International Journal of Computers and Informatics

Journal Homepage: https://www.ijci.zu.edu.eg

Int. j. Comp. Info. Vol. 11 (2026) 74–100

Paper Type: Review Article

Recent Advances and Challenges In Malware Detection For Internet of Things Systems: A Comprehensive Review

Nayera A. Alsayed [1,*], Mahmoud Khaled Abd-Ellah [1], Osama M. Elkomy [2] and Walaa M. EL-Hady [2]

[1] Faculty of Artificial Intelligence, Egyption Russian University, Cairo 11829, Egypt;
E-mails: nayera-alsayed@eru.edu.eg; Mahmoud-khaled@eru.edu.eg.
[2] Department of Information Technology, Faculty of Computers and Informatics, Zagazig University, Zagazig, Egypt.

Received: 09 Feb 2026 **Revised**: 01 Apr 2026 **Accepted**: 26 Jun 2026 **Published**: 30 Jun 2026

Abstract

The Internet of Things (IoT) ecosystem faces escalating security threats from sophisticated malware targeting diverse, resource-constrained devices. Despite advances, effective detection remains challenging due to evolving malware behaviors and heterogeneous environments. This review presents a comprehensive survey of over forty-three studies from 2018 to 2025, analyzing machine learning, deep learning, hybrid, and non-AI-based malware detection techniques. Our review reveals that deep learning and hybrid models generally outperform traditional methods by capturing complex behavioral patterns, yet issues like limited dataset diversity, computational demands, and explainability persist. We identify critical research directions including lightweight edge-compatible models, federated learning, multimodal feature fusion, and explainable AI integration. These insights provide a structured understanding of current approaches and guide the development of scalable, robust, and interpretable IoT malware detection systems, advancing cybersecurity in increasingly connected environments.

Keywords: IoT Malware Detection; Deep Learning; Hybrid Models; Federated Learning; Explainable AI (XAI).

1 | Introduction

The Internet of Things (IoT) has transformed digital ecosystems by interconnecting billions of devices, enabling seamless data exchange and intelligent automation across sectors such as healthcare, smart homes, transportation, and industry [1]. However, this extensive connectivity has introduced unprecedented security vulnerabilities, making IoT networks increasingly attractive targets for cyber attackers [2].These challenges are exacerbated by the resource constraints and heterogeneity of IoT devices, which hinder the effectiveness of traditional security measures in such dynamic and large-scale environments [3].

Malware attacks represent some of the most prevalent and severe cyber threats, compromising device integrity, stealing sensitive data, and disrupting critical services. Despite advances in cybersecurity, malware continues to evolve rapidly, exhibiting increasing complexity and stealth that hinder detection and analysis. Attackers employ techniques such as encapsulation and compression to conceal malicious activities, further

Corresponding Author: nayera-alsayed@eru.edu.eg

complicating analysis. The growing diversity of digital environments, communication protocols, and computing technologies enables malware to manifest in numerous forms. Consequently, effective malware analysis has become increasingly challenging, necessitating the development of advanced and adaptive detection solutions [4].

Malware detection plays a crucial role in safeguarding users and organizations from malicious software. Initially, signature-based detection methods were used, but they are ineffective against new or sophisticated malware. To address this, researchers have developed more advanced techniques, including behavior-based and heuristic detection, as well as pattern verification. With the continuous evolution of threats, data mining and machine learning have become widely adopted to strengthen malware detection and bolster security systems [5].

Malware detection has become a critical cybersecurity concern because it directly affects the legal, financial, and reputational stability of organizations [6]. Over the years, numerous studies have investigated this issue using both ML and DL techniques [7]. Traditional ML approaches have been employed to classify and identify malicious patterns, while recent research has emphasized the superior capability of DL models in learning complex representations of malware behaviors [8-10].

Deep learning, in particular, offers promising solutions for enhancing the performance of detection systems by automatically extracting meaningful and intricate features from raw data [6]. Models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks can capture both the static and dynamic characteristics of malware, enabling a more comprehensive understanding of malicious activities. Furthermore, DL-based systems minimize the dependency on manually crafted detection rules, improving adaptability, scalability, and efficiency in practical deployment scenarios [11, 12].

