SYSTEMATIC LITERATURE REVIEW ON MALWARE DETECTION AND CLASSIFICATION: TYPES, PLATFORMS, MITIGATIONS, LIMITATIONS AND OPEN ISSUES

JOSEPH ADEBAYO OJENIYI, BENITA CHIKODILI NWODO, IDRIS ISMAILA, OLUSANJO OLUGBEMI FASOLA, MOSES DOGONYARO NOEL & SULEIMAN AHMAD
Federal University of Technology, Minna, Nigeria
E-mail: Ojeniyija@futminna.edu.ng **Phone No:** +2347041639013

Abstract
This paper presents a PRISMA 2020–compliant systematic literature review of malware detection and classification studies published between 2023 and 2026. It synthesizes research across malware types, computing platforms, feature representations, detection architectures, and mitigation strategies from major digital libraries. The review identifies a significant shift toward deep learning and transformer-based models, which outperform traditional machine learning in capturing complex behavioral and structural patterns. Graph-based methods improve semantic relationship modeling, while federated learning enables privacy-preserving collaborative detection. Despite these advances, critical challenges persist, including dataset bias, temporal concept drift, adversarial vulnerability, and weak cross-platform generalization. Many studies rely on static datasets and random splits, leading to inflated performance estimates that do not reflect real-world conditions. Explainability, deployment feasibility, and adversarial robustness remain insufficiently addressed, limiting operational adoption in SOC environments. This review proposes a unified taxonomy and future research agenda focused on robustness-aware evaluation, temporal benchmarking, cross-platform generalization, and deployment-ready, adversary-aware detection frameworks.

Keywords: Malware detection; Malware classification; Systematic literature review; Machine learning; Deep learning; Transformer models; Graph neural networks; Adversarial robustness

Introduction

Malware continues to be one of the most persistent and economically damaging threats to modern computing environments. Contemporary malware exhibits advanced characteristics such as polymorphism, metamorphism, fileless execution, and the abuse of legitimate system utilities—commonly referred to as "living-off-the-land" techniques—which significantly reduce the effectiveness of traditional signature-based antivirus systems (Ling *et al.,* 2023; Deldar *et al.,* 2023). These techniques allow malicious code to evade static pattern matching and heuristic rules, resulting in prolonged detection latency and increased exposure to zero-day attacks.

From an operational perspective, these evolutions have placed significant strain on Security Operations Centers (SOCs). High false-positive rates, alert fatigue, and delayed threat attribution are widely reported challenges, particularly in large-scale enterprise and cloud environments (Song *et al.,* 2025). As malware increasingly blends benign and malicious behaviors, purely rule-based detection mechanisms struggle to maintain both precision and recall, motivating the widespread adoption of learning-based detection techniques.

In response to the shortcomings of traditional defenses, the research community has progressively shifted toward machine learning (ML) and, more recently, deep learning (DL) approaches for malware detection and classification. Early ML-based systems relied heavily on manually engineered features such as opcode frequencies, imported libraries, and permission sets. While these approaches improved detection accuracy, they remained vulnerable to feature manipulation and concept drift (Deldar *et al.,* 2023).

Between 2023 and 2026, deep learning architectures—including convolutional neural networks (CNNs), recurrent neural networks (RNNs), transformers, and graph neural networks (GNNs)—have become dominant in the literature due to their ability to learn hierarchical and contextual representations directly from raw or minimally processed data (Alshomrani & Albeshri, 2024; Kim *et al.,* 2025). Transformer-based models, in particular, have gained attention for modeling long-range dependencies in byte sequences and behavioral traces, drawing parallels between malware analysis and natural language processing (Kim *et al.,* 2025).

However, despite impressive empirical results, many proposed models report performance under idealized conditions that do not reflect real-world deployment scenarios. Accuracy-centric evaluations, static train–test splits, and the absence of adversarial threat models often lead to over-optimistic conclusions regarding model robustness and generalizability (Ling *et al.,* 2023).

Modern malware operates across a diverse range of platforms, including Windows (PE), Android (APK), Linux (ELF), Internet of Things (IoT) devices, and document-based formats such as PDF and Microsoft Office files. Each platform exhibits distinct execution models, system interfaces, and attack surfaces, which significantly influence feature extraction and detection strategies (Song *et al.,* 2025).

