Ranking Fraud Detection for Mobile Apps using Evidence Aggregation Method

Ms. Prajakta U. Gayke[1], Prof. Sanjay B. Thakare[2]
Department of Computer Engineering,
Savitribai Phule Pune University, Pune, India

bstract: **Nowadays everyone is using smart phone. There is need of various applications to be installed on smart Phone. To download application smart phone user has to visit Apps store such as Google Play Store, Apples store etc. which is the major target of fraud applications. The detection and removal of these apps from android is the major problem in now days. In mobile Apps business ranking fraud alludes to false or tricky operations which have a purpose behind knocking up the Apps in the leader board chart. To be sure, it turns out to be more continuous for App developers to utilize doubtful means, for expanding their mobile Apps' business. The starting aim of this project is to enhance the prevention of ranking frauds in mobile apps. In this work the leading event and group of neighboring events that is leading session of an app is identified from the collected historical records of mobile Apps. Then three different types of evidences are collected from the user feedbacks like comments namely ranking, rating and review based. These three evidences are aggregated by using evidence aggregation method. The output of aggregation is the mobile app which decides the app is false or not. At last, we assess the proposed framework with certifiable App information gathered from the App Storeroom for quite a while interval and also ranking fraud detection method with different services related to Apps such as recommendation of Apps for user that is to preventing false apps to be recommended to user, to learning more powerful fraud evidences and latent relation analysis on reviews.**

Index Terms **- Evidence aggregation method, fraudulent behavior, Mobile Apps, ranking rating and review evidences**.

I. INTRODUCTION

In all over the world for the mobile electronic devices are a very vast collection of millions of mobile Apps. These Apps developed by App Developer and Post at leaderboard for ranking purpose.The number of versatile mobile Apps has developed at a stunning speed in the course of recent years. For instance, at the month end of April 2013, there are 1.6 million and more than those apps at Apple's App store and Google Play. To fortify the advancement of portable Apps, numerous App stores introduced day by day App leaderboards, which show the graph positions of almost all well-known Apps. In fact, the App leaderboard is a standout amongst the most essential routes for advancing portable Apps. A top most position on the leaderboard more popular is the app is the fact. Top ranked app have more amount of downloads and earnings in million dollars. In this form, App designers have a tendency to investigate different ways for getting the higher position in leader board for example, promoting advertisement for their Apps keeping in mind the end goal to have their Apps ranked as top rank as possible in Application leaderboard.

Be that as it may, as a late pattern, rather than depending on conventional advertising, fraud App designers resort to some false intends to purposely help their Apps and in the long run control the outline positions on an App store. This is typically executed by utilizing "bot farms" or "human water armies" to blow up the App downloads, appraisals and audits in a brief while. For instance, a report from Venture Beat shows that, when an App was advanced with the assistance of positioning control, it could be pushed from amount 1,800 to the main 25 in Apple's without top leaderboard and more than 50,000-100,000 new clients could be procured inside of a few days.

In this work, we suggest to build up a position extortion recognition model for mobile Apps. Surely, our cautious perception uncovers that mobile Apps are not generally positioned top position in the leader board, but rather just in some driving occasions, which is form other leading sessions. Careful observation shows that the mobile Apps are not always at top most position in leader board. But only in some time period called leading event which is form different leading sessions means ranking fraud particularly occur in this leading sessions. Therefore detecting frauds in apps is nothing but detecting ranking fraud in leading sessions. This leading session identify from each app on the basis of historical record of mobile apps which is given to the mining algorithm. The evidences of fraud detection is then given to the three extracting functions ranking, review and rating then aggregation of these evidences is done by evidence aggregation method. The output gives mobile app with false or true result.in proposed system false apps are notify to users and study some effective evidences of mobile apps.

A.MOTIVATION

Now a day, in the mobile App business ranking fraud alludes to false or tricky exercises which have a motivation behind knocking up the Apps in the leaderboard chart. To be sure, it turns out to be more incessant for App designers to utilize shady means, for example, expanding their Apps business or sending false App evaluations, to confer positioning misrepresentation.