While malware detection and intrusion detection are two closely-related research topics in IoT cybersecurity, they deal with various security issues. Malware detection is mainly concerned with the identification of malicious software using the study of malware features and behaviors, while intrusion detection is concerned with the identification of malicious actions or access in IoT environments [13, 14]. However, because there is some overlap in terms of techniques used, datasets, and detection frameworks in both fields, intrusion detection papers relevant for IoT malware detection have been included in this survey.

Despite the widespread adoption of DL approaches for malware detection, several challenges persist. Many studies rely on small or narrowly focused datasets, which limits the generalizability of their models to real-world environments [15-18]. In addition, a lack of integration between different DL architectures has been observed, limiting their ability to model the complex and diverse nature of malware [4]. Several studies also fail to include the full range of IoT-related malware, focusing instead on limited samples that do not represent real-world threats [19]. Moreover, suboptimal preprocessing and feature extraction methods have often resulted in lower-performing models, as highlighted in [20].

This review extends the existing literature by providing an integrated analysis of IoT malware detection approaches, covering data preprocessing techniques, feature engineering methods, machine learning, deep learning, hybrid models, explainable artificial intelligence (XAI), and emerging research directions such as federated learning and lightweight edge intelligence. In addition to summarizing recent advances, this survey provides a comparative critical analysis of existing approaches, identifies current research gaps and practical deployment challenges, and discusses future research opportunities.

Accordingly, the main objectives of this review are to:

- Provide a comprehensive overview of IoT malware detection studies published between 2018 and 2025
- Provide a comparative analysis of preprocessing techniques, feature engineering methods, and malware detection approaches, including machine learning, deep learning, and hybrid models

- Highlight current research gaps and practical deployment challenges
- Discuss emerging trends and future research directions toward scalable, efficient, and trustworthy IoT malware detection systems.

The remainder of this paper is organized as follows. Section 2 presents the background of IoT malware detection and AI-based detection techniques. Section 3 describes the review methodology adopted for selecting and analyzing the literature. Section 4 presents the literature review together with a comparative critical discussion of existing approaches, research gaps, and future directions. Finally, Section 5 concludes the paper and summarizes the key findings and practical implications.

2 | Background

2.1 | IoT Malware Overview

The Internet of Things (IoT) links billions of smart devices together to make a huge, ever-changing network that lets machines talk to each other and automate tasks. But this connection has also made cybersecurity harder in several ways. Malware is any program that runs malicious applications on computers, smartphones, computer networks, and other devices that it targets on purpose. Ransomware, rootkits, worms, Trojan horses, and viruses are all types of malware. Every family and type of malware is made to have a different effect on the first victim's computer in different ways, such as by harming the system being targeted, allowing remote code execution, or stealing private data, etc [5, 21].

IoT malware refers to malicious software designed to exploit vulnerabilities in IoT devices or communication protocols, enabling attackers to gain unauthorized access, steal data, or disrupt network operations. Common examples include Aidra, Bashlite, and Mirai, which primarily leverage weak authentication or open ports to assemble large botnets capable of launching distributed denial-of-service (DDoS) attacks [13].

2.2 | AI Techniques for IoT Malware Detection

The integration of Artificial Intelligence (AI) into IoT malware detection has significantly enhanced the ability of security systems to identify and respond to cyber threats. Traditional methods, such as signature-based and anomaly-based detection, often struggle with the dynamic and evolving nature of IoT malware, especially zero-day and polymorphic attacks that frequently alter their code to evade detection. AI-driven approaches, including Machine Learning (ML) and Deep Learning (DL), enable systems to automatically learn complex behavioral and structural patterns from large and heterogeneous IoT data sources, thereby improving detection speed, adaptability, and accuracy [22, 23]. ML-based techniques, such as Decision Trees (DT), Support Vector Machines (SVM), and Random Forests (RF), require manual feature engineering and data preprocessing [24, 25],whereas DL-based methods like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) models can automatically extract hierarchical and complex representations from raw data, leading to higher detection accuracy.

