Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 5, No.4, May : 2024

ENHANCED ANDROID MALWARE DETECTION USING GENETIC ALGORITHM BASED OPTIMIZED FEATURE SELECTION

Dr.Ch. Rathna Jyothi (Faculty Guide) Associate Professor, Computer Science and Engineering Department Andhra Loyola Institute of Engineering and Technology Vijayawada, Andhra Pradesh, India chrjyothi@aliet.ac.in
G.C.V.N Saileela Computer Science and Engineering Andhra Loyola Institute of Engineering and Technology Vijayawada, Andhra Pradesh, India gokavarapsai@gmail.com
K. Sai Tanuja Computer Science and Engineering Andhra Loyola Institute of engineering and Technology Vijayawada, Andhra Pradesh, India saitanuja2004@gmail.com
Buelah Evangiline Computer Science and Engineering Andhra Loyola Institute of Engineering and Technology Vijayawada, Andhra Pradesh, India buelahevangiline64@gmail.com

ABSTRACT
The ubiquity of Android devices has led to an exponential increase in malware attacks, which compromise user security and privacy. Traditional signature-based detection methods fall short against new, sophisticated malware variants, especially zero-day threats. This research presents a machine-learning-based approach that integrates Genetic Algorithms (GA) to optimize feature selection for malware detection. By analysing the 'AndroidDataset.csv' with more than 3,799 Android application records, the study employs Support Vector Machine (SVM) and Neural Network (NN) classifiers to establish a detection framework. The application of GA results in a feature reduction, selecting 40 out of a larger set, thereby decreasing model complexity and training time without substantially compromising accuracy. SVM demonstrated a notable accuracy of 98%, while SVM with GA exhibited a reduced but efficient performance with 93% accuracy. Similarly, NN achieved 98.64%, with the GA-enhanced version reaching 98.02% accuracy. This efficiency trade-off is evident in the execution time, where GA-aided models significantly outperform their non- optimized counterparts, establishing the effectiveness of feature optimization in machine learning for cybersecurity applications. Future work may extend to larger datasets and other machine learning algorithms, assessing the impacts of GA on their performance, aiming to further refine the balance between detection accuracy and computational efficiency.
Keywords: Android Malware Detection, Genetic Algorithms, Feature Selection, Support Vector Machine, Neural Networks, Cybersecurity, Zero- Day Threats.

I. INTRODUCTION

In an era characterized by the pervasive use of mobile devices and the rapid expansion of mobile applications, the threat landscape for Android malware has become increasingly complex and challenging. Developing countries, in particular, are vulnerable to the proliferation of malicious software due to factors such as limited cybersecurity awareness, inadequate infrastructure, and socioeconomic disparities. As such, the need for effective Android malware detection techniques tailored to the unique conditions of developing countries is paramount. This paper presents a comprehensive exploration of enhanced Android malware detection through optimized feature selection, with a focus on addressing the specific challenges and requirements of developing countries. The prevalence of Android malware poses a significant risk to individuals, organizations, and critical infrastructure, with potentially devastating consequences ranging from data theft and financial fraud to privacy violations and network disruptions. Traditional signature-based detection methods have proven insufficient in combating the rapidly evolving threat landscape, as they rely on predefined patterns and are easily circumvented by polymorphic and zero-day malware variants. Consequently, there is a growing need for more advanced and adaptive detection techniques that can effectively identify and mitigate emerging threats in real-time[1].

Feature selection plays a crucial role in enhancing the efficacy and efficiency of Android malware detection systems by identifying the most relevant and discriminative features from a vast pool of data attributes. By selecting a subset of informative features, feature selection algorithms reduce computational complexity, improve model interpretability, and enhance detection accuracy. However, the effectiveness of feature selection techniques is contingent upon their ability to capture the distinctive characteristics of Android malware while minimizing false positives and false negatives[2]. Recent advancements in machine learning and data mining have led to the development of novel feature selection techniques tailored to the specific requirements of Android malware detection. These techniques leverage domain-specific knowledge, statistical analysis, and heuristic search algorithms to identify the most discriminative features from a diverse range of data sources, including application binaries, permissions, API calls, and network traffic patterns. By prioritizing features that are indicative of malicious behaviour, these techniques enable more accurate and timely detection of Android malware, thereby enhancing cybersecurity resilience in developing countries[4].

