Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Machine Learning Powered Detection and Blocking of Fraud Mobile Applications

Billa Tanuja, Cherukumalli Radhika, Dr. V. B. Ganapathy, Dr. P. Dhivya
Department of Computer Science and Engineering
Dr. M.G.R. Educational and Research Institute, Chennai 600095, India

Abstract - **The extensive application of mobile applications has bred novel opportunities of cyber fraud. There is a growing use of fraud apps that exploit trust and vulnerabilities of systems. Such applications tend to replicate trusted applications but they end up executing malicious activities. Such activities involve unauthorized permission usage, loss of sensitive information and fraudulent financial transactions. The existing detecting techniques, relying on the fixed rules or statical signatures, struggle to cope with the rapidly evolving and emerging threats. This paper introduces an early detection and prevention system of fraudulent mobile applications through a machine learning based platform. We integrate features of application permission, application execution behavior, and network interaction behavior to form a single perspective of application behavior. We also test and train some of the supervised learning models on a set of data with both benign and fraud applications. Experiments with our framework demonstrate that it has a good detection performance and low false alarm rate. In addition, sensitive actions and automatic blockage of apps can also be introduced, which can prevent the execution of applications upon recognition using an integrated blocking mechanism. The findings indicate that our system provides a good solution to the enhancement of the security of mobile applications, and it can be applied practically in the mobile ecosystems.**

Keywords - **Fraudulent Mobile Applications, Machine Learning, Mobile Security, Permission Analysis, Behavioral Mon- itoring, Malware Detection, Application Blocking**

- A framework for identifying fraudulent mobile applications based on machine learning.
- For precise fraud detection, permission analysis, behavioral monitoring, and network activity profiling are integrated.
- Using a Random Forest classifier to decrease false positives and increase detection accuracy.
- An automated blocking system that instantly stops fraud- ulent applications from being executed.

I. INTRODUCTION

The swift development of mobile computing platforms has transformed smartphone into communication, electronic payment, medical and business application tools. Meanwhile, mobile application ecosystem has seen a great increase in both official and third-party markets. Although this has made the users find things easier, it has also posed a great threat of security threats in the form of fraudulent mobile applications.

These programs tend to look like authoritative software programs with dangerous capabilities concealed be- neath them to seek to exploit user trust and vulnerabilities in the system. Fraud applications are usually characterized by mislead- ing and hazardous practices. These are over use of permissions, access to sensitive data without authorization, covert operations of financial transactions, and round-the-clock communication with servers of untrustworthy nature. A large number of users enable them without careful consideration of the dangers that enable fraudulent appli- cations to operate. The repercussion of such activities may include loss of money, identity theft, breaches of privacy and unauthorized access to vital system resources. Mobile fraud has continued to increase in numbers as mobile devices become a repository of personal and monetary data. Conventional mobile application security is mainly based on signature method and rule method of detection. These technology methods rely on pattern or known malware signature to identify malicious activity. Although they are useful against threat signature, they have difficulty in detecting new fraudulent applications which are based on obfuscation, dynamic payload delivery, or behavior evasion techniques. In addition, the techniques of static analysis, which evaluate application code and requested permissions without execution of the program, are not sufficient to capture runtime behavior indicative of fraud inspiration. Machine learning has come out as an encouraging solution to such issues since it is capable of analyzing high order patterns and learning on large data sets. Machine learn- ing models can distinguish more between legitimate and fraudulent applications by examining the properties of the applications, including permission requests, runtime behavioural properties, and network communication behaviour. This is opposed to traditional methods because learning- based systems can make generalization on new samples as well as adapt to changing threat environment. Nevertheless, most of the current ma- chine learning solutions are concerned only with detection, but they lack an effective mechanism of automated prevention after detecting a fraudulent application. In order to address these issues, this paper

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

proposes a machine learning-based system that will identify and prevent fraud- ulent mobile applications. This system incorporates permission analysis, behavioral monitoring, and a network activity profiler to come up with a clear image of the application behavior. Learned supervised models are trained on labeled data sets comprising of valid and fraudulent applications. Once a mobile app is classified as a fraudulent one, there is an automatic blockage mechanism in place. This stops execution and restricts access to sensitive resources of the system, minimizing the chances of harm.

