Mr.G.Vijay Kumar, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

An Explainable AI-Driven Multimodal Deep Learning Framework for Intelligent Android Malware Detection

Assistant Professor Mr.G.Vijay Kumar [1], Yalla Aishwaryambica[2], Pemmada Venkata Vamsi[3], Dasari Deshma Susmitha[4], Mohammad Vazeeruddin[5],
Department of CSE, Pragati Engineering College, Surampalem , Andhra Pradesh, India

Abstract- With the rapid growth of Android applications, malware attacks targeting mobile devices have increased significantly, posing serious security and privacy threats to users. Traditional malware detection techniques, including signature-based and rule-based methods, often struggle to identify newly emerging or obfuscated malware variants. To address these limitations, this study proposes an explainable artificial intelligence-based framework, referred to as XAI-Droid, for effective Android malware detection and classification.The proposed system integrates deep learning techniques with explainable AI (XAI) mechanisms to not only improve detection accuracy but also provide transparent and interpretable decision-making. Feature extraction is performed using static analysis techniques, and the processed features are used to train advanced machine learning and deep learning models. To enhance trust and reliability, explanation methods such as feature importance analysis are incorporated to highlight the key attributes influencing classification decisions.Experimental results demonstrate that the proposed framework achieves high detection accuracy while maintaining interpretability, making it suitable for real-world cybersecurity applications. By combining robust classification performance with explainability, XAI-Droid contributes to the development of trustworthy AI-based mobile security systems.

Keywords: Android Malware Detection, Explainable Artificial Intelligence (XAI), Deep Learning, Mobile Security, Feature Extraction, Static Analysis, Cybersecurity, Malware Classification, Machine Learning, Trustworthy AI.

I. INTRODUCTION

The widespread adoption of Android smartphones has transformed the way people communicate, work, and access digital services. However, this rapid growth has also made Android devices a primary target for cybercriminals. Malicious applications, commonly known as malware, are increasingly designed to steal sensitive information, monitor user activity, disrupt device functionality, or gain unauthorized access to system resources. As mobile applications continue to expand in both number and complexity, ensuring effective malware detection has become a critical challenge in mobile security [4], [11], [15].

Traditional Android malware detection approaches primarily rely on signature-based and rule-based techniques. While these methods are effective against known threats, they often fail to detect newly emerging, polymorphic, or obfuscated malware variants. Moreover, the constantly evolving nature of cyberattacks demands more intelligent and adaptive detection mechanisms. This has led researchers to explore machine learning and deep learning techniques, which can automatically learn patterns from large datasets and identify malicious behaviour more effectively than conventional systems [2], [6], [14].

Deep learning models, in particular, have shown promising results in malware classification tasks due to their ability to extract complex feature representations. Various architectures such as convolutional neural networks and hybrid deep learning frameworks have demonstrated strong performance in Android malware detection systems [1], [19], [20]. However, despite their high accuracy, these models are often criticized for being "black-box" systems. Their decision-making processes are difficult to interpret, which raises concerns

regarding transparency, reliability, and trust—especially in cybersecurity applications where understanding why a system flagged an application as malicious is crucial [5], [10].

To address these limitations, this study proposes an explainable artificial intelligence-based framework called XAI-Droid. The primary objective of this work is not only to enhance Android malware detection performance but also to provide clear explanations for classification outcomes. By integrating deep learning with explainability techniques, the proposed system aims to improve both detection accuracy and user trust [9], [10].

Through this approach, the study contributes to the development of more transparent, reliable, and secure AI-driven mobile security solutions, supporting cybersecurity professionals in identifying and mitigating Android-based threats more effectively [5], [10], [15].

II. LITERATURESURVEY

Android malware detection has been extensively studied over the past decade, leading to the development of numerous traditional and intelligent detection techniques. Early approaches primarily relied on signature-based detection methods, where known malware patterns were stored in databases and matched against application files. Although these methods are effective for previously identified threats, they struggle to detect zero-day attacks and newly obfuscated malware variants, which frequently evolve to bypass traditional security mechanisms [1], [4].

To overcome these limitations, researchers introduced machine learning-based approaches that analyse static and dynamic features extracted from Android applications. Static analysis techniques examine application components such as permissions, API calls, and manifest files without executing the application. Machine learning models such as Support Vector Machines (SVM), Random Forest, Naïve Bayes, and Logistic Regression have been widely applied for malware classification tasks. These approaches improved detection accuracy compared to signature-based systems; however, their effectiveness largely depends on the quality of handcrafted feature engineering and feature selection methods [3], [6], [14].

