

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Fraud App Detection Using Sentiment Analysis

Shivani V. Bandodkar¹

Dept. of Computer Engineering
Goa University
Shree Rayeshwar Institute of
Engineering & Information Technology,
Goa University,
Shiroda, Goa, India.

Saish S. Paradkar²

Dept. of Computer Engineering
Goa University
Shree Rayeshwar Institute of
Engineering & Information Technology,
Goa University,
Shiroda, Goa, India.

Parth A. Dalvi³

Dept. of Computer Engineering
Goa University
Shree Rayeshwar Institute of
Engineering & Information Technology,
Goa University,
Shiroda, Goa, India.

Sweta Kamat⁴

Dept. of Computer Engineering
Goa University
Shree Rayeshwar Institute of
Engineering & Information Technology,
Goa University,
Shiroda, Goa, India.

Poonam Sinai Kenkre⁵

Dept. of Computer Engineering
Goa University
Shree Rayeshwar Institute of
Engineering & Information Technology,
Goa University,
Shiroda, Goa, India.

Dr. Shailendra Aswale⁶

Dept. of Computer Engineering
Goa University
Shree Rayeshwar Institute of
Engineering & Information Technology,
Goa University,
Shiroda, Goa, India.

Abstract: Fake mobile application are software that mimic functionality of valid, reliable and genuine applications. once this applications are installed they perform malicious actions like aggressively display advertisements to get revenue from that, intercept sensitive data from your system, infect devices and so on. Most of the times the user cannot differentiate between the fake and legitimate applications hence before downloading any app people always enquire about the opinion of the app by the users. In this paper we are introducing a platform where people can enquire about the application before downloading it. The results are based on previous reviews and ratings given by the users which provides a chance to recognize user experience with a particular mobile app. Basically we will analyze the reviews using sentiment analyses which is a text classification technique which analysis text and tells whether the sentiment is positive, neutral or negative.

Keywords: User reviews, sentiment analysis, lexicon, NLP, Tokenization, stopwords removal.

I. INTRODUCTION

Sentiment is an emotion or attitude prompted by the feelings of the customer. Sentiment analysis is also referred to as opinion mining, as opinions are collected from customer is mined to reveal the rating of the app. The process of Sentiment analysis comes under machine learning. [1]

Information is gathered and is analyzed to determine the sentiment about the information such as negative or positive sentiment. Before purchasing the app people always enquire about the opinion of the app by the other users. [2]

The process of Sentiment analysis uses natural language processing (NLP) to collect and examine the opinion or sentiment of the sentence. It is popular as many people prefer to take some advice from the users. As the amount of opinions in the form of

reviews, blogs, etc. are increasing continuously, it is beyond the control of manual techniques to analyze huge amount of reviews & to aggregate them to a efficient decision. Sentiment analysis performs these tasks into automated processes with less user support. [3]

It is not always possible to have a one technique to fit in all solution because different types of sentences express sentiments/opinions in different ways. Sentiment words (also called as opinion words) (e.g., great, beautiful, bad, etc) cannot distinguish an opinion sentence from a non-opinion one. A conditional sentence may contain many sentiment words or sentences, but express no opinion. The type of sentences, i.e., conditional sentences, it have some unique characteristics which make it hard to determine the orientation of sentiments on topics/features in such of the sentences. By sentiment orientation, we mean positive, negative or neutral opinions. Conditional sentences are sentences which describe implications or hypothetical situations & their consequences. In English language, a variety of conditional connectives can be used to form these sentences. A conditional sentence contains two clauses: the condition clause and the consequent clause, that are dependent on each other. Their relationship has significant implications on whether the sentence describes an opinion. [4] As there are more than millions of apps on the App store, there is many competition between apps to be on top of the leader board on the basis of popularity. As leader board is the most important way for promoting apps. The higher rank on the leader board leads to huge number of downloads & million doll or of profit. Apps give advertisement to promote their apps on the leader board. Many apps use fraudulent means to boost their ranking on the leader board of the App store. There are various means to increase downloads & ranking of the app which is done by "bot farms" or "human water armies", human water armies are a group of internet ghostwriters who are paid to post fake reviews. The app is said to fraud on the basis of 3 parameters: Ranking, Rating & Review of the app. In ranking based we check the historical ranking of the app, there are 3 different ranking phases, rising phase, maintaining phases & recession phase. The apps ranking rising to peak position on leader board (ie. rising phase), to keep at the peak position on the leader board (ie. maintaining phase), & finally decreasing till the end of event (ie. recession phase). The reviews are taken from the dataset and are converted into tokens on which sentiment analysis is performed.

