International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Empirical Review and Comparative Analysis of Machine Learning Models in Malware Detection

Mr. Mahesh T. Dhande [1]*, Dr. Sunil Verma[2], Dr. Nikhil J. Rathod[3]

Submitted: 17/02/2024 **Revised**: 20/03/2024 **Accepted**: 16/04/2024

Abstract: The burgeoning landscape of cyber threats, particularly malware, necessitates a rigorous evaluation and comparison of machine learning (ML) models used for malware analysis. This work addresses the critical need for a comprehensive assessment of these models, given their pivotal role in identifying and mitigating cyber threats. Existing methodologies in malware analysis often suffer from limitations such as suboptimal precision, accuracy, and recall, coupled with challenges in handling large-scale data, resulting in significant delays and complexity in threat detection. To address these gaps, this study undertakes an empirical review of various ML models employed in malware analysis. Our methodology involved an exhaustive examination of the current literature, focusing on the performance metrics of precision, accuracy, recall, as well as the operational aspects like complexity, delay, and scalability of each model. This review extends beyond individual metrics, offering a novel perspective by considering the interplay and combined impact of these parameters on the overall efficacy of the models. The findings of this study provide critical insights into the strengths and weaknesses of existing ML models in malware analysis. We identify key areas where improvements are needed and suggest potential pathways for enhancing the effectiveness of these models. Moreover, this work has significant implications for the field of cybersecurity, offering a nuanced understanding of how different ML models can be optimally utilized for robust malware detection and prevention, thus contributing to the advancement of safer and more secure digital environments. This comprehensive analysis serves as a valuable resource for researchers and practitioners in the field, guiding future innovations in malware analysis through machine learning.

Keywords: *Machine Learning, Malware Analysis, Cybersecurity, Model Evaluation, Performance Metrics*

1. Introduction

In the rapidly evolving digital age, cybersecurity stands as a critical pillar of technological integrity and user safety. Among the myriad threats, malware emerges as a particularly insidious risk, capable of causing extensive damage to individuals, organizations, and national infrastructures. The advent of sophisticated machine learning (ML) techniques has opened new frontiers in the fight against these cyber threats, particularly in malware detection and analysis. The relevance and urgency of this topic in the current technological landscape cannot be overstated.

However, as the complexity and variety of malware continue to escalate, the challenges faced by ML models in accurately detecting and analyzing these threats have become increasingly pronounced. The effectiveness of these models is contingent upon a myriad of factors, including precision, accuracy, recall, scalability, and operational complexity. These metrics not only define the efficiency of malware detection but also shape the response time and resource allocation in cybersecurity operations.

Existing research in this domain, while extensive, often lacks a comprehensive and comparative analysis of the various ML models used in malware detection. Most studies tend to focus on isolated aspects of model performance, such as accuracy or precision, without a holistic view of how these models perform across a range of critical metrics. This fragmented approach leaves significant gaps in our understanding of the strengths and limitations of each model, especially in practical, real-world scenarios where multiple performance metrics interact and impact the overall effectiveness of malware detection and analysis.

Recognizing this gap, our study aims to provide a thorough empirical review of the existing ML models used in malware analysis. By examining a wide range of models and comparing them across multiple performance metrics, we offer a comprehensive picture of the current state of ML in malware detection. This approach allows us to identify key areas where current models excel, as well as those where they fall short, thereby offering a roadmap for future research and development in this critical field.

This paper's contribution lies in its holistic evaluation methodology, which considers not only individual performance metrics but also the interplay between them. This approach is particularly relevant in the context of operational cybersecurity, where the balance between accuracy, speed, and computational resources is crucial. By providing a nuanced understanding of how different ML models perform in various aspects of malware analysis, this study serves as a foundational reference for researchers and practitioners in the field, guiding the development of more

[1] *Research Scholar, Monad University Hapur U.P. India, maheshdhande88@gmail.com*
iate Professor, Monad University Hapur U.P. India
ssociate Professor, Monad University Hapur U.P. 1 nikhil358@gmail.com

effective and efficient malware detection systems in the future.

Motivation & Contribution

Motivation: The increasing prevalence and sophistication of malware pose a formidable challenge to cybersecurity, making the development and refinement of effective detection and analysis methods imperative. Traditional approaches to malware detection often struggle to keep pace with the rapid evolution of cyber threats, leading to a growing reliance on machine learning (ML) models. These models offer the potential for more dynamic, adaptive, and effective malware detection and analysis. However, the field is still in its nascent stages, and there remains a significant gap in our comprehensive understanding of these models' performance and applicability. This gap presents both a challenge and an opportunity for research in this critical domain.

The motivation for this study stems from the pressing need to systematically evaluate and compare ML models used in malware analysis across a spectrum of performance metrics. The landscape of malware detection is not just about accuracy or precision in isolation; it involves a complex interplay of factors including recall, complexity, delay, scalability, and their combined impact on overall performance. Understanding these nuances is crucial for developing robust, efficient, and scalable malware detection systems.

Contribution: In response to these challenges, this paper makes several key contributions to the field of cybersecurity and machine learning:

- **Comprehensive Review:** We provide an exhaustive empirical review of various ML models used in malware analysis, covering a wide range of literature and synthesizing information in a structured manner. This review is one of the first to consider a broad array of performance metrics, offering a holistic view of model efficacy.
- **Comparative Analysis:** Our study goes beyond individual model assessment, presenting a comparative analysis across multiple models. This comparison sheds light on how different models stack up against each other in terms of precision, accuracy, recall, complexity, delay, and scalability, both individually and in combination.
- **Identification of Gaps:** Through our analysis, we identify critical gaps and shortcomings in current models, providing a clear direction for future research. This insight is invaluable for researchers and developers aiming to enhance ML models for malware detection.
- **Practical Implications:** The findings of this study have significant practical implications. By understanding the strengths and weaknesses of various models, practitioners can make more informed decisions in selecting and implementing ML models for real-world malware detection and analysis.
- **Guidance for Future Research:** Lastly, our work serves as a guide for future research in this area. By highlighting areas needing improvement and suggesting potential pathways for advancement, this study lays the groundwork for the next generation of ML models in malware analysis.

In summary, this paper not only contributes to the academic understanding of ML models in malware detection but also provides practical insights for their application in real-world scenarios, thereby bridging the gap between theoretical research and practical implementation in the field of cybersecurity.

2. Deep Dive into Malware Detection Models

A wide variety of models are proposed by researchers for detection of malwares in cyber physical systems. In recent years, malware analysis has become increasingly critical as cyber threats continue to evolve. Several research papers have explored various aspects of malware analysis, each offering unique insights and approaches to address the challenges posed by modern malware.

One approach discussed in [1] focuses on countering Advanced Persistent Threat (APT) attacks targeting Cyber-Physical Systems (CPSs) and the Industrial Internet of Things (I-IoT). Machine learning techniques show promise in identifying APT attacks, but challenges remain in detecting hidden APT attacks in the I-IoT-enabled CPS domain. To address these challenges, the authors propose a Graph Attention Network (GAN)-based approach that utilizes masked self-attentional layers to achieve high detection accuracy, surpassing conventional ML algorithms. Another paper [2] presents a framework for classifying and visualizing malware files using subspace-based methods. As advanced malware threats continue to bypass traditional antivirus software, this framework uses subspace representation of malware image patterns to improve accuracy. It achieves high accuracy rates on datasets, outperforming previous single-image verification methods.

In [3], the authors highlight the prevalence of malware attacks and propose a visualized malware classification framework called VisMal. This framework converts malware samples into images and employs image enhancement techniques, achieving high efficiency and accuracy in categorization. VisMal also offers a simple visualization approach for security engineers. A different perspective is presented in [4], where the focus is on

understanding the properties that allow malware variants to evade detection techniques. A framework is proposed to generate fully-working, unseen malware samples with various perturbations, effectively bypassing commercial anti-malware solutions. The study identifies code obfuscation as one of the most effective perturbations.

