A Novel Recommendation System for Mobile Apps Based on Detection of Ranking Frauds

Mrs. Neelima Ambekar, Pratiksha Magar,Shweta Dhodare,Poonam Falley, Priyanka Sonawane

[1]*Mrs . Neelima Ambekar, Information Technology, MCOERC, Maharashtra, India*
[2] *Miss.Pratiksha Magar, Information Technology, MCOERC, Maharashtra, India*
[3] *Miss.Shweta Dhodare, Information Technology, MCOERC, Maharashtra, India*
[4] *Miss.Poonam Falley, Information Technology, MCOERC, Maharashtra, India*
[5] *Miss.Priyanka Sonawane Information Technology, MCOERC, Maharashtra, India*

ABSTRACT

Recently, Smartphone gains more and more popularity not because of its look and feel but peoples are addicted to use various applications of Smartphone. With the use of smartphone apps user also worried to reveal their identity while downloading and installing app in their mobile as many apps may require user information. For example, what's app messenger, hike messenger etc uses contact list information from user cell phone, Google map, drive apps like uber, Ola etc uses contacts, location information of user. Mobile apps mostly varied and less understood typically for their popular functionality. Users are neglecting to install apps as security and privacy point of view. To overcome security issues there is need of such system which gives recommended app list to the end user. User recommendation is based on popularity as well as security concern.

Keyword: - *Mobile Apps,Recommender Systems,Security and Privacy.*

1. INTRODUCTION

Generally, mobile apps are developed using computer programming languages such as, java, android, .net, X-code etc. Apps are designed in such way to match with device layout such as computer, mobile, tabs etc. After successful completion of application programs is using programming language, they have to plot into electronic devices like, mobile, tabs, android wear etc. Application distributed platform is used to relate device and develop apps. In digital market, there are lots of mobile apps are available, few of them are freeware and at the same time others are In-purchased apps. Usually, apps are downloaded and installed in target device such as, android, windows, iPhone etc. The app availability based on the public demand, so the designer tools herd fast and rapid extension of mobile app into various other kinds such as games in mobile, factory computerization, services based on

GPS and based on location, banking, ticket purchasing, etc. There is a challenging issue in mobile app recommendation, because of sudden increase in quantity and the variety of mobile apps which in turn led to the conception of broad range of review and creation sources including blogs, magazines and online app services [1]. Each and every day multiple app stores launched a daily app leader boards that outputs the app ranking chart as per app popularity. Leader board is one of the most popular ways to promote mobile apps [3]. Therefore, according to the ranking on leader board app get downloaded by mobile users. And this is the reason of more advertising of mobile apps by their developers. However, leafy App developers find some fraudulent means to purposely boost their Apps which eventually manipulate the chart rankings on an App store. Mobile app ranking is varied according to its functionality with respect to privacy and security [2].

To provide better experience for end user some apps used personal information of user such as, contact list, location based service to extract user's current location. For instance, users may not expect to share their location as well as their other personal information. According to recent survey, it seems that multiple peoples were not interested to download the apps that required personal information of user and also many of them were uninstalling apps from their mobile phones which have functionality to access private information of user [4-6]. Therefore, there is a need of such a kind of system that provides recommendation of apps for end user with security and privacy perspective. We are going to propose a

recommendation system which work against user privacy and secrecy and then recommend apps for end user. For implementation such kind of system we will have to evaluate risk score i.e. privacy factor. It may be security wise and popularity wise score [2].

However, for security and popularity of apps we need to explore app dataset and required to refine it. And have to evaluate certain kind of permissions as, normal permissions having minimal risk to other applications. Dangerous permissions provides access for private data and finally signature/System permissions which required system signature certifications like, ability to control the overall process of the system. In app mining process we give preferences for rating based apps, review based app and also trying for survey based customer reviews [7]. Which can further combine together for generate recommended app-list for end user. By evaluation of risk score and app mining method we will have to prove efficiency and capability of our system. Our final contribution is detection of ranking fraud detection and app recommendation.

