Fraud App Detection Using Sentimental Analysis Using Naive Bayes Algorithm

Daphne Jackson[1], Mrs.Dhamayanthi k[2]
[1]Dept of Master of Computer Applications
[2]Associate Professor,Dept of Master of Computer Applications
[1, 2] Francis Xavier Engineering College, Vannarpettai, Tirunelveli

Abstract- *With the expansion in the amount of mobile applications in the day to day life, it is crucial to keep track of which ones are risky and which ones aren't.Based just on the reviews that are presented for each application, one cannot determine how secure and reliable any application is. So it is vital to check and establish a system to make sure that the apps present are legitimate or scam. The purpose is to construct a web system in detecting fraud apps before the user downloads by employing sentimental analysis. Sentimental analysis can assist in identifying the emotional undertones of words spoken online. This technique assists in keeping an eye on social media and provides a quick overview of the general public's views on particular issues. On the internet, the user may not always find accurate or genuine reviews of the product. The reviews can be made up or real. We can tell if the software is real or not by analyzing the reviews that include both user and admin remarks. The machine can learn and interpret the sentiments, emotions, and other aspects of reviews and other texts using sentimental analysis. One of the main components of app ranking fraud is the manipulation of reviews. Analyzing reviews and comments can assist in choosing the best app for both Android and iOS by employing emotional analysis and support vector machines. Users nowadays prefer to use a mobile app over a website.The goal is to create a system that uses sentimental analysis and data mining to identify fraudulent apps before a user downloads them. Therefore, we are suggesting a web application that will process the data, feedback, and application review. So it will be easier to decide which application is fraud or not. Several applications can be processed at a time with the web application. Also, users may not always find accurate or reliable product reviews online. As a result, the administrator will evaluate the ratings and comments, making it simple for the administrator to determine whether an application is legitimate or fraudulent. Using sentimental analysis and data mining, the machine is able to learn and analyze the sentiments, emotions, and other aspects of reviews and other texts.One of the main components of app ranking fraud is the manipulation of reviews. We have utilized the LSTM model to forecast the results.*

Keywords- Mobile App, Review, Rating, Sentimental Analysis,Naive Bayes.

I. INTRODUCTION

Sentiment is an emotion or attitude brought on by the customer's feelings. Opinion mining is another name for sentiment analysis because it uses user reviews to determine how well-liked an app is. Sentiment analysis is a function of machine learning proposed by [1].Information is gathered and is evaluated to determine the sentiment regarding the information such as negative or positive emotion. People frequently ask about other users' opinions of an app before making a purchase,as proposed by [2].Sentiment analysis gathers and analyzes the opinion or sentiment of the sentence using natural language processing (NLP). It is well-liked since many individuals favour seeking assistance from other users. It is impossible for manual procedures to assess an enormous amount of reviews and to aggregate them into an effective choice because the number of opinions expressed in the form of reviews, blogs, etc. is continuously rising. Sentiment analysis converts these duties into automated processes with less user support,proposed by[3]. Narayanan et al[4] focuses on conditional sentences, which have several distinctive features that make it challenging to ascertain the sentimental orientation of such statements. It uses a classifier and a condition classifier that are based on the entire sentence.

Data is gathered from user forums for the following products: cell phones, cars, LCD TVs, audio systems, and medicine. A novel method of determining sentence polarity uses clustering to automatically group related words into categories proposed by [5]. The web application has an uploaded apk file for the mobile application. Information about the application, including reviews, ratings, and past performance, is extracted using an APK parser. Sentiment analysis of the reviews is done using natural language processing. It generates the graph results by applying the fraud detection rule to the application. A positive outcome is considered when there are more than three ratings. Moreover, a negative outcome is assumed if the rating count is fewer than 3. Techniques used are cloud stack, data mining and NLP proposed by [6]. Reviews of applications are extracted and