The primary contributions of this work are pointed out

below:

- Design of a machine learning system to identify and prevent fraud mobile apps.
- Multi-dimensional features that reveal per- mission usage, behavioral pattern and network activity.
- Comparison of various models of supervised learning to determine optimal performance of detection.
- Real- time prevention of fraudulent application execution through implementation of automated response system.
- Creation of a system that can develop and learn about new fraud trends to enhance its detection accuracy with time.

II. LITERATURE SURVEY

Initially, the research about suspect apps and frauds was predominantly based on data sorting and visualization. It was important to notice unusual clusters in numbers, as noted by the team of Keim before [1] who could notice how images of data could be used to identify suspicious action. When specialists mapped large volumes of information, patterns that were salient emerged. There the crew of Misarwala excavated [2] further with number- crunching machinery and demonstrated the way to group and label the data points and get a clue to the secrets hiding below. The invisible connections usually emerged when the methods were used to sort by similarity or type. The development of mobile apps compelled researchers to investigate app store fraud. In contrast to passive observation of traffic, Nowroji and Vanitha

[3] developed a scoring mechanism of shady apps with the use of IP addresses. This method was based on the detection of unusual network behavior of fake programs that masqueraded as legitimate programs. Although it worked well with catching strange data flows, it encountered issues with apps that switch IPs frequently or conceal the information with en- crypting. The subsequent work was on pre-running inspection of the app code, in order to increase the reliability of finding the fraud. Firdaus and team [4] employed a genetic search and statical analysis in selecting features helpful in detection of Android malware. By identifying the most suitable combi- nations of features, their method improved the effectiveness of classification of the apps but retained a limited number of features. Nevertheless, the techniques based solely on the static checks did not perform so well with scrambled code or concealed operations that will be executed later. this impaired the performance against sophisticated counterfeit applications. Newer strategies emerged in the face of such issues with richer data mining on top of smarter learning models. The detection of fraudulent apps received a boost by a group headed by Venkataramaiah [5] who devised a sophisticated mining-based session process, basing on user behavior hints and abnormal sequence of activities. With that said, SVM models allowed Avayaprathambiha and colleagues to identify fake rankings in phone applications. These techniques were more effective, but they were based on highly selected features and clean training datasets [6] . Innovation has also been geared towards detection of huge volumes of malware by designing intelligent algorithms. In one instance, the group of Yerima became Android threats users of various machine learning systems that were used simultaneously [7] . Findings indicated the quicker identification with the concurrent and grouped methods without loss of precision when the two were combined. Beginning with the work of Grover [8] attempts to identify modified search rankings were made by the use of specially-designed tools that aimed to make results truthful - not the common conducts of virus. The approach did not only involve searching the malicious code, but rather commenced searching the behavior of the apps online. Subsequently were cloud computing and digital forensic powered systems which gave strength to these methods of detection. A case in point is the concept of Patil and colleagues [9] : an arrangement with cloud support to mine Android applications, using pattern- finding techniques to mark suspicious ones. Although off- site processing was faster in number-crunching, it led to the concern of delays, and to whom the private information is available. Recently, the researchers started attracting the user created content and extended the net further than the programs do. Based on the reviews, Puram and Singh [10] extracted the meaning with the support of fuzzy logic, demonstrating how the words said by the users can be a clue to fraud. Pingale et al. [11] combined multiple detection techniques to create something more resistant to deception instead of applying a single approach. It is no less important to test these systems. scholars excavated the means of quantifying the effectiveness with which they do so. Azzopardi et al. [12] carefully observed the aspect of precision and recall and found out why such aspects are important when the real-world fraud cases are infrequent yet essential. Although a lot are pursues of an increased accuracy, the trend among the researches is biased towards such tools as machine learning and data mining to sharpen the detection. Nevertheless, the majority of solutions follow the pattern of threat detection without including auto-reply and stop options. The difference is its association of intelligent pattern recognition with immediate blockage of apps - the identification of suspicious apps followed by their closure with a single cohesive barrier.

