As per UGC guidelines an electronic bar code is provided to seure your paper

International Journal for Modern Trends in Science and Technology
Volume 10, Issue 04, pages 332-337.
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol10issue04.html
DOI: https://doi.org/10.46501/IJMTST1004050

Machine Learning-Powered Fraud App Detection: Safeguarding Google Play Store Integrity

S.K.Shankar[1] , Setti Hariharamanikanta[2], Gadi Divya Sri[2], Sarella Udaya Bhaskara Suresh[2], Revanth Kallakuri[2], Hareen Kumar Devu[2]

[1]Assistant Professor, Department of Computer Science Engineering, Pragati Engineering College, Surampalem , Andhra Pradesh, India.
[2]Department of Computer Science Engineering, Pragati Engineering College, Surampalem , Andhra Pradesh, India.

To Cite this Article
S.K.Shankar , Setti Hariharamanikanta, Gadi Divya Sri, Sarella Udaya Bhaskara Suresh, Revanth Kallakuri, Hareen Kumar Devu, Machine Learning-Powered Fraud App Detection: Safeguarding Google Play Store Integrity, International Journal for Modern Trends in Science and Technology, 2024, 10(04), pages. 332-337. https://doi.org/10.46501/IJMTST1004050

Article Info
Received: 06 April 2024; Accepted: 18 April 2024; Published: 26 April 2024.

Copyright © S.K.Shankar et al; This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT

As the range of mobile applications utilized in daily life grows, it becomes more important than ever to keep current and determine which are safe and which are not. It is difficult to form a judgment. Our approach predictions using four criteria: evaluations, feedback, in-app purchases, and the presence of advertisements. The system evaluates three models: Naïve Bayes, logistic regression, and decision tree classifier. These models were then evaluated based on four F1 score parameters. These are recall, precision, and accuracy. A high F1 score should be larger than 0.7, and a recall score greater than 0.5 suggests enhanced precision and accuracy. After analysis, we discovered that the decision tree model was an exceptional model, with an accuracy of 85% and an F1 score.

Keywords- Decision Tree, Google Play Store Apps, Reviews, Ratings, In app purchases, Contains Ad

1. INTRODUCTION

As technology has advanced, so has the use of mobile phones. The number of Play Store apps on numerous major platforms, including the popular Android and iOS, has exploded. It has become a significant difficulty in the business intelligence arena because to its rapid growth through everyday use, marketing, and development. The market is becoming more competitive as a result [7]. Companies and software developers fight intensely to demonstrate the quality of their goods, and they spend a large amount of time and money acquiring clients to secure their long-term viability. Customer feedback and updates on each software that customers can download play an extremely important function [5].

This enables engineers to detect and incorporate difficulties into the design of a new product that fulfils human needs. Instead, then depending on traditional marketing tactics, under the trees App producers may heavily promote their apps and eventually influence their ranking in the App Store. This is sometimes

performed by employing "bot ranch" or "water army" methods to boost the number of downloads and audits [15]. Occasionally, for the benefit of developers, groups of people are employed to commit fraud and leave spam comments and ratings on programs. Crowd turfing is the term used to describe this activity.

As an outcome, it is vital to provide users with accurate and authentic feedback before installing the app in order to prevent mistakes. To process and analyse the many comments and ratings received for each application, an automated method is necessary. Because mobile phones are in such high demand, it is vital that suspicious applications be flagged as fraudulent so that Play Store consumers may readily identify them. The user will be unable to tell if the remarks or ratings they read are fraudulent or authentic for their advantage. By providing a comprehensive view of ranking fraud detection systems, we describe a strategy to detect such fraudulent applications on the Google or Apple store [2].

We can tell if an app is fake or real, thus we provide a method that uses four features: in-app purchases, adverts, ratings, and reviews to determine whether an app is defrauding its consumers [5]. We begin the method by considering the four most significant elements in choosing the target. The scraped data is then trained using several categorization models based on these features before being chosen as the best and most accurate model for the system. During this stage of the selection process, we got a large number of models of variable accuracy. Naive Bayes (83%), Logistic Regression (84%), and Decision Tree (85%) [2].

2. LITERATURE SURVEY

Nevon Projects proposes a comprehensive framework for detecting quality fraud, which may be enhanced by domain-generated data. It is one of the most advanced initiatives for detecting fraudulent applications through information algorithms. This tool detects fraudulent applications with 75-80% accuracy.