B.CHALLENGES

1) Detect Fraud ranking in daily App leaderboards.
2) Avoid ranking manipulation.

II. RELATED WORK

Hui Xiong[1] discovered ranking fraud detection system for mobile Apps but it is still under study research. To fill this crucial lack, we propose to develop a ranking fraud detection system for mobile Apps. We also determine several important challenges. First challenge, in the whole life cycle of an App, the ranking fraud does not always happen, so we need to detect the time when fraud happens. Finally, due to the dynamic nature of chart rankings, it is difficult to find and verify the evidences associated with ranking fraud, which motivates us to discover some implicit fraud patterns of mobile Apps as evidences.

D. M. Blei[3] has proposed latent dirichelet allocation of generative probabilistic model for Accumulations of discrete data such as text corpora. LDA is a three-level various leveled Bayesian model, in which everything of an accumulation is displayed as a limited blend over an basic set of points. Every subject is, thus, demonstrated as a boundless blend over a basic arrangement of point probabilities. In the context of text modeling, the topic probabilities give an explicit representation of a document. We introduced efficient approximate inference techniques based on variation processes and an EM algorithm for empirical Bayes parameter estimation. We report results in record modeling, text classification, and collaborative filtering, comparing to a mixture of unigrams model and the Probabilistic LSI model.

Y. Ge, H. Xiong[2] has proposed taxi driving fraud detection system for Advances in GPS following innovation have en-abled us to introduce GPS beacons in city taxis to gather a lot of GPS follows under operational time requirements. These GPS follows give unparalleled chances to us to reveal taxi driving extortion exercises. In this paper, add to a taxi driving misrepresentation identification framework, which can efficiently explore taxi driving extortion. In this framework, first give capacities to discover two parts of confirmations: travel course proof and driving separation proof. Besides, a third capacity is intended to consolidate the two parts of proofs in view of Dempster-Shafer hypothesis. To actualize the framework, first recognize introducing destinations from a lot of taxi GPS logs.

A. Klementiev, D. Roth[7] An unsupervised learning algorithm for rank aggregation Many applications in data recovery, natural language processing, information mining, and related fields require a positioning of cases regarding indicated criteria instead of a grouping. Moreover, for some such issues, various set up positioning models have been all around concentrated on and it is alluring to consolidate their outcomes into a joint positioning, formalism indicated as rank accumulation. This paper exhibits a novel unsupervised learning algorithm for rank accumulation (ULARA) which gives back a direct blend of the individual positioning capacities of ranking functions in view of the standard of compensating requesting understanding between the rankers.

D. F. Gleich[15] has proposed Rank aggregation via nuclear norm minimization with the method of rank aggregation is informally interwoven with the structure of skew-symmetric matrices and applies recent approach in the theory and algorithms of matrix completion to skew-symmetric matrices. This mix of thoughts delivers another system for positioning an arrangement of things. The need of our plan is that a rank aggregation shows a partially filled skew-symmetric matrix. Here extend an algorithm for matrix completion to hold skew-symmetric information and utilize that to take out ranks for each item. This algorithm applies to both pairwise comparison and rating data. Because it is based on matrix completion, it is vigorous to both noise and inadequate information.

Klementiev, D. Roth, and K. Small[8] has proposed Unsupervised rank aggregation with distance-based models which needs to incorporate the arrangement of rankings regularly manages collecting and it just comes up when a specific positioned information is produced. Despite the fact that the different heuristic and managed learning ways to deal with rank total, a prerequisite of area information and directed positioned information exists. Along these lines, to resolve this issue, a structure is proposed for learning total rankings without supervision. This system is instantiated for the instances of permutations and combinations of top-k records.

