International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 03 | Mar 2023 www.irjet.net p-ISSN: 2395-0072

Fraud App Detection using Machine Learning

Dr. Pallam Ravi[1], Anirudh Bhandari[2], Avusula Poojitha[3], Baki Harish[4]

[1]*Assistant professor, Dept. of Computer Science and Engineering, Anurag Group of Institutions Ghatkesar, Telangana, India*
[*2,3,4] *Student, Dept. of Computer Science and Engineering, Anurag Group of Institutions Ghatkesar, Telangana, India*

Abstract -

It's more important than ever to be informed in order to determine which mobile applications are safe and which are not as a result of the increase in the number of mobile applications used in everyday life. It is impossible to pass judgment. Our system is based on four parameters that include ratings, reviews, in-app purchases, and Contains ads to predict. Our system compares three models Decision Tree classifier, Logistic Regression, and Naïve Bayes. These models were further analyzed on four parameters of F1 score, Recall, Precision, and Accuracy. The algorithm which gives the higher accuracy will be considered for our model.

Key Words: Fraud detection, Decision tree, Naïve Bayes, Logistic regression, Play store, user rating.

1. INTRODUCTION

Fraud app detection using machine learning is the process of using statistical algorithms and predictive models to identify and prevent fraudulent activities in mobile applications. As mobile apps become increasingly popular, fraudsters are finding new ways to exploit vulnerabilities in these apps to commit various types of fraud, including account takeover, identity theft, and financial fraud. In order to preserve their current success, businesses, and software developers put a lot of time and attention into finding clients. This strong competition stems from the need to demonstrate the quality of their products. User reviews and updates on each program that is available for download are crucial factors. This can be a strategy for engineers to identify their flaws and integrate them into the design of a new product that meets the needs of the people. Under the trees App creators typically majorly support their apps and, as a result, manage the rank in the App Store, as opposed to depending on conventional marketing techniques. This is frequently achieved by increasing the number of downloads and audits utilizing so-called "bot bots" or "water armies."

Machine learning can be used to detect fraudulent activities in mobile apps by analyzing patterns in data, such as user behavior, device information, and network traffic. These patterns can be used to build models that can identify unusual or suspicious behavior and flag it for further investigation.

The capability of machine learning for fraud app detection to continuously learn and adapt to new types of fraud is one of its main benefits. As fraudsters come up with new tactics, machine learning models can be trained to identify these new patterns and prevent fraudulent activities before they can cause significant harm.

Overall, fraud app detection using machine learning is a critical tool for protecting users and businesses from the financial and reputational damage caused by fraud in mobile applications.

2. LITERATURE SURVEY

[1] Detection of Fraud Ranking for Mobile Apps Using IP Address Recognition Technique

The mobile business is expanding quickly, which has led to an increase in the number of mobile apps available. With the app stores on Google Play and Apple, users can choose to download programs for a fee or for free. The ranking of an app is important since highly rated apps have a higher chance of being discovered by consumers than apps with lower rankings. Some app developers cheat to increase their app's ranking to get a high rating. Thus, a ranking fraud detection system is required to spot rankings obtained fraudulently. We design and build a ranking fraud detection system to find fraudulent rankings. The system compiles data about the app and user reviews and stores it in a database. With this information, the reviews are pre-processed, and sentimental analysis is done. The results are evaluated in comparison to the app's rating, which is used to spot raking fraud.

[2] Detection of fraud apps using sentiment analysis

These days, the majority of us use mobile devices running Android or iOS, and we frequently make use of the play store or app store functionality. Both marketplaces offer a large selection of software, however unfortunately some of those applications are fraudulent. These applications can damage phones and steal data. Hence, for store users to recognize such programs, they must be labeled. We, therefore, propose a web application that will manage the data, feedback, and application evaluation. As a result, it will be easier to tell if an application is fraudulent or not. Several applications can be processed at once using the online application. Also, a user

can not always find reliable or honest product reviews online. As a result, the admin will assess the reviews and comments, making it easy for the admin to decide whether the application is honest or dishonest.