Consequently, AI-driven malware detection has emerged as an essential element of contemporary IoT cyber security frameworks. These frameworks provide protection mechanisms that are data-driven, scalable, and flexible, beyond the capabilities of old methods. The implementation of this technical transition is a significant step toward the development of Internet of Things security systems that are intelligent, automated, and robust, and that are able to protect against the growing sophistication of cyberattacks[26].

3 | Review Methodology

The present literature review was performed with the help of an organized approach that will make sure that all the latest studies on malware detection approaches in the context of the Internet of Things (IoT) can be identified and analyzed. Studies that were performed between 2018 and 2025 are considered as part of this literature review because of the dynamic nature of IoT malware detection approaches.

3.1 | Literature Sources

Relevant literature sources were obtained via reputable scientific electronic databases such as IEEE Xplore, ScienceDirect (Elsevier), SpringerLink, MDPI, ACM Digital Library, Wiley Online Library, and Google Scholar. They were selected for containing a vast number of highly relevant peer-reviewed journal articles and conference proceedings in areas including cybersecurity, artificial intelligence, malware analysis, and IoT.

3.2 | Search Strategy

Keywords were carefully chosen in order to perform the literature search for articles associated with IoT malware detection and intelligent malware detection mechanisms. The following search keywords were used:

- "IoT Malware Detection"
- "Internet of Things Malware"
- "Machine Learning Malware Detection"
- "Deep Learning Malware Detection"
- "IoT Security"
- "Malware Classification"
- "Hybrid Malware Detection"
- "Explainable AI Malware Detection"

3.3 | Study Selection Criteria

Each study identified during the search process was then considered against specific inclusion and exclusion criteria.

Inclusion Criteria:

- Peer-reviewed journal articles and conference papers.
- Publications published between 2018 and 2025.
- Studies focusing on malware detection methods for IoT devices.
- Research using machine learning, deep learning, hybrid learning, feature engineering, or explainable AI methods.
- Studies published in English.
- Research with experimental validation and performance measures.

Exclusion Criteria:

- Duplicate papers.

- Non-English studies.
- Editorials, tutorials, theses, and books.
- Literature not focused on IoT malware detection.
- Articles that lack necessary experimental results or analyses.

3.4 | Screening Process

To ensure transparency and reproducibility, the study selection followed a structured screening pipeline inspired by the PRISMA guidelines. The initial database search across IEEE Xplore, ScienceDirect, SpringerLink, MDPI, ACM, Wiley, and Google Scholar yielded a total of 420 articles based on the specified keyword combinations. After removing 135 duplicate publications, 285 unique papers remained. A rigorous title and abstract screening phase filtered out 192 irrelevant studies that did not directly align with IoT-specific malware detection or lacked implementation details. The remaining 93 papers underwent full-text assessment against our strict inclusion and exclusion criteria. Ultimately, 43 core studies published between 2018 and 2025 were selected as the definitive basis for this comprehensive survey and comparative analysis.

4 | Literature Review

4.1 | Existing Review Papers on IoT Malware Detection

In recent years, quite a number of survey articles have examined IoT malware detection from different angles, ranging from machine learning and deep learning, intrusion detection systems to malware analysis. Despite the valuable contribution of these surveys towards consolidating the considerable progress made in this area, most of the surveys focus narrowly on individual aspects of IoT security but fail to offer an overall review of the malware detection process. Also, the inclusion of some novel research areas such as explainable artificial intelligence (XAI), federated learning, lightweight edge-based approaches, and deployment-related issues is lacking in quite a number of existing surveys. This comparison is provided in Table 1 in relation to selected review articles.

Table 1. Comparison of Existing IoT Malware Detection Review Papers with the Present Survey.