II. RELATED WORK

Various papers related to the proposed work were studied.

A web site is created and Data is collected from it ie. Star rating and Textual format

For the process of Data pre-processing techniques like part of speech tagging, negation phase identification algo, Feature extraction, K-means cluster algorithm is used and then Performance is evaluated using following equations:

$$\text{Total data count} = tp + fp + tn + fn$$

$$\text{Recall} = tp / (tp + fn)$$

$$\text{Precision} = tp / (tp + fp)$$

$$\text{Accuracy} = (tp + tn) / (tp + tn + fp + fn) [1]$$

Some movie reviews are taken and data is divided into training and testing data. In training data feature selection is performed using MI, IG, TF-idf, chi-square and both the training and testing data is classified using vector machine. In this paper they have also focused on unigram and bigram technique to extract sentiments as a result unigram is the best method to extract sentiment. [2]

In this paper authors they have discussed about 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. [3]

This paper focuses on conditional sentences, which have some unique characteristics that make it hard to determine the orientation of sentiments in such sentences. Consequent Whole-sentence-based classification: classifier and Condition classifier is used. [4]

Data is collected from user forums: Cell phone, Automobile, LCD TV, Audio systems and Medicine. Clustering based approach is a new way of detecting sentence polarity it helps to categorized similar words together without human interference. [5]

Apk file of mobile application is uploaded on the web application. APK parser is used to extract information about the application such as reviews, ratings and historical record. Natural Language Processing is used to perform sentiment analysis on the reviews. By applying rule for detection of fraud application, it generates the graph results. If the rating count is greater than 3 then it is considered as a positive result. And if the rating count is less than 3 then it is considered as a negative result. Methodology used are cloud stack, data mining and NLP. [6]

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

In this paper 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. [8]

User reviews are collected using open source scrapping 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. [9]

In this paper The Tweets Sentiment Analysis Model analyses tweets data. It can identify positive, negative or neutral sentiments and measure intensity of positive/negative opinions in regard to any category. The framework of the TSAM consists of three modules: Feature selection module that extracts the relevant words from each sentence Sentiment identification module that associates expressed opinions with each relevant entity in each sentence level. Sentiment aggregation and scoring module calculates the sentiment scores for each entity. [10]

Google API calculation approach is used to calculate the rank of the applications using Calculation algorithm where they take application ratings from play store and calculate the ranks using the calculations. [11]

Feature extraction in sentiment analysis is an emergent research field so in this paper we have concentrated on related work performed to identify directions for future work. There are many feature selection techniques, NLP based, Machine learning or clustering based, Statistical, Hybrid, are discussed. Features are categorized as syntactic, semantic, lexico-structural, implicit, explicit and frequent, making it easy for the future researchers to work on. Different pre-processing modules like POS tagging, stop word removal, stemming and lemmatization are discussed.

Finally we conclude that feature space reduction, redundancy removal and evaluating performance of hybrid methods of feature selection can be the future direction of research work for all researchers in the field of feature extraction in sentiment analysis.[14]

Developers have developed a ranking fraud detection system for mobile Apps. Specifically, we show that ranking fraud happened in the leading sessions and provided a method for mining leading sessions for each App from its historical ranking records. Then, we identify ranking based evidences and rating based evidences for detecting ranking fraud. Moreover, we proposed 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 validate the proposed system with extensive experiments on the real-world App data collected from the App store. Experimental results showed the effectiveness of the proposed approach.[15]

The main objective is fraud application detection using fuzzy logic to differentiate the actual fraud apps. The proposed system perform classification of apps & detect 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.[16]

Sentiment Analysis is major task of NLP (natural language processing). Data used as input are online app reviews. The objective content from the sentences are removed and subjective content is extracted. The subjective content consists of sentiment sentences. In NLP, part-of-speech (POS) taggers are developed to classify words based on POS. Adjective and verbs convey opposite sentiment with help of negative prefixes. Sentiment score is computed for all sentiment tokens.[17]

Related work:

Paper study can be summarized into a table which can be grouped into various categories for analyzing whether the app is fraudulent or not.