Despite this diversity, much of the malware detection literature remains platform-specific, with Windows-based malware dominating empirical evaluations. Android malware studies often rely on permission- or API-based features, while IoT malware research frequently focuses on network telemetry due to device constraints (Çıplak *et al.,* 2025). This fragmentation complicates cross-platform generalization and limits the applicability of proposed solutions in heterogeneous operational environments, where SOCs must simultaneously monitor endpoints, servers, mobile devices, and embedded systems.

Recent benchmark efforts, such as holistic and multi-format datasets introduced after 2023, highlight the need for unified evaluation frameworks capable of supporting multiple platforms and classification tasks (Joyce *et al.,* 2025). These developments underscore the growing recognition that siloed detection approaches are insufficient for real-world malware defense.

Several surveys and reviews on malware detection have been published prior to and during the early 2020s. While these studies provide valuable overviews of machine learning and deep learning techniques, many suffer from limitations that reduce their relevance to contemporary threats. First, a significant portion of earlier surveys predate the widespread adoption of transformer architectures and graph-based learning, which have reshaped representation learning in malware analysis since 2023 (Alshomrani & Albeshri, 2024).

Second, many existing reviews lack methodological rigor, often adopting narrative or ad hoc selection strategies rather than systematic protocols such as PRISMA. As a result, selection bias, incomplete coverage, and inconsistent synthesis are common issues (Deldar *et al.,* 2023). Third, prior surveys frequently emphasize algorithmic performance while neglecting adversarial evasion, concept drift, and operational deployment constraints—factors that are critical in real-world security environments (Ling *et al.,* 2023).

Consequently, there is a clear need for an updated, methodologically rigorous systematic literature review that captures recent advances while critically examining their limitations and practical implications.
Based on preliminary analysis of the recent literature, several critical research gaps motivate this systematic review:

1. Robustness and adversarial awareness: Many studies fail to evaluate models under realistic evasion scenarios or clearly define attacker capabilities (Ling *et al.,* 2023).
2. Temporal and distributional generalization: Static evaluation protocols dominate, despite evidence that malware distributions evolve rapidly over time (Joyce *et al.,* 2025).
3. Cross-platform generalization: Limited attention is given to unified detection strategies across heterogeneous platforms (Song *et al.,* 2025).
4. Operational explainability: Explainable AI techniques are often treated as optional add-ons rather than integral components of analyst workflows (Mohammadian *et al.,* 2025).
5. Deployment constraints: Computational cost, latency, and telemetry availability are frequently ignored, reducing real-world applicability.

Addressing these gaps requires not only novel detection techniques but also a comprehensive synthesis of existing work to guide future research and deployment.
This paper makes the following key contributions:

1. A PRISMA 2020–compliant systematic literature review of malware detection and classification research published between 2023 and 2026.
2. A unified taxonomy of malware types and target platforms, highlighting emerging trends and neglected areas.
3. A comparative synthesis of detection and classification techniques, including deep learning, transformer-based, graph-based, and federated approaches.
4. A structured analysis of mitigation strategies addressing adversarial evasion, concept drift, and deployment challenges.
5. An identification of open research issues and a future research agenda aligned with both academic and SOC operational needs.

The remainder of this paper is organized as follows. Section 2 describes the systematic review. Section 3 methodology and study selection process Section 4 reviews detection and classification techniques alongside mitigation strategies. Section 5 discusses limitations, open research challenges, and future directions for malware detection and classification research.

Related Works

Malware detection research published between 2023 and 2026 demonstrates a continued transition from signature-based and rule-driven systems toward learning-based and behavior-centric detection paradigms. Large-scale empirical studies and surveys report that machine learning (ML) and deep learning (DL) models increasingly outperform traditional approaches when evaluated on heterogeneous malware corpora, particularly for previously unseen variants (Maniriho *et al.,* 2024). This transition is driven by the rapid mutation rate of malware families, the commoditization of packing and obfuscation services, and the proliferation of fileless attack techniques that minimize static artifacts.

Recent literature further emphasizes the importance of temporal evaluation protocols, noting that random train–test splits inflate performance due to data leakage across malware variants that share code lineage. Longitudinal evaluations reveal significant performance decay over time, highlighting the impact of dataset aging and the need for drift-aware learning pipelines (Botacin *et al.,* 2021; Joyce *et al.,* 2025). As a result, benchmark design and evaluation methodology have become first-order research concerns alongside model architecture.
Contemporary malware research concentrates on a set of dominant malware categories defined by payload objective and operational behavior. Ransomware remains the most intensively studied category due to its financial impact and operational disruption, with studies analyzing encryption workflows, key management, lateral movement, and command-and-control (C2) behaviors. Trojans, particularly modular loaders and banking trojans, continue to

dominate initial-access and credential theft scenarios, frequently serving as delivery mechanisms for secondary payloads.