III. PROPOSED METHODOLOGY

The research methodology that is proposed will help to

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

identify and prohibit fraudulent mobile apps through machine learning technologies. It seeks to uphold efficiency, scalability and practical implementation in mobile security scenarios. This system adheres to a pipeline based on the analysis of application data, feature engineering, supervised learning, and auto- mated response mechanisms. Each step is streamlined to make it as efficient as possible in terms of detection, false positives and responses to fraud. The general process starts by gathering characteristics at the application-level when it is installed and running. These properties are processed and converted into features which represent application behavior. The features obtained are further processed through trained machine learning models to identify applications as benign and fraudulent. When fraud has been identified, it is blocked by an automatic system that is established to prevent additional ex- ecution and only allow sensitive system resources access. The system takes into account various application characteristics categories that form a whole behavioral pro- file:

- Permission usage patterns indicate that sensitive device resources have been accessed.
- Behavioral indicators are based on the execution activities of the application.
- There are network communication attributes indicate sus- picious ex- ternal interactions.
- Resource consumption characteristics Resource con- sumption characteristics are abnormal CPU or memory usage.
- A combination of these various characteristics makes the proposed frame-work useful to distinguish between fraud applications and legitimate software even as fraud tactics evolve.

A. Dataset

The training and evaluation data is la- beled records of mobile applications, which are benign and malicious applica- tions. The gathered evidence will include details on permission requests, the statistics of behavioral patterns observed during execution, and attributes of the network on the basis of application communication patterns. Actual usage situation is used as a source of data samples to capture the reality of the operating conditions and realistic fraudulent behavior. In order to enhance the generalization capabilities of the model, the set of applications demonstrating various fraud patterns and other legitimate usage patterns are added. The data collected is preprocessed with the steps of noise reduction, missing or inconsistent values, and normalization of feature distributions before training. These measures have the effect of making sure that there is uniform scaling of features, which leads to increased stability and accuracy of the learning process.

B. Feature Extraction

The role of feature extraction is to convert raw information in applications to useful numerical representations that can be used in machine learning. Characteristics that are discovered demonstrate much of the behavior of applications, including excessive attempts to get permission, suspicious background operations, abnormal resource use, and suspicious network communica- tion patterns. The pre- processing methods such as feature scaling and normalization are applied to provide consistency in different applications. The feature selection is used to retain the most significant attributes and reduce the redundancy and complexity. The resulting feature vectors pro- vide a small and concise representation of application behavior that makes the classification more efficient and accurate when training and in real-time applications.

C. Machine Learning Model

The model of fraud detection is a supervised learning model that is trained offline with labeled application data. We take into consideration various lightweight classification algorithms that determine the equilibrium between performance and com- putational efficiency in detection. During training, the model is trained on patterns which assist in the classification process to differentiate between fraud and benign applications with minimum classification errors. During real-time operation, the trained model takes the input application feature vectors and estimates the probability of the fraud. The model is designed to compromise accuracy, the speed of inference, and the consumption of resources and hence appropriate in mobile security settings. The result of classification forms the basis of the decision-making and response.

D. Automated Fraud Blocking Mechanism

The proposed framework combines an automated protection mechanism by fraud blocking along with the detection module to offer proactive protection. Whenever the system identifies an application as fraudulent it takes corrective measures to ensure that it does not cause additional damage. Such measures are denying installation/execution of the application, denying privileged permissions, and limiting access to vital system resources. Moreover, alerts are also created to inform users or system administrators that there were fraudulent cases. This combined reaction system will guarantee that the threats are mitigated in time and prevent possible losses due to fraudulent applications. Using detection and feature automatic response, the proposed system enhances the level of mobile security and reduces the use of human intervention.