E.-P. Lim [10] has proposed product review spammer detection to locate customers producing unsolicited mail reviews or evaluation spammers. He became aware of several feature behaviors of evaluate spammers and model those behaviors so as to discover the spammers. Particularly, they seek to model the following behaviors. First, spammers might also target unique merchandise or product organizations to be able to maximize their effect. They tend to deviate from the alternative reviewer of their rankings of products. They advocate scoring methods to degree the degree of unsolicited mail for every reviewer and apply them on an Amazon overview data set then pick a sub-set of notably suspicious reviewers for further scrutiny by using our user evaluates with the help of a web primarily based spammer evaluation software program particularly evolved for person assessment experiments.

III. PROBLEM DEFINITION

The number of Apps has designed or increased at a vast speed across a couple of years. To refreshing the development of Apps many App stores introduced everyday basis Apps leaderboards chart, which shows the positions of nearly all famous Apps. A Top most rank on the chart generally lead to several downloads and earnings in million dollars, instead of depending on traditional marketing ways. Fake App creators apply some fraud full actions to intentionally improve their Apps and in the end inflate the chart positions on an App Store. This is normally done by utilizing “bot farms” or “human water armies” to manipulate the App downloads rating and comments in an extremely limited time period.

IV. IMPLEMENTATION DETAILS

A. System Architecture :

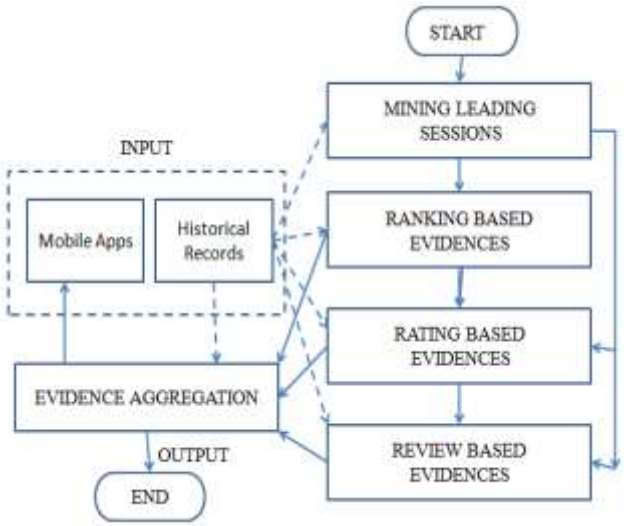


Fig. Ranking fraud detection system framework.

After careful understanding the system has been divided into the three evidences and at last aggregate these evidences as a result. There are two main phases of this system-
i) Identifying the leading sessions for mobile apps.
ii) Identifying evidences for ranking fraud detection.

Leading events:

(Leading Event)-Given a positioning limit a main occasion of App contains a time period range. Relating rankings of mobile App a, Note that we apply a positioning edge which is normally smaller than K , here on the grounds that K may be huge, and the positioning records past are not exceptionally helpful for recognizing the positioning controls. Moreover, we additionally find that a few Apps have a few nearby driving events. Especially, a main occasion which does not have other close-by neighbours can likewise be dealt with as an uncommon driving session.

Leading Sessions:

A main session of App contains a period range T_s and n adjoining driving occasion. Application speak to its times of fame, so the positioning control will just occur in these driving sessions. Along these lines, the issue of recognizing positioning extortion is to distinguish fake driving sessions. Along this line, the first assignment is the means by which to mine the main sessions of a horde. Leading session is calculated from closable leading event.

Identifying the Leading Sessions for Mobile APPs:

In mining leading session algorithm there are two important steps for calculating particular period in which fraud is happened for that particular mobile app. The first step is to search leading events from the mobile App's historical ranking records and second is for merging neighbouring leading events for developing leading sessions. Specifically, firstly extract individual leading event e for the given App from the starting time. For each and every extracted individual leading event e, we check the time span between e and the current leading session s to decide whether they include to the similar leading session. Thus, this algorithm can identify leading events and sessions by scanning mobile app a's historical records only once.