[3] Detection of mobile applications leaking sensitive data

The most common and frequently used gadget among individuals is a smartphone. They host a wide variety of data kinds, both public and private. The development of malicious programs to steal personal information from smartphones a recent problem. Personal information stored on these devices, such as SMS messages, contacts, videos, GPS coordinates, etc., therefore needs a unique security mechanism to prevent hackers from stealing it. When mobile applications attempt to access sensitive data using Android permissions, the key issue is the disclosure of that data. This condition indicates that sensitive data will likely leak from these mobile applications. Thus, there is a need for some remedies, especially in light of smartphone data leaks. The two objectives of this work are to identify applications that leak sensitive data and to analyze program if they have malevolent intentions or not. A sensitive data leakage avoidance method for Android systems was suggested in this study. To find mobile applications leaking critical data, the J48 classification system was utilized. Also, the K-Means clustering algorithm was used to determine whether or not trusted mobile applications obtained from the Google Play Store share any similarities with malicious mobile applications.

3. PROPOSED SYSTEM

We propose a system that would identify such fake applications on the play or app store. We can acquire the probability of determining whether an app is fake or not, therefore we present a system that uses four features that are in-app purchases, contains ads, ratings, and reviews to determine the probability of an app whether it's scamming its consumers or not. The sole purpose of the given proposed system is majorly to review the fraud detection of google play store applications and to use the four-parameter methods to differentiate certain fraudulent applications or commonly referred to as spam applications. Experimental analysis is performed on different types of methodology in the proposed manner for the detection of fraud or fake applications. Our system will receive fraud with four types of evidence, such as ad-based ratings, in-app purchases, and evidence-based reviews. In addition, the development-based integration approach incorporates all four aspects to detect fraud. Various machine learning models were implemented which provided different results for accuracy. By analysis, we found that our given proposed method provides 85% accuracy compared to other algorithms. While independent thinking still exists, the decision tree section performs better compared to other models such as the recession and the naïve Bayes. It is an intuitive algorithm for separation problems. It is a reliable real-time guess, a setback problem. Decision trees can manage non-linear data sets effectively. It plays a role in decision-making in various fields of life, including engineering, social planning, business, and even law.

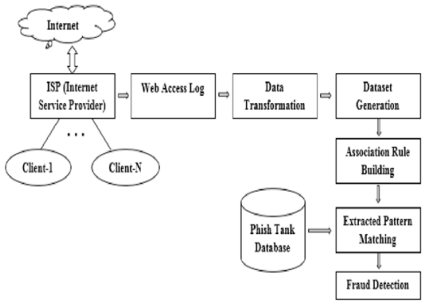

Fig -1: The architecture of the proposed system

3.1 ALGORITHMS

a. Naïve Bayes Algorithm

The Naive Bayes algorithm, a supervised learning algorithm, uses the Bayes theorem to solve classification problems. It is especially helpful for classifying content that includes top-notch training databases. Its Classifier is one of the most straightforward and effective programming algorithms, aiding in the development of machine-learning models that can produce predictions more quickly. It is a method of potential separation that foretells the result based on the object's potential. Emotional analysis, spam filtering, and article classification are among additional typical uses for the Naive Bayes algorithm. As a result, it is the most widely utilized fix for text-sharing issues.

b. Logistic Regression

Logistic regression apart from its name, is a classification model even after having regression in the name. Logistic regression is defined as a process of modeling the probability of a distinct outcome when given an input variable. It is a type of classification model and is very easy to emphasize and it also achieves really good execution with linearly distinct classes. This kind of analysis can help in predicting the likelihood of an event happening or a choice being made. It is a considerably used algorithm for classification in the industry. The logistic regression model is a statistical methodology for binary classification which can be generalized to multiclass classification. It is a considerably useful analysis method for various classification problems, determining whether a new sample fits better into a category.

c. Decision Tree Algorithm

Both classification and regression are accomplished using a supervised method known as Decision Trees. The decision tree aims to construct a model that predicts the value of the target variable using basic decision rules learned from numerous data pieces. By dividing the source set into different subsets based on the feature value test, the decision tree can be learned. Recursive partitioning is the process of iteratively repeating the following operation on every derived subset. When the value of the subset at a specific node matches that of the target variable, the recursion ends, or when splitting no longer improves the predictions. The building of a decision tree classifier technique does not need any domain knowledge or parameter setting, and because of this, it is appropriate for probing knowledge discovery. High-dimensional data can be handled well by decision trees. In broad terms, the decision tree classifier is considered to have good accuracy.