IV. SYSTEM ARCHITECTURE

Fig. 1 illustrates the system architecture of the suggested mobile fraud application detection and prevention framework. It is structured as a layer and modular framework to enable efficient fraud detection, prompt response and uninhibited integration in mobile security settings. The system comprises of four logical layers which are: the Application Data Source Layer, User Interface Layer, Mobile Security Processing Layer and the System Control and Response Layer. The Application Data Source Layer has mobile applications installed and or

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

requested by the users. This layer will offer raw application data, such as requested permissions, application metadata, behavioral logs, and network activity data when installing and running an application. Such sources of data are crucial in monitoring patterns of unusual or fraudulent behavior. The interface with the users or the administrators is done at the User Interface Layer. It shows warning messages, security messages, fraud detection messages, and application status messages. It provides an interface where the user can get warn- ings in case of suspicious applications, check the information of blocked applications and manage the basic security settings. This layer encourages transparency and creates awareness regarding mobile security threats to the user. The central processing is carried out in Mobile Security Processing and System Control Layer where the fraud detection and response are closely interconnected. This single layer is able to provide feature preprocessing, classification, decision-making and au- tomated responses. Raw application data is first processed by

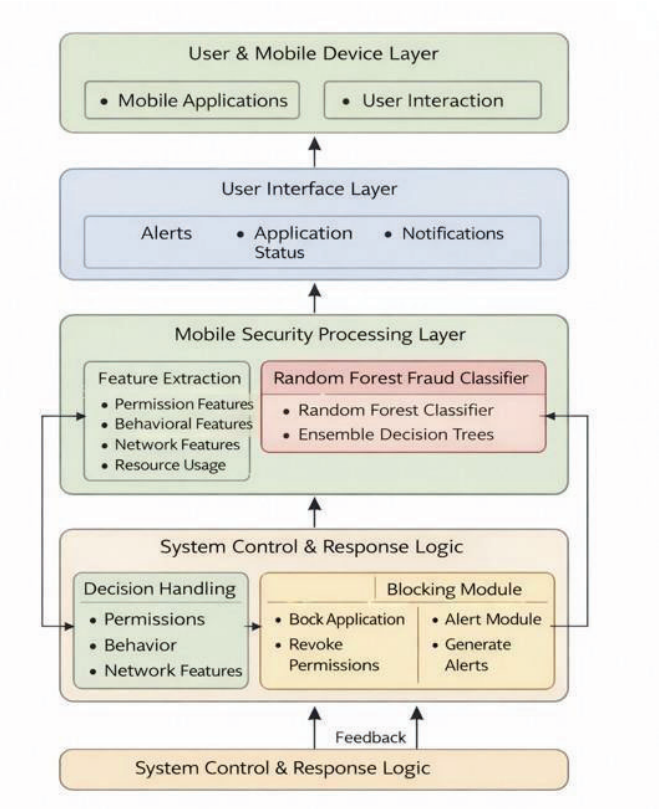

Fig. 1. System Architecture of the Machine Learning Powered Fraud Mobile Application Detection and Blocking Framework

the system to derive significant features such as frequency of permission usage, abnormal execution patterns, network com- munication characteristics and resource consumption trends. The primary machine learning model used in the detection of a fraud is a Random Forest classifier. The offline training of the classifier is done using labeled benign and fraudu- lent applications datasets. Random Forest is selected, since it is a strong algorithm, can handle high dimensional feature space; overfitting does not occur and its ability to operate in real- time and in mobile devices supports its application in these applications. In operation, the system transmits extractions of feature vectors to the model that has been trained, and that classifies an individual application as benign or a fraud. When an application is detected to be a fraud, the System Control and Response Logic initiates protection measures. Such measures can be blocking the execution or installation of applications, withdrawing sensitive permissions, and creating the alert to alert a user or system administrator. Such automatic response decreases the possibility of data leakage, loss of money and compromise of the system. Lastly, the identified findings and system executions are reported into the User Interface Layer, which creates a closed-loop security system. It is an integrated configuration that provides sustained monitoring, rapid fraud detection and preventive threats, and maintains low computational cost and scalability to real-world mobile systems.