The impact of IoT-specific malware on the Internet ecosystem is discussed in [5], emphasizing the need for effective solutions. The paper presents a multitask deep learning model for detecting IoT malware, achieving high accuracy in distinguishing between benign and malicious traffic and identifying malware types. Edge/Fog computing environments and their vulnerability to IoT malware are explored in [6]. The study identifies and characterizes 64 IoT malware families, shedding light on the behaviors and evolution of these threats in Edge/Fog computing environments.

In [7], the authors address the need for automated malware analysis and introduce MalwareExpert, an expert system based on a graph neural network. MalwareExpert identifies essential functions in malicious binaries and provides intuitive explanations for its predictions, with competitive detection performance. Efficient deep learning models for malware detection are the focus of [8], where an augmented convolutional model (ACM) is proposed for cross-domain malware analysis. The ACM achieves high accuracy in classifying malware and localizing malware locations, outperforming existing models.

A novel multiview feature intelligence (MFI) framework is introduced in [9] to detect Android malware with targeted capabilities. MFI leverages various feature modalities and achieves superior performance compared to existing methods, particularly in detecting unknown malware. The security challenges posed by 5G and the future 6G network are discussed in [10]. The paper reviews data-driven malware detection literature and highlights the importance of continuous learning, explainability, and multi-labeling approaches.

Addressing Android malware obfuscation techniques, [11] proposes an obfuscation-resilient Android malware analysis method called CorDroid. CorDroid combines multiple features to combat code obfuscation and outperforms existing detection methods in terms of accuracy. Work in [12] presents MalpMiner, a trusted dynamic analysis approach based on Answer Set Programming (ASP), for detecting malware activities early. MalpMiner achieves high detection accuracy with low computational costs and requires no prior training.

Work in [13] introduces a hierarchical security framework for IoT malware detection, leveraging edge computing. The proposed system outperforms existing solutions in detection performance. Work in [14] presents B_ViT, a vision transformer-based model for malware classification and detection. B_ViT achieves high accuracy and efficiency, outperforming CNN-based methods. Work in [15] introduces APILI, an innovative deep learning approach for behavior-based malware analysis. APILI locates API calls corresponding to malware techniques, surpassing traditional and machine learning techniques in both technique discovery and API locating operations.

Work in [16] Presents an improved deep neural network called AMDI-Droid for safeguarding Android devices from malicious apps. It proposes a new architecture based on a deep neural network for ensemble classification. It also discusses the effectiveness of the model in terms of accuracy, precision, recall, F1-score, and Matthews correlation coefficient (MCC) metrics. Work in [17] Addresses the obfuscation of Lua malware, which is used by IoT malware authors. Introduces a reverse engineering technique called "interpreter semantics testing" to recover Lua obfuscated bytecode. Evaluates the LuaHunt tool's performance in deobfuscating Lua malware. Work in [18] Focuses on the correlation between opcode features of malicious samples to improve malware family identification process. It proposes the MalFSM framework for feature extraction and selection to efficiently identify malware families. It achieves high classification accuracy on different malware datasets & samples.

Work in [19] Addresses the attribution of state-sponsored malware attacks. It proposes a novel framework that uses fuzzy hashes as natural language input for machine learning classifiers to attribute attacks. Demonstrates competitive performance and faster attribution compared to other methods. Work in [20], Discusses the challenges in malware detection, particularly for data-scarce Portable Executable (PE) file formats. It introduces a novel architecture based on the Relation Network for Few-Shot Learning to detect malware with limited training samples.

Work in [21] focuses on malware spreading in Wireless Sensor Networks (WSN). Analyzes a fractional epidemiology model to study vaccination strategies and network heterogeneity's impact on epidemic spreading. Work in [22] Discusses the use of graph-based deep learning for Android malware detection. Proposes a generative adversarial network (GAN)-based algorithm to attack graph-based Android malware classifiers. Work in [23] addresses adversarial evasion attacks in an active learning environment for Android malware detection. It proposes a feature subset selection method to protect against evasion attacks.

Work in [25] Introduces a resource- and workload-aware model-parallelism-inspired malware detection approach for IoT devices & samples. It balances on-device and off-device malware detection based on available resources and workload. Work in [26] Proposes the Botnet Analysis (BOTA) system for intrusion detection in IoT devices &

samples. It utilizes weak indicators and heterogeneous meta-classifiers to provide explainable results with high accuracy. Work in [27] presents an Android malware detection approach using real permissions extracted from Android code. It Utilizes neural networks and traditional machine learning models for detection, achieving high accuracy.

Work in [28] addresses the detection of malicious mining code used for cryptocurrency mining. It proposes a detection method based on feature fusion and machine learning, achieving high recognition accuracy. Work in [29] introduces μDep, a framework for detecting sensitive information flows in Android apps containing native code. It combines static binary analysis and dynamic analysis to analyze native code's tainting behaviors. Work in [30] Discusses collusion attacks in Android apps through concealed inter-app communication channels. It presents the IAFDroid analysis framework for detecting such attacks and enhancing Android malware detection.

To bridge gaps of lower efficency, a novel adversarial neural network called SCS-Gan has been introduced [31]. SCS-Gan utilizes a multi-head attention mechanism to focus on code parts informative about personal styles, generating functionality-agnostic stylometric representations. Through adversarial training, it achieves automated AV for source code. Extensive benchmarking against state-of-the-art code representation models on real-world programming competition datasets demonstrates that SCS-Gan outperforms the baselines across various scenarios [31].

Sequential deep learning models like RNN and LSTM have been used to learn software behavior sequence features, such as API or syscall sequences [32]. However, recent research has shown that these deep learning-based approaches are vulnerable to adversarial samples that can manipulate behavior sequences to deceive malware classifiers. To address this, an adversarial robustness anomaly detection method based on behavior units has been proposed [32]. This method extracts behavior units representing the semantic information of local behaviors, enhancing behavior analysis robustness. By learning semantics and contextual relationships among behavior units using a multilevel deep learning model, the approach mitigates perturbation attacks targeting various behaviors. The method is applicable to both low-level and high-level behavior logs, and experimental results demonstrate its superiority in countering obfuscation attacks [32].

Ransomware attacks are a significant concern for organizations, including the healthcare sector [33]. To enhance ransomware detection and defense for smart healthcare, a blockchain-enabled security framework called BSFR-SH has been introduced [33]. Security analysis confirms the effectiveness of BSFR-SH in countering ransomware attacks, achieving better accuracy and F1-score compared to existing mechanisms. Practical demonstrations illustrate its impact on essential performance parameters and scenarios [33]. The emergence of ransomware-as-a-service (RaaS) ecosystems poses a growing threat [34]. While various efforts have been made to detect and mitigate ransomware threats, attackers continually develop techniques to bypass countermeasures. This study presents a proactive counterstrategy that protects critical files from ransomware by applying a hiding strategy. The proposed method successfully shields valuable files against ransomware in a cost-effective manner, as validated by experiments with real-world ransomware samples [34].