On paper, they offered a comprehensive analysis of the facts as well as a proposed fraud detection methodology. They evaluated three forms of verification: quality-based assurances, rating-based guarantees, and review-based validation. In the paper, they consider only updates as parameters with the algorithm naive bayes.

They created a system that detects fake applications using emotive commentary and data processing. In the article, initially examine the app based solely on analytics evaluation to determine whether the software is genuine or fraudulent. When improving your e-mail, utilize to test the spelling of the file. We employ a simple set of criteria and a manage analyst to reveal app fraud, and based on user input, we deliver key-word analysis on a completely scaled basis. In this approach, we also aid with what the customer feels about our app.

On this fraud detection utility, the administrator also provides an app hyperlink that should be supplied to the software, so that when logged in, the user may view the app data and locate the hyperlink for that program. For us, managers upload the software and provide a link to it from the Apple software Store and Google Play. To develop our app, we can ask the consumer after the usage of our app to collect opinions. We use this type of remarks from the consumer and help us enhance the app where the consumer using our app will at once see all apps evaluated by the administrator and introduced by the administrator.

As a result, the user obtains all information about that app, including whether it is phony or not. While an administrator declares fraud rather than their own fraud, this saves the consumer time and provides personal safety. The JP INFOTECH project is an effective method for determining the optimal hours for each application based on record level. By examining the operating environment of applications, this system discovers that Fraud Apps often have different levels for each of the main session patterns than regular applications. A bogus proposal suggests using historical records of application levels to create three jobs based on a fake theme. In addition,

Two sorts of bogus proposals are presented depending on application rating and review history. On paper, the risk summary solely takes into account the application-specific risk signals. They have created a system in which the required elements of risk signals and a restricted number of hazards for Android apps remember the end objective on paper, implying a means to detect fraud fees through the use of IP addresses. The use of a cellular user IP address has also been one of the most recent emails surveyed. In the advertisement for a cellular app, an app named just perverted.

The app is in development. Today, reputation and expectations play a vital role in the cell business.

Experiments accumulate to provide a position in each application. However, IP snooping allows consumers to exchange IP addresses and price the utility many times. The creators of the study investigated the topic of detecting shilling attacks one and a half times while measuring data. Dependent philosophy can be used to propose trustworthy material and learning that is less controlled.

3. SYSTEM ANALYSIS

A. EXISTING SYSTEM

The present technique for the "Fraud App Identification of Google Play Store Applications Utilizing Decision Tree" study addresses the increasing need to distinguish legitimate from potentially fraudulent mobile apps. With the rise of smartphone apps, it is critical to evaluate their safety. The method is based on four important parameters: ratings, reviews, paid purchases in the app, and the presence of adverts in the apps. Three machine learning models were used: Decision Tree Classifier, Logistic Regression, and Naive Bayes Model. The above models are evaluated using four indicators of performance. F1 scores include recall, precision, and accuracy. A strong F1 score should exceed 0.7, and a recall score of above 0.5 is deemed sufficient, especially when combined with higher accuracy and precision levels. After study, the Decision Tree algorithm emerged as a strong choice, with an accuracy rate of 85%, an F1 score of 0.815, a recall value of 0.85, and a precision of 0.87.

DISADVANTAGES OF THE EXISTING SYSTEM

1. **Limited Feature Set:** The system is based on a limited number of features, namely ratings, reviews, in-app purchases, and ad presence. This may not capture all relevant aspects of app behaviour or user interactions, thus leading to errors in fraud detection.
2. **Data Imbalance:** The dataset utilized for training and evaluation may have a class imbalance, with much fewer fraudulent apps than legitimate ones. This imbalance can hinder the model's ability to generalize well and accurately detect fraud.
3. **Static Analysis:** The approach appears to focus on static features like ratings and reviews, but it does not take into account dynamic parts of app behaviour or changes over time. Fraudulent apps may evolve and change dynamically, impacting the model's efficacy.
4. **Dependency on User-Generated Content:** Ratings and reviews are user-generated information, thus their credibility varies. Users may give biased or deceptive information, resulting in inaccurate model predictions. Furthermore, the system may not incorporate cultural or linguistic variations that can influence the interpretation of user assessments.
5. **Model Interpretability:** While Decision Tree models are well-known for their interpretability, they may fail to capture complicated data linkages as well as more advanced models. The model's interpretability may limit its ability to recognize subtle patterns in the data, thereby hurting the identification of sophisticated fraudulent operations.