Sr No	Category	Journal	Methodology	Algorithms	DataSet	Advantages	Disadvantage	Efficiency
1	Sentiment analysis on online product review,	International research journal of engineering and technology	Machine learning NLP	K means cluster	Dataset used	Helps in finding accurate review of product	Most reviews have both positive and negative situation	89%
2	Evaluation of features on sentimental analysis	International conference on information and communication technologies(ICICT)	Support vector machine	Porter stemming algo	Online shopping website	Unigram is the best method to extract sentiment	Bigram with streaming of stop words gives less accuracy	82%
3	Sentiment analysis and complex natural language,	Springer journal	NLP Sentiment ala	Naive bayes	Movie review	Naïve bayes : assumes feature independence. Centroid evaluation is sensitive to noise. K_nearest neighbor: sensitive to irrelevant features.. Support vector machine: require more resources. Lexicon based: struggle with domain context	naïve bayes : simple and fast. Centroid evaluation classify on vector distance K_nearest neighbor: handle co-related features. Support vector machine: classify on hyperplane. Lexicon based: can identify new lexicon	Not Specified

4	Sentiment Analysis of Conditional Sentences	Conference on Empirical Methods in Natural Language Processing	Consequent Wholesentencebased classification: classifier: Condition classifier:	Part of speech tagging	Web 2.0	Canonical tense patterns- have been showed useful in classification.	Special conditional sentences that do not use easily recognizable conditional connectives are few but difficult to recognise them	Not Specified
5	A Clustering-based Approach on Sentiment Analysis	International research journal of engineering and technology	Matlab toolbox	Porters algorithm	User forums: Cellphone, Automobile, LCD TV, Audio systems and Medicine.	Well performed, efficient and non human participating approach on solving sentiment analysis problem. Produce accurate cluster results in short time.	Clustering results are unstable due to random selection of centroids in kmeans. The size of the document set might influence the outcome.	
6	MobSafe: Forensic Analysis For Android Applications Using Cloud Stack And Data Mining	International Journal of Advanced Research in Computer Engineering & Technology	Natural Language Processing Data Mining Cloud Stack Sentiment Analysis	Part Of Speech Apk Parser	Google Play Store	Able to compare two applications	Most reviews have both positive and negative situation	Not Specified
7	Revelation of Fraud Apps using Sentiment Analysis App Reviews	National Conference on Emerging Trends In Engineering & Technology	Natural Language Processing Data Mining Sentiment Analysis	Tokenization Feature Selection Sentiment Classification	Apple Store Google Play Store	Detect Fraud using rating, review and ranking evidences	APK file is needed	Not Specified
8	Comparison of Lexicon based and Naïve Bayes	International Journal for	Supervised Approach	Naive Bayes Semantic	Blogs Movie	Lexicon Based is more accurate	Ineffective to find fraud in	Lexicon bases =
	Classifier in Sentiment Analysis	Scientific Research & Development	Corpus Based Approach		Review		leading sessions	92% Naive Bayes= 86%
9	How Do Users Like This Feature? A Fine Grained Sentiment Analysis Of App Review	IEEE 2017	Natural Language Processing Sentiment Analysis	Topic Modelling Lexical Collocation Finding Algorithm Latent Dirichlet Allocation	Apple Store Google Play Store	Precision results were higher for short reviews	Naive Bayes is less accurate compared to lexicon based	Precision=91% Recall=73%
10	Sentiment Analysis On Tweets For Social Events	IEEE 2013	Sentiment Analysis NLP	Feature Selection and Extraction Sentiment Classification Sentiment Aggregation	Twitter	It is feasible and accurate	Non frequently mentioned features are not detected Lexical sentiment analysis has limited handling of negation and no handling of past tense and sarcasm	Not Specified
11	An examination of the current rating system used in mobile app store	IEEE	Google API calculation approach	Calculation algorithm	Google play	Finds accurate ranking if sufficient and updated data provided	POS Processing not used	Not Specified