Ref	Year	ML	DL	Hybrid Models	Preprocessing	Feature Engineering	XAI	Federated Learning	Lightweight Edge Models	Critical Comparative Analysis	Coverage Period
[27]	2021	✓	✓	✗	Limited	✗	✗	✗	✗	Limited	2010–2020
[28]	2022	✓	✓	✗	✗	✗	✓	✓	✗	Limited	Up to 2022
[29]	2023	Limited	✓	✗	✗	✗	✗	✗	✗	Limited	Up to 2023
[30]	2023	✓	✓	Limited	✓	Limited	✗	✗	✗	Limited	Up to 2023
[31]	2024	✓	✓	Limited	Limited	Limited	✗	Limited	✗	Limited	Up to 2024
[32]	2024	✗	✓	✗	Limited	Limited	✗	✗	✗	Limited	Up to 2024
[33]	2024	✓	Limited	✗	Limited	✗	✗	✗	✗	Limited	Up to 2024
This Review	**2026**	✓	✓	✓	✓	✓	✓	✓	✓	**Comprehensive**	**2018–2025**

As depicted in Table 1, previous reviews have predominantly centered on certain research paths like malware detection using machine learning, deep learning, IDS, or malware analysis. Although these literature reviews are rich sources of knowledge and useful for gaining insight into particular research areas, they rarely cover the entire IoT malware detection pipeline or the emerging trends in this area in depth. Specifically, there is very little information about preprocessing, feature engineering, hybrid malware

detection approaches, explainable artificial intelligence (XAI), federated learning, light edge-based solutions, and deployment aspects.

On the other hand, the proposed literature review offers a more holistic approach since it critically analyzes the recent IoT malware detection approaches presented from 2018 to 2025. The current literature review not only provides a summary of the state-of-the-art malware detection solutions but also conducts a comparative and critical analysis of the approaches to preprocessing, feature engineering, machine learning, deep learning, hybrid solutions, and research challenges. Thus, the proposed literature review can be considered an important source for researchers who are looking for malware detection solutions in IoT environment.

4.2 | Recent IoT Malware Detection Studies

The expansion of IoT has led to many interconnected smart devices, but this increased connectivity also exposes them to various cyber threats, including advanced malware attacks. As a result, there has been significant research into intelligent malware detection methods tailored for IoT systems[14].

Numerous studies have been performed to address malware detection difficulties in IoT systems, utilizing both ML and DL techniques. These studies differ significantly in terms of the datasets used, the algorithms used, the performance metrics evaluated, and the experimental conditions used. Benchmark datasets such as NSW-NB15, N-BaIoT, NSL-KDD, CSE-CIC-IDS2018, CICAndMal2019, and CICIoT2023 are now widely used for performance evaluation in IoT malware detection research. Nonetheless, despite promising results in many studies, model performance varies significantly due to diversity of IoT data, dataset imbalance, resource constraints at the edge, and a lack of real-time evaluation settings[32, 34]. This section provides a comprehensive and comparative overview of key research studies on IoT malware detection. As shown in Table 2, it illustrates the methodology, databases, and reported results used to identify current research trends, important limits, and suggested areas for future research.

4.3 | Machine Learning-based Methods

M. Azeem et al. (2024) detected malware using UNSW-NB15 using ML [4]. They extracted features using Information Gain, Gain Ratio, Bag-of-Words, Word2Vec, and TF-IDF. The model was trained with various classifiers. ET had the best accuracy at 99.98%. The study's drawbacks include its reliance on the UNSW-NB15 dataset, its inability to adequately capture complicated categorical variable interactions, and its lack of deep learning.

The MADP-IIME protocol was introduced for detecting malicious assaults in IoT networks using a ML approach in [35]. It integrates four methodologies: NB, LR, Artificial Neural Network (ANN), and RF, along with processes for encoding labels, feature engineering, scaling, and managing missing data. Tested on an 823.69 MB IoT network intrusion dataset, MADP-IIME achieved a 99.50% detection rate, outperforming other methods. However, the findings may lack broader applicability due to the specific dataset used for evaluation.

El-Ghamry et al. (2023) balanced 338 benign and 512 harmful pictures using SMOTE [17]. The Ant Colony Algorithm was used to pick features more complexly. The features were extracted using density distribution, gray-level run-length matrix (GLRLM), and invariant moments. Particle Swarm Optimization (PSO), based on SVM, was utilized to fine-tune model parameters. The model scored 95.56% accuracy, 96.43% recall, 94.12% precision, and 95.26% F1. Problems included employing a class with an imbalanced cost function and a single data set.

