International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

A Context-Aware Multimodal Explainable Deep Learning Framework for Robust Android Malware Detection and Proactive Threat Prevention

Mr .B.Janu Naik[1], Velugubanti Lakshmana Siva Ganesh[2], Vignesh Mullangi[3], Vanapalli Veera Satya Sai Praneesh[4], Maddula Veera Venkata Sai Pradeep[5]

[1]Assistant Professor, Department of CSE (Data Science) In Pragati Engineering College, Surampalem, Andhra Pradesh, India,
[2,3,4,5] UG Students Department of CSE (Data Science) In Pragati Engineering College, Surampalem, Andhra Pradesh, India.

Abstract- With the rapid expansion of Android applications, malware attacks targeting mobile devices have increased significantly, creating serious security and privacy concerns for users. Traditional malware detection approaches, such as signature-based and rule-based methods, often fail to detect newly emerging or obfuscated malware variants. To overcome these challenges, this study proposes an explainable artificial intelligence-based framework, named XAI-Droid, for effective Android malware detection and classification. The proposed system integrates deep learning techniques with explainable AI (XAI) methods to enhance detection accuracy while ensuring transparency and interpretability in decision-making. Feature extraction is carried out using static analysis techniques, and the extracted features are used to train advanced machine learning and deep learning models. To improve trust and reliability, explanation methods such as feature importance analysis are incorporated to identify the key attributes influencing classification outcomes. Experimental results demonstrate that the proposed framework achieves high detection accuracy while maintaining interpretability, making it suitable for practical cybersecurity applications. By combining strong classification performance with explainability, XAI-Droid contributes to the development of reliable and 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 use of Android smartphones has significantly changed the way people communicate, work, and access digital services. However, this rapid adoption has also made Android devices a major target for cyberattacks. Malicious applications, commonly referred to as malware, are increasingly developed to steal sensitive data, monitor user activities, disrupt device operations, or gain unauthorized access to system resources. As the number and complexity of mobile applications continue to grow, effective malware detection has become a critical concern in mobile security [4], [11], [15].

Conventional Android malware detection methods mainly rely on signature-based and rule-based approaches. Although these techniques are effective in identifying known threats, they often fail to detect newly emerging, polymorphic, or obfuscated malware variants. Additionally, the dynamic and evolving nature of cyber threats requires more advanced and adaptive detection mechanisms. As a result, researchers have turned to machine learning and deep learning methods, which can automatically learn patterns from large datasets and detect malicious behaviour more effectively than traditional techniques [2], [6], [14].

Deep learning models have shown strong potential in malware classification due to their ability to learn complex feature representations. Various architectures, including convolutional neural networks and hybrid deep learning models, have demonstrated high performance in Android malware detection systems [1], [19], [20]. However, despite their effectiveness, these models are often considered “black-box” systems because their decision-making processes are not easily interpretable. This lack of transparency raises concerns about reliability and trust, particularly in cybersecurity applications where

understanding the reason behind a classification decision is essential [5], [10].

To overcome these challenges, this study proposes an explainable artificial intelligence-based framework known as XAI-Droid. The main objective of this work is not only to improve malware detection accuracy but also to provide clear and interpretable explanations for classification results. By combining deep learning with explainability techniques, the proposed system aims to enhance both performance and user trust [9], [10].

Through this approach, the study contributes to the development of transparent, reliable, and secure AI-based mobile security solutions, enabling cybersecurity professionals to detect and mitigate Android malware threats more effectively [5], [10], [15].

II. LITERATURE SURVEY

Android malware detection has been widely researched over the past decade, resulting in the development of various traditional and intelligent detection techniques. Early methods mainly relied on signature-based approaches, where known malware patterns were stored in databases and matched against application files. Although these techniques are effective for detecting known threats, they are not capable of identifying zero-day attacks or newly obfuscated malware variants, which are frequently designed to evade traditional security systems [1], [4].

To address these limitations, machine learning-based approaches were introduced to analyse 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 including Support Vector Machines (SVM), Random Forest, Naïve Bayes, and Logistic Regression have been widely used for malware classification. These approaches improved detection accuracy compared to signature-based methods; however, their effectiveness depends heavily on the quality of feature engineering and feature selection processes [3], [6], [14].

Dynamic analysis techniques further improved malware detection by analysing application behaviour during runtime. These methods monitor system activities such as system calls, network communication, and file operations while the application is executing. By capturing runtime behaviour, dynamic analysis can identify malicious actions that may not be visible through static analysis alone. However, these approaches require higher computational resources, sandbox environments, and longer execution time, making real-time deployment more challenging [4], [8].

