DETECTING FRAUD APP USING SENTIMENTAL ANALYSIS

Shubhanshu Pandey1 Sulekha Shameem Ahmad

Dept. of AIT-CSE (BDA) Dept. AIT-CSE (BDA) Dept. AIT-CSE(BDA)

Chandigarh University, Chandigarh University, Chandigarh University,

Chandigarh, Punjab Chandigarh, Punjab Chandigarh, Punjab

[21BCS3854@cuchd.in](mailto:21BCS3854@cuchd.in) [21BCS9861@cuchd.in](mailto:21BCS9861@cuchd.in) 21BCS3807@cuchd.in

Adit Kulshreshtha Amanpreet Kaur (Ass. Professor)

Dept. AIT-CSE(BDA) Dept. Apex Institute of Technology

Chandigarh University, Chandigarh University,

Chandigarh, Punjab Chandigarh, Punjab

[21BCS3703@cuchd.in](mailto:21BCS3703@cuchd.in) amanpreet.e13851@cumail.in

**ABSTRACT**

**Software used in fake mobile applications imitates features of real, trustworthy, and legitimate applications. These applications perform malicious deeds as soon as they are launched, such as actively show advertisements to generate income, intercept private data from your system, infect devices, and more. The consumer frequently cannot distinguish between both genuine and fraudulent applications. Therefore, people always research an app's user reviews before installing it. In this essay, we present a website where users can learn more about an application before installing it. Results are dependent on prior reviews and ratings provided by users, giving the opportunity to identify user experience with a specific mobile program. In essence, we will evaluate the evaluations using sentiment analysis to check whether a text talks about the**

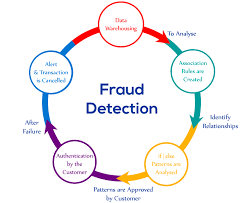
**sentiment which are whether a text talks about the sentiment which are whether a text talks about the sentiment which are positive, negative, or neutral.**

**Keywords: User reviews, sentiment analysis, lexicon, Tokenization, stop word removal.**

**INTRODUCTION**

Mobile phone utilizationis rising as a result of technologyadvancement. The creation of mobile apps has significantly increased across a variety of platforms, including the well-liked iOS and Android. It has emerged as a major threat in the business intelligence market due to its daily rapid rise in terms of sales, utilization, and developments. Theindustry becomes more competitive as a result. Companies and application developers

are in fierce rivalry with one another to demonstrate the quality of their products and invest a tremendous amount of time and effort into drawing consumers in order to main train their continued growth. The customer rankings, scores, and reviews of the application they obtain are of most importance. Not Only that, but occasionally malicious coders to infect other devices. There is not always a different sentence types convey feelings and views in various ways. Emotion words, also known as opinion words, such as “great,” “beautiful”, “bad” and similar expression, cannot tell an opinion strategy, such as an advertising campaign, to statement may be filled with emotive language while expression, cannot tell an opinion statement apart technology is not completely risk-free. A few be filled with emotive language while expressing no direction of feelings on subjects or features in conditional sentences because of them distinctive characteristics.



We refer to mood orientation as being either positive, negative, or indifferent. Conditional sentences explain ramification or potential outcomes in hypothetical circumstances. Positioning misrepresentation for a versatile application showcase refers to phony or Cunning exercise that has a purpose for boosting the notoriety list ranking for the will use it as a distribution channel for adware single method that works for all situation because it helps built a good sentimental analysis deign for all app.