V. PROPOSED ALGORITHM

The suggested algorithm is a map of the operation of the fraud mobile application detection and blocking system. It outlines how to gather application data, initial processing, feature creation, machine learning model classification, and an automated response. The algorithm is designed to operate effectively in mobile security settings and guarantee high accuracy of detection and low computational requirements.

Algorithm 1 Fraud Mobile Application Detection and Block- ing Algorithm

```
Collect mobile application data
Apply preprocessing and feature normalization
Extract permission, behavioral, and network features
Classify application using trained Random Forest model
Determine whether the application is benign or fraud
if application is fraud then
        Block application execution or installation
        Revoke sensitive permissions
        Generate alert to user or administrator
end if
```

The computational complexity of the algorithm mainly depends on feature extraction and model inference. Feature extraction increases linearly with the number of application attributes. Random Forest inference depends on the number of decision trees and their depth. Since model training is performed offline, the online detection and blocking process remains lightweight and suitable for real-time deployment.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

The experimental evaluation of the proposed fraud mobile application detection system used a dataset made up of both

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

benign and fraudulent applications collected from real-world environments. This dataset includes applications with differ- ent permission usage patterns, behavioral traits, and network activities. To ensure a fair comparison, the proposed system and existing detection methods were evaluated under the same conditions.

A. Quantitative Evaluation

Quantitative evaluation was done using standard performance metrics, including detection accuracy, precision, recall, F1-score, false positive rate, and average detection latency. Such measures are classification effectiveness and com- puta- tional efficiency. Table I compares the current system and the proposed detection framework based on the Random Forest. The findings indicate that the proposed system is much more accurate in detecting and more reliable in the classification of objects than the current systems. The results of the higher accuracy and recall are a positive sign that the number of false alarms was minimized without deteriorating the level of fraud detection. Also, the decreasing latency of detection indicates that the presented solution can be applied to the implementation of mobile security in real time.

TABLE I
QUANTITATIVE PERFORMANCE COMPARISON

Metric	Existing System	Proposed System
Detection Accuracy	89.4%	96.8%
Precision	0.88	0.96
Recall	0.87	0.95
F1-Score	0.87	0.95
False Positive Rate	Moderate	Low
Average Detection Latency	92 ms	45 ms

B. Qualitative Evaluation

Besides quantitative analysis, we also had a qualitative evaluation to determine the robustness of the system, its adaptability as well as its practical effectiveness. The current systems are predominantly static permission checks or rule based systems. They are not always able to identify the emerging fraud schemes and are ineffective in the inconsistent conditions of execution. The proposed system demonstrates that it is more robust in the sense that it utilizes the multi- dimensional feature analysis, which comprises permission usage, run-time behavior, resource consumption, and network communication pattern. This enables the system to detect fraud applications that conceal evil use as a usual behavior. The existing strategies tend to make inconsistent determinations in cases where applications demand borderline authorizations or present temporary behavioral transformations. On the con- trary, the proposed Random Forest classifier provides stable classification through combination of several decision trees, making the use of classifier stable and interpretable. The proposed system is more convenient in terms of user protection since the automated blocking and alert generation functions enhance user protection even without human intervention. Current systems tend to be based on user confirmation or offline scanning which may slack down the response and also expose the system to more exposure. fraud. In general, it can be concluded that the qualitative findings demonstrate that the proposed system is more adaptable, reliable, and practical in use in comparison to current solutions. It is applicable in the mobile security systems because it has low response times, high precision, and automatic reaction to detection.

C. Discussion

The numerical and descriptive findings are combined to clearly demonstrate the advantages of the suggested framework of detecting fraud. The system counteracts the flaws of the previous traditional, static and rule- based methods by applying machine learning and detailed feature analysis. The findings affirm that the suggested system can be effective in offering correct, effective, and scalable mobile application-based fraud detection.