In recent years, deep learning methods have gained significant attention due to their ability to automatically learn hierarchical feature representations from large datasets. Models such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and hybrid CNN–LSTM architectures have shown strong performance in Android malware detection. These models can capture complex relationships within application features and behavioural patterns, leading to improved classification accuracy [2], [19], [20]. More recently, transformer-based models and multimodal learning approaches have been explored to capture long-range dependencies and combine multiple data sources for better detection performance [12].

Despite their high accuracy, deep learning models are often criticized for their lack of interpretability. Their decision-making process is not easily understandable, which raises concerns about transparency, trust, and accountability in cybersecurity systems. Security analysts require clear explanations for why an application is classified as malicious in order to validate automated decisions and understand evolving threat patterns [5], [10].

To overcome this limitation, Explainable Artificial Intelligence (XAI) techniques have been introduced in cybersecurity applications. Methods such as SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations) provide insights into feature importance and model decision-making, allowing analysts to better understand prediction outcomes [9], [10]. However, many existing studies focus either on improving detection performance or on enhancing interpretability separately, rather than combining both within a single framework.

Additionally, most previous research primarily relies on single data sources, such as static features or dynamic behavioural data. Limited attention has been given to multimodal approaches that integrate multiple sources of information.

Combining heterogeneous data sources can significantly improve the robustness and accuracy of malware detection systems by capturing complex attack patterns more effectively [12], [13].

Motivated by these limitations, the proposed XAI-Droid framework integrates multimodal feature extraction, advanced deep learning models, and explainable AI techniques within a unified system. By combining high detection performance with interpretability, the proposed approach aims to develop reliable, transparent, and trustworthy Android malware detection systems.

III. SYSTEM ANALYSIS

A. EXISTING SYSTEM

Traditional Android malware detection systems mainly rely on signature-based and rule-based approaches. These methods compare application files with a database of known malware signatures to determine whether an application is malicious. While these techniques are effective for detecting previously known threats, they are unable to identify newly emerging or obfuscated malware, especially those using polymorphic techniques to bypass signature-based defences [1], [4].

With the advancement of artificial intelligence, machine learning-based detection systems have been introduced to enhance malware detection. These systems utilize features such as permissions, API calls, opcode sequences, and network behaviour logs. Common algorithms including Naïve Bayes, Support Vector Machines (SVM), Decision Trees, Random Forest, and Logistic Regression have been widely used for classification tasks. These models analyse patterns within application data to distinguish between benign and malicious software, offering improved performance compared to traditional signature-based methods [3], [6], [14]. In addition, some studies have explored ensemble learning techniques to further improve prediction accuracy and reliability in malware detection systems [8].

More recently, deep learning approaches such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) networks have been applied to automatically learn complex feature representations from Android applications. These models are capable of capturing hierarchical structures and sequential behaviours within data, leading to improved detection accuracy compared to conventional machine learning techniques [2], [19], [20].

However, most of these systems primarily focus on improving classification accuracy and treat the model as a black-box. Limited attention has been given to interpretability, transparency, and trust, which are critical factors in cybersecurity applications where analysts need to understand the reasoning behind model decisions. The lack of explainability in deep learning-based malware detection systems has therefore become a significant challenge [5], [10]. Furthermore, many existing approaches depend on a single type of data source, such as static features or dynamic behavioural logs. This reliance on a single representation may reduce system robustness when dealing with advanced or evolving malware that uses multiple evasion techniques [12], [13].

DISADVANTAGES OF THE EXISTING SYSTEM

- **Lack of Interpretability:**

Deep learning models often operate as black-box systems, making it difficult to understand the reasoning behind their predictions. In cybersecurity applications, analysts require clear explanations for why an application is classified as malicious. The absence of interpretability reduces trust in automated systems and limits their adoption in real-world security environments [5], [10].

- **Limited Generalization Capability:**

Many malware detection models are trained on specific datasets and may not perform well when exposed to new or unseen malware families. Advanced malware techniques such as obfuscation, code mutation, and polymorphism can reduce the effectiveness of models trained on limited or outdated datasets [4], [14].

- **Single-Modality Dependency:**

Several existing detection frameworks rely on a single type of data source, such as static features or dynamic behavioural logs. This dependency may lead to incomplete detection, as important indicators from other data sources are not considered. Integrating multiple feature sources can improve robustness and detection accuracy [12], [13].

- **Overfitting and Underfitting Issues:**

Inadequate training or lack of data diversity can result in overfitting or underfitting. Overfitting occurs when a model memorizes training data instead of learning general patterns, while underfitting happens when the model fails to capture complex malware behaviours. Both issues reduce system reliability [6], [14].

- **High Computational Cost:**

Advanced deep learning models, especially those used for dynamic analysis, require significant computational resources for training and inference. This high resource requirement can limit their deployment in real-time or resource-constrained environments [2], [8].