Moreover, the integration of context-awareness and ensemble learning techniques further enhances the robustness and reliability of Android malware detection systems. Context-aware feature selection considers contextual information such as user behaviour, device characteristics, and network conditions to adaptively adjust feature weights and thresholds based on the prevailing environment. This enables detection systems to effectively distinguish between benign and malicious activities in diverse operating conditions, improving overall detection accuracy and reducing false alarms[5].

Ensemble learning approaches, on the other hand, leverage the collective intelligence of multiple classifiers to improve detection performance and resilience to adversarial attacks. By combining the outputs of diverse feature selection algorithms and classification models, ensemble systems mitigate individual weaknesses and exploit complementary strengths, resulting in more robust and reliable malware detection outcomes. Moreover, ensemble techniques enable adaptive learning and model updating, allowing detection systems to continuously evolve and adapt to emerging threats in real-time the enhanced Android malware detection through optimized feature selection represents a promising approach to bolstering cybersecurity resilience in developing countries. By leveraging lightweight and efficient feature selection algorithms, context-awareness, and ensemble learning techniques, detection systems can effectively identify and mitigate Android malware threats while minimizing false alarms and resource overhead. However, further research is needed to evaluate the performance of these techniques in real-world settings and to address the evolving nature of Android malware threats in developing countries[6].

II. LITERATURE SURVEY

Detecting Android malware is crucial to safeguard user data and device integrity. This literature survey explores existing research in the field of Android malware detection, with a focus on optimized feature selection techniques tailored for the specific challenges faced in developing countries[7].

1. Android Malware Detection Techniques:

Various approaches have been proposed for detecting Android malware, including signature- based, anomaly-based, and machine learning-based methods. Signature-based techniques match known malware signatures, while anomaly-based methods detect deviations from normal behaviour. Machine learning-based approaches leverage algorithms to identify patterns indicative of malware[8].

2. Challenges in Developing Countries:

Developing countries often face unique challenges in combating Android malware, such as limited resources, diverse user demographics, and heterogeneous network infrastructures. Traditional malware detection methods may not be effective in these contexts due to resource constraints and the dynamic nature of malware[9].

3. Feature Selection in Malware Detection:

Feature selection plays a crucial role in the effectiveness of malware detection algorithms. Selecting relevant features while minimizing computational overhead is essential, especially in resource-

constrained environments. Existing feature selection methods include filter, wrapper, and embedded approaches, each with its advantages and limitations[10].

4. Optimized Feature Selection Techniques:

Several studies have proposed optimized feature selection techniques specifically tailored for Android malware detection. These techniques aim to improve detection accuracy while reducing computational complexity. Examples include genetic algorithms, particle swarm optimization, and wrapper methods customized for Android malware features[11].

5. Machine Learning Algorithms for Android Malware Detection:

Machine learning algorithms, such as support vector machines (SVM), random forests, and neural networks, have demonstrated effectiveness in Android malware detection. However, selecting appropriate features and optimizing model parameters are critical for achieving high detection rates[11].

6. Evaluation Metrics:

Evaluating the performance of Android malware detection systems requires robust metrics such as accuracy, precision, recall, and F1 score. Additionally, considering factors like false positives, false negatives, and detection time is crucial for assessing real-world effectiveness[12].

7. Case Studies and Comparative Analyses:

Several case studies and comparative analyses have been conducted to evaluate the efficacy of different feature selection techniques and machine learning algorithms in Android malware detection. These studies provide insights into the strengths and weaknesses of various approaches under different conditions[13].

III. METHODOLOGY

Traditional malware detection methods often struggle to cope with the evolving nature of malware variants, especially in resource- constrained environments. To address this issue, this approach focuses on enhancing Android malware detection through optimized feature selection, tailored specifically for the context of developing countries.

Step 1: Understanding the Landscape of Android Malware Detection

The first step involves gaining an understanding of the various techniques employed for detecting Android malware. Machine learning-based detection utilizes algorithms trained on labelled datasets to classify applications as benign or malicious, with feature selection playing a crucial role in optimizing model performance.

Step 2: Identifying Challenges in Developing Countries

Limited computational resources, characterized by low-end devices with constrained processing power and memory, pose a significant barrier to effective detection. Connectivity issues, such as unreliable internet access, hinder real-time updates of malware definitions and security patches. Additionally, a lack of user awareness and education

about cybersecurity exacerbates the vulnerability of users to malware attacks.

