72	0.74

The following **Fig 4.1** describes experimental result for Clique and Coarse Cluster analysis. The figure contains finding number of Google App usage for attacks in social environments are shown.

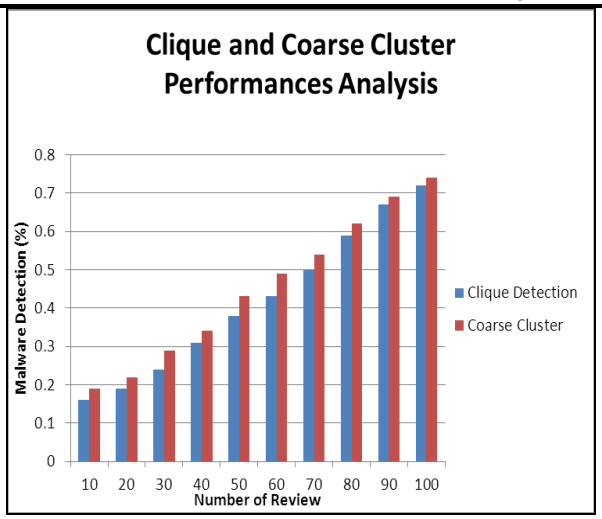


Fig 4.1 Clique and Coarse Cluster Performance Analysis

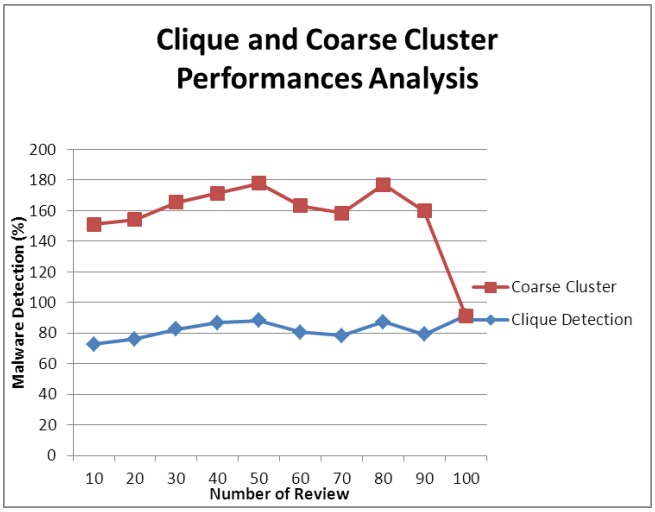
The following **Table 4.2** describes experimental result for Clique and Coarse Cluster error rate analysis. The table contains Number application review and average percentages for CT and CC using malware detection are shown.

Table 4.2 Reduced Error Rate for Clique Detection and Coarse Cluster

Mobile Review	Clique Techniques (%)	Coarse Cluster (%)
10	72.54	78.62
20	76.13	78.11
30	82.42	83.13
40	86.66	84.67
50	88.13	89.78
60	80.44	82.66
70	78.33	80.21
80	87.22	89.76
90	79.22	80.65
100	91.22	92..62

The following **Fig 4.2** describes experimental result for Clique and Coarse Cluster error rate analysis. The figure contains Number application review and average percentages for CT and CC using malware detection are shown.

.



Results:

- The statistical analysis of Malware app injection attacks data if prepared can be used for research development.
- N number of review can be found out easily where the injections are easy found out.
- The multimedia app attacks can also be detected
- The efficiency of the paper is further improved by improving coding efficiency
- In future, the time taken to complete the task is minimized
- Multitasking can also performed

V CONCLUSION

The experiments on the twitter posts, have shown that a high percentage of fraud users are found. In addition, it recommendation for FairPlay's ability to discover non-fraud users also Also, directly discussions are taken for problem area discovery. Live Text streams, for example, dialog messages can be followed and characterization can be embraced. The new framework is structured to such an extent that those improvements can be incorporated with current modules effectively with less mix work and it ends up important if the past upgrades are made in expectation.

REFERENCES

[1] Mahmudur Rahman, Mizanur Rahman, Bogdan Carbunar, and Duen Horng Chau, "Search Rank Fraud and Malware Detection in Google Play", IEEE Transactions On Knowledge And Data Engineering, Vol. 29, NO. 6, June 2017.