transformed into tokens. Tokenization is the process of turning a stream of text into tokens, which are collections of words, phrases, and symbols. The preprocessing input for these tokens is. The technology determines the user's emotions after preprocessing the reviews. A positive review increases the score by 1, and a bad review decreases the score by 1. This will evaluate each review's score and validate if the application is legitimate or fraudulent, proposed by [7].Two sentiment analysis techniques are contrasted. Machine learning and lexicon-based approaches. The vocabulary-based technique, which has two branches: corpus-based approach and dictionary-based approach, focuses on finding the sentiment words from the sentence and comparing them to lists of words already in use. Whereas naive bayes require a training set, the lexicon-based approach does not. When a sentence is fully processed using training set data, the lexicon-based technique is more accurate than a Naive Bayes classifier,proposed by[8]. Using free scraping software, I gather user reviews and store them in my SQL database. From the saved dataset, titles and comments are extracted. The NLTK toolkit's collocation discovery method is used to extract characteristics from user evaluations. A general score is assigned to the detected features based on user feelings that are taken from all reviews. Lastly, high-level characteristics that are more meaningfully grouped into fine-grained features are created using topic modeling techniques,proposed by [9]. The Tweets Sentiment Analysis Model analyzes twitter data in this essay. When it comes to any category, it can quantify the intensity of favourable and negative opinions as well as identify positive, negative, and neutral attitudes. Three modules make up the TSAM's framework: A feature selection module that takes each sentence's pertinent words out of its Sentiment identification module that links opinions voiced with each significant sentence-level element. The sentiment scores for each entity are determined by the sentiment aggregation and scoring module,proposed by [10]. The Google API calculation method, which takes application ratings from the Play Store and uses calculations to get ranks, is used to determine the rank of applications, as proposed by [11].

Fraud app detection involves analyzing user reviews and feedback about a mobile application to determine if there is any fraudulent activity associated with it. Sentimental analysis is a technique that uses natural language processing and machine learning to analyze the emotional tone and attitude of user-generated content such as reviews, comments, and feedback.In the context of fraud app detection, sentimental analysis can be used to identify patterns and anomalies in user reviews that indicate fraudulent behavior, such as fake reviews or reviews from bots. By analyzing the sentiment of reviews, the app developer or app store can identify potential fraudsters and take appropriate action to protect users from fraudulent activity.Overall, fraud app detection using sentimental analysis is an important tool for ensuring the safety and security of mobile app users and maintaining the integrity of the app ecosystem..

II. RELATED WORK

Daniel A. Keim et al [1], in their paper asserted that Visual information investigation has high potential and numerous applications, for example, misrepresentation discovery also, information mining will utilize data representation innovation for an improved information examination.

Fuzail Misarwala et al [2], proposed that by utilizing various data mining techniques and algorithms, it would become easier for us to determine our backend retrieval of data. Fraud can be classified into various types which are the applications of data mining. With the end goal of grouping, extortion has been separated into four general classifications: budgetary misrepresentation, media communications extortion, PC interruption and protection misrepresentation.

Esther Nowroji et al [3], identified the source's uniqueness; it wasn't quite efficient considering the fact that IP snooping can be done. This IP snooping allows the users to change their IP address and allow them to rate an app more than once.

Muhammad Taimoor Khan et al [4], proposed that various natural language processing challenges at document level, sentence level, feature level and lexicon level. They have also compared different techniques and approaches to solve the natural language processing challenges such as naïve bayes, k-nearest neighbor, centroid, support vector machine, lexicon based,statistical based.

Rohini V et al [5], asserted that two methods of sentiment analysis are compared. Lexicon based approach and machine learning approach.Lexicon based approach deals with searching the sentiment words form the sentence and comparing with existing list of words,it has two branches dictionary and corpus based approach. Lexicon based approach does not require training set whereas naïve bayes requires training set Lexicon based method is accurate than Naïve bayes classifier when sentence is processed completely with training set data

Neha Puram et al [6], proposed that The main objective is fraud application detection using fuzzy logic to differentiate the actual fraud apps. The proposed system performs classification of apps & detects their group whether

they belong to good, bad, neutral, very good, very bad.Different class value & threshold value gives different results of accuracy of time required for execution.

Tichkule et al [7],proposed that Application reviews are extracted and converted into tokens. Tokenization is the process of converting a stream of text into words, phrases, symbols known as tokens. These tokens are the input for pre-processing. After preprocessing of reviews, the system determines the user's emotions. Positive reviews add 1 to positive score and negative review adds 1 to negative score. With this it will determine the score of every review and confirm whether the application is real or fake.

Israel J. Mojica Ruiz et al [8], proposed that Google API calculation approach is used to calculate the rank of the applications using a Calculation algorithm where they take application ratings from play store and calculate the ranks using the calculations.