2. LITERATURE SURVEY

In the year 2008 B Zhou, J.Pie and Z.Tang published paper titled "A spamicity approach to web spam detection". In that paper he differentiated the spam mail detection as how we classify the mail as spam or ham. By using the stop words or special words which generally occur in the mail so we can sort them by those words [1]

In year 2009 W. Enck, M. On tang, and P. McDaniel published paper titled On lightweight mobile phone application certification. In that paper the proceedings of the 16th ACM conference on Computer
and communications security was focused on the security level which should be set for apps.[2]

In the year 20011 A.P. Felt, E. Chin, S. Hanna, D. Song, and D.Wagner published Android permissions demystified. In that paper proceedings of the 18th ACM conference on Computer and communications
Security which was purely based on the security issues. [3]

In the year 2012 H. Zhu, H. Cao, E. Chen, H. Xiong, and J. Tian published paper on Exploiting enriched contextual information for mobile app classification. In that paper he made the classification of mobile app depending on there work.[4]

In the year 2013 S. Xie, G. Wang, S. Lin, and P. S. Yu published paper on Review spam detection via temporal pattern discovery which broadly was based on the review detection.[5]

In the year 2013 Matthijs.J Warrens published a paper on Comparison of Cohens Kappa in which he made the aggregation of fraud detection possible[6].

In the year 2015 Hengshu Zhu, Hui Xiong published paper on Discovery of Ranking Fraud for Mobile Apps in which he classified the apps as ranking, rating and review based and found out the leading sessions.[7]

3. PROPOSED SYSTEM

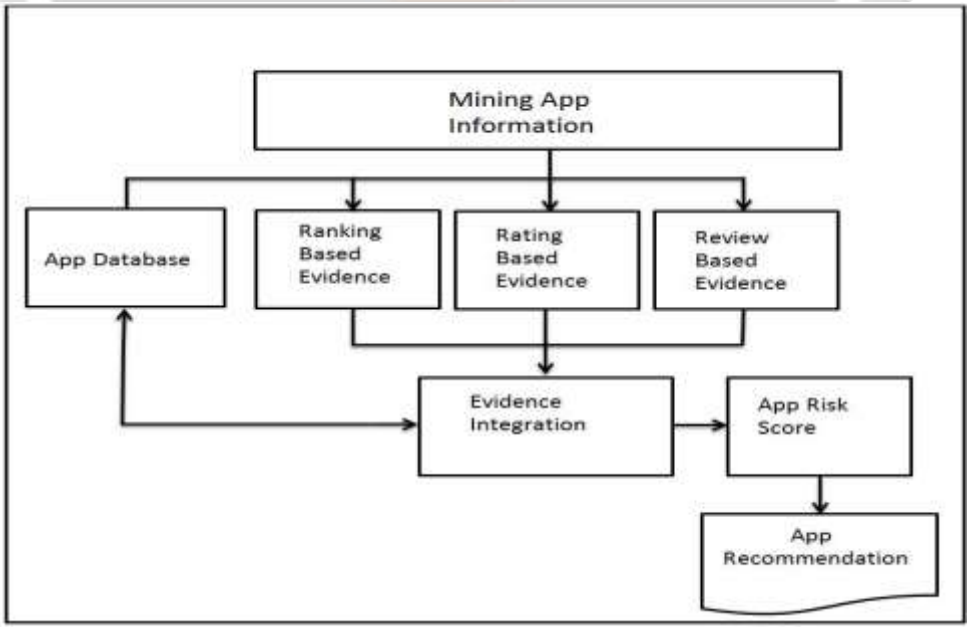


Fig -1. Architecture

3.1. App Mining:

App mining is the procedure of extracting technical, business and customer review and rating information for analysis and prediction i.e. to capture uncertain events. In system architecture diagram app mining method is divided into three parts such as, ranking based, rating based, survey based. It is described as follow:

Ranked based: It contains app information having certain rise and falls in their popularity.