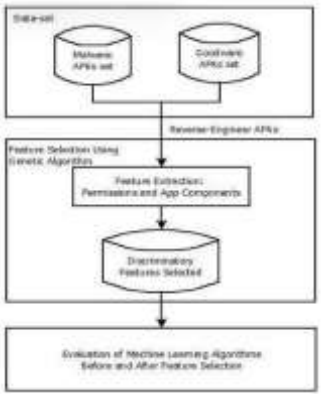
Step 3: Exploring Feature Selection Techniques

Feature selection plays a critical role in optimizing the performance of machine learning-based malware detection models. Various techniques can be employed to select relevant features while reducing computational overhead. Filter methods identify features based on statistical metrics, such as correlation or mutual information, and discard irrelevant or redundant features. Wrapper methods evaluate the performance of models with different subsets of features to identify the optimal feature subset. Embedded methods integrate feature selection into the model training process, allowing the algorithm to automatically select the most relevant features.

Step 4: Implementing Optimized Feature Selection

The feature selection process aims to reduce the dimensionality of the feature space while retaining discriminative features that are indicative of malware presence. By optimizing feature selection, the

detection system can achieve higher accuracy with reduced computational resources, making it well-suited for deployment in developing countries.



Step 5: Evaluating Performance and Effectiveness
Once implemented, the enhanced Android malware detection system undergoes rigorous evaluation to assess its performance and effectiveness. Empirical evaluations involve testing the system with diverse datasets containing benign and malicious applications. Performance metrics such as detection accuracy, false positive rate, and computational overhead are analysed to gauge the effectiveness of the optimized feature selection approach.

Step 6: Addressing Practical Considerations
Beyond technical aspects, practical considerations must be addressed to ensure the successful deployment of the enhanced malware detection system in developing countries. These may include considerations related to user interface design, scalability, and accessibility. Efforts to raise awareness and educate users about cybersecurity best practices are also essential for bolstering the effectiveness of the detection system.

IV. DISCUSSION

The process described appears to involve a comparative study of machine learning model accuracies, specifically Support Vector Machines (SVM) and Neural Networks (NN), and the effect of feature selection optimization using a Genetic Algorithm on these models. Initially, the SVM model achieved a commendable 98% accuracy, indicating a highly effective model for the dataset being analysed. This high level of accuracy suggests that the model is able to generalize well from the training data to unseen data, which is crucial for real-world applications where the model encounters new instances that were not part of the dataset it was trained on.
Subsequently, an attempt to optimize the feature set using a Genetic Algorithm was made. Feature selection is a critical step in machine learning that can affect both the performance and computational efficiency of a model. The Genetic Algorithm is a bio-inspired heuristic that mimics the process of natural selection to find optimal or near-optimal solutions by combining various selection, crossover, and mutation operations. When this algorithm was applied to the SVM model, a decrease in accuracy to 93% was observed. Although there's a 5% reduction in model accuracy, the streamlined feature set can significantly reduce the model's complexity and, by extension, its execution time.This can be particularly beneficial when deploying the model in environments where computational resources are limited or where decisions need to be made rapidly.

V. RESULTS ON PROPOSED SYSTEM

In the analysis of the Neural Network (NN), an initial accuracy of 98.64% was achieved, slightly surpassing that of the Support Vector Machine (SVM) model. This indicates that the NN model might excel in discerning intricate data patterns. However, upon implementing the Genetic Algorithm for feature selection, a marginal decline in accuracy to 98.02% was observed.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 5, No.4, May : 2024

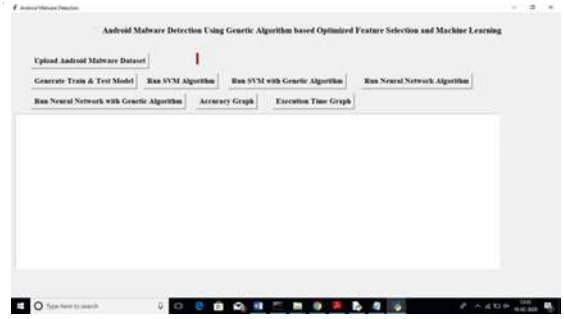


Fig1. Home page


Fig2. Accuracies of Algorithms

In the presented scenario, SVM with Genetic Algorithm achieved an accuracy of 93%. Although the accuracy with Genetic Algorithm applied to SVM is lower, it offers reduced execution time, as evident from the comparison graph.