Guzman et al [9], proposed that User reviews are collected using open source scraping tools and stored in my SQL database. Titles and comments are extracted from stored dataset. Collocation finding algorithm provided by NLTK toolkit is used for extraction of features from user reviews. User sentiments are extracted about the identified features and given them a general score across all reviews. Finally topic modeling techniques are used to group fine grained features into more meaningful high-level features.

Safrin et al [10], proposed a model to extract valuable information of mobile applications based on user reviews, ratings and ranking and aggregation of this evidence to detect fraud applications. Our system will use each review of the use along with the rating which will help to identify the user sentiment towards the topic.

III. THEORY

Many users frequently use Android applications to carry out a wide variety of tasks. Unfortunately, a number of apps have been based on fictitious behaviour that does not match their predicted behaviour. Where the prevalence of false negatives is substantial, the current pertinent procedures that develop these applications have performance issues. These methods lack scalability and offer very little flexibility when it comes to questioning supported sets of dubious permissions in order to identify a collection of particularly pertinent known abnormal applications. The authorization of the recognised most pertinent application category is then examined to see if there is significant overlap or to avoid designating related applications as perhaps abnormal.

Sentiment analysis and user emotions are the foundations for fraud app detection. Sentiment analysis deals with the client's emotions. If this analysis's findings are favourable, the app is safe to use. The app is a false statement app if the outcome of this analysis is negative. If the outcome of this study is neutral, the app is neither very good nor particularly bad to use. The basic goal is to create a system that notices reviews based mostly on evidence, with optimization assisted by aggregation to combine the evidence for fraud detection. As a result, it will be simpler to identify applications that are fraudulent. It's imperative to "keep on top of your game" as a developer, ie.,Maintain your app's updates by adding the most popular features and bug fixes. However, the majority of app shops merely assign each app an average rating (out of 5).

A 1. *Research Methodology*

Using a machine learning model for sentiment analysis in fraud app detection, a dataset of app reviews is often collected, with the app reviews being classified as either fraudulent or real. The model is then put to the test on a different set of reviews in order to gauge how well it can identify fake apps. By employing feature engineering approaches to extract more insightful data from the reviews, such as detecting frequent words and phrases used in fake reviews, the efficiency of the model can be further increased. Overall, sentiment analysis presents a way to identify fraudulent apps that show promise, and further research in this field is expected to produce even more accurate detection techniques.

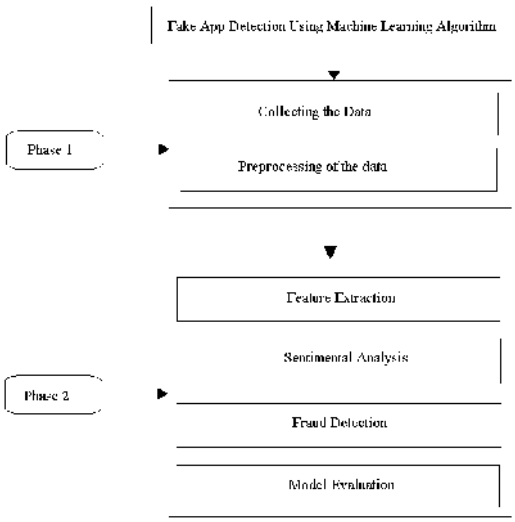


Figure 1. Research Methodology

A 2. *Algorithm Implementation*

Naive Bayes is a classification algorithm based on Bayes' theorem. It assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature, and that each feature contributes independently to the probability that an item belongs to a certain class.The algorithm works by first training on a set of labeled data to learn the probabilities of each feature in each class. Then, given a new item to classify, it calculates the probability of that item belonging to each class based on the probabilities of its features in each class. The class with the highest probability is then assigned to the item..Naive Bayes is often used in natural language processing applications such as spam filtering, sentiment analysis, and text classification. It has a number of advantages, including being computationally efficient and requiring relatively little training data. However, its assumption of feature independence can be unrealistic in some cases, and it may not perform as well as more complex algorithms on certain types of data.