Identifying evidences for ranking fraud detection:

Identifying different evidences for ranking fraud detection is applied on output of mining leading session algorithm. Step by step three evidences applied are ranking, rating and review based.
1) In ranking based evidences specific ranking pattern is always satisfied by app ranking behavior. This includes rising phase, maintaining phase and recession phase.
Ranking pattern for rising and recession phases:-

$$\theta_e^1 = \arctan\left(\frac{K^* - r_b^a}{t_b^e - t_a^e}\right), \quad \theta_e^2 = \arctan\left(\frac{K^* - r_c^a}{t_d^e - t_c^e}\right) \qquad (1)$$

Fraud signature for leading session:

$$\bar{\theta}_s = \frac{1}{|E_s|} \sum_{e \in s} (\theta_e^1 + \theta_e^2) \qquad (2)$$

Ranking pattern for maintaining phase:-

$$\Delta t_m^e = (t_c^e - t_b^e + 1) \quad (3)$$

Fraud signature for leading session:

$$x_s = \frac{1}{|E_s|} \sum_{e \epsilon s} \frac{K^* - \bar{r}_m^e}{\Delta t_m^e} \quad (4)$$

2) In rating based evidences rating pattern is used for ranking fraud detection in app. This rating is done after downloading the app by user and then user gives rating to that app. If the rating is high in the leader board of app industry then that app is attracted by more mobile app users. In this the fraud occurred during rating is performed in leading session. An App with rating fraud might have surprisingly high ratings in the fraudulent leading sessions. Thus, Fraud signature:

$$\Delta R_s = \frac{\bar{R}_s - \bar{R}_a}{\bar{R}_a}, (s \epsilon a) \quad (5)$$

3) In review based evidences reviews are the textual comments that is given by mobile app users after using or downloading that app. Before downloading the app user always preferred to view these comments given by most users. Based on previous work on review spam detection there are still some issues for locating local anomaly in leading events e for ranking fraud detection system.

B. Algorithm:

Algorithm [1]: Mining Leading Sessions.

Input 1: a's historical ranking records R_a;
Input 2: ranking threshold K***;**
Input 2: merging threshold □;
Output: set of a's leading sessionsS_a;

1. R_a= {r_1^a,r_2^a,…,r_n^a};
2. r_i^a= {1,……., K,+∞};
   Where,
   r_i^a is the ranking of a at time ti,
   +∞= a is not ranked in the top K list,
   N is the number of all ranking records.
3. Leading event and session is a tuples of:
   <t_{start}^e; t_{end}^e>
   < t_{start}^s; t_{end}^s; E_s>
   Where,
   E_s is the set of leading events e in session s.
4. If $r_i^a \leq$ K* and t_{start}^e ==0 then
   t_{start}^e = ti;
5. Else if $r_i^a \geq$ K* and $t_{start}^e \neq 0$ then
   // found new event;
   e = < t_{start}^e ; t_{end}^e >;
   if E_s == □ then
   E_s ∪ =e; t_{start}^s =t_{start}^e; t_{end}^s =t_{end}^e;
   else if (t_{start}^e - t_{end}^s) < □ then
   E_s ∪ = e; t_{end}^s = t_{end}^e;
   else then
   // found new session;
   s = <t_{start}^s;t_{end}^s; E_s >;
   S_a ∪ = s;
6. Return S_a;

Algorithm: EVIDENCE AGGREGATION
1. Analyze the historical records of mobile apps.
2. Differentiate the evidences as Ranking based, Rating based, Review based.
3. Aggregate these evidences.
4. Design Android application framework.

V. MATHEMATICAL MODEL

Let S, be a system that describes detection of ranking Fraud for Mobile Apps- S= {I, P, O}

Where,

1) Input (I): Historical data for Apps,

I = $\{i_1; i_2; i_3; i_4; i_5; i_6; i_7; i_8; i_9\}$; where,

I = {i_1: Application Title,
i_2: Version,
i_3: Uploaded be,
i_4: Web Portal details,
i_5: Certificate for app,
i_6: Downloaded by,
i_7: Rating,
i_8: Review,
i_9: Device Id
}

2) Process (p): $\{p_1; p_2; p_3; p_4; p_5\}$, Where,

- p_1 = MLS:
  R_a=$\{r_1^a, r_2^a, \ldots, r_n^a\}$;
  r_i^a= {1,…, K, +∞}; where,

R_a= is a's historical ranking records,
r_i^a= is the ranking of a at time ti,
+∞= a is not ranked in the top K,
n = number of all ranking records.