In [36], various machine learning techniques were utilized, including Random Forest (RF), XGBoost (XGB), Decision Tree (DT), Gradient Boosting (GB), Logistic Regression (LR), Support Vector Machine

(SVM), and K-Nearest Neighbour (KNN), using malware samples from VirusTotal, AnyRun, and PolySwarm. The RF model achieved the highest accuracy at 93.3%, followed by XGBoost at 93.0%, DT at 90.9%, and GB at 90.0%. LR recorded 86.7% accuracy, while the accuracies of KNN and SVM were unspecified. The study noted challenges faced by ML models due to insufficient training samples, especially for fileless malware, which may affect their detection capabilities.

In [37], DWARF Mongoose Optimization with ML-Driven Ransomware Detection (DWOML-RWD) was introduced for ransomware detection. It employed the Enhanced Krill Herd Optimization (EKHO) and dynamic opposition-based learning (DOBL) for feature selection, while the Extreme Learning Machine (ELM) model was used for detection, optimized by DWO Mongoose Optimization (DWO). The performance of DWOML-RD was compared against Adaboost-M1, Bagging, RF, Rotation Forest, and DT, achieving notable results with 99.40% accuracy, recall, and F1 score, although the database scope was limited.

Researchers in [38] have created a machine learning-based way to find bad data in IoT networks. They employed KNN, RF, and Gaussian Naïve Bayes (GNB) and evaluated the algorithms on a sample of fake IoT traffic that had both good and bad traffic. The KNN model was more accurate, with an accuracy of 94.44%. The RF model was 88.8% accurate, however the GNB model was only 77.78% accurate. The study has several problems, though. For example, it can't uncover new malware that hasn't been found yet, and the data source documentation isn't clear, which makes it hard to check or repeat the results.

In [39], researchers employed SVM, NN, RF, and KNN to analyze Android device power consumption to discover ransomware software programs. Three Android cellphones were monitored for power consumption to build the dataset. The KNN model was most accurate at 94.27 %. The SVM model placed second at 91.19%, the Random Forest model third at 87.56%, and neural networks last at 73.33%. However, the study noted that energy use habits can change quickly and that standard strategies are ineffective.

The researchers in [40] applied machine learning methods and an ensemble architecture to detect malware in network data, specifically using the AdaBoost technique to combine Artificial Neural Networks (ANN), Naive Bayes (NB), and Decision Trees (DT) into a unified model. Their experiments on the UNSW-NB15 dataset demonstrated impressive performance, with the AdaBoost ensemble achieving an accuracy of 99.54%, compared to 95.32% for DT, 92.61% for ANN, and 91.17% for NB. Further testing on the NIMS Botnet Dataset and simulated IoT sensor network traffic produced an AdaBoost accuracy of 98.29%, while DT, ANN, and NB reached 96.10%, 94.22%, and 88.28% respectively. Despite these strong results, the study highlighted ongoing challenges related to the increasing complexity of modern attacks and the reliance on high-quality, generalizable feature sets.

4.4 | Deep Learning-based Methods

In [41], the researchers introduced the I-MCM framework, using a Tiny ANN to identify IoT malware , employing the Operation Code (Opcode) Purification Technique. They utilized the IoT dataset, which comprises 3000 malware samples (ARM, MIPS, X86) sourced from MalwareBazaar, with benignware obtained from OpenWRT, OpenIPC, and the Raspberry Pi Store. This framework has an accuracy score of 97.1.

In [42], B. Taşcı et al. (2024) applied machine learning algorithms to the CIC IoT 2023 dataset, comparing various models including Naive Bayes, KNN, SVM, DT, and a 1D CNN. The 1D CNN achieved the highest accuracy at 98.36%. Despite these strong results, the study highlighted limitations such as the need for validation on larger datasets and further evaluation of the model's effectiveness and practicality in real-time scenarios, particularly on devices with limited resources.