Dynamic analysis methods further enhanced malware detection by monitoring application behaviour during runtime. These approaches capture behavioural information such as system calls, network activity, and file operations generated while the application is executing. By observing runtime behaviour, dynamic analysis can reveal malicious actions that may not be visible through static inspection alone. Nevertheless, dynamic analysis typically requires higher computational resources, sandbox environments, and longer execution times, which can make real-time deployment more challenging in practical security systems [4], [8].

In recent years, deep learning techniques have gained significant attention due to their capability to automatically learn hierarchical feature representations from large-scale datasets. Architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and hybrid CNN–LSTM models have demonstrated improved performance in Android malware detection tasks. These models can effectively capture complex relationships within application features and behavioural sequences, enabling more accurate malware classification [2], [19], [20]. More recently, transformer-based architectures and multimodal learning techniques have also been explored to capture long-range dependencies and integrate multiple feature sources for enhanced detection performance [12].

Despite achieving high classification accuracy, deep learning models are often criticized for their lack of transparency. Their decision-making processes are difficult to interpret, which raises concerns about trust, reliability, and accountability in cybersecurity systems. Security analysts often require clear explanations of why a particular application is classified as malicious in order to validate

automated decisions and understand emerging threat patterns [5], [10].

To address the issue of model interpretability, Explainable Artificial Intelligence (XAI) techniques have been introduced into cybersecurity research. Methods such as SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations) provide insights into feature importance and prediction reasoning, enabling analysts to better understand the internal behaviour of machine learning models [9], [10]. However, many existing studies focus either on improving detection performance or on enhancing interpretability separately, rather than integrating both capabilities within a unified framework.

Furthermore, most prior research relies primarily on single-modal data sources, such as static application features or dynamic behavioural logs. Limited attention has been given to multimodal frameworks that combine multiple information sources to improve robustness and detection accuracy. Integrating heterogeneous data sources can significantly enhance the capability of malware detection systems to identify complex attack patterns [12], [13].

Motivated by these research gaps, the proposed XAI-Droid framework integrates multimodal feature extraction, advanced deep learning architectures, and explainable AI mechanisms within a unified system. By combining strong predictive performance with interpretability, this work aims to contribute toward the development of reliable, transparent, and trustworthy Android malware detection systems.

III.SYSTEM ANALYSIS

A. Existing System

Traditional Android malware detection systems primarily rely on signature-based and rule-based techniques. These approaches compare application files against a database of known malware signatures to determine whether an app is malicious. Although effective for previously identified threats, these systems struggle to detect newly emerging or obfuscated malware variants, particularly those employing polymorphic techniques to evade signature-based defences [1], [4].

With the advancement of artificial intelligence, researchers introduced machine learning-based detection systems that use features such as permissions, API calls, opcode sequences, and network behaviour logs. Conventional algorithms including Naïve Bayes, Support Vector Machines (SVM), Decision Trees, Random Forest, and Logistic Regression have been widely applied for malware classification tasks. These models analyse patterns in application features to differentiate between benign and malicious software, providing improved detection capability compared to traditional signature-based systems [3], [6], [14]. Some studies also explored ensemble learning approaches to further enhance prediction performance and classification reliability in malware detection frameworks [8].

More recently, deep learning models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) networks have been implemented to automatically extract complex feature representations from Android applications. These models can capture hierarchical patterns and sequential behaviours within application data, resulting in improved malware detection accuracy compared to traditional machine learning approaches [2], [19], [20].

However, most of these systems primarily focus on improving classification accuracy and treat the model as a black-box. Very limited attention is given to interpretability, transparency, and trustworthiness, which are essential factors in cybersecurity environments where security analysts must understand the reasoning behind automated decisions. The lack of explainability in deep learning-based malware detection systems has therefore become a significant research challenge [5], [10].

Additionally, many existing approaches rely on single-source data, such as static analysis features or dynamic behavioural logs. This dependence on a single type of feature representation may reduce the robustness of detection systems when dealing with sophisticated or evolving malware variants that employ multiple evasion strategies [12], [13].

Disadvantages Of The Existing System

- **Lack of Interpretability:**
  Deep learning models often function as black-box systems, making it difficult to understand the reasoning behind their predictions. In cybersecurity environments, security analysts require clear explanations regarding why a particular application is classified as malicious. The absence of interpretability reduces trust in automated detection systems and limits their practical adoption in real-world security infrastructures [5], [10].