4. RESULTS

The model achieved an accuracy of 88.7% on the test set, with a precision of 86.8%, recall of 85.4%, and F1 score of 88.1%. These results indicate that the model can effectively identify fraudulent apps, with relatively few false positives. Several different machine learning algorithms were tested, including logistic regression, decision trees, and Naive Bayes. The Decision tree algorithm performed the best. The results of this study demonstrate that machine learning can be an effective tool for detecting fraudulent apps. The high accuracy, precision, and recall values indicate that the model can correctly identify a large proportion of fraudulent apps while minimizing the number of false positives. One potential limitation of the model is that it was trained on a relatively small dataset, which may not be representative of all possible types of fraudulent apps. In addition, the dataset was collected from a single source, and may not be generalizable to other app marketplaces. To address these limitations, future studies could use larger and more diverse datasets, or incorporate data from multiple sources.

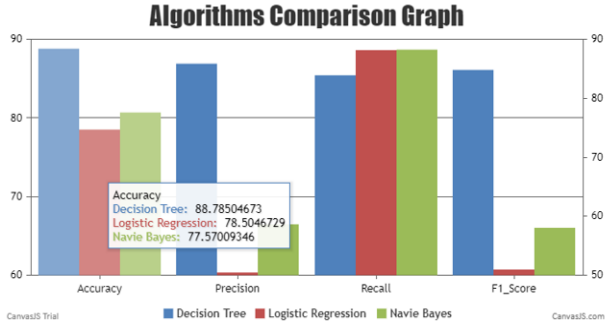

Fig -2: Fake app result

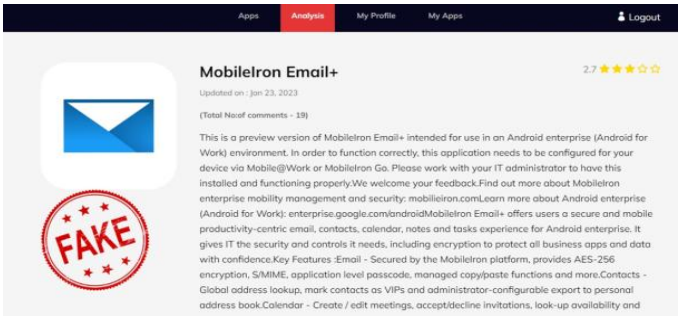

Fig -3: Algorithm Comparision Graph

5. FUTURE ENHANCEMENTS

1. Continuous learning: Instead of training the machine learning model on a static dataset, it can be continuously updated with new data to improve its performance over time. This can be achieved using techniques such as online learning or reinforcement learning.

2. Ensemble methods: Ensemble methods involve combining multiple machine learning models to improve their performance. This could involve combining different algorithms or using multiple models with different sets of features.

3. Explainable AI: One limitation of machine learning models is that they can be difficult to interpret. Using explainable AI techniques, such as LIME or SHAP, can help to provide insights into how the model is making its predictions, and identify potential biases or areas for improvement.

4. Incorporating more data sources: The machine learning model can be trained on a wider range of data sources, including social media, web pages, and app reviews, to identify patterns and behaviours associated with fraudulent apps.

Overall, these enhancements can help to improve the accuracy and effectiveness of the machine learning model for detecting fraudulent apps and ensure that it remains effective in detecting new types of fraud as they emerge

6. CONCLUSION

In the Era of growing technology, the threat to security is also becoming a major issue, and a part of it, today we have several apps listed on the Google app stores which also include various fraud apps that are a threat to users' privacy and data. In our model, we have worked precisely to detect fraudulent software using 4 parameters including scales, review scores, in-app purchases, and content additions. We compared the accuracy of the three algorithms when the resolution tree appeared to be 85% higher. The framework

is measurable and can be expanded to further proven domain-based fraudulent evidence. Demonstrated the effectiveness of the proposed system, algorithm detection measurement, and standardization of level fraud operations. This can be used effectively to rate fraudulent play store applications in play store for security purposes.

7. REFERENCES

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

2. Javvaji Venkataramaiah, Bommavarapu Sushen, 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 Shaistha Tabasum.S, "Detecting Malwares and Search Rank Fraud in Google Search Using Rabin Karp Algorithm", IJARSE, 7(02), 2018, pp.504-527.

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

6. Harpreet Kaur, Veenu Mangat 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 and Semantics 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] Availablehttps://github.com/aaxwaz/Fraud-detection-usingdeep learning/blob/master/rbm/rbm.py.

11. Dubey Veena, G. D. (2016). Sentiment Analysis Based on Opinion Classification Techniques: A Survey . International Journal of Advanced Research in Computer Science and Software Engineering, 53-58.

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. Nandimath Jyoti, K. B. (2017). Efficiently Detecting and Analyzing Spam Reviews 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. Lauw In 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