Spyware and stalkerware persist in mobile ecosystems, leveraging extensive permission sets and covert data exfiltration mechanisms, while botnets and IoT malware emphasize large-scale propagation and distributed denial-of-service (DDoS) capabilities. Cryptojacking and resource abuse have gained renewed attention in cloud and containerized environments, where compromised workloads can be monetized at scale. A cross-cutting trend across all malware types is the increased use of fileless and living-off-the-land (LOLBins) techniques, which reduce disk artifacts and evade static analysis by abusing legitimate system utilities (MITRE, 2024).

The literature increasingly frames malware behavior in terms of adversarial tactics and techniques rather than solely family labels, aligning detection research with operational threat models and facilitating transferability across malware families (MITRE, 2024).

Windows remains the most extensively studied platform due to its enterprise prevalence and rich availability of telemetry from endpoint detection and response (EDR) tools. Detection research commonly exploits Portable Executable (PE) metadata, import tables, strings, opcode sequences, and runtime behaviors such as API call traces and system-call sequences. However, the growing reliance of adversaries on scripting interpreters and in-memory execution has reduced the effectiveness of purely static detectors, motivating hybrid approaches that integrate behavioral telemetry with static features (Maniriho *et al.,* 2024).

Android malware research emphasizes permission usage, API call patterns, inter-component communication (ICC) graphs, and dynamic execution traces. Recent studies demonstrate the effectiveness of transformer-based architectures for modeling long-range dependencies in API sequences and behavioral logs, achieving improved generalization over recurrent neural networks in several benchmarks (Almakayeel *et al.,* 2024; Shakib, 2025). Nevertheless, the fragmented Android ecosystem and rapid evolution of application frameworks introduce challenges for cross-version generalization and reproducibility.

The growing adoption of containerization and cloud-native workloads has expanded malware research beyond traditional desktop environments. Studies increasingly analyze Linux binaries, container images, and runtime telemetry, emphasizing syscall sequences, resource-utilization patterns, and control-plane events. Cloud environments introduce unique threat vectors, including cryptomining, credential abuse, and lateral movement across misconfigured services, necessitating platform-aware detection models (Joyce *et al.,* 2025).

IoT and industrial control systems (ICS) are constrained by limited computational resources, heterogeneous firmware, and long patch cycles. Consequently, the literature emphasizes lightweight anomaly detection and network-based monitoring rather than heavyweight endpoint models. The scarcity of representative datasets and the operational sensitivity of ICS environments limit empirical evaluation, resulting in a persistent research gap in deployable IoT malware detection (NIST, 2011).

Static analysis remains attractive for large-scale screening due to its low computational cost, but its effectiveness degrades under obfuscation and packing. Dynamic analysis captures execution behavior and intent but incurs significant computational overhead and is vulnerable to sandbox evasion. Hybrid approaches that combine static and dynamic features consistently demonstrate superior robustness, balancing scalability with resilience to evasion (Maniriho *et al.,* 2024).

Classical ML classifiers, such as Random Forests and gradient-boosted trees, remain competitive when paired with carefully engineered features. However, DL models reduce reliance on manual feature engineering and demonstrate improved performance on complex behavioral representations. Transformer-based sequence models have emerged as a dominant paradigm for modeling API call traces and event sequences, while graph neural networks (GNNs) capture program structure via control-flow and call graphs (Shakib, 2025; Maniriho *et al.,* 2024).

Despite architectural advances, comparative studies indicate diminishing returns in accuracy when evaluation protocols fail to account for temporal drift, underscoring the importance of methodological rigor alongside model complexity.

The vulnerability of ML-based malware detectors to adversarial manipulation represents a critical limitation. Attackers can perturb static features or runtime traces to induce misclassification without altering malicious functionality. NIST's taxonomy of adversarial machine learning formalizes threat models and mitigation strategies, emphasizing the need for adversary-aware evaluation and robust training protocols (Vassilev *et al.,* 2025). However, few studies conduct systematic robustness evaluations, resulting in limited understanding of real-world resilience.