In [43], researchers created the ACLR model to find threats that come from the network. The model utilized four different DL architectures: ANN, CNN, LSTM, and RNN. They used the UNSW-NB15 dataset. The suggested ACLR model has an accuracy rate of 97.49%, a precision rate of 97.7%, a recall rate of 97.17%, and an F1-Score of 97.23%. The study has several disadvantages, one of which is that the performance relies on labeled data, which may not necessarily be representative.

T. Shi et al. (2024) used TF-IDF and one-hot encoding in [44] to look at data and show patterns in NetFlow data. To find strange things, they used Autoencoder and Isolation Forest models. On all of the test datasets, the Isolation Forest model got a recall rate of 100% and a precision rate of more than 80%. The deep autoencoder model was also very good at what it did. The study acknowledged how hard it is to train autoencoders on large datasets.

In [45], M. Amin et al. (2022) introduced advanced techniques for detecting malicious android applications using DL and MLmodels. Their approach involved preprocessing APK files through one-hot encoding and bytecode extraction, followed by feature selection and extraction using the Chars2Vec method. They evaluated two deep learning models: a Fully Connected Neural Network (FCN) and a Long Short-Term Memory (LSTM) network, both of which achieved an accuracy of 98.9%. However, the study encountered challenges such as the limited effectiveness of some feature detectors against code obfuscation and reduced performance when dynamic code loading occurred during runtime.

In [46], although the study focuses on intrusion detection rather than malware detection, study used logistic regression and ResNet-18 models to find risks on a network. The data was turned into network flows, and the features were found and saved in.pcap format. They picked the best 15 attributes from different datasets and put them into two groups. The ResNetDDoS-1 model was the best, with an accuracy of 98.70%, a precision of 97.53%, a recall of 97.96%, and an F1-score of 97.74%. The study has a number of shortcomings, including the fact that it largely focused on specific types of scanning and DDoS attacks, which doesn't cover all conceivable dangers. To make the models more resilient, they may need to be fine-tuned and validated on a wider range of datasets.

The authors developed a CNN-DMA malware detection system using deep learning and image processing in [15]. This CNN-based malware classification model extracts textural features. The Malimg dataset contains 9,339 grayscale malware samples. The model found 99% of malware, but malware class imbalance and small image size limited it.

In [47], the researchers utilized an ANN to identify the Mirai, employing a dataset comprising 115 characteristics and 49,548 sets for both Mirai and Benign classifications. They cleansed the data first, and then they trained the model. Their model achieved an accuracy of 92.9% and False Negative rate of 0.3.

Fuzzy neural networks and software-defined networking (SDN) were combined by researchers in [16] to create new ways to find threats. ANFIS, or the Adaptive Neuro-Fuzzy Inference System, makes recognition more accurate and flexible. Phishing, memory use, SNR, error messages, server overload, and SSH alarms are all covered in the NSL-KDD dataset. Even though it has some flaws, like a limited testing environment and dataset reach, the framework can find 83% of attacks.

In [48] , the DAIMD approach uses dynamic analysis to identify IoT malware during execution [40]. ZFNet CNN sorts data by features. A bat approach equalizes data and prevents viruses from being too specialized. The testing dataset comprises 561 files, some malicious and some not. The training dataset comprises 840 images of malware behavior over time. The DAIMD model was 99.28% correct. The study found that the training dataset's quality and diversity affect the model's efficacy and ability to generalize to new malware strains.

The study introduced the MTHAEL model, a collective learning strategy for discovering dangerous software using multiple models [49]. Information Gain and Opcode Dictionary revealed key features. The

researcher tested the device with ASUS, D-Link, and TP-LINK firmware routers and harmful malware. The approach scored 99.98%, indicating good data security.

Malicious or benign system calls were examined using RNNs and data preprocessing [18]. Researchers extracted key system call features using n-gram and TF-IDF. 200 samples of malware and benign software are in the dataset. When distinguishing malicious and benign system calls, the RNN model reached 98.712% accuracy after five epochs. However, the dataset's failure to cover all malware versions and the model's dependence on a single hidden layer, which limits its capacity to recognize complicated patterns, were noted.