Rating Based: It recognizes fake ratings about app i.e. random comments of users. It contains over wise popularity apps. For more download developer provides some discount or over some gift for users where user are attracted and gives positive comments regarding app.

Review based: Comment wise evidence. In this fraud detection is takes place using stop words to remove it. Basically normalized vector is used to calculate similarity between app functionalities.

3.2. Evidence Integration:

In this the result which is generated through ranking, rating and survey is considered and then it is integrated and that information is passed for calculating the risk score.

3.3. Calculation of Risk Score:

Two types of principle are involved to calculate risk score:

Security principle: App risk score is evaluated and arranged it in ascending order. If risk score is similar of more than one app then it is evaluated or ranked by popularity score.

Popularity principle: Ranking apps are evaluated as per popularity score and arrange it in descending order. If risk score is similar of more than one app then it is evaluated or ranked by security score.

3.4. App Recommendation:

After all processing done regarding security and popularity issues of mobile apps, final stage is to recommend sorted apps to the end user. So, that user can satisfy their need and worries
About app usage.
In our system we use four major components: App Category List, Recommendation List, Search Application, and Update Profile.

1] App Category List: The app category consists of the apps which are fetched from the Google play store and the apps which the user contains in his mobile.

2] Recommendation Application: The recommendation application consists of two types of notification list: First according to the user's age, profession and gender. Second consist of notifying to uninstall the apps which are no longer used by the user.

3] Search Application: It consists of the link of Google play store so that user can download the app according to his/her interest.

4] Update Profile: It contains the user data like age, profession and gender which is used to recommend the app.

4. ALGORITHM

4.1 Ranking Algorithm:

Input 1: a's historical ranking records R_a;

2: the ranking threshold K*;

3: the merging threshold Ø;

Output: the set of a's leading sessions S_a;

Initialization:

S_a = Ø; 1: E_s = Ø; e = Ø; s = Ø; t^e_{start} = 0;

2: for each i ϵ [1, $|R_a|$] do

3: if $r_i{}^a \leq K^*$ and $t^e_{start} == 0$ then

4: $t^e_{start} = t_i$;

5: else if $r_i{}^a > K^*$ and $t^e_{start} \neq 0$ then

6: $t^e_{end} = t_{i-1}$; e=< t^e_{start}, t^e_{end}>;

7: if E_s == Ø then

8: E_s U = e; $t^s_{start} = t^e_{start}$; $t^s_{end} = t^e_{end}$;

9: else if ($t^e_{start} - t^s_{end}$) < Ø then

10: E_s U = e; $t^s_{end} = t^e_{end}$;

11: else then

12: s = < t^s_{start}, t^s_{end}, E_s >;

13: S_aU = s; s = Ø is a new session;

14: E_s = {e}; $t^s_{start} = t^e_{start}$; $t^s_{end} = t^e_{end}$;

15: t^e_{start} = 0; e = Ø is a new leading event;

16: return S_a;

4.2 Fraud Detection Algorithm for Fake Reviews

Step 1: Comparing customer 1 reviews and customer 2 review.
Step 2: Then checking which one is more similar type of reviews exact review then from given reviews one review from that similar review is considered for a particular product.
Step 3: Keeping a criteria that only registered users can give review.
Step 4: Giving preference to top 10 reviews for a particular product that will help the users to decide whether to download that app or not.