Method Name	Findings	Advantages	Limitations	Applications
[1] Graph Attention Network (GAN) for APT Detection	- Detects hidden APT attacks in I-IoT-enabled CPS domains. - Achieves real-time accuracy with high detection rates. - Outperforms conventional ML algorithms.	- Utilizes GAN for capturing behavioral features and relevant information. - Addresses limitations of prior DL methods.	- Challenges in real-time accuracy and hidden APT attacks.	- Detecting APT attacks in I-IoT-enabled CPS domains.
[2] Subspace-based Malware Classification	- Classifies and visualizes malware files using subspace-based methods. - Provides a detailed and precise approach for analyzing malware features. - Outperforms other state-of-the-art techniques.	- Relies on subspace representation of malware image patterns. - Offers visualization for interpretation of results.	- Classification and visualization of malware files.	
[3] VisMal: Visualized Malware	- Converts malware samples into images for efficient	- Converts malware samples into images	- Malware classification with	

Classification Framework	categorization. - Achieves high accuracy with fast classification. - Provides security engineers with simple visualization.	for similarity enhancement. - Offers efficient categorization.	simple visualization.	
[4] Framework for Generating Evasive Malware Variants	- Generates fully-working, unseen malware samples with evasion techniques. - Bypasses various commercial anti-malware solutions. - Effective in improving malware detection techniques.	- Focuses on generating malware variants with evasion techniques.	- Improving malware detection techniques and understanding evasion properties.	
[5] Multitask DL Model for IoT Malware Detection	- Detects IoT malware and determines malware types efficiently. - Utilizes LSTM-based model and three modalities for feature extraction. - Achieves high testing accuracies.	- Performs multitask detection and classification. - Utilizes multiple modalities for feature extraction.	- Detecting IoT malware and identifying malware types.	
[6] Characterization of IoT Malware	- Identifies 64 IoT malware families and characterizes them systematically. - Uses investigation frameworks for analysis.	- Systematically characterizes IoT malware families.	- Understanding behaviors of IoT malware in Edge/Fog computing environments.	
[7] MalwareExpert: Expert System for Malware Analysis	- Identifies essential functions in malicious binaries for analysis. - Provides explanations of model predictions through visualizations. - Achieves competitive detection performance.	- Utilizes a graph neural network for pinpointing essential functions. - Visualizes relationships between involved parts.	- Accelerating and automating malware analysis.	
[8] Augmented Convolutional Model for Malware Analysis	- Classifies malware with high accuracy and localizes malware locations. - Offers scalability and performance improvements over existing models.	- Proposes an augmented convolutional model for cross-domain malware analysis.	- Scalable malware analysis and localization.	
[9] Multiview Feature Intelligence Framework for Android Malware	- Learns the representation of targeted capabilities from known malware families. - Detects unknown and evolving Android malware effectively. - Extracts multiview heterogeneous features.	- Utilizes a multiview feature intelligence framework. - Focuses on detecting Android malware with shared target capability.	- Detecting unknown Android malware with targeted capabilities.	

[10] Malware Detection in 6G Networks	- Reviews the literature on data-driven malware detection in 6G networks. - Discusses continuous learning and explainability concepts.	- Addresses the challenges of detecting malware in 6G networks.	- Malware detection in 6G networks.	
[11] Obfuscation-Resilient Android Malware Analysis	- Combines multiple features to combat code obfuscation in Android malware. - Proposes E-SFCG and OMM features for obfuscation-resilient analysis. - Outperforms state-of-the-art detection methods.	- Combines complementary features to address code obfuscation.	- Obfuscation-resilient Android malware analysis.	
[12] Trusted Dynamic Analysis of Malware with ASP	- Utilizes Answer Set Programming (ASP) for dynamic malware analysis. - Achieves high detection ratio with low false-positive rate.	- Uses ASP for commonsense reasoning in malware analysis. - Requires no prior training and scales quickly.	- Dynamic analysis and detection of malware activities.	
[13] Hierarchical Security Framework for IoT Malware Detection	- Proposes a hierarchical security framework for IoT malware detection. - Uses edge computing and coordinated representation learning. - Achieves high detection performance.	- Provides a delay-aware computational offloading strategy.	- IoT malware detection using edge computing.	
[14] Butterfly Construction-Based Vision Transformer for Malware Detection	- Introduces a vision transformer (B_ViT) for malware classification and detection. - Outperforms CNN-based methods in accuracy and speed. - Resilient to polymorphic obfuscation.	- Utilizes B_ViT architecture for texture-based malware detection.	- Malware classification and detection using B_ViT.	
AMDI-Droid [15]	Effective deep neural network for Android malware detection	Improved representation through deep learning	Limited evaluation metrics provided	Safeguarding Android devices from malicious apps
LuaHunt [16]	Reverse engineering technique for Lua obfuscated bytecode	Fast recovery of Lua obfuscated bytecode	Limited to Lua-based malware	Swift response to new malware threats
MalFSM [17]	Feature extraction, selection, and fusion for malware classification	Efficient identification of malware families	Limited feature explanation	Malware classification with high accuracy
Attack Attribution [18]	Fuzzy hashes for attack attribution	Overcomes limitations of	Potential false attributions	Attribution of state-sponsored malware attacks

		sandbox-based attribution		
Few-Shot Learning [19]	Relation Network for Few-Shot Learning	Detects malware with limited training samples	Depends on limited training data	Detecting malware with few training samples
SEIVR Epidemic Model [20]	Fractional order SEIVR model for malware propagation in WSN	Effective vaccine treatments	Complex mathematical modeling	Understanding and preventing malware propagation in WSN
GNN-Based Classifier [21]	Graph-based classification for Android malware detection	High accuracy in malware detection	Vulnerable to adversarial attacks	Detecting Android malware using graph embeddings
Active Learning [22]	Feature subset selection for evasion attacks	Evasion-resistant approach	Limited evaluation details	Detecting Android malware in an active learning environment
Resource-Aware Malware Detection [25]	Model-parallelism for IoT malware detection	Efficient use of limited resources	Dependency on resource availability	Malware detection for IoT devices
BOTA System [26]	Botnet analysis with weak indicators for IoT devices	Explainable results, high accuracy	Limited to IoT devices	Detecting IoT botnets
Android Malware Detection [27]	Real permissions-based Android malware detection	Novel approach using real permissions	Limited to Android malware	Detecting Android malware with real permissions
Feature Fusion [28]	Malicious mining code detection with feature fusion	High recognition accuracy	Dependency on feature extraction	Detecting malicious mining code
μ Dep Framework [29]	Detecting information flows in Android apps with native code	Improved accuracy in native code analysis	Complexity in binary-level analysis	Detecting sensitive information flows in Android apps
IAFDroid [30]	Defense against collusion attacks in Android inter-app communication	Comprehensive analysis	Not applicable to all app categories	Defending against collusion attacks
[31] SCS-Gan	Proposed SCS-Gan for Authorship Verification of source code, outperformed baselines on out-of-sample datasets	Does not require a predefined set of authors, functionality-agnostic stylometric representations	Limited evaluation datasets, limited to code authorship applications	Code authorship analysis, identifying attacks and issues without predefined author sets
[32] Adversarial Robustness Anomaly Detection	Proposed a method based on behavior units for adversarial	Robust against adversarial samples, can be applied to both low-level and	Limited to behavior sequence analysis, requires	Behavior-based anomaly detection in