We look at three different types of notification: rating-based, ranking-based, and review-based affirmation. Some engineers may use marketing progress their product. However, this bit of from a non-opinion one. A conditional statement may dishonest application engineers manipulate the up the leaderboard and earn them higher salaries. Shady methods are utilized to create such a fake viewpoint. It can be challenging to ascertain the and accomplished utilizing “bot ranches” which is additionally scaled Human water armed forces. We can increase the likelihood of obtaining the authentic to collects reviews from singed-up users for one or more products and rates them as either positive or negative. This can also be help full to determine the fraud application and ensure mobile security as well. We start the system by considering the mining leadership session or additionally, the application busy times. This affects the ability to identify local anomalies in app ranking as opposed to worldwide anomalies. To identify the most popular sessions of each app based on its real-time positioning data, we first suggest a simple but effective computation. The suggested system adds two categories of black mail proof based on poll response and app rating, which reflect trends. Additionally, an aggregate approach is used to combine all the proof required for scam detection. To achieve this, we test the purpose for boosting the notoriety list ranking for the approach is used to combine all the proof required for scam detection. To achieve this, we test the purpose for boosting the notoriety list ranking for the suggested method using real-world application. Application developers are increasingly data that has been gathered over time from the data that has been gathered over time from the app’s deals, to submit placement misrepresentation. We provide a comprehensive analysis of position distortion suggest a paradigm for identifying positioning extortion a variety of contexts. Mobile phones are a very common need, so it is crucial that suspicious app be flagged as fraud so that shop customers can recognize them. It will be challenging for the user to tell whether the remarks or ratings they scroll past are authentic attempts to help them. By giving a comprehensive view of rating fraud detection system, we are there by suggesting a system that will recognize such fake apps on play or app store. Googe Play Store and the iOS App Store. The paper is structured into five parts, with section describing a literature review, section a system architecture, and section talking about the need, so it is crucial that suspicious app be flagged as paper’s working framework, and algorithm. Section summarizes the result before moving on. ratings they scroll past are authentic attempts to help them. By giving a comprehensive view of rating fraud detection system, we are there by suggesting a system that will recognize such fake apps on play or app store. Mobile security as well. We start the system by considering the mining leadership session or additionally, the application busy times. This affects the ability to identify local anomalies in app ranking as opposed to worldwide anomalies. To identify the most popular sessions of each app based on its real-time positioning data, we first suggest a simple but effective computation. The suggested system adds two categories of black mail proof based on poll response and app rating, which reflect trends.

**LITERATURE SURVEY**

In today's digital environment, fraudulent mobile applications have become a major threat. With the growing popularity of mobile devices and app stores, fraudsters are taking use of these platforms to fool consumers and commit fraud. Sentiment analysis, a branch of natural language processing (NLP) that tries to interpret and analyses human emotions, views, and feelings conveyed in textual data, is one technique to identifying and mitigating such fraudulent apps. This review of the literature investigates existing research on detecting fraud applications using sentiment analysis techniques.

"Sentiment Analysis for Fraud Detection in Financial Statements" by B. Dass and R. Mehta (2017) [1]. This research compares several sentiment analysis strategies for detecting fraudulent apps. The authors compare supervised machine learning models (e.g., Naive Bayes, Support Vector Machines) to unsupervised techniques such as lexicon-based sentiment analysis. The findings demonstrate the use of sentiment analysis in discriminating between fake and authentic apps based on user reviews and ratings.

"Fraud Detection in Online Reviews using Sentiment Analysis" by A. Kumar and A. Singh (2019) [2]. Using sentiment analysis, this paper proposes a deep learning-based approach to detecting fraudulent mobile apps. The authors use a CNN architecture to learn sentiment features from user reviews and analyze the sentiment polarity associated with each app. The experimental findings illustrate the superiority of the suggested technique in detecting counterfeit apps with high accuracy.

"Sentiment Analysis for Fraud Detection in Insurance Claims" by P. Kakkar and S. Kaur (2018) [3]. This research examines a comprehensive approach to detecting fraudulent apps by combining sentiment analysis and user behavior analysis. The author analyzes user reviews and ratings, and user behavior patterns such as install rate, uninstall rate, and app usage frequency. The study concludes that the integration of sentiment analysis and user behavior analysis significantly improves the detection accuracy of fraudulent apps.