5.MATHEMATICAL MODEL

Sys = {ψ1, ψ2, ψ3, ψ4, ψ5}
ψ1 = Data collector app.
This data is collected from sample population decided at start of designing of system.
ψ2= Actual recommender application for end user that analyse user profile and recommend based on clusters
ψ3= Web based handshaking interfaces or services. Handshake between Android and web based app information is necessary
ψ4= Application for super user of the system.
Ψ5= Application permission and security.
ψ1= {ψ1Ip, ψ1Op, ψ1Fn}
Here, ψ1Ip is Input,
Ψ1Op is Output and
Ψ1Fn is Function
WHERE,
ψ1Ip = {ψ1Ip1, ψ1Ip 2}
ψ1Ip 1 = Add information of user belongs to sample population
ψ1Ip 2= Access Logs Details of selected sample user
ψ1Fn = {ψ1Fn1, ψ1Fn2, ψ1Fn3, ψ1Fn4, ψ1Fn5}
ψ1Fn1= Save personnel information
ψ1Fn2= Get installed app details
ψ1Fn3= Filter system required app
ψ1Fn4= Get access logs details as per application
ψ1Fn5= Add these logs details after analyzing them
ψ1Op = {ψ1Op1}
ψ1Op1= Get analytical results of added logs
2. Ψ4 = {ψ4Ip, ψ4Op, ψ4Fn}
ψ4Ip = {ψ4Ip1}
ψ4Ip1= Manage / Generate .apk file in case of any changes in respective apps
ψ4Fn = {ψ4Fn1, ψ4Fn2, ψ4Fn3, ψ4Fn4, ψ4Fn5, ψ4Fn6}
ψ4Fn1= Get App Permissions using API
ψ4Fn2= Add app permission
ψ4Fn3 = Update app permission list
ψ4Fn4= Manage app details
ψ4Fn5 = Manage app category list
ψ4Fn6 = Generate Clusters based on logs information and permission matches and reviews.
ψ4Op = {ψ4Op1, ψ4Op2, ψ4Op3}
ψ4Op1= Update App Permissions
ψ4Op2= Update app permission list
ψ4Op3 = Update app category list
3. Ψ3 = {ψ3Ip, ψ3Op, ψ3Fn}
ψ3Ip = {ψ3Ip1, ψ3Ip2, ψ3Ip3, ψ3Ip4}
ψ3Ip1 = Application set
ψ3Ip2 = Permission Set
ψ3Ip3 = Categories set
ψ3Ip4 = User request
ψ3Fn = {ψ3Fn1, ψ3Fn2, ψ3Fn3, ψFn4, ψ3Fn5}
ψ3Fn1=Extract web based feature
ψ3Fn2= Web based features preprocessing
ψ3Fn3 = Get application category
ψ3Fn4 = User request web service
ψ3Fn5 = App recommendation list
ψ3Op = {ψ3Op1, ψ3Op2}
ψ3Op1= Classification of app
ψ3Op2= Recommended App

4. Ψ2= {ψ2Ip, ψ2Op, ψ2Fn}
ψ2Ip = {ψ2Ip1, ψ2Ip2, ψ2Ip3, ψ2Ip4}
ψ2Ip1= Personnel Details
ψ2Ip2= Trendy Apps requests
ψ2Ip3= Current Apps requests
ψ2Ip4= Personnel App recommendation based on profile
ψ2Fn = {ψ2Fn1, ψ2Fn2, ψ2Fn3}
ψ2Fn1= Get app recommendation request

ψ2Fn2= Get user profile based app details
ψ2Fn3= Get Trendy app list by calculating and analyzing the request
ψ2Op = {ψ2Op1, ψ2Op2, ψ2Op3}
ψ2Op1 = List of current recommended application based on profile with category, review and risk score
ψ2Op2 = List of trendy app
ψ2Op3 = List of personalized recommendation app

6. EXPERIMENTAL RESULTS

We have taken top 20 paid apps and top 20 free apps for our analysis. We have maintained two databases first is the dynamic database where we have provided the link of the Google play store and we are continuously updating the database with the top trending free and paid apps. Second database is of the registered user, when the user registers himself and login to the app then all the apps data related to which apps user is using is stored in the database.
Also we made some synthetic data in which we are manually creating the fake ranks, reviews and ratings for some apps to check whether our system is detecting it as fraud app or not and recommending to uninstall. The app is providing recommendation in 3 types:

1. Using age, profession and gender the app is recommending those apps which are of user interest.

2. If a app is installed on user mobile and it is fraud or not in top 20 free or paid apps we are using the integration of ranking ,rating and review for that app and if all the condition are true then our app will recommend user to uninstall that suspicious or fraud app.