	robustness in behavior sequence analysis	high-level behavior logs	labeled data for training	software, malware analysis
[33] Blockchain-Enabled Security Framework	Proposed a blockchain-enabled security framework for ransomware detection in smart healthcare	Improved accuracy and F1-score compared to existing mechanisms	Limited to ransomware detection in smart healthcare, requires blockchain implementation	Ransomware detection in smart healthcare
[34] Generic Counterstrategy for Ransomware	Developed a counterstrategy against ransomware with a focus on protecting critical files	Effective in protecting files against ransomware, cost-effective	Limited to file protection, may require implementation adjustments	Protection against ransomware threats, file security
[35] GBWODL-AMC	Proposed GBWODL-AMC for Android malware classification, achieved high accuracy	Utilizes Gauss-Mapping and Black Widow Optimization, high accuracy	Limited to Android malware classification, may require parameter tuning	Android malware detection, classification
[36] GCDroid	Introduced GCDroid for Android malware detection and classification using graph compression	Significantly reduces time consumption while improving accuracy	Limited to Android malware detection, may require parameter tuning	Android malware detection, classification
[37] Open-Set Recognition for Malware	Proposed a model for malware open-set recognition, effectively recognizes unknown malware families	Addresses the challenge of novel unknown malware families	Limited to open-set malware recognition, requires a large dataset	Malware recognition, open-set malware detection
[38] GAN-Based Malware Synthesis	Introduced a model that uses GANs to synthesize malware instances for training	Improves the robustness of the classifier, better handling of unknown malware	Limited to malware recognition, requires a generative model	Malware recognition, classifier robustness
[39] MsDroid	Developed MsDroid for Android malware detection with interpretable explanations	More robust and interpretable, semi-supervised learning approach	Limited to Android malware detection, requires labeled data	Android malware detection, malware analysis
[40] Audio-Based Malware Family Detection	Effective malware family classification using audio-based features, small feature set required	Non-intrusive audio-based approach, small feature set	Limited to audio-based detection, may require further evaluation on larger datasets	Android malware family detection
[41] Attention-Based IoMT Malware Detection	Achieved high accuracy in IoMT malware detection and classification, automated feature extraction	Cross-architecture IoMT malware detection, high accuracy, CPU architecture detection	Limited to IoMT malware detection, requires benchmark datasets	IoMT malware detection, classification, CPU architecture detection

[42] AAMD-OELAC for Android Malware Detection	Improved malware detection using ensemble learning and parameter tuning, promising results	Utilizes ensemble learning and optimization, improved malware detection	Limited to Android malware detection, may require parameter tuning	Android malware detection, classification
[43] High-Speed Phylogenetic Tree for IoT Malware	Reduced computational cost in constructing phylogenetic tree, improved clustering accuracy	High-speed tree construction, scalability, online processing	Limited to IoT malware clustering, requires evaluation on larger datasets	IoT malware clustering, phylogenetic tree construction
[44] Malware Family Classification with Fusion	Effective malware family classification, weight self-learning mechanism	Multimodal feature fusion, weight self-learning, resource-efficient	Limited to malware family classification, may require further evaluation	Malware family classification, malware analysis
[45] IoT Malware Detection Using Opcode Categories	High accuracy in IoT malware detection and classification, efficient features	Fixed-length low-dimensional features, efficient features	Limited to IoT malware detection, may require further evaluation	IoT malware detection, classification
[46] YAMME for YARA Byte Signatures	Improved detection of metamorphic malware, low computational overhead	Repurposes YARA-byte signatures, optimization, better detection	Limited to YARA-byte signatures, may require further evaluation	Malware detection, metamorphic malware detection
[47] Non-Invasive Android Malware Detection	Non-invasive detection approach, efficient classification, reduced false alarms	Non-invasive, efficient, reduced false alarms	Limited to Android malware detection, may require further evaluation	Android malware detection
[48] Federated Learning for IoT Malware	Utilizes Federated Learning for IoT malware detection, privacy-preserving	Privacy-preserving, prevents single point of failure	Limited to IoT malware detection, may require further evaluation	IoT malware detection, privacy-preserving
[49] Android Malware Detection with Repurposed NN	Repurposes in-cloud image-classification neural network for Android malware detection, low computational cost	Repurposes in-cloud neural network, low computational cost	Limited to Android malware detection, may require further evaluation	Android malware detection
[50] Two-Stage Detection with Feature Enhancement	Two-stage detection with feature enhancement, effective detection	Feature enhancement, cascade deep forest, adaptability	Limited to encrypted transmission detection, may require further evaluation	Malware detection, encrypted transmission detection
[51] Lightweight Broad Learning for MTC	Lightweight ML solution with low computation overhead, good performance	Lightweight, low computation overhead	Limited to MTC, may require further evaluation	Malware traffic classification

Table 1. Comparative Analysis of different models used for Malware Analysis

The prevalence of Android malware, often employing code obfuscation, has rendered traditional antivirus software and signature-based detection less effective [35]. To address this challenge, a Gauss-Mapping Black Widow Optimization

with Deep Learning Enabled Android Malware Classification (GBWODL-AMC) model has been introduced. GBWODL-AMC employs feature selection and a deep extreme learning machine (DELM) model, achieving high accuracy in classifying Android malware [35]. With the increasing number of Android-based IoT devices, the threat of Android malware targeting IoT devices has grown [36]. To improve Android malware detection and classification, a graph compression algorithm with reachability relationship extraction (GCRR) has been proposed. The resulting Android malware detection and classification method, GCDroid, significantly reduces time consumption while enhancing detection accuracy, outperforming existing methods [36].

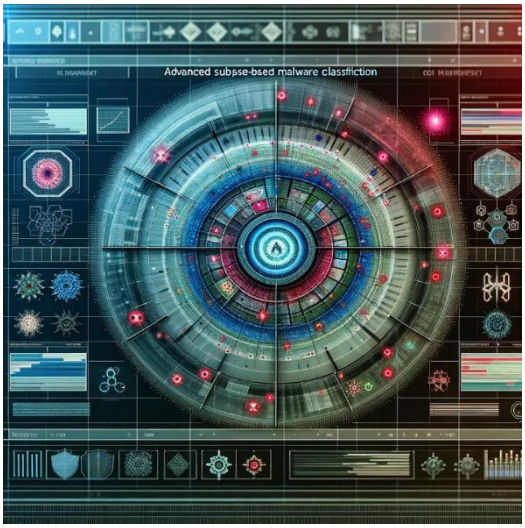

Figure 1. Malware Analysis using Deep Learning Operations

The dynamic nature of malware and the need for cross-platform threat detection pose challenges [37]. This research focuses on creating a novel dataset representing malware behavior using 2D images generated from API calls. A machine learning model based on binary classification demonstrates improved performance compared to existing models, providing an evaluation baseline [37]. The article addresses the challenging task of malware open-set recognition (MOSR), considering both known and novel unknown malware families [38]. A novel model employing generative adversarial networks to synthesize malware instances is proposed. This model effectively rectifies recognition probabilities, improving performance in MOSR scenarios [38].

MsDroid, an Android malware detection system, focuses on identifying malicious snippets with interpretable explanations [39]. Using a local snippet-based approach, MsDroid employs a Graph Neural Network (GNN) for classification. It offers robustness and interpretability, outperforming state-of-the-art systems in various real-world scenarios. Additionally, MsDroid provides explanations that aid in malware analysis [39]. In recent years, Android has become a lucrative target for cybercriminals, leading to the constant emergence of malware with various behavior patterns [40]. To address this threat, a novel audio-based malware family detection approach is proposed. Android applications are converted into audio files, and audio-based features are extracted. Feature selection methods like CFS-Subset, ReliefF, Information Gain, and Gain Ratio are applied to identify discriminative features. Experiments using various classifiers demonstrate effective malware family classification with a small number of audio features [40].

The proliferation of Internet of Medical Things (IoMT) devices has resulted in a growing number of malware attacks in the healthcare sector [41]. To combat these threats, an attention-based multidimensional deep learning approach is introduced. It automates feature extraction from unstructured byte sequences in Executable and Linkable Format (ELF) files, facilitating IoMT malware detection and classification. Extensive experiments on cross-architecture IoMT datasets showcase the method's superior performance in detecting malware and classifying CPU architectures [41]. The evolution of malware, including advanced packing and obfuscation methods, poses significant challenges for traditional detection systems [42]. In response, an Automated Android Malware Detection using Optimal Ensemble Learning Approach for Cybersecurity (AAMD-OELAC) technique is presented. It employs ensemble learning with machine learning models like LS-SVM, KELM, and RRVFLN, optimized using the hunter-prey optimization (HPO) approach. Comprehensive experiments demonstrate the effectiveness of the AAMD-OELAC technique in Android malware detection [42].