[2] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani , "Crowdroid: Behavior-based Malware detection system for Android," in Proc. ACM SPSM, 2011, pp. 15–26.

[3] G.Kesavaraj, S.Sugumaran, "A Study on Classification techniques in data mining" Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT),IEEE Xplore, 4-6 July 2013,

[4] H. Peng, et al., "Using probabilistic generative models for ranking risks of Android Apps," in Proc. ACM Conf. Comput. Commun. Secur., 2012, pp. 241–252.

[5] S. Yerima, S. Sezer, and I. Muttik, "Android Malware detection using parallel machine learning classifiers," in Proc. NGMAST, Sep. 2014, pp. 37–42.

[6] J. Sahs and L. Khan, "A machine learning approach to Android malware detection," in Proc. Eur. Intell. Secur. Inf. Conf., 2012, pp. 141–147.

[7] B. Sanz, I. Santos, C. Laorden, X. Ugarte-Pedrero, P. G. Bringas, and G. Alvarez, "Puma: Permission usage to detect malware in android," in Proc. Int. Joint Conf. CISIS12-ICEUTE' 12-SOCO' Special Sessions, 2013, pp. 289–298.

[8] J. Ye and L. Akoglu, "Discovering opinion spammer groups by network footprints," in Machine Learning and Knowledge Discovery in Databases. Berlin, Germany: Springer, 2015, pp. 267–282.

[9] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine." Computer Networks and ISDN Systems, vol. 30, nos. 1-7, pp. 107-117, 1998.

[10] R. Cai, J.-M. Yang, W. Lai, Y. Wang, and L. Zhang, "iRobot: An Intelligent Crawler for Web Forums," Proc. 17th Int'l Conf. World Wide Web, pp. 447-456, 2008.

[11] A. Dasgupta, R. Kumar, and A. Sasturkar, "De-Duping URLs via Rewrite Rules," Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 186-194, 2008.

[12] C. Gao, L. Wang, C.-Y. Lin, and Y.-I. Song, "Finding Question-Answer Pairs from Online Forums," Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 467-474, 2008.

[13] L. Zhang, B. Liu, S.H. Lim, and E. O'Brien-Strain, "Extracting and Ranking Product Features in Opinion Documents," Proc. 23rd Int'l Conf. Computational Linguistics, pp. 1462-1470, 2010.

[14] M.L.A. Vidal, A.S. Silva, E.S. Moura, and J.M.B. Cavalcanti, "Structure-Driven Crawler Generation by Example," Proc. 29thAnn. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 292-299, 2006.

[15] J.-M. Yang, R. Cai, Y. Wang, J. Zhu, L. Zhang, and W.-Y. Ma, "Incorporating Site-Level Knowledge to Extract Structured Data from Web Forums," Proc. 18th Int'l Conf. World Wide Web, pp. 181-190, 2009.

[16] Y. Zhai and B. Liu, "Structured Data Extraction from the Web based on Partial Tree Alignment," IEEE Trans. Knowledge Data Eng., vol. 18, no. 12, pp. 1614-1628, Dec. 2006.

[17] Mahmudur Rahman, Mizanur Rahman, Bogdan Carbunar, and Duen Horng Chau, "Search Rank Fraud and Malware Detection in Google Play", IEEE Transactions On Knowledge And Data Engineering, Vol. 29, NO. 6, June 2017.

[18] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani , "Crowdroid: Behavior-based Malware detection system for Android," in Proc. ACM SPSM, 2011, pp. 15–26.

[19] A. Shabtai, U. Kanonov, Y. Elovici, C. Glezer, and Y. Weiss, "Andromaly: A behavioral malware detection framework for Androiddevices,"Intell.Inform.Syst.,vol.38,no.1,pp.161–190,2012.

[20] G.Kesavaraj, S.Sugumaran, "A Study on Classification techniques in data mining" Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT),IEEE Xplore, 4-6 July 2013,

[21] H. Peng, et al., "Using probabilistic generative models for ranking risks of Android Apps," in Proc. ACM Conf. Comput. Commun. Secur., 2012, pp. 241–252.

[22] S. Yerima, S. Sezer, and I. Muttik, "Android Malware detection using parallel machine learning classifiers," in Proc. NGMAST, Sep. 2014, pp. 37–42.