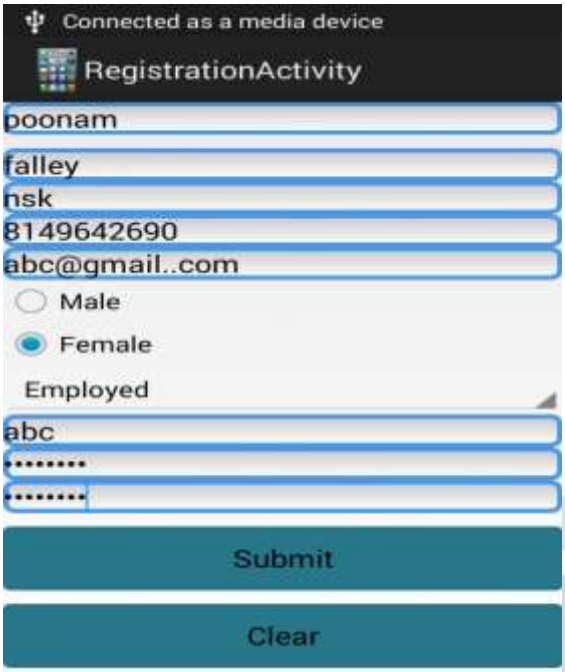
3. By using the app usage count if an app in user mobile is not used for longer time than it should notify the user to uninstall that app.

Using is analysis we have assumed following values:
1. For paid apps if the rate count of an app suddenly increases by **5000** every time then that app is suspicious app.
2. for free apps if the rate count of an app suddenly increases by **1, 50,000** every time then that app is suspicious app.
3. If there is a jump of 0.2 in rating for an app then that app is considered as suspicious.

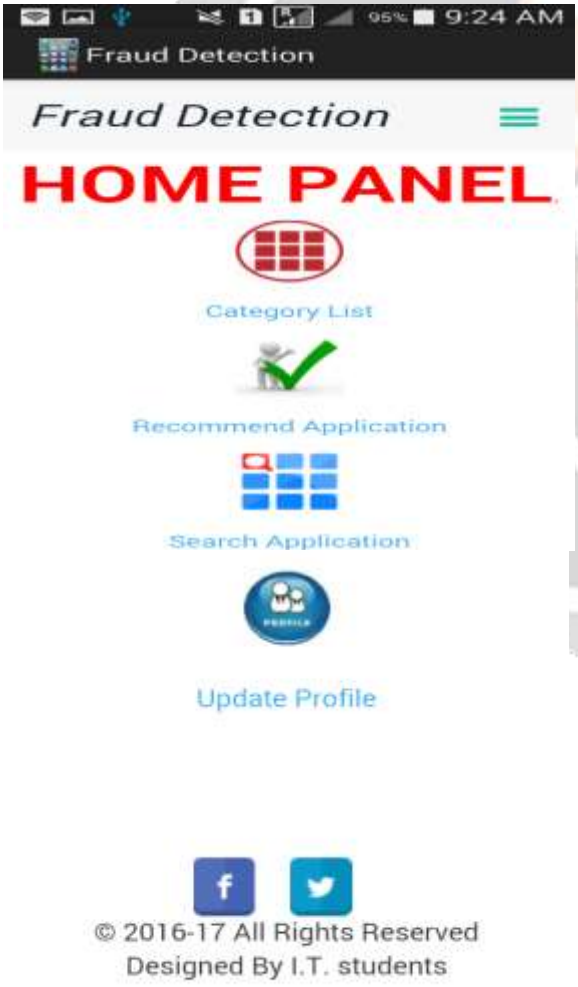
6.1 Screenshots:

6.1.1 Login

6.1.2 Registration Page

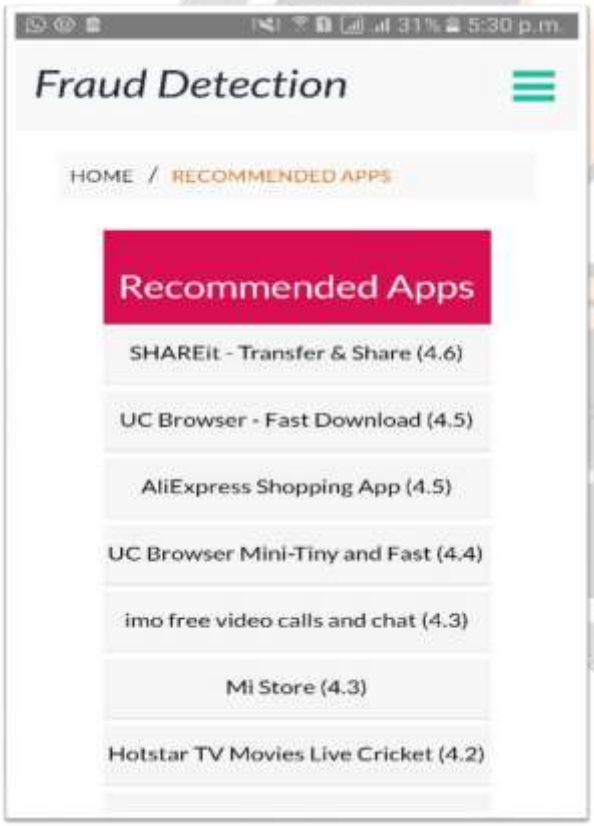


6.1.3 Home Panel



6.1.4 App Category List

6.1.5 Recommendation of apps to user based on age, profession and gender of user.



6.2 Performance Evaluation

For performance evaluation, we have tested our system for 20 apps. Time required for processing is captured for Rank, Rating and Review processing so the total processing time is calculated from it**.**

Number Of Apps	Rank and rating Processing Time	Review Processing Time	Total Processing Time
5	0.488	2.56	3.048
10	0.915	5.293	6.208
15	1.393	10.145	11.538
20	1.672	16.027	17.699