Naive Bayes is a commonly used algorithm in fraud detection, as it can effectively identify patterns in large datasets and make accurate predictions about the likelihood of fraud.In fraud detection, the algorithm can be trained on a large set of labeled data that includes both fraudulent and non-fraudulent transactions. The algorithm will then use this training data to learn the probabilities of various features, such as transaction amount, location, and time of day, in relation to fraudulent activity.When a new transaction is received, the algorithm will use these probabilities to calculate the likelihood that the transaction is fraudulent. If the probability exceeds a certain threshold, the transaction will be flagged for further review or declined.

One of the advantages of using Naive Bayes in fraud detection is its ability to handle large datasets and quickly identify patterns in the data. It can also adapt to new data as it becomes available, allowing for ongoing refinement and improvement of the fraud detection system.However, it is important to note that Naive Bayes assumes that features are independent, which may not always be the case in fraud detection. In addition, it may not perform as well as more complex algorithms in detecting certain types of fraud. Therefore, it is often used in combination with other algorithms and techniques to create a more comprehensive fraud detection system.

IV. EXPERIMENTS AND RESULTS

A 1. *Simulation Environment*

Jupyter Notebook is an open source web application that you can use to create and share live code, equations, visualizations, and text documents. Jupyter Notebooks are maintained by Project Jupyter staff. This is a random project from his IPython project which had an IPython notebook project itself. The name Jupyter comes from the core programming languages it supports: Julia, Python, and R. Jupyter comes with an IPython kernel that can be used to write Python programs, but over 100 other kernels are available. Well done. Jupyter notebooks are especially useful for doing computational physics or doing a lot of data analysis using computer tools as a scientific lab notebook.

Google Colab, also known as Colaboratory, is a free Jupyter notebook environment that requires no configuration and runs entirely in the cloud. Free GPU and TPU support for users. Colaboratory allows you to write and run code, store and share your analysis, and access powerful computing tools from your browser, all for free. As the name suggests, collaboration is guaranteed in the product. A Jupyter notebook that uses the function of linking with Google Docs. And since it runs on Google servers, you don't need to update anything. Notebooks are stored in your Google Drive account. It provides a platform that allows anyone to develop deep learning applications using commonly used libraries such as PyTorch, TensorFlow, and Keras. It provides a computer-friendly way to avoid the burden of intensive training of ML operations.

A 2. *Architecture diagram*

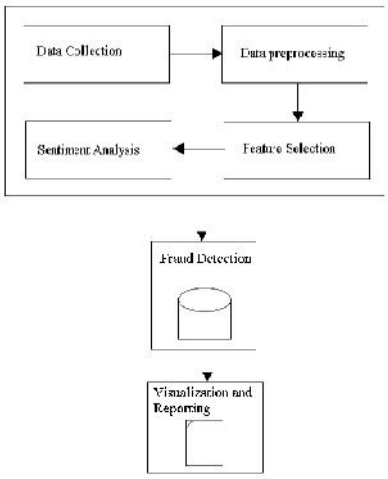


Figure 2. Architecture Diagram

Figure 3. Creating training and testing data..