The study in [50]detects successive attacks using J48 DT, ANN, and NB. The N-BaIoT dataset had 849,234 benign samples and 831,298 assaults. In detecting "Junk" attacks, Naive Bayes performed poorly with 61.52% accuracy. At 85.81%, "ACK" accuracy was better. With over 99% accuracy across all attack types, the J48 DT and ANN outperformed other methods. NB's accuracy increased to 99.10% for Junk attacks and 99.09% for ACK attacks after feature selection, demonstrating the importance of feature selection for cyberattack detection.

In [51], the researchers converted APK files into color images and used a Deep Convolutional Neural Network (DCNN) to extract features from these images for malware detection. They utilized the Leopard Mobile IIOT dataset, which includes 14,733 malware samples and 2,486 benign samples. The proposed method achieved an accuracy of 97.81%. However, limitations of the study include the relatively small size of the dataset and the lack of comparison with other deep learning models.

In [52], the researchers introduced a framework for malware detection, utilizing Radare2 to extract control flow graphs and compute graphical metrics for feature selection during preprocessing. They tested several models—CNNs, RFs, SVMs, and LR on a dataset of 2,962 malware samples randomly selected from CyberIOCs. Among these, the CNN achieved the highest accuracy at 99.66%. Despite the promising results, the study acknowledged several limitations, such as the possibility that static analysis miss malware behaviors that only manifest during execution and the risk that malware could use obfuscation strategies to evade detection at the program level.

The study [53] used techniques such as tokenization, stemming, and stopword removal to build a system for finding malware and illegal software downloads. A deep convolutional neural network and the TFIDF and LogTF weighting methods were used to look at color pictures from malware executable files. The framework was 97.46% accurate at finding malware and piracy, but it could only find malware from known families and couldn't figure out which families weren't known.

An anomaly detection system (ADS) based on DL was introduced in [54]to detect fuzzers, analysis, backdoor, DoS, generic, exploits, reconnaissance, shellcode, and worm attacks in industrial IoT. The suggested ADS initialized the deep feed-forward neural network (DFFNN) during training and testing using DAE results. The authors found old and novel threats using NSL-KDD and UNSW-NB15 datasets. A 99% detection rate was found with the proposed model.

4.5 | Hybrid Frameworks

The authors used Compute Unified Device Architecture (CUDA)-Accelerated Hybrid CNN-DNN to detect IoT network attacks in [55]. They employed CNN for feature extraction and deep neural networks for detection and classification on the kitune dataset. Their model had 98.41% precision and 98.56% recall.

In [56], the research presented a proposed hybrid DL model by using CNN and LSTM. The authors utilized multiple DL architectures on the CICIoT 2023 dataset; nevertheless, the hybrid CNN-LSTM model attained the highest accuracy and performance at 99.23%.

In [57], the authors developed an ensemble classifier for IoT malware detection by integrating LSTM, RNN, and CNN models. They trained their approach using just 10% of the BoT-IoT dataset, which

contains five output categories, 46 features, and 3,000 samples related to keylogging attacks. Validation was performed on the CSE-CIC-ID2018 dataset. The ensemble model surpassed the performance of each individual model, achieving an accuracy of 97.67%, precision of 97.72%, recall of 97.68%, and an F1-score of 97.70%.

CNN and LSTM were used to create a hybrid deep learning model for malware detection utilizing behavioral analysis [58]. Dynamic behavioral data is collected using a log parser analyzer, API monitoring, and extension checker. All modules pre-process. The CNN-LSTM model then finds undesirable behavior using these attributes. The algorithm found 96% of 2,500 malware samples and 1,000 benign samples, proving it can detect advanced and cunning malware threats.

In [59], the study introduced CPL-Net, a malware detection model that proficiently recovers spatiotemporal information from malware texture images through the integration of CNN and LSTM networks. The model's strength was improved during the pre-processing stage by using data improvement methods as rotation, flipping, brightening, darkening, and adding Gaussian noise. The Malimg dataset was used to test the model, which got an accuracy of 98.7% and an F1-score of 98.6%.