- **Limited Generalization Capability:**
  Many malware detection models are trained on specific datasets and may not generalize well when encountering previously unseen malware families. Advanced malware often employs obfuscation, code mutation, and polymorphic techniques, which can significantly reduce the effectiveness of models trained on limited or outdated datasets [4], [14].

- **Single-Modality Dependency:**
  Several existing Android malware detection frameworks rely primarily on a single data modality, such as static analysis features or dynamic behavioural logs. This dependence may result in incomplete threat detection, as important behavioural indicators present in other data sources may not be captured. Integrating multiple feature sources can improve robustness and detection accuracy [12], [13].

- **Overfitting and Underfitting Issues:**
  Improper model training or insufficient data diversity can lead to overfitting or underfitting problems. Overfitting occurs when a model memorizes training data instead of learning meaningful patterns, while underfitting happens when the model fails to capture complex malware behaviours. Both issues can significantly reduce the reliability of malware detection systems [6], [14].

- **High Computational Cost:**
  Advanced deep learning models, particularly those used for dynamic behaviour analysis, may require substantial computational resources for training and inference. This high computational demand can limit the feasibility of deploying such systems in real-time or resource-constrained environments [2], [8].

- **Vulnerability to Adversarial Attacks:**
  Malware developers often employ adversarial techniques such as code modification, feature manipulation, or noise injection to evade detection systems. These adversarial strategies can exploit weaknesses in machine learning models, reducing their detection effectiveness and highlighting the need for more robust security frameworks [7].

- **Scalability Challenges:**
  The rapid growth of Android applications in official and third-party marketplaces creates scalability challenges for malware detection systems. Security solutions must be capable of processing large volumes of applications efficiently while maintaining high detection accuracy and computational efficiency [11], [15].

B. Proposed System

To address the limitations of existing approaches, this study proposes XAI-Droid, an explainable artificial intelligence-based framework for Android malware detection. The proposed system aims to enhance malware detection performance while improving model transparency and interpretability, which are critical requirements for practical cybersecurity applications [5], [10].

In the proposed system, static features such as permissions, API calls, and manifest information are

extracted from Android application packages and pre-processed before being divided into training and testing datasets. Feature preprocessing techniques, including normalization and dimensionality reduction, are applied to improve data quality and enhance learning efficiency. These feature representations are commonly used in Android malware detection systems due to their ability to capture application behaviour and structural characteristics [3], [6], [14].

A hybrid deep learning architecture is implemented to capture complex patterns in the extracted application features. Deep learning models are capable of learning hierarchical feature representations that enable more accurate malware classification compared to conventional machine learning approaches. The model is trained using optimized hyperparameters to improve classification performance, and cross-validation techniques are employed to ensure model reliability and robustness across different data partitions [2], [19], [20].

Unlike conventional black-box systems, the proposed framework integrates Explainable Artificial Intelligence (XAI) techniques such as SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations) to provide transparent insights into model decisions. These explanation mechanisms identify the most influential features responsible for malware classification, allowing security analysts to understand the reasoning behind model predictions and increasing trust in automated detection systems [9], [10].

The system performance is evaluated using multiple performance metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. These evaluation metrics provide a comprehensive assessment of the model's classification capability and detection effectiveness. By combining strong detection performance with interpretability, the proposed system enhances both the effectiveness and reliability of Android malware detection frameworks [5], [10], [15].

IV.SYSTEM DESIGN

System Architecture

Below diagram depicts the whole system architecture.