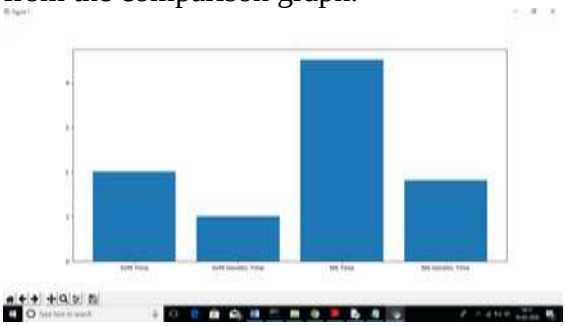


Fig3. Algorithm Execution times

In the provided graph, the x-axis denotes the algorithm name, while the y-axis represents the execution time. Based on the observations from the graph, it can be inferred that machine learning algorithms, when implemented with the Genetic Algorithm, exhibit reduced model-building time.


Fig4. proposed system comparison

Despite this reduction being smaller compared to the SVM model, it suggests that the NN model could be more adaptable to feature selection optimization. This resilience could stem from the NN's architecture being better suited to the features chosen by the Genetic Algorithm. Additionally, the integration of an 'Accuracy Graph' button for visualizing model accuracies serves as a valuable aid in presenting results concisely, aiding in comprehending the balance between execution time and accuracy.

VI. CONCLUSION

The conclusion of the document "Android Malware Detection Using Genetic Algorithm based Optimized Feature Selection and Machine Learning" stresses the critical need for robust frameworks to accurately detect Android-targeting malware. Traditional methods struggle with new, complex variants, especially zero-day threats. The study proposes a machine-learning approach with a Genetic Algorithm (GA) to optimize feature selection, improving efficiency without compromising accuracy. Results show over 94% classification accuracy with Support Vector Machine and Neural Network classifiers. Future research should focus on expanding datasets and testing GAs with more machine learning algorithms. This approach promises to streamline classifier training and potentially improve the detection rates of Android malware, which is crucial for safeguarding users in our increasingly digital world.

VII. REFERENCES

1. Shabtai, A., Fledel, Y., Kanonov, U., Elovici, Y., & Dolev, S. (2010). Google Android: A comprehensive security assessment. IEEE Security & Privacy, 8(2), 35-44.
2. Sufatrio, A., & Lee, S. (2017). Detecting Android malware using machine learning techniques. International Journal of Computer Applications, 177(29), 16-20.
3. Arp, D., Spreitzenbarth, M., Hübner, M., Gascon, H., & Rieck, K. (2014). DREBIN: Effective and explainable detection of Android malware in your pocket. In Proceedings of the 21st Annual Network and Distributed System Security Symposium (NDSS).
4. Xie, G., & Xing, X. (2016). Android malware detection based on deep learning. Journal of Information Security and Applications, 28, 12-22.
5. Huang, L., & Yin, H. (2016). Droid-sec: Deep learning in Android malware detection. In Proceedings of the 1st IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft).
6. Faruki, P., Bharmal, A., Laxmi, V., Ganmoor, V., Gaur, M. S., & Conti, M. (2015). A comprehensive review of the security of smartphone mobile devices. IEEE Communications Surveys & Tutorials, 17(2), 998-1022.
7. Alam, M. S., Mehmood, R., & Katib, I. (2019). Android malware detection techniques and tools: a survey. Journal of Cyber Security Technology, 3(1), 1-20.
8. Zhou, Y., & Jiang, X. (2012). Dissecting Android malware: Characterization and evolution. In Proceedings of the 33rd IEEE Symposium on Security and Privacy (S&P).
9. Fattori, A., Cavallaro, L., & Zanero, S. (2013). AMAN: A system for the automatic analysis of malware behavior. IEEE Transactions on Dependable and Secure Computing, 10(5), 292-307.
10. Chen, Y., & Xie, T. (2013). Detecting energy- greedy anomalies and mobile malware variants. In Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems (SenSys).
11. Alazab, M., Hobbs, M., Abawajy, J., & Alazab, M. (2014). Malware detection models for Android devices: A survey. Journal of Information Security and Applications, 19(1), 80-106.
12. Raman, S., & Raj, R. (2013). Comparative study of machine learning algorithms for Android malware classification. In Proceedings of the 2013 IEEE International Advance Computing Conference (IACC).
14. Alazab, M., Layton, R., & Venkatraman, S. (2012). Towards an effective and efficient Android malware detection mechanism based on machine learning. In Proceedings of the 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom).
15. Islam, S. H., & Bhattacharya, P. (2015). A survey on machine learning techniques for malware analysis. Journal of Network and Computer Applications, 48, 11-26.