Figure 4. Insert reviews

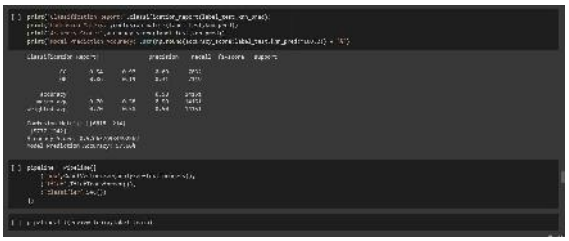


Figure 5. Classification report

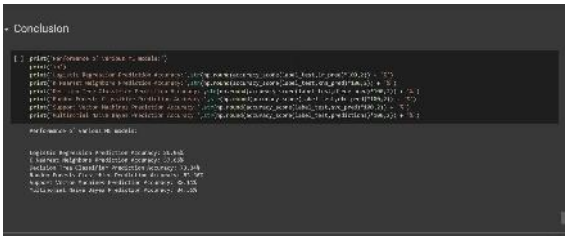


Figure 6. Performance of ML model

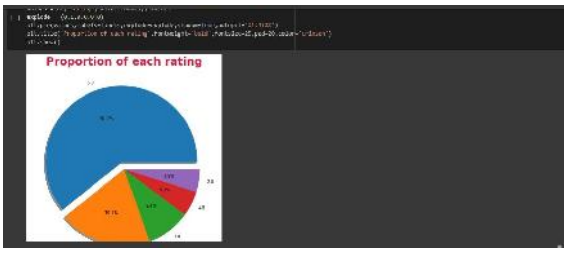


Figure 7. Proportion of rating

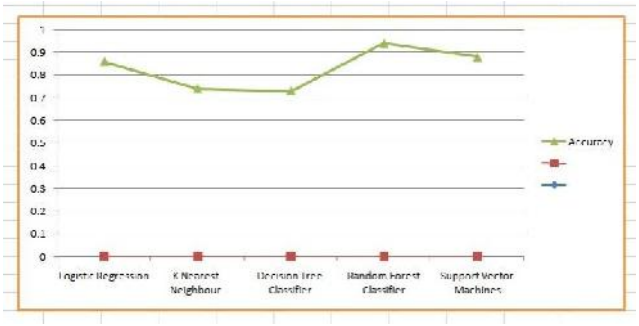
Figure 8. Model Accuracy

A 3. *Performance Metrics*

TABLE 1. REVIEW LEVEL RATIO

Algorithm	Accuracy
Logistic Regression	86%
K Nearest Neighbour	74%
Decision Tree Classifier	73%
Random Forest Classifier	94.00%
Support Vector Machines	88%

TABLE 2. REVIEW LEVEL CORRELATION OF METRICES



V. DISCUSSION AND CONCLUSION

The conclusion is about using sentiment analysis and the support vector machine idea to identify fraudulent applications. It was backed up by the architecture diagram, which provided information on the project's implementation of the algorithm and processes. Data is gathered and kept in the database, which is subsequently assessed using the specified supporting algorithms. This is a distinctive method in which the evidence is combined and focused on a single conclusion. The suggested architecture can be expanded to include other domain-generated evidence for review fraud detection and is scalable. The testing findings demonstrated the viability of the suggested system, the scalability of the detection algorithm, and some regularity in the ranking of fraud activities.

VI. FUTURE SCOPE

Sentiment analysis can be a useful tool for detecting fraud in various applications. It involves analyzing text data to determine the sentiment or emotional tone of the content. This can be used to identify potential fraudulent behavior, such as manipulation or deception.One potential enhancement for a fraud detection app using sentiment analysis is to incorporate machine learning algorithms to improve the accuracy of the analysis. This could involve training the algorithm on a large dataset of fraudulent and non-fraudulent content, so that it can learn to recognize patterns and make more accurate

predictions.Another potential enhancement would be to incorporate other types of data, such as behavioral or transactional data, into the analysis. By combining sentiment analysis with other types of data, the app could potentially identify fraudulent behavior more accurately and quickly.Additionally, the app could be designed to provide real-time alerts to users when potentially fraudulent behavior is detected. This could help prevent fraud from occurring in the first place, or at least minimize the damage caused by it.Overall, there are many potential enhancements that could be made to a fraud detection app using sentiment analysis. By leveraging the latest technology and data analysis techniques, it is possible to develop a highly accurate and effective tool for detecting and preventing fraud.

REFERENCES

[1] Daniel A. Keim, “Information Visualization and Visual Data Mining” IEEE Trans. Visualization and Visual Data Mining, vol. 8,Jan-Mar 2002.

[2] Fuzail Misarwala, Kausar Mukadam, and Kiran Bhowmick, “Applications of Data Mining in Fraud Detection”, vol. 32015.

[3] Esther Nowroji., Vanitha., “Detection Of Fraud Ranking For Mobile App Using IP Address Recognition Technique”, International Journal for Research in Applied Science & Engineering Technology, vol. 4, 2016.

[4] Muhammad Taimoor Khan, Mehr Durrani, Armughan Ali, Irum Inayat, Shehzad Khalid & Kamran Habib Khan “Sentiment analysis and complex natural language” (2016) a springeropen journal

[5] Rohini, V., and M. Thomas. "Comparison of Lexicon based and Naïve Bayes Classifier in Sentiment Analysis." International Journal for Scientific Research & Development 3, no. 4 (2015)

[6] .Neha Puram, Neha M., and K. R. Singh. "An Implementation to Detect Fraud App Using Fuzzy Logic."

[7] Tichkule, Ashwini, Nidhi Nikhar, Dewanand Kapgate, and Omkar Dudhbure. "Revelation of Fraud Apps using Sentiment Analysis App Reviews.