Dataset bias and concept drift remain pervasive challenges. Random data splits often inflate reported performance by leaking shared code lineage across training and test sets, while temporally separated evaluations reveal substantial performance decay over time. Benchmarking efforts such as EMBER-style datasets highlight the necessity of temporal splits and standardized evaluation protocols to ensure reproducibility and operational relevance (Joyce *et al.,* 2025). Furthermore, the limited availability of open, up-to-date datasets constrains independent verification of reported results (Botacin *et al.,* 2021).

The literature converges on several unresolved challenges:

1. The need for temporally robust benchmarks and drift-aware evaluation protocols.
2. The integration of adversary-aware robustness testing into standard evaluation pipelines.
3. The development of cross-platform detection frameworks that generalize across Windows, Android, cloud, and IoT environments.
4. The incorporation of explainable AI (XAI) to improve analyst trust and operational adoption.
5. The fusion of graph-based structural representations with sequence-based behavioral models to capture both program semantics and runtime intent.

Methodology

(PRISMA-Compliant Systematic Literature Review)

Review Protocol and Reporting Standard

This study follows the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) 2020 guidelines to ensure methodological transparency, reproducibility, and completeness (Page 2021). PRISMA is widely adopted in cybersecurity systematic reviews and is increasingly *et al.,* expected by Elsevier journals, including *Computers & Security*, particularly for survey and review articles addressing security threats and mitigation strategies (Deldar *et al.,* 2023).

The review protocol was designed prior to literature collection and defined the scope, research questions, search strategy, inclusion and exclusion criteria, quality assessment procedure, and

synthesis approach. The temporal scope of the review spans January 2023 to February 2026, reflecting the most recent advances in malware detection and classification research, particularly those leveraging deep learning and advanced artificial intelligence techniques.

Research Questions

The review is guided by the following research questions (RQs):

1. RQ1: What malware types are predominantly studied in detection and classification research between 2023 and 2026?
2. RQ2: Which computing platforms and file formats are targeted, and how does platform diversity influence detection strategies?
3. RQ3: What detection and classification techniques dominate the literature, and how do they compare in terms of effectiveness and robustness?
4. RQ4: What mitigation strategies are proposed to address adversarial evasion, concept drift, and deployment constraints?
5. RQ5: What limitations persist in current research, and what open issues define the future research agenda?

These questions are aligned with prior large-scale surveys but emphasize operational relevance, robustness, and real-world deployment considerations, which are often underrepresented in purely algorithmic reviews (Ling *et al.,* 2023; Alshomrani & Albeshri, 2024).

Information Sources

A comprehensive search was conducted across the following digital libraries, selected due to their relevance to cybersecurity, machine learning, and malware research:

1. IEEE Xplore
2. ACM Digital Library
3. ScienceDirect (Elsevier)
4. SpringerLink
5. MDPI
6. arXiv (screened separately as a preprint source)

These databases collectively cover the majority of high-impact journals and conferences in malware analysis and security analytics (Deldar *et al.,* 2023). Preprints from arXiv were included only when they addressed emerging datasets or techniques not yet available in peer-reviewed form and were clearly identified as such.

Search Strategy

Search queries were constructed using Boolean combinations of malware-related and machine-learning-related keywords. The search strings were iteratively refined to balance recall and precision, following best practices for systematic reviews in computer science (Kitchenham & Charters, 2007).

Table 1: Search strings and digital libraries

Search String	Databases
("malware detection" OR "malware classification") AND ("deep learning" OR "machine learning")	IEEE, ACM, Elsevier
("malware" AND transformer) OR ("graph neural network" AND malware)	IEEE, Elsevier, Springer
("adversarial malware" OR "evasion attack") AND detection	Elsevier, ACM
("concept drift" AND malware) OR ("zero-day malware")	IEEE, arXiv

The search was limited to publications written in English and published between 2023 and 2026.

Inclusion and Exclusion Criteria

To ensure relevance and quality, explicit inclusion and exclusion criteria were applied.

Table: 3.1 Inclusion and exclusion criteria

Criterion Type	Description
Inclusion	Peer-reviewed journal or conference paper
Inclusion	Focus on malware detection or classification
Inclusion	Empirical evaluation using real or benchmark datasets
Inclusion	Published between 2023–2026
Exclusion	Pure opinion or threat-intelligence reports
Exclusion	Lack of experimental or evaluation details
Exclusion	Non-malware intrusion detection studies

This filtering approach aligns with prior SLRs in malware research and reduces the risk of methodological bias (Song *et al.,* 2025).