12	Mining comparative sentences and relations	Department of computer science university of Illinois at Chicago	Data mining	Part of speech tagging CSR GRNERATION	Comments	Helps to calculate polarity of comparative sentence	Due to cumulative nature of the current store rating it is difficult to climb up from an initial poor rating after huge number of raters have rated the app.	Not Specified
13	Discovery Of Ranking Fraud for Mobile Application	International Journal of Engineering Science and Computing	Data Mining,	Pattern Analysis, Natural Language Processing, POS tagging, Naive Bayes classifier	Google Play Store	Evidences can be modeled by statistical Hypothesis test, Easy to extend with other evidences from domain knowledge	After extorting three times of evidences, there is no step to merge them	Not Specified
14	A Review of Feature Extraction in Sentiment Analysis	Journal of Basic and Applied Scientific Research	Data mining, Sentiment Analysis, Feature Extraction	POS tagging, Clustering,	Google Play Store	Many feature selection techniques are discussed	Many issues are faced in feature extraction	Not Specified
15	Ranking Fraud Detection For Mobile Apps	ACM international conference	Data Mining, Natural Language Processing, Sentiment Analysis	Permutation based model, score based, Dumpster-Shafer Rules, Gaussian Approximation	Google Play Store & Apple App Store	Detect Fraud Ranking, Anomaly Detection	Ineffective to find fraud in leading sessions	Not Specified
16	An Implementation to Detect Fraud App Using Fuzzy Logic	International Journal on Future Revolution in Computer Science & Communication Engineering	Fuzzy Logic	Tokenization, Fuzzy logic algorithm, Ontology,	Google Play Store	Classification of Apps & detect their group(good, bad, neutral)	Ranking parameter not used	93.75%
17	Sentiment analysis using product review data	Springer	Sentiment analysis or opinion mining	POS tagging, Negative phrase identification, Sentiment score computation,	App store	Tackles problems on Sentiment analysis, sentiment	Not work well for reviews that purely contains implicit sentiment	Not specified
				Feature vector formation		polarity categorization		
18	Information Extraction for Mobile Application User Review.	IEEE	Filtering Content Classification Topic Modelling Sentiment Analysis	Naïve Bayes SVM Logistic Regression Non Negative Matrix Latent Dirichlet allocation	User reviews from app store	SVM work better than other classification algorithm comparing with Naïve Bayes and Logistic Regression, SVM achieved best F-measure using unigram feature	Only used existing methods for filtering, classification, sentiment analysis and topic modelling	83.5%

Table2.1: Comparison table for different methodology

We have compared methods based on four different classification methods:

1. Multinomial naïve bayes
2. linear SVM
3. K-neighbor
4. Perception

Sr no	Method	Accuracy	Training time	Prediction time
1	Multinomial naïve bayes	89.9%	0.008 sec	0.001 sec
2	linear SVM	90%	0.075 sec	0.001 sec
3	K-neighbor	85.8%	0.003 sec	0.003 sec
4	Perception	88.8%	0.019 sec	0.002 sec

Table2.2 : Comparison table

As seen in above table multinomial naïve bayes has an accuracy of 89.9% and less computing time hence it is more preferred over other methods. Our fraud application detection method uses it as classification method.

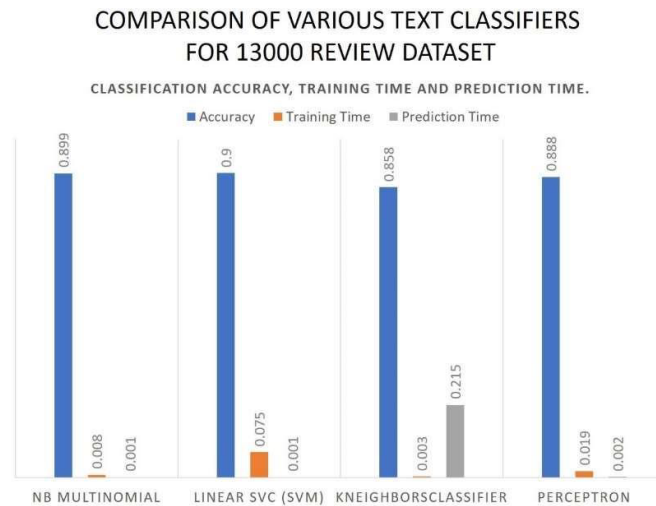


Fig2.1: Comparison of text classifiers

III. PROPOSED METHODOLOGY

The main purpose of our paper is propose a model to extract valuable information of mobile applications based on user reviews, ratings and ranking and aggregation of this evidences 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. [1]

DATA COLLECTION DETAILS

Data collection is an important part of Machine Learning. Data collection is the process of gathering and measuring information for different available sources. Machine Learning requires a huge set of data having multiple attributes, to be able to classify some input parameters more accurately. Data collection is the important aspect that makes the algorithm training possible. It has been observed that greater number of attributes yields a better result.