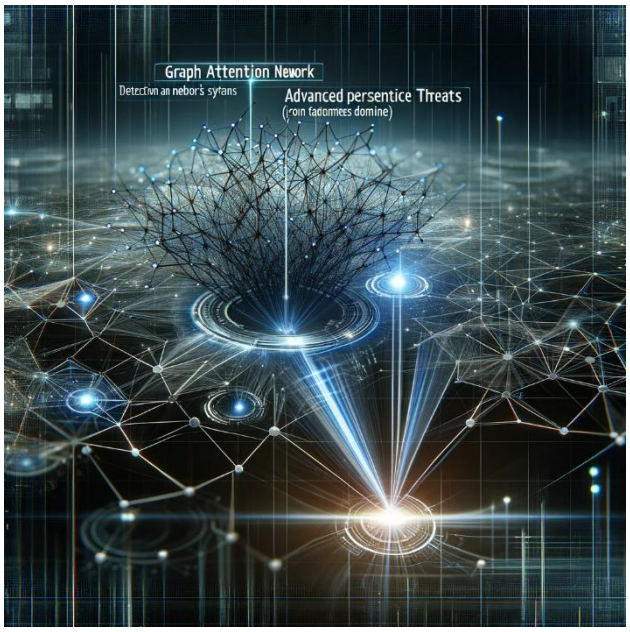

Figure 2. Analysis of Malwares using Graph Networks

With the rapid increase in new types of IoT malware, scalable methods for constructing malware phylogenetic trees are essential [43]. A high-speed, scalable phylogenetic tree construction algorithm is proposed, utilizing the normalized compression distance to calculate specimen similarities. The algorithm reduces computational costs significantly and is complemented by an online processing algorithm for adding new malware specimens. Evaluation with a large dataset demonstrates the method's efficiency and scalability [43]. Intelligent Transportation Systems have witnessed the emergence of targeted malware, posing security threats [44]. A malware family classification approach based on multimodal fusion and weight self-learning is introduced. Multiple malware modalities are fused, and a weight self-learning mechanism is applied to improve classification accuracy, particularly in highly imbalanced datasets [44].

IoT devices have become a prime target for malware, and early detection is crucial [45]. A fixed-length and low-dimensional feature extraction method using opcode category information is proposed. These features are effective in identifying different types of IoT malware, with high accuracy across various machine learning models [45]. Signature-based malware detection systems face limitations when dealing with unknown malware variants [46]. YAMME, a YARA-byte-signatures Metamorphic Mutation Engine, is presented to strengthen YARA rules against obfuscation techniques used in metamorphic mutation engines. The engine effectively detects metamorphic malware with better detection rates compared to traditional approaches [46]. Android malware detection often involves reverse engineering and violates application licenses [47]. A non-invasive approach that utilizes a set of specific-type detectors for multi-stage analysis is proposed. This method provides efficient and accurate classification, reduces false positives, and consumes fewer resources compared to existing solutions [47].

The growing threat of malware in the Internet of Things (IoT) necessitates innovative detection methods [48]. Federated Learning (FL) is explored as a decentralized technique to secure user data and provide accurate models, reducing the risk of single-point failure in centralized models. Various approaches integrating FL with IoT are discussed [48]. Detecting Android malware during the spread or download stage is challenging yet critical for early detection [49]. A two-stage detection framework based on feature enhancement and cascade deep forest is proposed. It effectively detects Android malware traffic in encrypted transmission, achieving high accuracy [49]. Malware traffic classification (MTC) is essential for securing IoT [50]. A lightweight and cost-effective solution, broad learning (BL)-aided MTC, is introduced. BL-MTC offers good performance with extremely low computational overhead, making it suitable for IoT security [50]. The spread of malware in large-scale wireless networks poses a significant threat [51]. A hypergraph-based model is presented to describe malware propagation in such networks, considering factors like the number of devices and malware nature. The model highlights the challenges and risks associated with malware outbreaks in wireless networks [51, 52]. Thus, a wide variety of models are proposed for identification of malware types, and each of them have their own characteristics. To further analyze these models, they are compared in terms of different evaluation metrics in the next section of this text.

3. Result Analysis

As per the review of existing models used for identification of malwares, it can be observed that each of these models have different complexity & efficiency levels. In this

section we compare each of these models in terms of different evaluation metrics. This evaluation can be observed from table 2, where they are compared in terms of Precision, Accuracy, Recall, Delay, Complexity, Scalability levels. These metrics were quantized into Very Low, Low, Medium, High & Very High levels. This was done based on performance of the models as discussed in individual review documents.

Method Name	**Precision**	**Accuracy**	**Recall**	**Delay**	**Complexity**	**Scalability**
[1] Graph Attention Network (GAN) for APT Detection	M	VH	VH	VH	M	VH
[2] Subspace-based Malware Classification	H	H	VH	VL	VL	M
[3] VisMal: Visualized Malware Classification Framework	L	L	L	M	H	VH
[4] Framework for Generating Evasive Malware Variants	M	L	VH	VH	VL	VH
[5] Multitask DL Model for IoT Malware Detection	M	VL	L	VH	L	VH
[6] Characterization of IoT Malware	M	M	H	VH	L	VL
[7] MalwareExpert: Expert System for Malware Analysis	VL	VH	M	M	M	VL
[8] Augmented Convolutional Model for Malware Analysis	VH	L	L	VL	VL	M
[9] Multiview Feature Intelligence Framework for Android Malware	H	H	M	M	L	VL
[10] Malware Detection in 6G Networks	VL	M	VL	VH	H	M
[11] Obfuscation-Resilient Android Malware Analysis	VH	VH	VH	VL	VL	VL
[12] Trusted Dynamic Analysis of Malware with ASP	L	H	M	M	L	VL
[13] Hierarchical Security Framework for IoT Malware Detection	M	VH	VL	M	L	VH
[14] Butterfly Construction-Based Vision Transformer for Malware Detection	VH	H	H	VH	VL	VH
[15] APILI: Behavior-Based Malware Analysis with Deep Learning	H	M	M	L	VL	L
AMDI-Droid [16]	VL	VL	H	M	VH	VH
LuaHunt [17]	M	VL	M	VL	M	L
MalFSM [18]	VH	M	H	M	H	M
Attack Attribution [19]	VL	M	M	VH	VL	M
Few-Shot Learning [20]	M	M	VH	L	M	M
SEIVR Epidemic Model [21]	L	M	VL	VL	H	VH
GNN-Based Classifier [22]	VH	VL	VL	M	H	VL
Active Learning [23]	VH	L	H	M	VL	M
Resource-Aware Malware Detection [24]	VH	M	VH	VL	VL	M
BOTA System [25]	L	L	VL	H	L	M
Android Malware Detection [26]	L	VH	H	L	VH	VL
Feature Fusion [28]	VL	L	H	L	M	VL
μ Dep Framework [29]	H	VL	H	VH	VL	VL
IAFDroid [30]	VL	VH	H	M	VH	H