B. PROPOSED SYSTEM

The proposed approach for improving the "Fraud App Detection of Google Play Store Apps Using Decision Tree" project seeks to overcome the highlighted constraints while also increasing the accuracy and reliability of fraudulent app detection. Additional dynamic features will be added to the feature set to reflect the ever-changing nature of mobile applications. This could include real-time monitoring of app behaviour, tracking changes over time, and analysing user interaction patterns. To reduce the influence of data imbalance, new sampling approaches or ensemble learning methods could be studied. The suggested system would also use sentiment analysis and natural language processing to better comprehend and interpret user feedback, taking into account any biases and linguistic variances. Furthermore, the model architecture will be optimized, maybe by experimenting with more powerful machine learning techniques or deep learning approaches to capture subtle correlations in the data. Finally, focus will be placed on creating a user-friendly interface to allow for easy interpretation of the model's predictions, hence increasing transparency and user trust in the fraud detection system. The suggested system's modifications aim to create a more robust and flexible solution for detecting fraudulent apps on the Google Play Store.

ADVANTAGES OF THE PROPOSED SYSTEM

Certainly, here are five potential advantages of the proposed system for the "Fraud App Detection of Google Play Store Apps Using Decision Tree":

1. **Enhanced Feature Set:** A more broad and dynamic set of features, in addition to ratings and reviews, allows for a more complete insight of app behaviour. This upgrade can result in more accurate fraud detection by capturing a broader variety of attributes associated with both legal and counterfeit apps.
2. **Real-time Monitoring:** The suggested system's emphasis on real-time monitoring enables the detection of changing patterns and behaviours in mobile applications. This ensures that the model continues to adapt to changes in fraudulent techniques, giving a proactive approach to fraud detection rather than relying exclusively on static features.
3. **Improved Model Robustness:** The proposed system tries to improve the model's ability to grasp complicated correlations in data by experimenting with advanced machine learning techniques or deep learning methodologies. This can lead to a more robust and accurate fraud detection system, particularly in situations when fraudulent activity reveals sophisticated patterns.
4. **Addressing Data Imbalance:** The system's use of enhanced sampling techniques or ensemble learning methods helps to overcome the issue of data imbalance. This ensures that the model is trained on a more representative dataset, avoiding biases toward the majority class and enhancing its sensitivity to the minority class of fraudulent apps.
5. **User-Friendly Interface:** A user-friendly interface improves the system's overall efficacy by making it accessible and understandable to users. Clear insights into the model's predictions increase user trust and knowledge, encouraging collaboration between the system and users in flagging potentially fraudulent apps on the Google Play Store.

4. SYSTEM DESIGN

SYSTEM ARCHITECTURE

Below diagram depicts the whole system architecture.

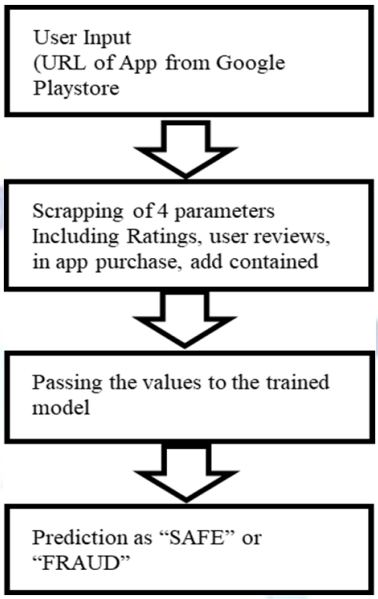

Fig 1. System Architecture

5. SYSTEM IMPLEMENTATION

MODULES

Data Collection and Preprocessing: This module collects pertinent data from the Google Play Store, such as app ratings, reviews, in-app purchase information, and ad presence. The acquired data goes through preparation to handle missing values, delete duplicates, and convert textual material into a format appropriate for analysis.

Feature Engineering and Selection: In this module, the system focuses on expanding the feature set by incorporating new dynamic elements that reflect the changing nature of mobile applications. Feature selection approaches can be used to discover the most relevant attributes for training the fraud detection model and improving its performance.

Machine Learning Models: This subject covers the implementation and training of machine learning models, such as Decision Tree, Logistic Regression, and Naïve Bayes. Hyperparameter tweaking and model evaluation approaches are used to assure peak performance. The investigation found the Decision Tree model as effective, which would be a major component.