Table -1

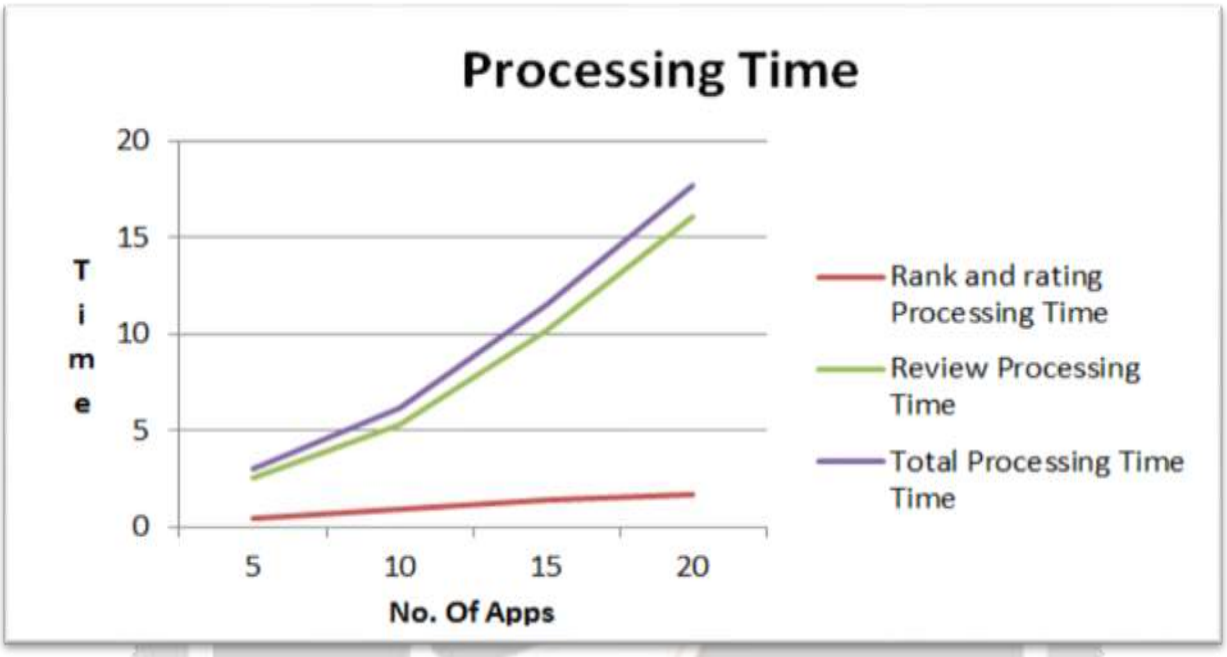


Chart -1: Graphical Representation

7. CONCLUSIONS

Specifically, we first showed that ranking fraud happened in leading sessions and provided a method for mining leading sessions for each App from its historical ranking records. Then, we identified ranking based evidences, rating based evidences and review based evidences for detecting ranking fraud. Moreover,an optimization based aggregation method to integrate all the evidences for evaluating the credibility of leading sessions from mobile Apps.An unique perspective of this approach is that all the evidences can be modeled by statistical hypothesis tests, thus it is easy to be extended with other evidences from domain knowledge to detect ranking fraud.

Finally, we have provided more effective fraud evidences and analysed the latent relationship among rating, review and rankings. Moreover,we have extended our ranking fraud detection approach with other mobile App related services, such as mobile Apps recommendation, for enhancing user experience. We have implemented app recommendation system which recommends in two ways: First according to the age,profession and gender it notifies user which apps the user should use.Second notification is given to the user about the apps which user has not used for longer in his mobile and current rates and reviews for that app.

8. ACKNOWLEDGEMENT

Words are inadequate in offering our thanks to our respected Guide in carrying out project in starting phase of the proposed system like requirement analysis and literature survey .We are thankful to our Director Dr. Kunal Darade for his motivation throughout this project.We also express our most gratitude to Dr. G. K. Kharate Principal and Prof. N. L. Bhale Head of department of Information Technology Engg, for their valuable co-operation in selecting the project topic and guiding us stepwise ahead. Inspiration and guidance are invaluable in every aspect of life, especially in field of education,which we have received from our respected project guide Mrs. Ambekar N. S.

who helped a lot in topic selection the information gathering,guiding us throughout project and gave earnest co-operation whenever required.
We would like to express sincere gratitude towards her. At last, we would like to take this opportunity to convey thanks to our entire staff members, who directly or indirectly encouraged and helped us in project work. Finally yet importantly, we would like to express heartfelt thanks to our beloved parents for giving blessing, our friends/ classmates for their help and wishes.

9. REFERENCES

[1]Hengshu Zhu, Hui Xiong, Yong Ge, Enhong Chen," Mobile App Recommendations with Security and Privacy Awareness".

[2] S.K.Ram Kumar, Dr. N. Lakshmi Narasimman, "Security Awareness of Mobile Application for Discovering Fraud Rank", PG Scholar, Computer Science Engineering Department, K.L.N. College of Engineering, Pottapalayam, Sivagangai 630 612, INDIA

[3] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth. Taintdroid: an information-flow tracking system for realtime privacy monitoring on smartphones. In Proceedings of the 9th USENIX conference on Operating systems design and implementation, OSDI'10, pages 1–6, Berkeley, CA, USA, 2010. USENIX Association.

[4] W. Enck, M. Ongtang, and P. McDaniel. On lightweight mobile phone application certification. In Proceedings of the 16th ACM conference on Computer and communications security, CCS '09, pages 235–245, New York, NY, USA, 2009. ACM.

[5] A. P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner. Android permissions demystified. In Proceedings of the 18th ACM conference on Computer and communications security, CCS '11, pages 627–638, New York, NY, USA, 2011. ACM.

[6] T. Joachims."Optimizing search engines using clickthrough data". In Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '02, pages 133–142, New York, NY, USA, 2002. ACM.

[7] http://en.wikipedia.org/wiki/google play.