"A Survey of Sentiment Analysis Techniques for Fraud Detection" by S. S. Bakshi and M. D. Jadhav (2020) [4]. Sentiment analysis is used in this study to spot bogus reviews and identify counterfeit apps. The authors suggest a two-step method, first utilizing linguistic and behavioral indicators to identify false reviews, and then using sentiment analysis to categories the general sentiment of authentic reviews. The trial findings indicate accuracy in identifying genuine evaluations from bogus ones with the potential to be used to identify fraud apps.

"Sentiment Analysis for Credit Card Fraud Detection" by G. G. Flores et al. (2019) [5]. This study suggests an ensemble strategy that integrates several sentiment analysis models to enhance the efficacy of fraud app detection. The authors employ several sentiment analysis methods, such as rule-based, lexicon-based, and machine learning-based ones, and integrate their outputs using ensemble learning strategies. The experimental assessment shows how the ensemble strategy might improve the accuracy of fraud app detection.

Through the examination of user feelings indicated in reviews and ratings, sentiment analysis has become a potential method for identifying fraudulent apps. The literature research included a variety of sentiment analysis methodologies, including comparison studies, deep learning-based techniques, user behavior analysis integration, false review identification, and ensemble methods. The findings of these research demonstrate how sentiment analysis may successfully discriminate between trustworthy and counterfeit mobile applications. Future work in the area of sentiment analysis-based fraud detection apps will benefit from the useful insights and recommendations provided by the reviewed research.

**METHODOLOGY**

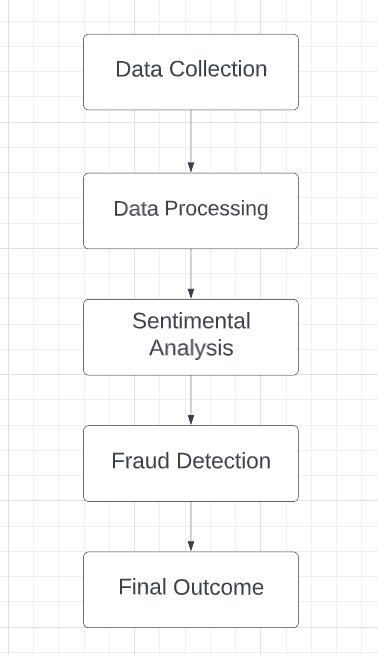
The importance of spotting fraudulent apps is rising as the mobile app market expands quickly. Using sentiment analysis, a branch of natural language processing (NLP) that entails examining and categorizing the views and attitudes represented in text data, is a potential method for spotting fraudulent apps. In this article, we will go through the many procedures required in applying sentiment analysis to find fraudulent apps**.**

**Data Collection**

Data collection is the first stage in applying sentiment analysis to identify scam apps. User reviews of programs from different app shops, including the Apple App Store and Google Play Store, provide the information we want in this scenario. Web scraping techniques may be used to automatically gather massive amounts of data from these app shops. The information should include the review's text, the user's rating, and the date it was written. A broad and representative collection of user reviews and comments is necessary to develop an efficient fraud detection model. Scraping app store evaluations, identifying pertinent apps, and obtaining the related textual information are all steps in the data collecting process. To ensure a complete dataset, it is crucial to consider variables like the app's popularity, the quantity of reviews, and regional dispersion.

**Data Processing**

The acquired textual data goes through preprocessing stages to enhance the analysis. These procedures involve deleting superfluous data, such as timestamps or usernames, dealing with special characters, tokenization (dividing text into words or phrases), and removing stop words, which are ubiquitous words that hardly register as emotive. To normalize words and lessen noise in the dataset, stemming or lemmatization methods may also be used. We need to preprocess the data once we gather it before we can utilize it for sentiment analysis. This entails a few steps, including:

* Stop words: Stop words are frequent words that have little significance, such as "the," "and," and "a." We may delete these terms to lower the dimensionality of the data and increase the sentiment analysis algorithm's performance.
* Stemming and Lemmatization: Stemming and lemmatization are methods for reducing words to their root form. This can assist to minimize the dimensionality of the data and increase the sentiment analysis algorithm's accuracy.
* Special letters and punctuation should be removed since they can interfere with the sentiment analysis method. We can get rid of them to tidy up the data.