Real-time Monitoring and Analysis: To address the dynamic nature of mobile apps, this module employs real-time monitoring of app behaviour, updates, and

user interactions. Continuous analysis guarantees that the model remains flexible to developing trends and is capable of quickly detecting fraudulent activities as they evolve over time.

User Interface and Reporting: The system has a user-friendly interface that allows users to interact with and interpret the model's predictions. This module enables users to enter app details or questions and return clear and understandable fraud risk assessments. Visualizations and short summaries help to present the results, increasing user knowledge and trust in the system.

6. EXPERIMENTAL RESULTS

The basic goal of the suggested approach is to investigate fraud detection in the Google Play store for apps and use four parameter strategies to discriminate between different fake apps, commonly referred to as spam apps. The recommended technique for detecting fraudulent or false applications includes performing experimental research using a number of methodologies. Our method will detect fraud by examining four sorts of data: ad-based ratings, in-app payments, and evidence-based reviews. Furthermore, the development-based integrated strategy employs all four criteria for detecting fraud. Several artificial intelligence (AI) models were used, each with varying levels of accuracy. Our investigation revealed that the one we recommended approach outperforms existing algorithms by 85%. While autonomous thinking survives, the Decision Tree element outperforms other models, including the financial crisis and Naive Bayes. It is an easy way to divide challenges. It is an accurate real-time guess, which comes with a drawback. Decision trees are effective for dealing with nonlinear data sets. It influences judgments in a variety of sectors, notably technology, social organizing, business, and even the law.

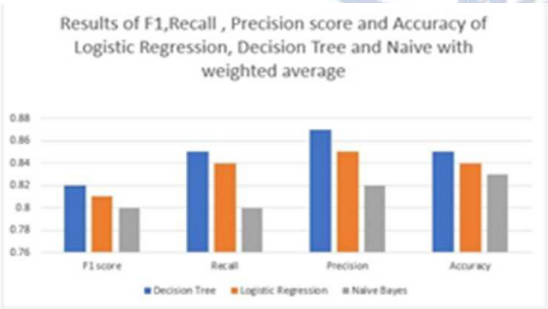

fig 2. Results Graph

The Results of F1, Recall, Precision and Accuracy Scores of Logistic Regression, Decision Tree and Naive with weighted average.

TABLE 2. COMPARATIVE ANALYSIS ON DIFFERENT MACHINE LEARNING ALGORITHMS FRAUD APP DETECTION

	F1	Recall	Precision	Accuracy
Decision Tree	0.815	0.85	0.87	0.85
Logistic Regression	0.81	0.84	0.85	0.84
Naïve Bayes	0.8	0.83	0.82	0.83