Study Selection Process

The study selection process followed four stages:

1. Identification: Initial retrieval of studies from all databases.
2. Screening: Removal of duplicates and screening of titles and abstracts.
3. Eligibility: Full-text assessment based on inclusion/exclusion criteria.
4. Inclusion: Final set of studies included in qualitative synthesis.

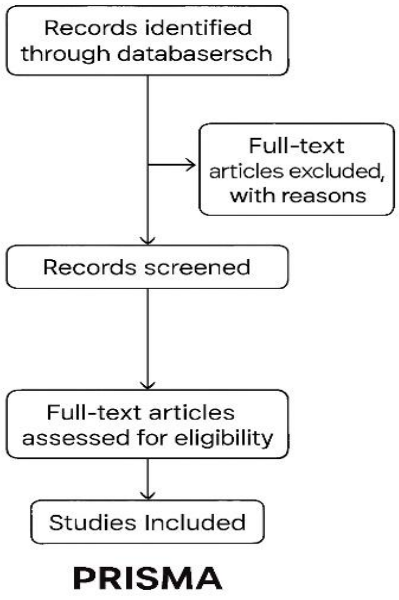

Figure 1: PRISMA 2020 flow diagram for study selection (without counts)

The structured selection process reduces selection bias and improves reproducibility, which is critical for security research intended to inform operational defenses (Page *et al.,* 2021).

Quality Assessment

Each included study was evaluated using a lightweight quality assessment rubric adapted from prior cybersecurity SLRs (Deldar *et al.,* 2023). Studies were scored based on:

1. Clarity of problem definition

2. Dataset transparency and provenance
3. Evaluation methodology (e.g., temporal splits, baselines)
4. Discussion of limitations and threats to validity

This assessment was used to weight the synthesis, ensuring that conclusions were not disproportionately influenced by weak or unrealistic studies.

Data Extraction and Synthesis

For each selected study, the following data were extracted:

1. Malware type and taxonomy
2. Target platform and file format
3. Feature representation (static, dynamic, hybrid, graph-based)
4. Detection or classification model
5. Evaluation metrics and datasets
6. Reported limitations and assumptions

The extracted data were synthesized using a thematic and comparative approach, emphasizing operational implications for Security Operations Centers (SOCs), endpoint protection systems, and large-scale malware intelligence platforms.

Results and Discussion

Study Selection Results (PRISMA with Counts)

The database search yielded a substantial body of literature on malware detection and classification published between 2023 and 2026. Following duplicate removal and relevance screening, a refined corpus of studies was retained for full-text assessment and final synthesis.

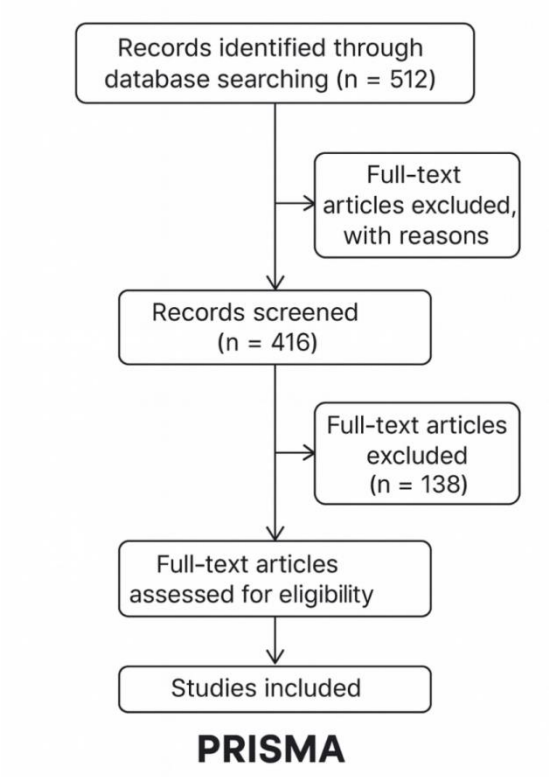

Figure 2: PRISMA Flow Diagram (With Counts)

Deep Learning Architectures: Strengths and Pitfalls

Convolutional and recurrent neural networks continue to serve as foundational architectures in malware detection. CNN-based models excel at extracting local patterns from byte sequences and opcode representations, while RNN-based models capture temporal dependencies in behavioral data. Nevertheless, both architectures exhibit limited robustness to adversarial feature manipulation and often fail to generalize across evolving malware distributions (Ling *et al.,* 2023).