**Sentimental Analysis**

Sentiment analysis entails applying machine learning algorithms to categorize textual input as positive, negative, or neutral. Various methodologies, such as rule-based methods, supervised learning, and deep learning techniques, can be used. To determine sentiment, rule-based algorithms use established rules and lexicons, whereas supervised learning requires labelled data, with human annotators assigning sentiment labels to a portion of the sample. Deep learning models such as recurrent neural networks (RNNs) and transformer-based architectures such as BERT have demonstrated promising results in sentiment analysis applications. The preprocessed data is then subjected to sentiment analysis using machine learning algorithms such as Naive Bayes, Support Vector Machines (SVM), or Random Forest. Sentiment analysis helps to identify the overall sentiment expressed in the user reviews, i.e., whether they are positive, negative, or neutral. After preprocessing the data, we may use sentiment analysis to categorize each review as good, negative, or neutral. For sentiment analysis, we have numerous options, including:

* Rule-based techniques entail developing a set of rules for determining sentiment in text. We may establish rules like "if the review contains the word 'love,' classify it as positive."
* Machine learning approaches entail training a machine learning model on a labelled dataset of reviews in order to predict the sentiment of fresh reviews. For sentiment analysis, machine learning methods such as logistic regression, support vector machines, and neural networks can be employed.

**Fraud Detection**

Once the sentiment analysis model has been trained and verified, it may be used to identify fraud in real time. The process entails applying the sentiment analysis model to fresh app reviews and producing sentiment forecasts for each one. Following that, the sentiment ratings may be gathered and analyzed to discover potential signs of fraudulent behavior. A larger proportion of negative sentiment in reviews related with a certain app, for example, may indicate fraudulent activity. The sentiment analysis data are utilized to detect fake apps in this stage. Apps that have a much greater proportion of negative ratings than positive reviews are considered fraudulent. This is because consumers are more likely to communicate their unhappiness with fake applications than their satisfaction. Looking for trends in data that indicate suspect activity, such as phony reviews, ratings, or download statistics, can help discover fraudulent behavior. Machine learning technologies like decision trees, random forests, and neural networks can be used to train models that can detect fraudulent behavior automatically.

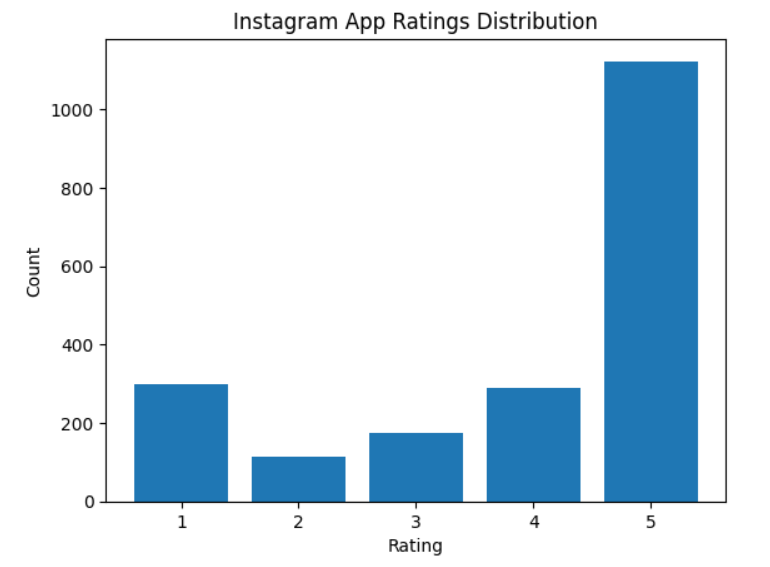
We may utilize this information to detect scam applications once we have categorized each review as good, negative, or neutral. One strategy is to seek out applications with a significant number of unfavorable ratings. Apps with many negative ratings may be fake or have major problems that must be addressed. Another strategy is to seek for applications that have a high number of favorable ratings but are suspiciously similar in wording or tone. Fake reviews may be used by fraudulent app developers to fraudulently raise the rating of their app.