Training data for fraud app detection is obtained from

Training data

Training dataset was used for training the algorithm so that algorithm learns and produce results. Training dataset consist of 13000 entries (reviews, sentiment value). Training dataset consists 50% of positive and 50% of negative reviews.

Testing dataset: Testing dataset was used for evaluating the model/algorithm with trained dataset. Testing dataset is real time dataset which is extracted from google playstore.

DATA PREPROCESSING

The process of converting data to something a computer can understand is called Preprocessing. The dataset which is obtained in data collection is not in the form which can be used by the classifier. Various Data preprocessing and feature extraction techniques must be performed on the dataset to make it suitable for generation of classification model. The python library Pandas is used to perform the preprocessing techniques on the dataset. Preprocessing steps are:

Tokenization

Tokenization basically refers to splitting up a large body of text into smaller lines, words or even creating words for a non-English language. Various tokenization functions are inbuilt into nltk module itself.

Stopwords Removal

Stop Word Removal is a process of filtering out useless data. In NLP, useless words are referred to as stopwords.

Lowercase conversion

In this all the upper case letters are converted to lower case.

Tfidf Vectorizer

The Tfidf Vectorizer will tokenize documents, learn the vocabulary and inverse document frequency weightings, and allow you to encode new documents.

PROPOSED RESEARCH DESIGN

The proposed approach for the system can be carried out by using corpus based and Naïve Bayes based approach to detect fraud application. First the dataset is prepared so that it can be used for the classifier. The dataset is first stored in a data structure dataframe which can be made by the pandas library. By using the tfidfVectorizer function, various features are extracted based on which the classifier is prepared which is then used for detection of fraud applications. For testing data we are using real time user reviews extracted from the Google play store using Google play scraper. Along with the reviews, ratings of the application are also extracted. On this reviews preprocessing is done by using tfidfVectorizer. This input is then given to the naïve bayes classifier which predicts the polarity for each review. The input to the model is the name of the application. The model extracts the reviews and gives it to the classifier for prediction. If the review is positive then score given to it is 1 and if the review is negative than the score given to it is 0. Finally, on the basis of sentiments and ratings of the application pie chart and ratings are displayed and recommendation is given to the user.

Algorithm: Basic steps describing the proposed algorithm are as follows:

1. Data Preprocessing
 - Tokenization
 - Stopwords Removal
 - Lowercase conversion
 - Feature Extraction
2. Naïve Bayes Classification
 - Sentiment Score Generation
3. Online Review Extraction
 - Data preprocessing
 - Naïve Bayes Classifier
 - Prediction
4. Online Rating Extraction
5. Final Predicted Recommendation