VII. CONCLUSION

In this paper, a machine learning system that can identify and block fraudulent mobile apps soda will be presented. The system addresses the growing risk of fraud through the behavior of applications and training to identify behavior through

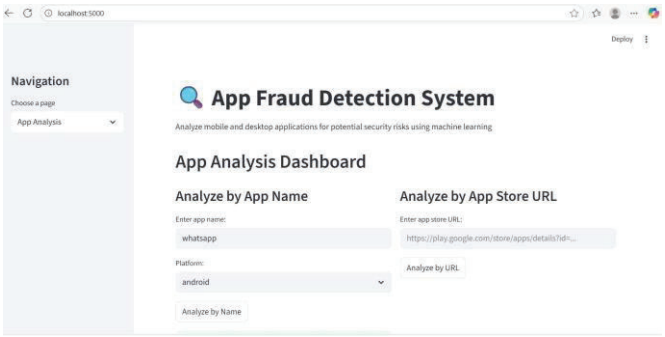

Fig. 2. Web-Based Interface for Monitoring and Blocking Fraud Mobile Applications

controlled classification. The framework combines intelligent detection with automatic blocking system to overcome the limitations of the conventional signature-based systems and enables fraud activities to be detected at an early stage. Experimental findings indicate that the methodology attains high detection accuracy at low detection latency rendering it to be appropriate in real-time mobile security applications. The addition of an online monitoring interface enhances ease of use and openness that enables the user and administrator to view detection outcomes and system activities. In general, the given solution offers an efficient and convenient approach towards the improvement of the mobile application security in contemporary mobile settings.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

AUTHOR CONTRIBUTIONS

- **Billa Tanuja** helped with the manuscript preparation, machine learning model implementation, and system design.
- **Cherukumalli Radhika** helped with feature extraction, dataset preparation, and experimental assessment.
- **Dr. V. B. Ganapathy** oversaw the study, verified the methodology, and offered technical advice.
- **Dr. P. Dhivya** helped to improve the research method- ology, reviewed the manuscript, and contributed to the design of the system architecture.