- **Vulnerability to Adversarial Attacks:**

Malware developers often use adversarial techniques such as code modification, feature manipulation, and noise injection to evade detection systems. These methods exploit weaknesses in machine learning models and reduce their effectiveness, highlighting the need for more robust detection frameworks [7].

- **Scalability Challenges:**

The continuous growth of Android applications across official and third-party platforms creates scalability challenges for malware detection systems. Effective solutions must be capable of processing large volumes of applications efficiently while maintaining high accuracy and performance [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].

IV. SYSTEM DESIGN

SYSTEM ARCHITECTURE

Below diagram depicts the whole system architecture.

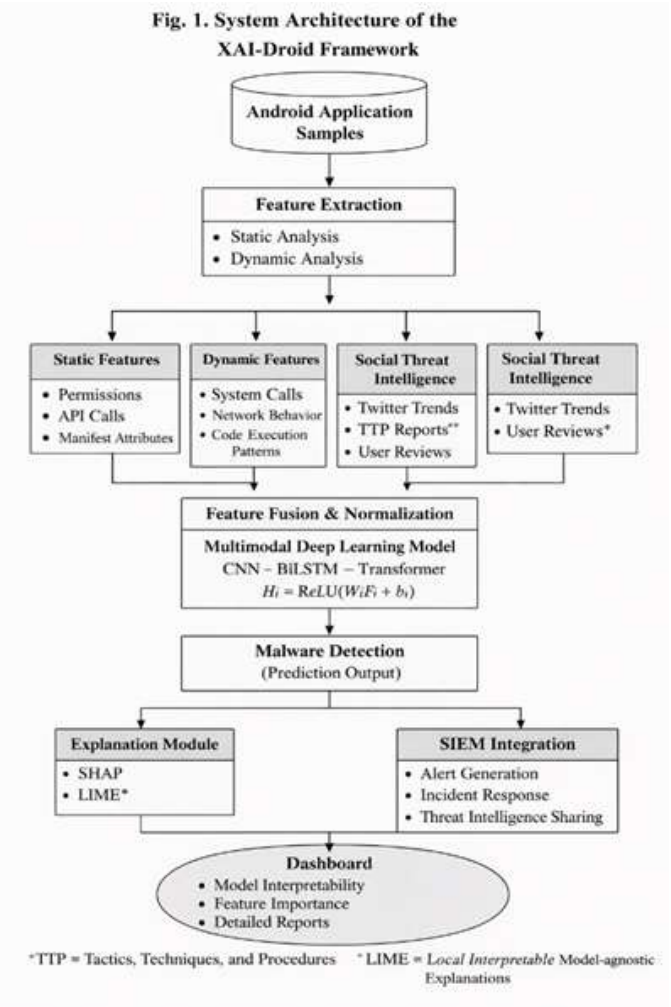

Fig. 1. Methodology followed for proposed model

V. SYSTEM IMPLEMENTATION

MODULES

Data Collection and Preprocessing:

The initial stage involves collecting Android application datasets that include both benign and malicious samples. Static features such as permissions, API calls, and manifest attributes are extracted from APK files. The extracted data is then processed through preprocessing steps including data cleaning, normalization, encoding, and dimensionality reduction. These steps help eliminate irrelevant or redundant information and prepare the dataset for efficient 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 stage, the most important 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 formats that can be effectively utilized by deep learning models. Proper feature representation plays a crucial role in improving the accuracy and robustness of malware detection systems [4], [11].

Deep Learning Model Training:

Advanced deep learning models, particularly Convolutional Neural Network (CNN)-based architectures, are trained using the processed dataset. The model learns to differentiate between benign and malicious applications by identifying hidden patterns and relationships within the extracted features. Hyperparameter tuning techniques are applied to optimize model performance and reduce overfitting during the training process [2], [19], [20].

Explainability Integration (XAI Module):

Unlike traditional 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 improves transparency by identifying the most influential features responsible for classification decisions. These explanations help cybersecurity analysts understand the reasoning behind model predictions and increase 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, relevant features are extracted and processed through the trained model to generate predictions quickly. 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 metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Continuous monitoring is also implemented to track model performance over time and update the system as new malware patterns emerge. This ensures that the detection framework remains effective and adaptable in evolving cybersecurity environments [4], [15].

VI. RESULTS AND DISCUSSION

This section presents the experimental evaluation and performance analysis of the proposed XAI-Droid framework for Android malware detection. Multiple machine learning and deep learning models were trained and assessed using stratified cross-validation to ensure reliable and unbiased evaluation.