[8] Israel J. Mojica Ruiz ,Meiyappan Nagappan, Bram Adams, Thorsten Berger, Steffen Dienst, Ahmed E. Hassan “An examination of the current rating system used in mobile app store” IEEE

[9] Guzman, Emitza, and Walid Maalej. "How do users like this feature? a fine grained sentiment analysis of app reviews." In 2014 IEEE 22nd international requirements engineering conference (RE), pp. 153-162. IEEE, 2014.

[10] Safrin, Raheesa, K. R. Sharmila, TS Shri Subangi, and E. A. Vimal. "Sentiment analysis on online product review." Int. Res. J. Eng. Technol 4, no. 04 (2017)

[11] Mudinas, Andrius, Dell Zhang, and Mark Levene. "Combining lexicon and learning based approaches for concept-level sentiment analysis." In Proceedings of the first international workshop on issues of sentiment discovery and opinion mining, p. 5. ACM, 2012

[12] .Avayaprathambiha. P, Bharathi. M, Sathiyavani. B, Jayaraj. S “" To Detect Fraud Ranking For Mobile Apps Using SVM Classification " International Journal on Recent and Innovation Trends in Computing and Communication, vol. 6, February2018

[13] .Rao, Shivani, and Misha Kakkar. "A rating approach based on sentiment analysis." In 2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence, pp. 557-562. IEEE, 2017.

[14] P. Rohini, K. Pallavi, J. Pournima, K. Kucheta, and P. P. Agarkar, “MobSafe : Forensic Analysis For Android Applications And Detection Of Fraud Apps Using CloudStack And Data Mining,” vol. 4, no. 10, pp. 3779–3782, 2015.

[15] Javvaji Venkataramaiah, Bommavarapu Sushen, Mano. R, Dr. Gladispushpa Rathi, “An enhanced mining leading session algorithm for fraud app detection in mobile applications” InternationalJournal of Scientific Research in Engineering., April2017.

[16] Siddharth Grover,“Malware detection: developing a system engineered fair play for enhancing the efficacy of stemming search rank fraud”, International Journal of Technical Innovation in Modern Engineering & Science,Vol. 4, October2018.

[17] Suleiman Y. Yerima, Sakir Sezer, Igor Muttik, “Android Malware Detection Using Parallel Machine Learning Classifiers”, 8th International Conference on Next Generation Mobile Applications, Services and Technologies,Sept.2014.

[18] Suprayogi, Erry, Indra Budi, and Rahmad Mahendra. "Information Extraction for Mobile Application User Review." In 2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS), pp. 343-348. IEEE, 2018.

[19] PatilRohini, Kale Pallavi, Jathade Purnima,KudaleKucheta,Prof.PankajAgarkar,“MobSafe: Forensic Analysis ForAndroid Applications And Detection Of Fraud Apps Using CloudStack And Data Mining”, International Journal of Advanced Research in Computer Engineering&Technology,Vol.4, October2015.

[20] MahmudurRahman,MizanurRahman, BogdanCarbunar,andDuenHorngChau,“Search Rank Fraud and Malware Detection in Google Play”, IEEE Transactions on Knowledge and Data Engineering,Vol. 29, June2017.

[21] Dr.R.Subhashini and Akila G,"Valence arousal similarity based recommendation services ", IEEE International

Conference on Circuit, Power and Computing Technologies, ICCPCT 2015.

[22] Gladence, L. Mary, M. Karthi, and V. Maria Anu. "A statistical comparison of logistic regression and different Bayes classification methods for machine learning." ARPN Journal of Engineering and Applied Sciences 10, no. 14 (2015): 5947-5953

[23] Tahura Shaikh#1, Dr. DeepaDeshpande,"Feature Selection Methods in Sentiment Analysis and Sentiment Classification of Amazon Product Reviews", International Journal of Computer Trends and Technology (IJCTT),Vol. 36, June2016.

[24] D.Janet, Vikrant Chole ,"A Review on Ranking Based Fraud Detection in Android Market", International Journal of Science and Research,Vol. 6, January2017.

[25] VivekPingale, Laxman Khile, Pratik Phapale, Pratik Sapkal, Prof. Swati Jaiswal,"Fraud Detection & Prevention of Mobile Apps using Optimal Aggregation Method", International Journal of Advanced Research in Computer Science and Software Engineering,Vol. 8, March2016.