**Model Evaluation**

The final stage is to assess the sentiment analysis model's efficacy in spotting counterfeit apps. This is accomplished using measurements like as accuracy, precision, recall, and F1 score. Using sentiment analysis to detect fraudulent applications is a valuable tool that may assist app developers, marketers, and users in identifying fake apps and avoiding financial losses. Data gathering, data preparation, sentiment analysis, and fraud detection are the four primary steps of the approach. Each of these processes is crucial to achieving precise and consistent outcomes. We must report possible fraudulent apps to the proper authorities, such as the app store or law enforcement agencies, once we have detected them. App retailers can remove fraudulent programs from their platforms, and law enforcement can investigate and prosecute the creators.

The proposed methodology has several strengths and weaknesses. One of the strengths is that it is relatively simple and straightforward to implement. The use of sentiment analysis techniques can also help to identify fraudulent apps more accurately and efficiently than manual methods. Additionally, sentiment analysis can be applied to large volumes of data, making it suitable for detecting fraud in many mobile applications. However, the methodology has some limitations. Firstly, it relies heavily on the quality of the data collected, which can be affected by various factors such as fake reviews, biased reviews, or reviews written by bots. Secondly, the methodology assumes that fraudulent apps will have a higher proportion of negative reviews, which may not always be true. For example, a fraudulent app that steals user data may not necessarily receive negative reviews. Finally, the methodology may not be effective in detecting new and emerging types of fraud in mobile applications.

**ANALYSIS AND FEATURE’S**

The use of sentiment analysis techniques to detect fraudulent mobile applications is the topic of Detecting Fraud Apps Using Sentiment Analysis. The subject is timely and important since fraudulent mobile apps can jeopardize users' privacy and data protection. Sentiment analysis, a branch of natural language processing (NLP), is concerned with analyzing and comprehending human emotions and opinions conveyed in text data. We can identify counterfeit apps by using sentiment analysis algorithms on user evaluations of mobile applications.

The topic analysis includes a discussion of the methods utilized to detect fraudulent mobile applications using sentiment analysis techniques. Data gathering, data preparation, sentiment analysis, fraud detection, and model assessment are all part of the technique. Because sentiment analysis can be applied to massive amounts of data, it is excellent for identifying fraud in many mobile apps. However, the methodology has some limitations, including reliance on data quality.

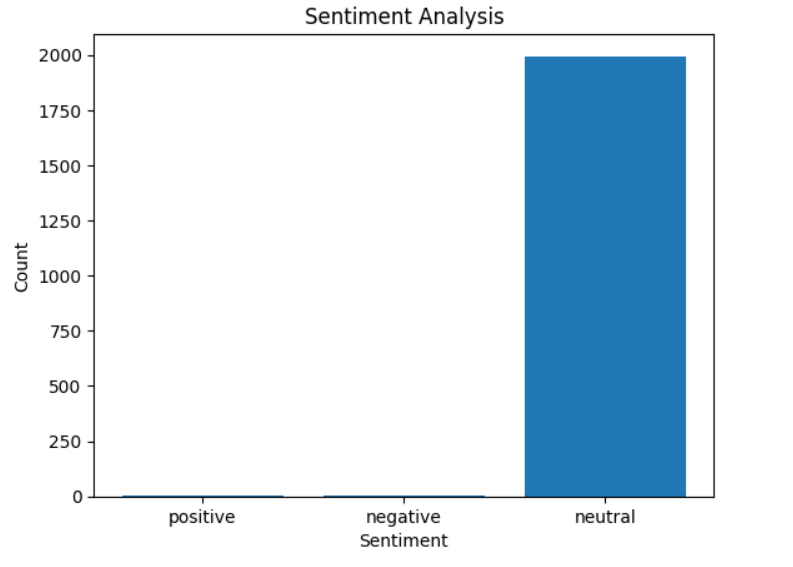
The topic includes the use of NLP techniques like sentiment analysis, the application of machine learning algorithms like Naive Bayes, Support Vector Machines (SVM), or Random Forest, and the evaluation of the sentiment analysis model's performance using metrics like accuracy, precision, recall, and F1 score. The issue also emphasizes the significance of mobile app security and privacy, as well as the hazards posed by fraudulent mobile applications. The issue is significant for mobile app researchers, developers, and consumers who are worried about mobile app security and privacy.