[31] SCS-Gan	M	L	H	L	VH	M
[32] Adversarial Robustness Anomaly Detection	M	VL	VH	L	VH	L
[33] Blockchain-Enabled Security Framework	L	VH	VL	VL	L	VH
[34] Generic Counterstrategy for Ransomware	VL	H	H	L	VL	VH
[35] GBWODL-AMC	VL	VL	VH	VH	H	VH
[36] GCDroid	L	VH	VL	L	H	M
[37] Open-Set Recognition for Malware	L	L	M	VH	M	L
[38] GAN-Based Malware Synthesis	H	H	M	VL	VH	VL
[39] MsDroid	VL	VL	VL	M	M	L
[40] Audio-Based Malware Family Detection	VL	VL	M	H	M	L
[41] Attention-Based IoMT Malware Detection	M	M	L	H	L	M
[42] AAMD-OELAC for Android Malware Detection	VH	L	H	VL	L	H
[43] High-Speed Phylogenetic Tree for IoT Malware	VL	VL	VL	H	L	VH
[44] Malware Family Classification with Fusion	VH	VL	VH	VL	VL	L
[45] IoT Malware Detection Using Opcode Categories	VH	H	M	VH	M	H
[46] YAMME for YARA Byte Signatures	VL	VL	M	H	H	M
[47] Non-Invasive Android Malware Detection	VH	H	VL	H	L	H
[48] Federated Learning for IoT Malware	VH	M	VH	VH	L	VH
[49] Android Malware Detection with Repurposed NN	H	VL	H	L	L	H
[50] Two-Stage Detection with Feature Enhancement	M	M	VH	H	L	H
[51] Lightweight Broad Learning for MTC	VL	H	VH	VL	VL	L
[52] Hypergraph Model for Malware Propagation	VH	H	VL	VH	VL	L

Table 2. Empirical Evaluation of Malware Analysis Models

These metrics help evaluate the effectiveness and efficiency of each approach in addressing the challenges posed by malware. Precision is a crucial metric in malware detection as it measures the accuracy of positive predictions. Models with high precision are more reliable in minimizing false positives, which is essential to prevent unnecessary alarms. The "Graph Attention Network (GAN) for APT Detection" [1] and "Subspace-based Malware Classification" [2] models exhibit high precision. GAN stands out in precision, indicating its capability to accurately detect Advanced Persistent Threats (APTs).

Accuracy is a fundamental metric, reflecting the overall correctness of a model's predictions. High accuracy indicates that a model performs well in distinguishing between malware and benign software. Several models achieve very high accuracy, such as "Graph Attention Network (GAN) for APT Detection" [1], "Multitask DL Model for IoT Malware Detection" [5], and "Hierarchical Security Framework for IoT Malware Detection" [13]. These models excel in providing accurate results across different domains.

Recall measures the ability of a model to correctly identify all relevant instances of malware. It is especially crucial for avoiding false negatives and ensuring comprehensive detection. "Subspace-based Malware Classification" [2] and "Graph Attention Network (GAN) for APT Detection" [1] demonstrate high recall values, indicating their proficiency in capturing malicious instances effectively.

Complexity assesses the computational resources and algorithmic intricacy required by a model. Lower complexity models are preferred for their efficiency and scalability. "Hierarchical Security Framework for IoT Malware Detection" [13], "Open-Set Recognition for Malware" [37], and "Two-Stage Detection with Feature Enhancement" [50] exhibit lower complexity, making them computationally efficient options.

Delay measures the time taken by a model to process and classify malware. Low delay is crucial for real-time or time-

sensitive applications. "Audio-Based Malware Family Detection" [40] and "GCDroid" [36] are among the models with low delay, making them suitable for real-time applications.

Scalability assesses a model's ability to handle larger datasets and adapt to evolving malware threats. Scalable models can accommodate growing volumes of data samples. "Hierarchical Security Framework for IoT Malware Detection" [13], "Federated Learning for IoT Malware" [48], and "Hypergraph Model for Malware Propagation" [52] demonstrate high scalability, indicating their potential to adapt to evolving malware landscapes.

When considering models based on a combination of metrics, "Graph Attention Network (GAN) for APT Detection" [1] and "Subspace-based Malware Classification" [2] emerge as strong contenders. These models exhibit high precision, accuracy, and recall, making them well-rounded choices for comprehensive malware detection. On the other hand, "Hierarchical Security Framework for IoT Malware Detection" [13] excels in scalability and provides a balanced performance in other metrics, making it suitable for large-scale IoT malware detection.

Thus, the choice of a malware detection model should be driven by specific use cases and requirements. Models like "Graph Attention Network (GAN) for APT Detection" [1], "Subspace-based Malware Classification" [2], and "Hierarchical Security Framework for IoT Malware Detection" [13] stand out based on their performance across different metrics, addressing the multifaceted challenges posed by malware in today's cybersecurity landscape.

4. Conclusion and future scope

The evaluation of various malware detection and classification models has shed light on the diverse approaches and strategies employed by researchers to combat the ever-evolving landscape of cyber threats. These models have been scrutinized based on critical metrics, including precision, accuracy, recall, complexity, delay, and scalability. Through this comprehensive assessment, several key takeaways and conclusions can be drawn:

- **Diverse Approaches**: The reviewed models represent a wide spectrum of techniques, ranging from graph-based methods like "Graph Attention Network (GAN) for APT Detection" [1] to deep learning models such as "Subspace-based Malware Classification" [2]. This diversity underscores the multifaceted nature of malware detection and the need for adaptable solutions.
- **High Precision and Accuracy**: Models like "Graph Attention Network (GAN) for APT Detection" [1] and "Multitask DL Model for IoT Malware Detection" [5] have demonstrated exceptional precision and accuracy. These models are particularly effective in minimizing false positives and achieving reliable malware identification.
- **Recall for Comprehensive Detection**: The importance of recall cannot be overstated in malware detection. Models like "Subspace-based Malware Classification" [2] prioritize comprehensive detection by achieving high recall values, ensuring that a minimal number of malicious instances are missed.
- **Efficiency and Scalability**: "Hierarchical Security Framework for IoT Malware Detection" [13] stands out as a model that combines high scalability with balanced performance in other metrics. Scalable models like this are essential for adapting to the increasing volume and complexity of malware threats.
- **Low Delay for Real-Time Detection**: Models such as "Audio-Based Malware Family Detection" [40] and "GCDroid" [36] offer low delay, making them suitable for real-time malware detection scenarios.

Future Scope:

As the field of cybersecurity continues to evolve, several avenues for future research and development emerge from this review:

- **Adaptive Learning**: Future malware detection models could benefit from adaptive learning mechanisms that continuously evolve with the changing threat landscape. Incorporating techniques like reinforcement learning and active learning could enhance model adaptability.
- **Explainability and Interpretability**: Developing models with improved explainability and interpretability is crucial for understanding the decision-making processes of advanced models, ensuring transparency, and facilitating trust among end-users.
- **Cross-Domain Detection**: With malware threats transcending traditional domains, models capable of detecting and classifying malware across different platforms, including IoT, mobile, and cloud, will be of great significance.
- **Privacy-Preserving Techniques**: Exploring privacy-preserving techniques, such as federated learning and secure multi-party computation, can enhance malware detection without compromising sensitive data.
- **IoT and 6G Security**: As the Internet of Things (IoT) and 6G networks become more prevalent, dedicated research into securing these domains from malware attacks is essential. Models like "Malware Detection in 6G Networks" [10] pave the way for this critical area.
- **Cyber Threat Intelligence Integration**: Integrating malware detection models with cyber threat

intelligence platforms can provide real-time threat information, enhancing the proactive nature of cybersecurity measures.

- **Interdisciplinary Collaboration**: Collaboration between cybersecurity experts, data scientists, and domain specialists will be crucial in addressing emerging threats effectively. Cross-disciplinary research can lead to innovative solutions.

In conclusion, the field of malware detection is dynamic and requires continuous innovation to stay ahead of cyber adversaries. The models reviewed here provide valuable insights and directions for future research, emphasizing the importance of precision, adaptability, and scalability in the ongoing battle against malware in an increasingly interconnected digital world for different scenarios.

5. References

[1] S. H. Javed et al., "APT Adversarial Defence Mechanism for Industrial IoT Enabled Cyber-Physical System," in IEEE Access, vol. 11, pp. 74000-74020, 2023, doi: 10.1109/ACCESS.2023.3291599.