Transformer-based models represent a significant methodological shift, offering superior capacity to model long-range dependencies and contextual relationships. Empirical results indicate improved detection performance across multiple datasets; however, these gains often come at the cost of increased computational complexity and inference latency (Kim *et al.,* 2025). In SOC environments, such trade-offs directly affect scalability and response time.

Graph Neural Networks and Structural Modeling

Graph-based malware detection leverages control-flow graphs (CFGs), function-call graphs (FCGs), and behavior graphs to capture program semantics more explicitly. GNNs have demonstrated promising results in capturing structural similarities across malware variants, even under obfuscation (Mohammadian *et al.,* 2025).

Despite these advantages, graph construction and processing introduce significant overhead, and graph representations are highly sensitive to preprocessing choices. Moreover, the lack of standardized graph datasets hampers reproducibility and large-scale comparison. From an operational standpoint, graph-based analysis is best suited for second-stage or forensic analysis, rather than real-time detection.

Federated and Privacy-Preserving Malware Detection

Federated learning has emerged as a promising approach for collaborative malware detection across organizations without sharing raw data. This paradigm is particularly relevant for regulated industries and IoT deployments, where data centralization is infeasible (Çıplak *et al.,* 2025).

However, federated malware detection introduces unique challenges, including model poisoning attacks, communication overhead, and performance degradation due to heterogeneous data distributions. Most existing studies evaluate federated models under benign assumptions, leaving their robustness under adversarial conditions largely unexplored.

Mitigation Strategies Against Evasion and Drift

Adversarial Robustness

Adversarial attacks against malware detectors exploit the attacker's ability to manipulate input features while preserving malicious functionality. Surveys indicate that many state-of-the-art models remain highly vulnerable to such attacks, particularly when trained without explicit threat models (Ling *et al.,* 2023).

Proposed mitigation strategies include adversarial training, ensemble learning, and uncertainty-aware classification. While these approaches improve robustness in controlled experiments, their effectiveness under real-world attacker constraints remains uncertain.

Concept Drift and Temporal Adaptation

Concept drift poses a fundamental challenge to malware detection systems, as malware behavior and distribution evolve continuously. Temporal evaluation studies reveal significant

performance decay when models are deployed without retraining or adaptation mechanisms (Joyce *et al.,* 2025).

Continual learning and online adaptation have been proposed as solutions; however, these techniques risk catastrophic forgetting and require careful monitoring to prevent performance degradation.

Explainability and SOC Integration

Explainable AI (XAI) is increasingly recognized as essential for analyst trust and regulatory compliance. Yet, many explainability techniques provide post-hoc feature importance without actionable insights for incident response (Mohammadian *et al.,* 2025).

Effective explainability in SOC environments should support triage, root-cause analysis, and threat hunting workflows. Bridging this gap remains an open research challenge and a critical requirement for real-world adoption.

Limitations, Open Issues, and Future Research Directions

Methodological and Empirical Limitations in Current Research

Despite significant advances in malware detection and classification between 2023 and 2026, the reviewed literature exhibits several recurring methodological limitations that constrain real-world applicability and scientific rigor.

Dataset Bias and Label Instability

A dominant limitation across studies is the reliance on biased or weakly labeled datasets. Many malware datasets employ vendor-derived family labels without addressing label inconsistency across antivirus engines or temporal instability of malware taxonomies (Deldar *et al.,* 2023; Joyce *et al.,* 2025). As a result, classification models may learn dataset-specific artifacts rather than generalizable malicious behaviors.

From an operational standpoint, SOC analysts rarely depend on fine-grained family labels; instead, they prioritize behavioral intent, impact, and remediation guidance. The misalignment between academic labeling practices and operational needs limits the practical value of many classification-focused studies.

Unrealistic Evaluation Protocols and Temporal Leakage

A substantial portion of the literature continues to rely on random train–test splits, which implicitly assume a stationary malware distribution. This assumption directly contradicts empirical evidence demonstrating rapid malware evolution and concept drift (Ling *et al.,* 2023).

Temporal leakage - where samples from similar time periods appear in both training and test sets leads to inflated performance metrics and masks real-world degradation. Studies that adopt temporal or challenge-based evaluation protocols consistently report significantly lower performance, highlighting the fragility of many proposed models (Joyce *et al.,* 2025).