- p_2= RnBE :

i. for rising and recession phase:

$\bar{\theta}_s = \frac{1}{|E_s|}\sum_{e \epsilon s}(\theta_e^1 + \theta_e^2)$, where,
$\bar{\theta}_s$= Fraud signature for s,
θ_e^1 , θ_e^2 = shape param. from eq. (1),(2),
$|E_s|$ = no. of e's in session s,

ii. for maintaining phase:

$x_s = \frac{1}{|E_s|}\sum_{e \epsilon s}\frac{K^* - \bar{r}_m^e}{\Delta t_m^e}$, where ,
x_s = Fraud signature for s,
K^* = ranking threshold,
$\bar{r}_m^e$ = avg. rank in this phase,
Δt_m^e = maintaining phase of eq.from eq.(3).

- p_3= RtBE :

$\Delta R_s = \frac{\bar{R}_s - \bar{R}_a}{\bar{R}_a}$, (s ϵ a), where,
ΔR_s = fraud signature,
$\bar{R}_s$ = avg. rating in leading session s,
$\bar{R}_a$ = avg. historical rating of App a,

- p_4 = ReBE:
  Reviews analysis for review based evidences.

- p_5 = EA: Linear combination of all existing evidences.

3) Output (O): $\{o_1, o_2, o_3, o_4\}$, where,

o_1: Top k-Ranked Apps,
o_2: Historical ranking records,
o_3: Evidence Details,
o_4: App Review

Table 1: Memorization Parameters

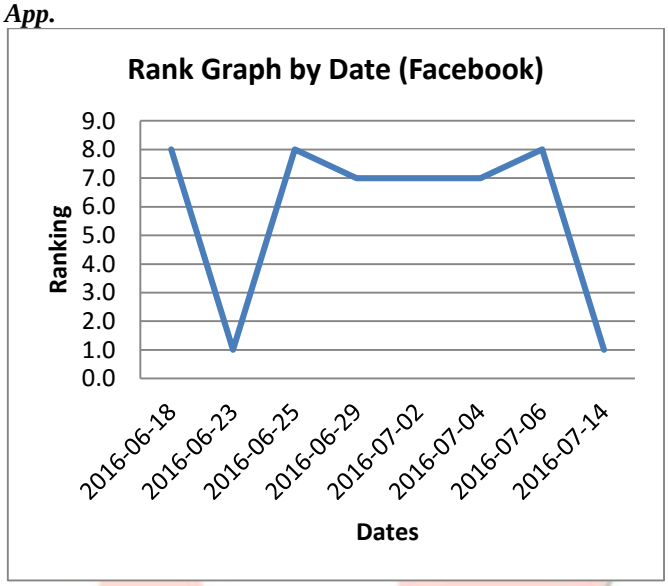
Symbol	Meaning
S	System that describes detection of ranking fraud system as a whole.
I	Input to the system as mobile apps.
i_1	Historical ranking records of mobile apps such as reviews/dataset details

P	Identify process as P.
MLS	Mining Leading Sessions
RnBE	Ranking Based Evidence
RtBE	Rating Based Evidence.
ReBE	Review Based Evidence.
EA	Evidence Aggregation.
O	Output as classified dataset, Top-K ranked Apps