The analysis focuses on comparing model performance, evaluating prediction accuracy, and examining the classification capability of the proposed system. Machine learning and deep learning approaches have been widely adopted in mobile security due to their effectiveness in identifying complex behavioural patterns in Android applications and improving detection performance [4], [11], [14].

A. Accuracy Comparison of Detection Models

Several classification algorithms were evaluated to identify the most effective model for Android malware detection. The models considered include Logistic Regression, Decision Tree, Support Vector Machine (SVM), Gradient Boosting, and a Convolutional Neural Network (CNN)-based deep learning model. Performance was evaluated using standard metrics such as 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 results, the CNN-based deep learning model achieved the highest classification accuracy of 96.8%, outperforming all traditional machine learning models. This improved performance is due to the model's ability to automatically learn hierarchical feature representations from application data such as permissions, API calls, and manifest attributes. Deep learning models are particularly effective in capturing complex relationships within features, leading to improved malware detection accuracy compared to conventional methods [2], [19], [20].

The accuracy comparison of the evaluated models is illustrated in Fig. 2, where advanced models such as Gradient Boosting and CNN demonstrate superior performance compared to traditional approaches like Logistic Regression and Decision Tree. These findings are consistent with previous studies that highlight the effectiveness of deep learning techniques in Android malware detection [2], [11], [20].

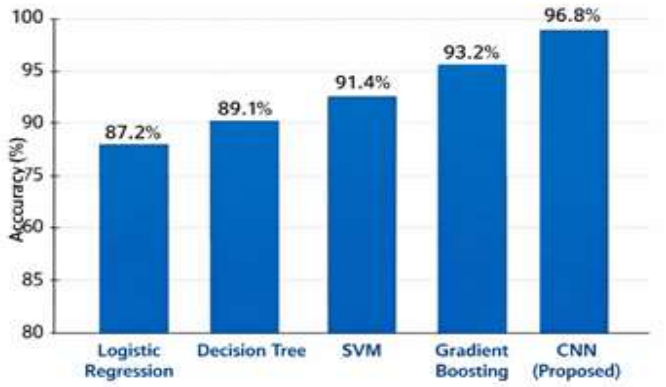

Fig. 2. Model Accuracy Comparison of Malware Detection Algorithms

B. ROC Curve Analysis

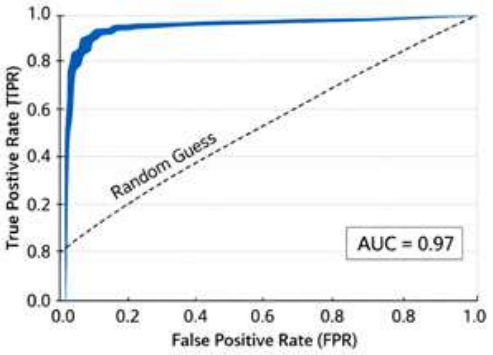

Fig. 3. ROC Curve for Android Malware Detection Model

In this study, the proposed CNN-based model achieved a ROC–AUC score of 0.97, indicating excellent classification performance. A ROC curve that approaches the top-left corner of the graph represents a model with high sensitivity and specificity, demonstrating strong capability in distinguishing between benign and malicious applications. ROC-based evaluation is commonly used in malware detection systems to assess model reliability across different threshold values [12], [14].

The ROC analysis further shows that the proposed framework maintains strong predictive performance even when handling imbalanced datasets, which is a common challenge in cybersecurity applications. The high ROC–AUC value confirms that the model produces reliable predictions while maintaining a low false-positive rate.

Overall, the experimental results demonstrate that the proposed XAI-Droid framework effectively detects Android malware with high accuracy and strong classification capability. The integration of deep learning with explainable AI techniques enhances both detection performance and model transparency, contributing to the development of reliable and trustworthy malware detection systems [5], [10], [15].

VII. CONCLUSION AND FUTURE WORK

This study introduced XAI-Droid, an explainable deep learning-based framework for Android malware detection. The proposed system integrates advanced feature extraction techniques, optimized deep learning models, and explainable AI methods to achieve both high detection accuracy and interpretability. Experimental results confirm that the model

delivers reliable classification performance while maintaining transparency in its decision-making process.

By providing clear explanations for each prediction, the system improves trust and usability in cybersecurity applications. This interpretability enables security analysts to better understand model decisions and supports more effective threat analysis.

Future work can focus on incorporating dynamic behavioural analysis to enhance robustness against advanced and evolving malware. Additionally, integrating adversarial defence strategies and real-time cloud-based deployment can improve scalability and system resilience. Expanding the dataset to include newer malware families can further strengthen the model's generalization capability.

Overall, the proposed framework contributes to the development of intelligent, transparent, and trustworthy AI-driven solutions for mobile security.

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. 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. 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.