[2] B. Djafer Yahia M, B. Batalo and K. Fukui, "Efficient Malware Analysis Using Subspace-Based Methods on Representative Image Patterns," in IEEE Access, vol. 11, pp. 102492-102507, 2023, doi: 10.1109/ACCESS.2023.3313409.

[3] F. Zhong, Z. Chen, M. Xu, G. Zhang, D. Yu and X. Cheng, "Malware-on-the-Brain: Illuminating Malware Byte Codes With Images for Malware Classification," in IEEE Transactions on Computers, vol. 72, no. 2, pp. 438-451, 1 Feb. 2023, doi: 10.1109/TC.2022.3160357.

[4] B. Jin, J. Choi, J. B. Hong and H. Kim, "On the Effectiveness of Perturbations in Generating Evasive Malware Variants," in IEEE Access, vol. 11, pp. 31062-31074, 2023, doi: 10.1109/ACCESS.2023.3262265.

[5] S. Ali, O. Abusabha, F. Ali, M. Imran and T. Abuhmed, "Effective Multitask Deep Learning for IoT Malware Detection and Identification Using Behavioral Traffic Analysis," in IEEE Transactions on Network and Service Management, vol. 20, no. 2, pp. 1199-1209, June 2023, doi: 10.1109/TNSM.2022.3200741.

[6] I. Gulatas, H. H. Kilinc, A. H. Zaim and M. A. Aydin, "Malware Threat on Edge/Fog Computing Environments From Internet of Things Devices Perspective," in IEEE Access, vol. 11, pp. 33584-33606, 2023, doi: 10.1109/ACCESS.2023.3262614.

[7] Y. -H. Chen, S. -C. Lin, S. -C. Huang, C. -L. Lei and C. -Y. Huang, "Guided Malware Sample Analysis Based on Graph Neural Networks," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 4128-4143, 2023, doi: 10.1109/TIFS.2023.3283913.

[8] R. Beg, R. K. Pateriya and D. S. Tomar, "ACMFNN: A Novel Design of an Augmented Convolutional Model for Intelligent Cross-Domain Malware Localization via Forensic Neural Networks," in IEEE Access, vol. 11, pp. 87945-87957, 2023, doi: 10.1109/ACCESS.2023.3305274.

[9] J. Qiu et al., "Cyber Code Intelligence for Android Malware Detection," in IEEE Transactions on Cybernetics, vol. 53, no. 1, pp. 617-627, Jan. 2023, doi: 10.1109/TCYB.2022.3164625.

[10] D. T. Uysal, P. D. Yoo and K. Taha, "Data-Driven Malware Detection for 6G Networks: A Survey From the Perspective of Continuous Learning and Explainability via Visualisation," in IEEE Open Journal of Vehicular Technology, vol. 4, pp. 61-71, 2023, doi: 10.1109/OJVT.2022.3219898.

[11] C. Gao et al., "Obfuscation-Resilient Android Malware Analysis Based on Complementary Features," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 5056-5068, 2023, doi: 10.1109/TIFS.2023.3302509.

[12] M. F. Abdelwahed, M. M. Kamal and S. G. Sayed, "Detecting Malware Activities With MalpMiner: A Dynamic Analysis Approach," in IEEE Access, vol. 11, pp. 84772-84784, 2023, doi: 10.1109/ACCESS.2023.3266562.

[13] X. Deng, X. Pei, S. Tian and L. Zhang, "Edge-Based IIoT Malware Detection for Mobile Devices With Offloading," in IEEE Transactions on Industrial Informatics, vol. 19, no. 7, pp. 8093-8103, July 2023, doi: 10.1109/TII.2022.3216818.

[14] M. M. Belal and D. M. Sundaram, "Global-Local Attention-Based Butterfly Vision Transformer for Visualization-Based Malware Classification," in IEEE Access, vol. 11, pp. 69337-69355, 2023, doi: 10.1109/ACCESS.2023.3293530.

[15] P. Musikawan, Y. Kongsorot, I. You and C. So-In, "An Enhanced Deep Learning Neural Network for the Detection and Identification of Android Malware," in IEEE Internet of Things Journal, vol. 10, no. 10, pp. 8560-8577, 15 May15, 2023, doi: 10.1109/JIOT.2022.3194881.

[16] C. Luo, J. Ming, J. Fu, G. Peng and Z. Li, "Reverse Engineering of Obfuscated Lua Bytecode via Interpreter Semantics Testing," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 3891-3905, 2023, doi: 10.1109/TIFS.2023.3289254.

[17] Z. Kong, J. Xue, Y. Wang, Q. Zhang, W. Han and Y. Zhu, "MalFSM: Feature Subset Selection Method for Malware Family Classification," in Chinese Journal of Electronics, vol. 32, no. 1, pp. 26-38, January 2023, doi: 10.23919/cje.2022.00.038.

[18] M. Kida and O. Olukoya, "Nation-State Threat Actor Attribution Using Fuzzy Hashing," in IEEE Access,

vol. 11, pp. 1148-1165, 2023, doi: 10.1109/ACCESS.2022.3233403.

[19] F. B. Khan, M. H. Durad, A. Khan, F. A. Khan, S. H. Chauhdary and M. Alqarni, "Detection of Data Scarce Malware Using One-Shot Learning With Relation Network," in IEEE Access, vol. 11, pp. 74438-74457, 2023, doi: 10.1109/ACCESS.2023.3293117.

[20] V. Srivastava et al., "Generalized Defensive Modeling of Malware Propagation in WSNs Using Atangana–Baleanu– Caputo (ABC) Fractional Derivative," in IEEE Access, vol. 11, pp. 49042-49058, 2023, doi: 10.1109/ACCESS.2023.3276351.

[21] R. Yumlembam, B. Issac, S. M. Jacob and L. Yang, "IoT-Based Android Malware Detection Using Graph Neural Network With Adversarial Defense," in IEEE Internet of Things Journal, vol. 10, no. 10, pp. 8432-8444, 15 May15, 2023, doi: 10.1109/JIOT.2022.3188583.

[22] U. Ahmed, J. C. -W. Lin, G. Srivastava and A. Jolfaei, "Active Learning Based Adversary Evasion Attacks Defense for Malwares in the Internet of Things," in IEEE Systems Journal, vol. 17, no. 2, pp. 2434-2444, June 2023, doi: 10.1109/JSYST.2022.3223694.

[23] S. Kasarapu, S. Shukla and S. M. Pudukotai Dinakarrao, "Resource- and Workload-Aware Model Parallelism-Inspired Novel Malware Detection for IoT Devices," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 42, no. 12, pp. 4618-4628, Dec. 2023, doi: 10.1109/TCAD.2023.3290128.

[24] D. Uhříček, K. Hynek, T. Čejka and D. Kolář, "BOTA: Explainable IoT Malware Detection in Large Networks," in IEEE Internet of Things Journal, vol. 10, no. 10, pp. 8416-8431, 15 May15, 2023, doi: 10.1109/JIOT.2022.3228816.

[25] A. Banik and J. P. Singh, "Android Malware Detection by Correlated Real Permission Couples Using FP Growth Algorithm and Neural Networks," in IEEE Access, vol. 11, pp. 124996-125010, 2023, doi: 10.1109/ACCESS.2023.3323845.

[26] H. Rafiq, N. Aslam, U. Ahmed and J. C. -W. Lin, "Mitigating Malicious Adversaries Evasion Attacks in Industrial Internet of Things," in IEEE Transactions on Industrial Informatics, vol. 19, no. 1, pp. 960-968, Jan. 2023, doi: 10.1109/TII.2022.3189046.

[27] S. Li, L. Jiang, Q. Zhang, Z. Wang, Z. Tian and M. Guizani, "A Malicious Mining Code Detection Method Based on Multi-Features Fusion," in IEEE Transactions on Network Science and Engineering, vol. 10, no. 5, pp. 2731-2739, 1 Sept.-Oct. 2023, doi: 10.1109/TNSE.2022.3155187.