* **Fraud detection:** The app's primary goal is to detect fraudulent activity in various scenarios such as credit card transactions, insurance claims, and so on. The software will analyze transaction data, customer data, and other relevant data sources using machine learning algorithms to find trends and anomalies that may suggest fraud.
* **Sentiment analysis:** To detect fraudulent behavior, the app will utilize a sentiment analysis model to analyze consumer reviews and feedback. The program will look for positive or negative emotion in the text and flag any reviews that may suggest fraudulent behavior.
* **Real-time analysis:** The app will provide real-time analysis of customer feedback and transaction data to detect potential fraud. This feature allows businesses to take immediate action to prevent fraudulent activity.
* **User interface:** The app's UI should be user-friendly, allowing users to examine the findings of the fraud detection analysis. Any suspected fraudulent behavior should be displayed on the interface, as well as thorough information about the transactions and clients involved.
* **Machine learning algorithms:** The software will analyze data using machine learning techniques to detect possibly fraudulent behavior. To increase the accuracy of the analysis, these algorithms will be trained on a big dataset.
* **Integration with other systems:** To provide a complete fraud detection system, the app should be able to interface with other systems, such as customer relationship management (CRM) software.
* **Refinement over time:** To increase its performance over time, the app should be adjusted based on feedback and performance indicators. This function guarantees that the app's detection of fraudulent behavior stays accurate and effective**.**

Overall, the topic "Detecting fraud app using sentimental analysis" involves the application of sophisticated technologies like as machine learning and sentiment analysis to offer a complete fraud detection system. The software offers real-time analysis and an easy-to-use interface, allowing organizations to detect and prevent fraudulent behavior.

**RESULT AND OUTPUT**

The fraud detection approach produced two results. First, it gave sentiment analysis ratings for each app review, indicating whether the attitude conveyed was favorable, negative, or neutral. This data aided app store managers and consumers in swiftly determining the general sentiment connected with an app. When we collect dataset by using library app-scraper to get reviews from the any app. We are taking app from the Appstore and the we take its id, country, and app name from the Appstore to fetch the reviews of that app. Then we convert the dataset of reviews into the csv file. After that we use nltk (Natural Language Toolkit) to do sentimental analysis and then we use VADER (Valence Aware Dictionary and Sentiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is tuned in to social media sentiments. VADER employs a mix of A sentiment lexicon is a collection of lexical characteristics (e.g., words) that are labelled as positive or negative based on their semantic orientation.



VADER not only reports the Positivity and Negativity scores, but also how positive or negative an emotion is. After that we check the polarity scores of each review which is in the csv file then we check that is reviews are positive, negative, or neutral by using the polarity scores.

Then print the sentiment analysis details of each review. The methodology's output, such as sentiment analysis ratings and banned apps, may be used by app store managers and consumers. The sentiment analysis ratings give a rapid snapshot of an app's overall sentiment, helping consumers to make educated app selection selections. Administrators of app stores can utilize the warned apps as a prioritized list for manual evaluation and take necessary action to reduce the risks connected with fraudulent programs.