REFERENCES

[1] D. A. Keim, "Information visualizing and visual data mining," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, pp. 1–8, Jan.–Mar. 2002.
[2] F. Misarwala, K. Mukadam, and K. Bhowmick, "Applications of data mining in fraud detection," *International Journal of Computer Applications*, vol. 32, 2015.
[3] E. Nowroji and Vanitha, "Detection of fraud ranking for mobile app using IP address recognition technique," *International Journal for Re- search in Applied Science & Engineering Technology*, vol. 4, 2016.
[4] A. Firdaus, N. B. Anuar, A. Karim, and M. F. A. Razak, "Discovering optimal features using static analysis and a genetic search based method for Android malware detection," *Frontiers of Information Technology & Electronic Engineering*, 2018.
[5] J. Venkataramaiah, B. Sushen, M. R., and G. P. Rathi, "An enhanced mining leading session algorithm for fraud app detection in mobile applications," *International Journal of Scientific Research in Engineering*, Apr. 2017.
[6] A. P. Prathambiha, M. Bharathi, B. Sathiyavani, and S. Jayaraj, "To detect fraud ranking for mobile apps using SVM classification," *Inter- national Journal on Recent and Innovation Trends in Computing and Communication*, vol. 6, Feb. 2018.
[7] S. Y. Yerima, S. Sezer, and I. Muttik, "Android malware detection using parallel machine learning classifiers," in *Proc. 8th Int. Conf. on Next Generation Mobile Applications, Services and Technologies*, Sept. 2014.
[8] S. Grover, "Malware detection: Developing a system engineered fair play for enhancing the efficacy of stemming search rank fraud," *International Journal of Technical Innovation in Modern Engineering & Science*, vol. 4, Oct. 2018.
[9] P. Rohini, P. Kale, P. Jathade, K. Kudale, and P. 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, Oct. 2015.
[10] N. M. Puram and K. R. Singh, "Semantic analysis of app review for fraud detection using fuzzy logic," *International Journal of Computer & Mathematical Sciences*, vol. 7, Jan. 2018.
[11] V. Pingale, L. Kuhile, P. Phapale, P. Sapkal, and S. Jaiswal, "Fraud detection and prevention of mobile apps using optimal aggregation method," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 8, Mar. 2016.
[12] L. Azzopardi, M. Girolami, and K. van Rijsbergen, "Investigating the relationship between language model perplexity and IR precision–recall measures," in *Proc. 26th Int. Conf. on Research and Development in Information Retrieval*, 2003.
[13] Y. Li, J. Jang, and X. Hu, "Android malware detection using permission-based analysis," *IEEE Access*, 2019.
[14] M. Zhang, Y. Duan, and H. Yin, "Behavioral analysis for Android fraud detection," *IEEE Transactions on Information Forensics and Security*, 2020.
[15] D. Arp *et al.*, "DREBIN: Effective and explainable detection of Android malware," in *Proc. NDSS*, 2014.
[16] M. Grace *et al.*, "RiskRanker: Scalable and accurate zero-day Android malware detection," in *Proc. MobiSys*, 2012.
[17] P. Wang, L. Zhang, and W. Liu, "Ensemble learning for Android fraud detection," *IEEE Transactions on Information Forensics and Security*, 2021.
[18] S. Hou, Y. Ye, and M. Song, "Robust Android malware detection using deep learning," *IEEE Access*, 2020.
[19] J. Feng *et al.*, "Android malware detection using hybrid machine learning techniques," *Journal of Network and Computer Applications*, 2019.
[20] K. Tam, S. J. Khan, A. Fattori, and L. Cavallaro, "CopperDroid: Automatic reconstruction of malware behaviors," in *Proc. NDSS*, 2015.
[21] M. Lindorfer *et al.*, "ANDRUBIS: Large-scale analysis of Android applications," in *Proc. Virus Bulletin*, 2011.
[22] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. KDD*, 2016.
[23] J. Ma *et al.*, "Learning to detect malicious URLs," *ACM Transactions on Intelligent Systems and Technology*, 2011.
[24] C. Yang, Z. Xu, and G. Gu, "DroidMiner: Automated mining and characterization of malicious Android apps," in *Proc. ESORICS*, 2014.
[25] A. Shabtai *et al.*, "Andromaly: Behavioral malware detection framework," *Journal of Intelligent Information Systems*, 2012.
[26] W. Enck *et al.*, "A study of Android application security," in *Proc. USENIX Security Symposium*, 2011.
[27] J. Sahs and L. Khan, "A machine learning approach to Android malware detection," in *Proc. IEEE ISI*, 2012.
[28] Z. Fang, W. Han, and Y. Li, "Permission-based Android security: Issues and countermeasures," *Computers & Security*, 2014.
[29] H. Peng *et al.*, "Using probabilistic generative models for Android malware detection," *IEEE Transactions on Dependable and Secure Computing*, 2018.
[30] S. Jerome *et al.*, "Mobile fraud detection using classification techniques," *International Journal of Computer Networks*, 2017.
[31] R. Sato and K. Suzuki, "Behavioral profiling of Android applications," *IEEE Consumer Electronics Magazine*, 2019.
[32] P. Vinod *et al.*, "Survey on malware detection techniques," *ACM Computing Surveys*, 2019.
[33] N. Milosevic, A. Dehghantanha, and K. R. Choo, "Machine learning aided Android malware classification," *Future Generation Computer Systems*, 2017.
[34] C. Kolbitsch *et al.*, "Effective and efficient malware detection at scale," in *Proc. ACM CCS*, 2012.
[35] A. Feizollah *et al.*, "An appraisal of Android malware detection techniques," *IEEE Communications Surveys & Tutorials*, 2017.
[36] M. Gupta and R. Sinha, "Fraud detection in mobile ecosystems," *Journal of Cyber Security*, 2018.
[37] V. Choudhary and S. Gupta, "Android security using machine learning," *International Journal of Information Security*, 2020.
[38] M. A. Amin *et al.*, "Behavioral analysis for detecting malicious Android apps," *Journal of Information Security*, 2019.
[39] P. Singh and K. Kaur, "Survey of Android malware detection techniques," *International Journal of Computer Applications*, 2018.
[40] A. Sharma and R. Kumar, "A comprehensive study on mobile application fraud detection," *IEEE Access*, 2021.