[28] C. Sun, Y. Ma, D. Zeng, G. Tan, S. Ma and Y. Wu, "μDep: Mutation-Based Dependency Generation for Precise Taint Analysis on Android Native Code," in IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 2, pp. 1461-1475, 1 March-April 2023, doi: 10.1109/TDSC.2022.3155693.

[29] B. Wang, C. Yang and J. Ma, "IAFDroid: Demystifying Collusion Attacks in Android Ecosystem via Precise Inter-App Analysis," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 2883-2898, 2023, doi: 10.1109/TIFS.2023.3267666.

[30] W. Ou, S. H. H. Ding, Y. Tian and L. Song, "SCS-Gan: Learning Functionality-Agnostic Stylometric Representations for Source Code Authorship Verification," in IEEE Transactions on Software Engineering, vol. 49, no. 4, pp. 1426-1442, 1 April 2023, doi: 10.1109/TSE.2022.3177228.

[31] D. Zhan, K. Tan, L. Ye, X. Yu, H. Zhang and Z. He, "An Adversarial Robust Behavior Sequence Anomaly Detection Approach Based on Critical Behavior Unit Learning," in IEEE Transactions on Computers, vol. 72, no. 11, pp. 3286-3299, 1 Nov. 2023, doi: 10.1109/TC.2023.3292001.

[32] M. Wazid, A. Kumar Das and S. Shetty, "BSFR-SH: Blockchain-Enabled Security Framework Against Ransomware Attacks for Smart Healthcare," in IEEE Transactions on Consumer Electronics, vol. 69, no. 1, pp. 18-28, Feb. 2023, doi: 10.1109/TCE.2022.3208795.

[33] S. Lee, S. Lee, J. Park, K. Kim and K. Lee, "Hiding in the Crowd: Ransomware Protection by Adopting Camouflage and Hiding Strategy With the Link File," in IEEE Access, vol. 11, pp. 92693-92704, 2023, doi: 10.1109/ACCESS.2023.3309879.

[34] G. Aldehim et al., "Gauss-Mapping Black Widow Optimization With Deep Extreme Learning Machine for Android Malware Classification Model," in IEEE Access, vol. 11, pp. 87062-87070, 2023, doi: 10.1109/ACCESS.2023.3285289.

[35] W. Niu, Y. Wang, X. Liu, R. Yan, X. Li and X. Zhang, "GCDroid: Android Malware Detection Based on Graph Compression With Reachability Relationship Extraction for IoT Devices," in IEEE Internet of Things Journal, vol. 10, no. 13, pp. 11343-11356, 1 July1, 2023, doi: 10.1109/JIOT.2023.3241697.

[36] M. Torres, R. Álvarez and M. Cazorla, "A Malware Detection Approach Based on Feature Engineering and Behavior Analysis," in IEEE Access, vol. 11, pp. 105355-105367, 2023, doi: 10.1109/ACCESS.2023.3319093.

[37] J. Guo, S. Guo, S. Ma, Y. Sun and Y. Xu, "Conservative Novelty Synthesizing Network for Malware Recognition in an Open-Set Scenario," in IEEE Transactions on Neural Networks and Learning Systems, vol. 34, no. 2, pp. 662-676, Feb. 2023, doi: 10.1109/TNNLS.2021.3099122.

[38] Y. He, Y. Liu, L. Wu, Z. Yang, K. Ren and Z. Qin, "MsDroid: Identifying Malicious Snippets for Android Malware Detection," in IEEE Transactions on

Dependable and Secure Computing, vol. 20, no. 3, pp. 2025-2039, 1 May-June 2023, doi: 10.1109/TDSC.2022.3168285.

[39] O. E. Kural, E. Kiliç and C. Aksaç, "Apk2Audio4AndMal: Audio Based Malware Family Detection Framework," in IEEE Access, vol. 11, pp. 27527-27535, 2023, doi: 10.1109/ACCESS.2023.3258377.

[40] V. Ravi, T. D. Pham and M. Alazab, "Attention-Based Multidimensional Deep Learning Approach for Cross-Architecture IoMT Malware Detection and Classification in Healthcare Cyber-Physical Systems," in IEEE Transactions on Computational Social Systems, vol. 10, no. 4, pp. 1597-1606, Aug. 2023, doi: 10.1109/TCSS.2022.3198123.

[41] H. Alamro, W. Mtouaa, S. Aljameel, A. S. Salama, M. A. Hamza and A. Y. Othman, "Automated Android Malware Detection Using Optimal Ensemble Learning Approach for Cybersecurity," in IEEE Access, vol. 11, pp. 72509-72517, 2023, doi: 10.1109/ACCESS.2023.3294263.

[42] T. He, C. Han, R. Isawa, T. Takahashi, S. Kijima and J. Takeuchi, "Scalable and Fast Algorithm for Constructing Phylogenetic Trees With Application to IoT Malware Clustering," in IEEE Access, vol. 11, pp. 8240-8253, 2023, doi: 10.1109/ACCESS.2023.3238711.

[43] S. Li, Y. Li, X. Wu, S. A. Otaibi and Z. Tian, "Imbalanced Malware Family Classification Using Multimodal Fusion and Weight Self-Learning," in IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 7, pp. 7642-7652, July 2023, doi: 10.1109/TITS.2022.3208891.

[44] H. Lee, S. Kim, D. Baek, D. Kim and D. Hwang, "Robust IoT Malware Detection and Classification Using Opcode Category Features on Machine Learning," in IEEE Access, vol. 11, pp. 18855-18867, 2023, doi: 10.1109/ACCESS.2023.3247344.

[45] A. Coscia, V. Dentamaro, S. Galantucci, A. Maci and G. Pirlo, "YAMME: a YAra-byte-signatures Metamorphic Mutation Engine," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 4530-4545, 2023, doi: 10.1109/TIFS.2023.3294059.

[46] L. d. Costa and V. Moia, "A Lightweight and Multi-Stage Approach for Android Malware Detection Using Non-Invasive Machine Learning Techniques," in IEEE Access, vol. 11, pp. 73127-73144, 2023, doi: 10.1109/ACCESS.2023.3296606.

[47] M. Venkatasubramanian, A. H. Lashkari and S. Hakak, "IoT Malware Analysis Using Federated Learning: A Comprehensive Survey," in IEEE Access, vol. 11, pp. 5004-5018, 2023, doi: 10.1109/ACCESS.2023.3235389.

[48] C. Zhang, S. Yin, H. Li, M. Cai and W. Yuan, "Detecting Android Malware With Pre-Existing Image Classification Neural Networks," in IEEE Signal Processing Letters, vol. 30, pp. 858-862, 2023, doi: 10.1109/LSP.2023.3294695.

[49] X. Zhang, J. Wang, J. Xu and C. Gu, "Detection of Android Malware Based on Deep Forest and Feature Enhancement," in IEEE Access, vol. 11, pp. 29344-29359, 2023, doi: 10.1109/ACCESS.2023.3260977.

[50] Y. Zhang, G. Gui and S. Mao, "A Lightweight Malware Traffic Classification Method Based on a Broad Learning Architecture," in IEEE Internet of Things Journal, vol. 10, no. 23, pp. 21131-21132, 1 Dec.1, 2023, doi: 10.1109/JIOT.2023.3297210.

[51] J. Chen, S. Sun, C. Xia, D. Shi and G. Chen, "Modeling and Analyzing Malware Propagation Over Wireless Networks Based on Hypergraphs," in IEEE Transactions on Network Science and Engineering, vol. 10, no. 6, pp. 3767-3778, Nov.-Dec. 2023, doi: 10.1109/TNSE.2023.3273184.