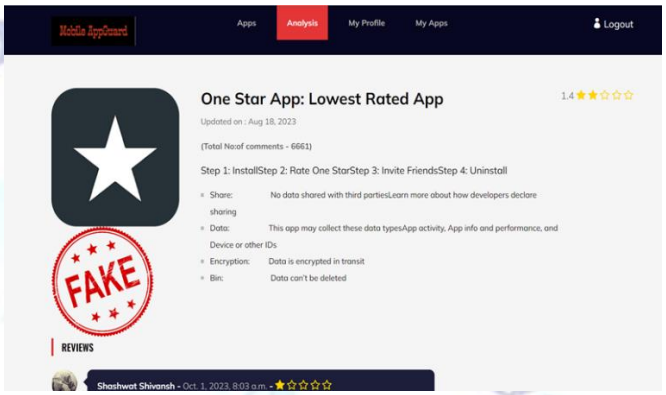

Fig 3. Negative Result

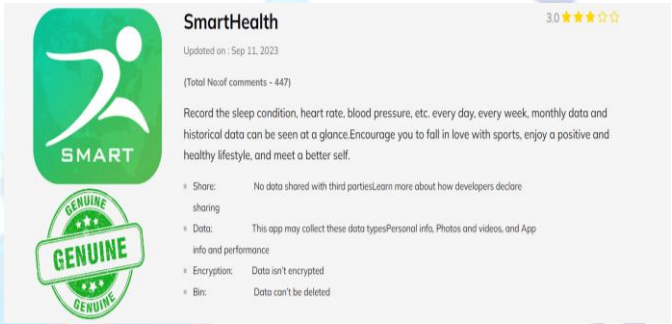

Fig 4. Positive Result

7. CONCLUSION AND FUTURE WORK

The danger to security has become a severe concern in the age of modern technology, as a result, here has become a large number of apps available on the Play store on Google, containing a variety of fraud applications that jeopardize user privacy and data. We detect counterfeit software based on four distinctive features: scales, review scores, app purchases, and the content additions. We assessed the accuracy of the three techniques when the resulting tree seemed to be 85% bigger. The approach is quantifiable and might be expanded to include more proven domain-based false evidence. The suggested system, algorithms detection measurements, and standardisation of level fraud operations have all been shown to be successful. This might be used to accurately analyse counterfeit Play Store apps on the Play Store.

Future Enhancement

In terms of application security, improving our fraud detection project yields good outcomes. Beyond decision trees, techniques such as random forest models and Deep Learning may improve accuracy. Deeper features engineering, such as employing NLP for app descriptions or assessing user interactions, may produce more exact results. Adaptive systems include real-time analysis, unambiguous AI judgments, and continuous learning. Combining user input, geographic information, and multi-platform recognition can broaden the reach. Regulatory inspection and engagement with experts promote ethical and effective development. You can increase app verification by knowing blockchain and applying smart contracts. As innovation evolves, the project's agility and creativity are going to be vital in addressing emerging fraud approaches.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] Esther Nowroji, Vanitha, "Detection Of Fraud Ranking For Mobile App Using IP Address Recognition Technique", vol. 4.

[2] JavvajiVenkataramaiah, BommavarapuSushen, Mano. R, Dr.GladispushpaRathi, "An enhanced mining leading session algorithm for fraud app detection in mobile applications"

[3] S.R.Srividhya, S.Sangeetha – "A Methodology to Detect Fraud Apps Using Sentiment Analysis"

[4] Keerthana. B, Sivashankari.K and ShaisthaTabasum.S, "Detecting Malwaresand Search Rank Fraud in Google Search Using Rabin Karp Algorithm", IJARSE,7(02), 2018, pp.504-527.

[5] Shashank Bajaj, Nikhil Nigam, PriyaVandana, Srishti Singh, "Detection of fraud apps using sentiment analysis", International Journal of Innovative Science and Research Technology.

[6] HarpreetKaur, VeenuMangat and Nidhi, — "A Survey of Sentiment Analysis techniques"

[7] International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2017, pp.921

[8] Jing Wan, Mufan Liu, Junkai Yi and Xuechao Zhang, "Detecting Spam Webpages through Topic andSemantics Analysis", IEEE Global Summit onComputer and Information Technology (GSCIT), 2015, pp. 83-92.

[9] Navdeep Singh, Prashant Kr. Pandey and Mr.Srinivasan, — "Improved Discovery of Rating Fake for Cellular Apps", IEEE International Conference on Science Technology Engineering and Management (ICONSTEM), 2016, pp. 135-140.

[10] Weiman Wang, Restricted Boltzmann Machine. GitHub. Aug 2017. [Online] Available: https://github.com/aaxwaz/Fraud-detection-usingdeep-learning/blob/master/rbm/rbm.py.

[11] DubeyVeena, G. D. (2016). Sentiment Analysis Based on Opinion Classification Techniques: A Survey .International Journal of Advanced Research in Computer Science and Software Engineering, 5358.

[12] Ranking fraud Mining personal context-aware preferences for mobile users. H. Zhu, E. Chen, K. Yu, H. Cao, H. Xiong, and J. Tian. In Data Mining (ICDM), 2012 IEEE 12th International Conference on, pages1212–1217, 2012.

[13] NandimathJyoti, K. B. (2017). Efficiently Detecting and Analyzing SpamReviews Using Live Data Feed. International Research Journal of Engineering and Technology (IRJET) , 1421-1424.

[14] Detecting product review spammers using rating behaviors. E.-P. Lim, V.-A.Nguyen, N. Jindal, B. Liu, and H. W. LauwIn Proceedings of the 19th ACM international conference on Information and knowledge management, CIKM "10 pages 939–948, 2013.

[15] Detection for mobile apps H. Zhu, H. Xiong, Y. Ge , and E. Chen. A holistic view. In Proceedings of the 22nd ACM international conference on Information and knowledge management, CIKM "13, 2013.