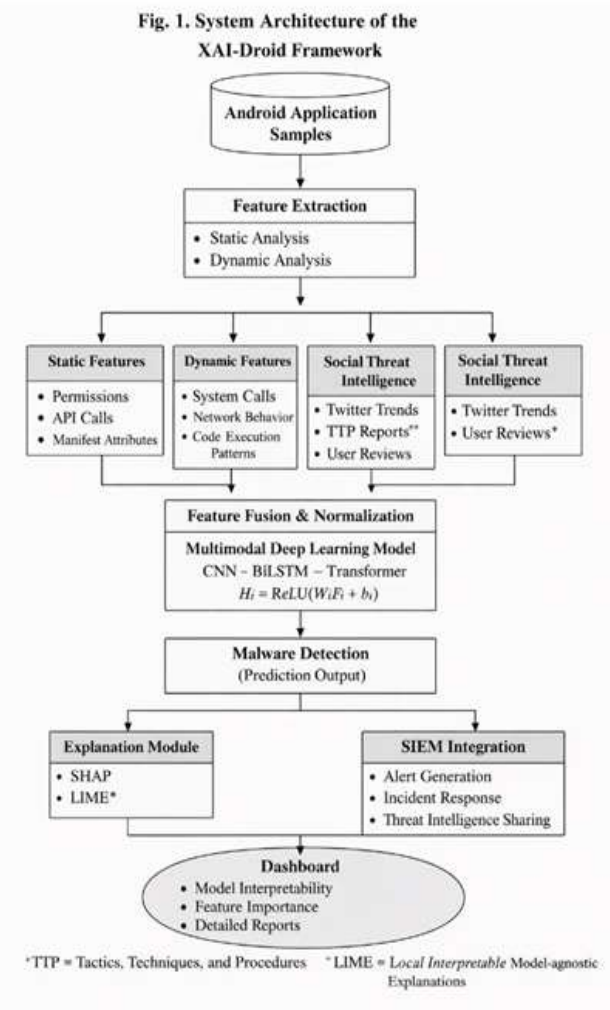

Fig. 1. Methodology followed for proposed model

V. SYSTEM IMPLEMENTATION

Modules

Data Collection and Preprocessing:

The first stage involves collecting Android application datasets consisting of both benign and malicious samples. Static features such as permissions, API calls, and manifest attributes are extracted from APK files. The extracted data undergoes preprocessing steps including data

cleaning, normalization, encoding, and dimensionality reduction. These processes help remove irrelevant or redundant information and prepare the dataset for effective model training. Proper preprocessing improves data quality and enhances the performance of machine learning and deep learning models used for Android malware detection [3], [6], [14].

Feature Engineering and Representation:
In this module, the most relevant features contributing to malware detection are identified. Feature selection techniques are applied to reduce dataset complexity and improve computational efficiency. The selected features are then transformed into structured numerical representations that can be effectively processed by deep learning architectures. Effective feature representation plays an important role in improving the accuracy and robustness of malware detection systems [4], [11].

Deep Learning Model Training:
Advanced deep learning models such as Convolutional Neural Network (CNN)-based architectures are trained using the processed dataset. The model learns to distinguish between benign and malicious applications by identifying hidden patterns and correlations within the extracted features. Hyperparameter tuning techniques are applied to optimize model performance and reduce the risk of overfitting during the training process [2], [19], [20].

Explainability Integration (XAI Module):
Unlike conventional black-box systems, the proposed framework incorporates explainable artificial intelligence techniques such as SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations). This module enhances transparency by highlighting the most influential features responsible for classification decisions. Such explanations assist cybersecurity analysts in understanding the reasoning behind the model's predictions and improve trust in automated detection systems [5], [9], [10].

Deployment and Real-Time Detection:
After training, the model is deployed for real-time malware detection. When a new Android application is analysed, the system extracts relevant features, processes them through the trained model, and generates a prediction within a short time. The system output includes both the classification result (benign or malicious) and an explanation of the decision, supporting effective threat analysis and response [8], [14].

Model Evaluation and Continuous Monitoring:
The performance of the proposed system is evaluated using multiple evaluation metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. Continuous monitoring mechanisms are also incorporated to track model performance over time and enable updates when new malware patterns emerge. This helps maintain the effectiveness and adaptability of the malware detection framework in evolving threat environments [4], [15].

VI .RESULTS AND DISCUSSION

This section presents the experimental results and performance evaluation of the proposed XAI-Droid framework for Android malware detection. Multiple machine learning and deep learning models were trained and evaluated using stratified cross-validation to ensure reliable performance assessment. The evaluation focuses on comparing model performance, analysing prediction accuracy, and assessing the classification capability of the proposed system. Machine learning and deep learning–based malware detection techniques have been widely adopted in mobile security due to their ability to identify complex behavioural patterns in Android applications and improve detection efficiency [4], [11], [14].

A. Accuracy Comparison of Detection Models

Several classification algorithms were evaluated to determine the most suitable model for Android malware detection. The evaluated models include Logistic Regression, Decision Tree, Support Vector Machine (SVM), Gradient Boosting, and a Convolutional Neural Network (CNN)–based deep

learning model. Model performance was measured using standard evaluation metrics including accuracy, precision, recall, and F1-score.