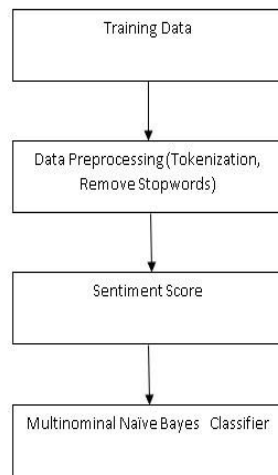


Fig3.1 Training data

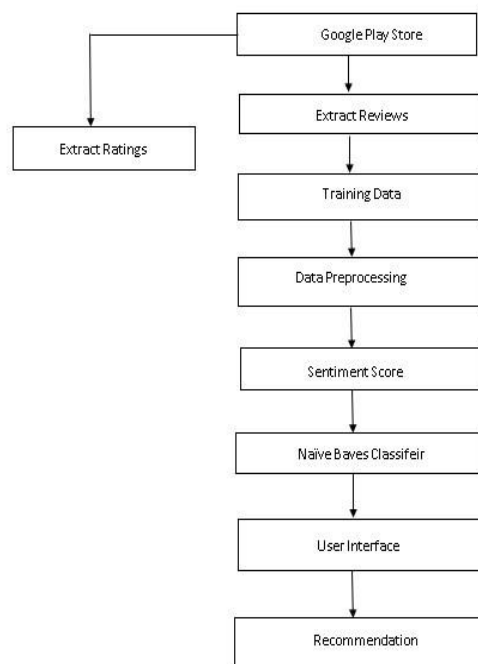


Fig3.2 Testing data

RESULT ANALYSIS

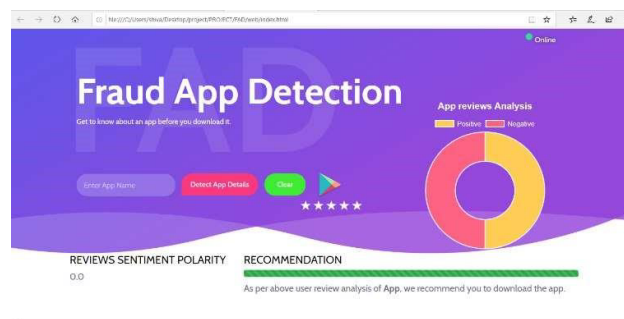


Fig4.1 Display page

When the user enters the app name and clicks on the detect app details button on the index.html webpage, It takes the text stored in the textbox and performs prediction function on it. From the playstore, the real-time reviews are collected and result is displayed.



Fig 4.2 App Detection

IV. CONCLUSION

In this paper, we have conducted a survey regarding different methodologies used in reviewing the status of application and predicting whether it is fraud or not. Our proposed methodology deals with sentiment analysis which has an advantage over the other methods due to the fact that lexicon based analysis is more accurate and fast than other approaches. The main advantage is that the reviews are real time and are extracted directly from playstore.

ACKNOWLEDGEMENT

We would like to thank my batch mates. We remember the journey we took to get this thing done we stuck together and did our best.

References

1. 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).
2. Shahana, P. H., and Bini Omman. "Evaluation of features on sentimental analysis." *Procedia Computer Science* 46 (2015): 15851592.
3. "Muhammad Taimoor Khan, Mehr Durrani, Armughan Ali, Irum Inayat, Shehzad Khalid & Kamran Habib Khan "Sentiment analysis and complex natural language" (2016) a springeropen journal
4. Narayanan, Ramanathan, Bing Liu, and Alok Choudhary. "Sentiment analysis of conditional sentences." In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pp. 180-189. Association for Computational Linguistics, 2009.
5. Li, Gang, and Fei Liu. "A clustering-based approach on sentiment analysis." In *2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering*, pp. 331-337. IEEE, 2010.
6. Xu, Jianlin, Yifan Yu, Zhen Chen, Bin Cao, Wenyu Dong, Yu Guo, and Junwei Cao. "MobSafe: cloud computing based forensic analysis for massive mobile applications using data mining." *Tsinghua science and technology* 18, no. 4 (2013): 418-427.
7. Tichkule, Ashwini, Nidhi Nikhar, Dewanand Kapgate, and Omkar Dudhbure. "Revelation of Fraud Apps using Sentiment Analysis App Reviews,
8. 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).
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. Zhou, Xujuan, Xiaohui Tao, Jianming Yong, and Zhenyu Yang. "Sentiment analysis on tweets for social events." In *Proceedings of the 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pp. 557-562. IEEE, 2013.
11. 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*
12. Jindal, Nitin, and Bing Liu. "Mining comparative sentences and relations." In *AAAI*, vol. 22, no. 13311336, p. 9. 2006.
13. Zhu, Hengshu, Hui Xiong, Yong Ge, and Enhong Chen. "Discovery of ranking fraud for mobile apps." *IEEE Transactions on knowledge and data engineering* 27, no. 1 (2014): 74-87.
14. Asghar, Muhammad Zubair, Aurangzeb Khan, Shakeel Ahmad, and Fazal Masud Kundi. "A review of feature extraction in sentiment analysis." *Journal of Basic and Applied Scientific Research* 4, no. 3 (2014): 181-186.
15. Zhu, Hengshu, Hui Xiong, Yong Ge, and Enhong Chen. "Ranking fraud detection for mobile apps: A holistic view." In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pp. 619-628. ACM, 2013..
16. Neha Puram, Neha M., and K. R. Singh. "An Implementation to Detect Fraud App Using Fuzzy Logic."
17. Fang, Xing, and Justin Zhan. "Sentiment analysis using product review data." *Journal of Big Data* 2, no. 1 (2015):

-
18. 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.
 19. Cambria, Erik, and Bebo White. "Jumping NLP curves: A review of natural language processing research." IEEE Computational intelligence magazine 9, no. 2 (2014): 48-57.
 20. Qiu, Guang, Bing Liu, Jiajun Bu, and Chun Chen. "Opinion word expansion and target extraction through double propagation." Computational linguistics 37, no. 1 (2011): 9-27.
 21. 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.
 22. 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.