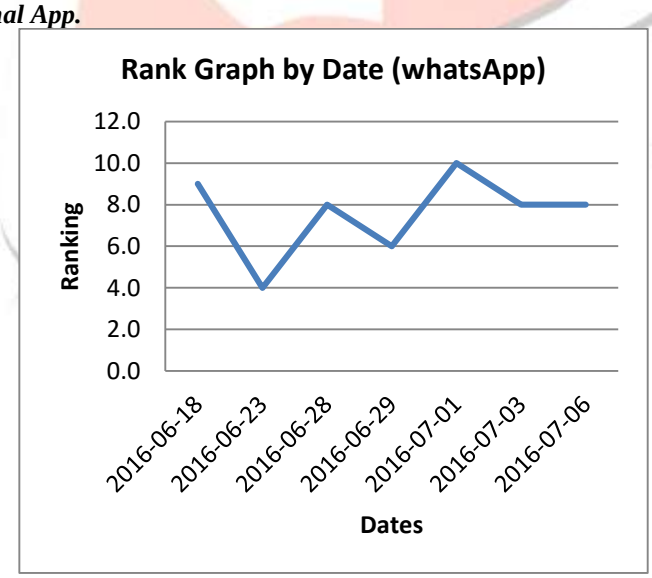
VI. EXPERIMENTAL RESULT

Here, main attention is on extracting different evidences such as reviews, ratings, ranking and download information from historical records of data set. Data set contains the historical reviews, ratings of mobile apps. In the result parts calculates and merge the evidences with help of evidence aggregation method.

Rank Graph By Date For Fraud App.



Rank Graph By Date For Normal App.



VII. CONCLUSION

In this paper, we develop ranking fraud detection system for mobile apps. It reviews various existing strategies used for internet or web spam detection, which is associated with the rating fraud for mobile Apps. Also, we've seen references for online review unsolicited mail detection and mobile App advice. By using mining the main sessions of mobile Apps, we aim to locate the ranking fraud. The leading classes works for detecting the nearby anomaly of App ratings. The machine targets to locate the ranking frauds based on three styles of evidences, including rating based evidences, ranking based evidences and comment based evidences. In addition, an optimization based totally aggregation method combines all of the three evidences to hit upon the fraud.

ACKNOWLEDGMENT

We are glad to express our sentiments of gratitude to all who rendered their valuable guidance to us. We would like to express our appreciation and thanks to the Principal of our college. We are also thankful to the Head of Department. We thank to the anonymous reviewers for their valuable comments.

REFERENCES

[1] Hengshu Zhu, Hui Xiong, Senior Member, IEEE, Yong Ge, and Enhong Chen, Senior Member, IEEE "Discovery of Ranking Fraud for Mobile Apps" IEEE Transactions On Knowledge And Data Engineering, Vol. 27, No. 1, January 2015.
[2] 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.
[3] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," J. Mach. Learn. Res., pp. 993–1022, 2003
[4] T. L. Griffiths and M. Steyvers, "Finding scientific topics," Proc. Nat. Acad. Sci. USA, vol. 101, pp. 5228–5235, 2004.
[5] 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.
[6] N. Jindal and B. Liu, "Opinion spam and analysis," in Proc. Int.Conf. Web Search Data Mining, 2008, pp. 219–230.
[7] 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.
[8] 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.
[9] 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.
[10] E.-P. Lim,V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers use rating behaviors," in Proc.19thACMInt. Conf. Inform. Knowl. Manage., 2010, pp. 939–948.
[11] 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.
[12] 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.
[13] 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.
[14] 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.
[15] 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.

Author Profile

Ms. Prajakta Uttamrao Gayke received B.E degree in Computer Science and Engineering from S. N. D. College of Engineering and Research center from Savitribai Phule Pune University, India in 2010 and pursuing ME degree in Computer Science and Engineering from JSPMs Rajarshi Shahu College of Engineering from Savitribai phule Pune University, India.

Mr. Sanjay Bapu Thakare Pursuing PhD from Computer Engineering Department of Dr. Babasaheb Ambedkar Technological University, Lonere M Tech-Information Technology from Bharati Vidyapeeth Deemed University College of Engineering, Pune BE from Computer Science and Engineering Department, Walchand College of Engineering, Sangli Currently working as Asst Prof at Computer Engineering Department, in RSCOE, Pune, Savitribai Phule Pune University Area of interest- Data Mining, Social Network Analysis