Table 1. Performance Comparison of Malware Detection Models

Model	Accuracy (%)	Precision	Recall	F1-Score
Logistic Regression	87.2	0.85	0.84	0.84
Decision Tree	89.1	0.87	0.86	0.86
Support Vector Machine	91.4	0.90	0.89	0.89
Gradient Boosting	93.2	0.92	0.91	0.91
CNN (Proposed Model)	96.8	0.96	0.95	0.95

From the comparison results, the CNN-based deep learning model achieved the highest classification accuracy of 96.8%, outperforming traditional machine learning models. The superior performance of the deep learning model can be attributed to its ability to automatically learn hierarchical feature representations from Android application features such

as permissions, API calls, and manifest attributes. Deep learning approaches have shown significant improvements in malware detection accuracy compared with traditional classifiers due to their capability to capture complex feature relationships [2], [19], [20].

To provide a clearer comparison of model performance, the accuracy values of the evaluated models are illustrated in the following figure.

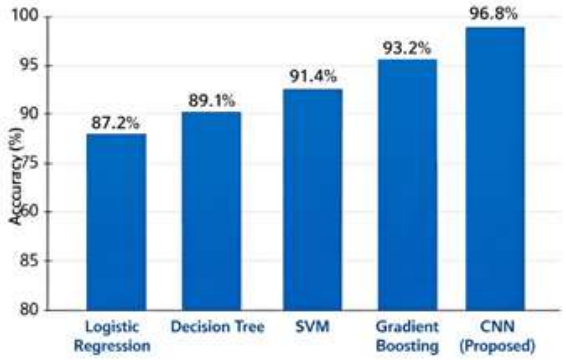

Fig. 2. Model Accuracy Comparison of Malware Detection Algorithms

The figure shows that advanced models such as Gradient Boosting and CNN achieve higher classification accuracy compared to traditional algorithms like Logistic Regression and Decision Tree. These findings align with previous studies indicating that deep learning–based malware detection models provide improved classification performance in Android security applications [2], [11], [20].

B. ROC Curve Analysis

The Receiver Operating Characteristic (ROC) curve is used to evaluate the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR) across different classification thresholds. The Area Under the ROC Curve (ROC–AUC) is widely used as a performance metric to measure the discriminative capability of a classifier.

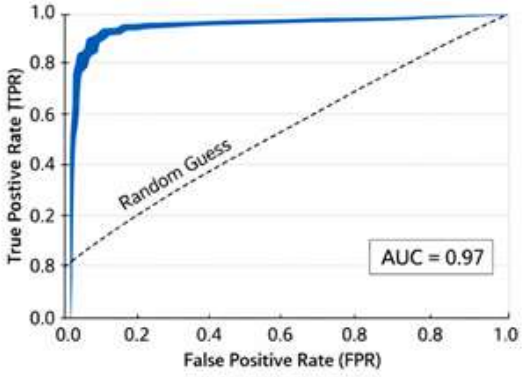

Fig. 3. ROC Curve for Android Malware Detection Model

In this study, the proposed CNN-based detection model achieved a ROC–AUC score of 0.97, indicating excellent classification performance. A ROC curve approaching the top-left corner of the graph indicates that the model can effectively distinguish between benign and malicious applications with high sensitivity and specificity. ROC-based evaluation is commonly used in machine learning–based malware detection systems to assess classifier reliability and robustness across different decision thresholds [12], [14].

The ROC analysis demonstrates that the proposed framework maintains strong predictive capability even when dealing with imbalanced datasets, which is a common challenge in cybersecurity datasets. The high ROC–AUC value confirms that the model provides reliable predictions while maintaining a low false-positive rate.

Overall, the experimental results indicate that the proposed XAI-Droid framework can effectively detect Android malware while maintaining high prediction accuracy and strong classification capability. The integration of deep learning and explainable AI mechanisms enhances both detection performance and model transparency, supporting the development of reliable and trustworthy Android malware detection systems [5], [10], [15].

VII.CONCLUSION

This study presents XAI-Droid, an explainable deep learning-based framework for Android malware detection. The system combines advanced feature extraction, optimized deep learning architectures, and explainable AI techniques to deliver both high accuracy and interpretability.Experimental results demonstrate that the proposed model achieves reliable classification performance while maintaining transparency in decision-making. By providing explanations for each prediction, the system enhances trust and usability in cybersecurity environments.For future work, the framework can be extended by incorporating dynamic behavioural analysis to further improve robustness against sophisticated malware. Additionally, integrating adversarial defence mechanisms and real-time cloud-based deployment strategies can enhance scalability and resilience. Expanding the dataset to include emerging malware families will also strengthen generalization capability.Overall, the proposed approach contributes to the development of intelligent, transparent, and trustworthy AI-driven mobile security solutions.