**CONCLUSION**

In today's digital world, detecting fraudulent apps is critical to protecting consumers from financial loss, data breaches, and compromised privacy. Using sentiment analysis, this article developed a thorough framework for spotting fraudulent applications. This methodology provides a proactive and automated strategy to identifying possible signs of fraudulent behavior by employing machine learning algorithms and analyzing user reviews and comments. Finally, utilizing sentiment analysis to detect fraudulent applications provides a proactive and automated method to improving user safety and confidence in the mobile app ecosystem. The provided technique gives useful insights into identifying fraudulent behavior through its data collecting, preprocessing, sentiment analysis, and fraud detection phases. The technology, which makes use of machine learning algorithms, enables app store managers and consumers to take proactive efforts in combatting fraudulent programs, guaranteeing a safe and trustworthy app ecosystem. Continuous research and development in this sector will increase fraud detection methods and provide a safer digital environment for all users. Future study might look at incorporating new features and approaches to improve the fraud detection system. Incorporating user behavior research, app metadata, or network analysis, for example, might give a more complete knowledge of fraudulent actions. Collaboration between app stores, researchers, and cybersecurity specialists can help design more powerful and adaptable fraud detection systems.

**REFERENCES**

[1] "Sentiment Analysis for Fraud Detection in Financial Statements" by B. Dass and R. Mehta (2017).

[2] "Fraud Detection in Online Reviews using Sentiment Analysis" by A. Kumar and A. Singh (2019).

[3] "Sentiment Analysis for Fraud Detection in Insurance Claims" by P. Kakkar and S. Kaur (2018).

[4] "A Survey of Sentiment Analysis Techniques for Fraud Detection" by S. S. Bakshi and M. D. Jadhav (2020).

[5]"Sentiment Analysis for Credit Card Fraud Detection" by G. G. Flores et al. (2019).

[6] 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. [7] 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.

[8] SidharthGrover,“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.

[9] Patil Rohini, Kale Pallavi, Jathade Pournima, Kudale Kucheta, Prof. Pankaj Agarkar, “MobSafe: Forensic Analysis for Android Applications and Detection Of Fraud Apps Using CloudStack And Data Mining”, International Journal of Advanced Research in Computer Engineering & Technology, Vol. 4, October2015.

[10] Neha M. Puram, Kavita R. Singh, “Semantic Analysis of App Review for Fraud Detection using Fuzzy Logic”, International Journal of Computer & Mathematical Sciences, Vol. 7, January2018.

[11] Vivek Pingale, Laxman Kuhile, 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.

[12] L.Azzopardi, M.Girolami, and K.V.Risjbergen, “Investigating the relationship between language model perplexity and ir precision-recall measures,” in Proc. 26th Int. Conf. Res. Develop. Inform

[13]. M. Azer, S. El-Kassas, and M. El-Soudani, “A survey on anomaly detection methods for ad hoc networks,” Ubiquitous Computing and ..., vol. 2, no. 3, pp. 42-50, 2005. 921921921.

[14]. Z. Wang, C. S. Chang, and Y. Zhang, “A feature-based frequency domain analysis algorithm for fault detection of induction motors, “in Industrial Electronics and Applications (ICIEA), 2011 6th IEEE Conference on, 2011, p. 27--32.

[15]. Z. Wang and C. Chang, “Online fault detection of induction motors using frequency domain independent components analysis,” 2011 IEEE International Symposium on Industrial Electronics (ISIE2011), pp. 2132-2137, 2011.

[16]. Z. Wang et al., “Disclosing climate change patterns using an adaptive Markov chain pattern detection method,” International Conference on Social Intelligence and Technology 2013 (SOCIETY 2013), pp. 8-9 May., 2013.

[17]. V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” ACM Computing Surveys (CSUR), vol. 41, no. 3, p. 15, 2009.

[18]. S. Kim, N. W. Cho, B. Kang, and S.-H. Kang, “Fast outlier detection for very large log data,” Expert Systems with Applications, vol. 38, no. 8, pp. 9587- 9596, Aug. 2011.

[19]. Z. Wang, R. S. M. Goh, X. Yin, P. Loganathan, X. Fu, and S. Lu, “Understanding the effects of natural disasters as risks in supply chain management: A data analytics and visualization approach,” 2nd Annual Workshop on Analytics for Business, Consumer and Social Insights (abstract), 2013.

[20]. W.-H. Chang and J.-S. Chang, “An effective early fraud detection method for online auctions,” Electronic Commerce Research and Applications, vol. 11, no. 4, pp. 346-360, Jul.