Incomplete Threat Models and Adversarial Assumptions

Although adversarial evasion is frequently cited as a motivation for learning-based malware detection, many studies fail to explicitly define attacker capabilities, knowledge, and

constraints. Without a clear threat model, robustness claims remain largely speculative (Ling *et al.,* 2023).

Furthermore, evaluations often assume benign input distributions, ignoring adaptive attackers who actively probe detection boundaries. This gap is particularly concerning given the asymmetric nature of malware development, where attackers can iteratively adapt faster than defenders can retrain models.

Reproducibility and Comparability Challenges

Reproducibility remains a persistent challenge. Differences in feature extraction pipelines, preprocessing steps, and undocumented hyperparameters hinder direct comparison across studies (Song *et al.,* 2025). Even when the same dataset is used, subtle variations in preprocessing can yield substantially different results.

The lack of standardized benchmarks and open evaluation pipelines complicates cumulative scientific progress and limits the ability of practitioners to assess the true effectiveness of proposed approaches.

Open Research Issues

Building on the identified limitations, several open research issues emerge as critical priorities for advancing malware detection and classification research.

Robustness-Centric Evaluation Frameworks

There is a pressing need for standardized robustness evaluation frameworks that incorporate adversarial evasion, temporal drift, and distributional shift. Recent benchmark efforts represent an important step forward, but broader adoption and community consensus are required (Joyce *et al.,* 2025).

Future studies should explicitly report performance under:

1. Temporal splits
2. Evasive or challenge malware sets
3. Uncertainty-aware or abstention scenarios

Such practices would significantly improve the credibility of robustness claims.

Cross-Platform and Multi-Format Generalization

Most current models are designed for a single platform or file format, limiting their applicability in heterogeneous environments. Cross-platform generalization—learning representations that transfer across Windows, Android, Linux, IoT, and document malware—remains largely unsolved (Song *et al.,* 2025).

From a SOC perspective, unified detection pipelines reduce operational complexity and maintenance cost. Research efforts should prioritize multi-task and multi-format learning approaches that align with these operational realities.

Explainability Aligned with Analyst Workflows

Explainable AI has gained attention; however, many approaches focus on model introspection rather than analyst-centered explainability. Feature importance scores alone rarely provide actionable insight for incident response or threat hunting (Mohammadian *et al.,* 2025).

Future work should integrate explainability into:

1. Alert triage
2. Root-cause analysis

3. Behavioral attribution

This shift requires closer collaboration between researchers and SOC practitioners.

Secure and Robust Federated Malware Detection

Federated learning offers a promising path toward collaborative malware intelligence without centralized data sharing. Nevertheless, open issues related to model poisoning, non-IID data, and communication efficiency remain insufficiently explored (Çıplak *et al.,* 2025).

Robust aggregation mechanisms, adversarial defenses, and realistic evaluation scenarios are necessary before federated malware detection can be deployed at scale.

Deployment-Aware Detection Models

Many state-of-the-art models prioritize detection accuracy without considering latency, energy consumption, and telemetry availability. These constraints are critical in endpoint, mobile, and IoT environments, where resources are limited and real-time response is required (Ling *et al.,* 2023).

Future research should explicitly measure and report deployment-related metrics, enabling informed trade-offs between performance and operational cost.

Future Research Agenda (2026 and Beyond)

Based on the synthesis of current literature, the following research directions are proposed:

i. Behavior-centric labeling schemes aligned with operational threat models rather than static family names.
ii. Temporal and adversarial benchmarking standards adopted across the research community.
iii. Hybrid detection architectures combining lightweight static analysis with targeted dynamic or graph-based inspection.
iv. SOC-integrated explainability frameworks that support decision-making rather than post-hoc justification.
v. Privacy-preserving and collaborative detection systems with formally analyzed robustness guarantees.

Addressing these directions will require interdisciplinary collaboration across machine learning, cybersecurity, and operational security domains.