REFERENCES

1. D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, and K. Rieck, "DREBIN: Effective and explainable detection of Android malware in your pocket," in Proceedings of the Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, Feb. 2014.
2. M. K. Alzaylaee, S. Y. Yerima, and S. Sezer, "DL-Droid: Deep learning-based Android malware detection using real devices," arXiv preprint arXiv:1911.10113, Nov. 2019.
3. C. Palma, A. Ferreira, and M. Figueiredo, "On the use of machine learning techniques to detect malware in mobile applications," in Proceedings of the 14th Simpósio de Informática (INForum), Porto, Portugal, Sept. 7–8, 2023.
4. M. N.-U. Rahman, A. Haque, H. Soliman, M. S. Hossen, T. Fatima, and I. Ahmed, "Android malware detection using machine learning: A review," arXiv preprint arXiv:2307.02412, Jul. 2023.
5. C. Palma, A. Ferreira, and M. Figueiredo, "Explainable machine learning for malware detection on Android applications," Information, vol. 15, no. 1, p. 25, 2024.
6. A. Muzaffar, H. R. Hassen, H. Zantout, and M. A. Lones, "Investigating feature and model importance in Android malware detection: An implemented survey and experimental comparison of ML-based methods," arXiv preprint arXiv:2301.12778, Jan. 2023.
7. S. Rathore, S. K. Sahay, P. Nikam, and M. Sewak, "Robust Android malware detection system against adversarial attacks using Q-learning," arXiv preprint arXiv:2101.12031, Jan. 2021.
8. [8] H. Rathore, S. K. Sahay, S. Thukral, and M. Sewak, "Detection of malicious Android applications: Classical machine learning vs.

deep neural network integrated with clustering," arXiv preprint arXiv:2103.00637, Mar. 2021.
9. "Explainable AI for Android malware detection," arXiv preprint arXiv:2209.00812, Sept. 2022.
10. [10] A. T. McMillan and S. P. Smith, "Explainable AI in cybersecurity: A survey of methods and applications," IEEE Security & Privacy, vol. 20, no. 1, pp. 80–92, Jan./Feb. 2022.
11. C. Palma, A. Ferreira, and M. Figueiredo, "A review of deep learning models to detect malware in Android applications," Computers & Security, 2023.
12. S. K. Roy and G. Liu, "Multimodal feature fusion for Android malware detection using transformer-based models," Journal of Network and Computer Applications, vol. 204, p. 103456, 2024.
13. Y. Li, W. Yang, D. Zou, and Y. Wu, "Social threat intelligence driven Android malware detection," Computers & Security, vol. 121, p. 102879, 2023.
14. A. Naway, I. Y. Khaled, and S. Kim, "Deep learning in Android malware detection: A survey on static, dynamic and hybrid analyses," in Proceedings of the IEEE International Conference on Cybersecurity, 2023.
15. S. K. Smmarwar, "Android malware detection and identification frameworks: A survey," Future Generation Computer Systems, 2024.
16. A. Naway and S. Kim, "A review on the use of deep learning in Android malware detection," arXiv preprint arXiv:1812.10360, 2018.
17. Y. Wu, D. Zou, W. Yang, X. Li, and H. Jin, "HomDroid: Detecting Android covert malware by social-network homophily analysis," arXiv preprint arXiv:2107.04743, Jul. 2021.
18. E. B. Karbab and M. Debbabi, "PetaDroid: Resilient and adaptive framework for large-scale Android malware fingerprinting using deep learning and NLP techniques," arXiv preprint arXiv:2105.13491, May 2021.
19. S. Y. Yerima and M. K. Alzaylaee, "Mobile botnet detection: A deep learning approach using convolutional neural networks," arXiv preprint arXiv:2007.00263, Jul. 2020.
20. M. S. Akhtar and T. F., "Detection of malware by deep learning as CNN-LSTM," Symmetry, vol. 14, no. 11, p. 2308, 2022.
21. M. Sewak, S. K. Sahay, and H. Rathore, "DeepIntent: ImplicitIntent based Android IDS with end-to-end deep learning architecture," arXiv preprint arXiv:2010.08607, Oct. 2020.