International Journal of Digital Crime and Forensics
Volume 18 • Issue 1 • January-December 2026

Research on Deep Learning-Based Android Malware Detection Systems

Xixiang Yin
https://orcid.org/0009-0008-8662-1639
AnHui Business College, China
Received: September 22nd, 2025 | **Accepted:** January 30th, 2026

ABSTRACT

To combat evolving Android malware, this paper proposed a lightweight deep learning detection system leveraging Drebin, AndroZoo Lite, and Canadian Institute for Cybersecurity MalDroid 2020 datasets. The approach fused static (permissions, call graphs) and dynamic (behavior logs) features via a hybrid convolutional neural network-Transformer model with bidirectional information flow for end-to-end training. To meet mobile device constraints, joint optimization through network pruning, quantization, and attention distillation was applied. Evaluated via five-fold cross-validation, the method outperformed baselines (support vector machine, long short-term memory, BERTroid (BERT-based Android Malware Detection Model), convolutional neural network-Vision Transformer) in precision, recall, F1, area under the curve, and inference latency, achieving high accuracy with low delay. It remains robust against polymorphic and obfuscated variants. Error analysis reveals the critical impact of feature fusion weights on decision-making, offering insights for real-time mobile threat defense.

KEYWORDS

Deep Learning, Android, Malware, Detection System

INTRODUCTION

With the in-depth penetration of the mobile internet, Android, leveraging its open-source nature and wide device compatibility, dominates the global mobile operating system market, with third-party apps in its ecosystem growing exponentially. However, its open permission mechanism and app store fragmentation have become a breeding ground for malware (Huang et al., 2021). In recent years, Android malware has become more concealed and mutates faster (Tam et al., 2017; Wei et al., 2017); attackers use code obfuscation, dynamic loading, and multistage encryption to evade detection, while malicious behaviors, like permission abuse, backdoor implantation, and privacy theft, have evolved into chained triggering and remote control, threatening user data security, device stability, and even critical information infrastructure. Thus, building an efficient and accurate Android malware detection system is crucial for safeguarding the mobile ecosystem and user rights, with high research and application value.

Current Android malware detection methods cover multiple technical paths (Dhalaria & Gandotra, 2021). Traditional signature-matching methods compare samples against predefined malicious code feature libraries for low-latency detection but fail to handle unknown malware and variants. Shallow machine learning methods use hand-designed features (e.g., feature hashing) with algorithms, like support vector machine (SVM), to build models, working in specific scenarios. With

DOI: 10.4018/IJDCF.401332

deep learning development, deep model-based solutions have become a focus: Convolutional neural networks (CNNs), good at local feature capture, analyze static features (e.g., Dalvik Executable (DEX) byte streams, control flow graphs) from decompiled Android Package Kits (APKs) or convert them into grayscale images; recurrent neural networks/long short-term memory (LSTM) networks process time-series data (e.g., syscall sequences) for dynamic behavior analysis; models, like Vision Transformer (ViT) and BERTroid(BERT-based Android Malware Detection Model), introduce attention mechanisms to build long-range feature dependencies and improve accuracy.

Yet, existing methods have universal flaws that hinder real-time mobile protection. Technically, single-modal solutions are limited: Static detection is vulnerable to obfuscation/packing and misses dynamic malicious behaviors (Pan et al., 2020); dynamic detection relies on sandboxes and fails to cover all trigger scenarios (Gajrani et al., 2015); and multimodal solutions struggle with poor feature coordination and information redundancy/loss. Performance-wise, higher accuracy of deep models comes with sharply increased parameters and complexity—e.g., BERTroid has 110 MB parameters and over 500 ms inference latency, while LSTM and CNN-ViT face similar issues—making deployment on resource-constrained mobile terminals difficult. Additionally, deep models' "black-box" nature leads to poor interpretability, lacking transparent decision bases for security auditing and traceability (Mohale & Obagbuwa, 2025).

These problems stem from the contradiction between technical characteristics and application needs. There is an inherent tension between "accuracy" and "speed": Better accuracy requires complex models and multimodal features, increasing computation; mobile hardware, however, demands lightweight, low-latency models (Chandran et al., 2025). Moreover, deep models are poorly matched with Android malware features—CNNs fail to build long-range logical connections of malicious behaviors, recurrent neural networks have redundant parameters and slow convergence, and Transformers' self-attention complexity grows quadratically with sequence length. Meanwhile, public datasets have inconsistent labels, unbalanced family distribution, and no unified evaluation standards, hindering horizontal comparison of research results (Dahiya et al., 2025).

To address these issues, this paper proposes a deep learning-based Android malware detection system integrating static and dynamic features. Based on Drebin (Arp et al., 2014), AndroZoo, and CIC-MalDroid 2020 datasets(Canadian Institute for Cybersecurity MalDroid), it builds a three-parallel lane feature extraction framework: Static lane obtains permission vectors and grayscale images via APK parsing/decompilation; dynamic lane captures behavioral logs with a self-developed sandbox; and network lane extracts traffic statistical features. A lightweight hybrid CNN-Transformer serves as the core classifier, combining CNN's local perception and Transformer's long-range dependency modeling to map multimodal features to a unified semantic space. Joint optimization (pruning, quantization, and attention distillation) reduces model size and latency while ensuring accuracy. End-to-end training via a bidirectional information flow framework and real-time feedback from the metrics eval module enhance robustness against variants and obfuscation. Five-fold cross-validation compares it with SVM, LSTM, BERTroid, and CNN-ViT in precision, recall, F1, area under the curve (AUC), and latency, verifying its superiority in real-time mobile protection and providing a new approach to the "accuracy-speed-deployability" dilemma. Beyond technical performance, the proposed system also provides forensic value for digital investigations. By combining multisource evidence, e.g., static code structures, behavioral traces, and network interactions, the detection results can serve as verifiable indicators in digital forensic workflows. The model's interpretability and attention-based attribution facilitate the identification of malicious code segments, network endpoints, and behavioral triggers, which can be used to reconstruct attack timelines, attribute threats to specific malware families, and support evidentiary reporting in judicial proceedings. Thus, this research not only enhances detection capability but also strengthens the evidentiary foundation for digital crime analysis.