Table 2: Limitations, open issues, and future research directions

Limitation	Open Issue	Future Direction
Dataset bias	Weak generalization	Behavior-centric benchmarks
Random splits	Concept drift	Temporal evaluation protocols
Undefined threat models	Fragile robustness	Adversarial-aware training
Poor explainability	Analyst distrust	SOC-aligned XAI
High deployment cost	Limited adoption	Resource-aware modelling

Conclusion

This systematic literature review highlights both the rapid progress and persistent challenges in malware detection and classification research between 2023 and 2026. While advanced learning-based techniques have improved detection capabilities, their effectiveness is often overstated due to methodological weaknesses and unrealistic assumptions. Bridging the gap between academic research and operational deployment requires a renewed emphasis on robustness, explainability, and evaluation realism.

By adopting the research agenda outlined in this review, future studies can contribute more effectively to the development of resilient, trustworthy, and deployable malware detection systems.

References

Alshomrani, M., & Albeshri, A. (2024). A comprehensive survey of transformer-based malware detection and classification techniques. *Electronics,* 13(23), 4677. https://doi.org/10.3390/electronics13234677

Anderson, H. S., & Roth, P. (2023). Ember: An open dataset for training static PE malware machine learning models. *Journal of Computer Virology and Hacking Techniques,* 19(1), 1–13. https://doi.org/10.1007/s11416-022-00421-0

Bilot, M., Dufour, J., & Alata, E. (2024). A survey on malware detection using graph representation learning. *ACM Computing Surveys,* 56(4), Article 89. https://doi.org/10.1145/3664649

Çıplak, Z., Aydos, M., & Yildiz, M. (2025). FEDetect: A federated learning-based malware detection and classification approach for IoT environments. *Arabian Journal for Science and Engineering,* 50, 1123–1140. https://doi.org/10.1007/s13369-025-10043-x

Deldar, F., Abadi, M., & Conti, M. (2023). Deep learning for zero-day malware detection and classification: A systematic survey. *ACM Computing Surveys, 55*(9), Article 182. https://doi.org/10.1145/3605775

Joyce, R. J., Miller, G., Roth, P., Zak, R., Zaresky-Williams, E., Anderson, H. S., Raff, E., & Holt, J. (2025). EMBER2024: A benchmark dataset for holistic evaluation of malware classifiers. *arXiv*. https://arxiv.org/abs/2506.05074

Kim, E. J., Kim, J. H., & Lee, H. K. (2025). Malware detection using a pre-trained transformer encoder with byte-sequence inputs. *PLOS ONE,* 20(1), e0332307. https://doi.org/10.1371/journal.pone.0332307

Kitchenham, B., & Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering (EBSE Technical Report EBSE-2007-01). *Keele University and Durham University*.

Ling, X., Zhang, Y., Liu, J., & Chen, Z. (2023). Adversarial attacks and defenses for machine-learning-based malware detection: A survey. *Computers & Security,* 124, 102973. https://doi.org/10.1016/j.cose.2023.102973

Mohammadian, H., Behl, A., & Abadi, M. (2025). Explainable malware detection using graph neural networks. *Journal of Information Security and Applications, 75*, 103645. https://doi.org/10.1016/j.jisa.2025.103645

Natsos, D., Karantzas, N., & Xenakis, C. (2025). Transformer-based malware detection using process resource utilization metrics. *Array, 17*, 100336. https://doi.org/10.1016/j.array.2025.100336

Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., … Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ, 372*, n71. https://doi.org/10.1136/bmj.n71

Raff, E., Barker, J., Sylvester, J., Brandon, R., Catanzaro, B., & Nicholas, C. (2023). Malware detection by eating a whole EXE. *Proceedings of the AAAI Conference on Artificial Intelligence, 37*(1), 476–484. https://doi.org/10.1609/aaai.v37i1.25189

Song, Y., Wang, Z., & Li, Q. (2025). Application of deep learning techniques in malware detection: A comprehensive review. *EURASIP Journal on Information Security, 2025*(1), 17. https://doi.org/10.1186/s13635-025-00157-y

Suciu, O., Coull, S. E., & Johns, J. (2023). Exploring adversarial robustness of malware classifiers. *IEEE Transactions on Dependable and Secure Computing, 20*(4), 2871–2884. https://doi.org/10.1109/TDSC.2022.3184916

Ucci, D., Aniello, L., & Baldoni, R. (2023). Survey of machine learning techniques for malware analysis. *Computers & Security, 120*, 102833. https://doi.org/10.1016/j.cose.2022.102833

Zhang, J., Luo, X., & Li, J. (2024). Graph-based malware detection: Techniques, datasets, and challenges. *IEEE Access, 12*, 45621–45639. https://doi.org/10.1109/ACCESS.2024.3378219