The study in [60] combined LSTM and CNN networks to construct a hybrid deep learning model. The model extracted features using Double-Density Discrete Wavelet Transform (D3WT). This approach was tested against Microsoft BIG-2015, Malimg, and IoT malware. The suggested model had 99.98% IoT malware dataset, 96.97% Microsoft BIG-2015, and 99.96% Malimg detection accuracy.

In [61] , although the study focuses on intrusion detection rather than malware detection, introduced a hybrid Deep Learning model combining CNN and LSTM, tested on two public datasets: NSL-Botnet and UNSW-NB15. The model achieved a detection accuracy of 99.4% with the NSL-Botnet dataset and 93% with UNSW-NB15.

In [12] ,this study introduced a hybrid deep learning model using LSTM and CNN. For data preprocessing, the model uses NLP. The authors used Kaggle data. The CNN-LSTM model outperformed traditional classifiers with an F1-score of 1.0, 99% accuracy, 99% precision, and 99% recall. Furthermore, DT was 98% and SVM was 95%. But the pre-processing pipeline and lack of dataset property data are restrictions.

In [62], the study focused on an IoT application running on an ARM CPU, RNN with long LSTM units to detect malware within the IoT network. The LSTM model outperformed other classifiers, RF, NB, SVM, MLP, KDD, DT, and AdaBoost. The dataset used consisted of 281 malware samples and 270 benign samples related to the IoT application, and the model was further evaluated using 100 additional IoT malware samples. The detection accuracy reached up to 98%.

4.6 | Explainable AI Approaches

In [63], this study introduced XAI-AMD-DL, which is a framework integrating Bidirectional Gated Recurrent Units (Bi-GRU) and CNN in the field of Explainable Artificial Intelligence (XAI). When tested on the CICAndMal2019 Android malware dataset, the model did well, getting 97.98% accuracy, 97.75% precision, 97.76% recall, and a 97.75% F1-score.

Table 2. Comparative Summary of Recent IoT Malware Detection Studies (2018–2025).

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
[41]	2025	OPCODE purification technique	--	I-MCM framework with a	97.1	--	--	--	IoT Malware & Benignware (3000 malware +	--	--

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
				Tiny ANN					benignware, ARM/MIPS/X86)		
[55]	2025	--	CNN	CUDA-Hybrid Model (CNN-DNN)	98.41		98.41	98.56	collected from the kitune dataset	--	--
[56]	2025	Label Encoding Z-score normalization	--	CNN-LSTM (Hybrid DL model)	99.23	99	--	--	CICIoT 2023	--	--
[42]	2024	Normalize Encoding Categorical Variables	--	1D CNN	98.36	99.95	100	99.96	CICIoT 2023	Limited multi-dataset validation Real-time deployment not evaluated	Low computational overhead
[4]	2024	BoW	Entropy-(IG) and (GR)	ET	99.98	99.9	99.98	99.98	UNSW-NB15	Single-dataset evaluation, Limited feature representation and No deep learning model.	High detection accuracy and Effective feature engineering.
				KNN	99.97	--	--	--			
		Word2Vec		RF	99.96	--	--	--			
				LR	99.96	--	--	--			
		TF-IDF		DT	99.95	--	--	--			
				MLP	99.96	--	--	--			
[57]	2024	Transform Normalize	-	CNN-RNN-LSTM Ensemble	97.67	97.7	97.72	97.68	Bot-IoT	High computational cost, Limited training data and Complex implementation	Improved ensemble performance sand Cross-dataset validation
					95.03	95.07	95.09	95.05	CSE-CIC-ID2018		
[43]	2024	--	--	ACLR	97.49	97.23	97.7	97.17	UNSW-NB15	Depends on labeled data and Limited real-world generalization	Good scalability
				ANN	75.68	75.59	78.17	75.68			
				CNN	94.40	94.39	94.44	94.40			
				LSTM	96.51	96.51	96.51	96.51			
				RNN	95.22	95.21	95.23	95.22			

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
[44]	2024	TF-IDF	One-Hot	Isolation Forest	--	--	80	100	NetFlow	Autoencoder training on large datasets can be challenging and time-consuming, often requiring significant computer power, such as GPUs or distributed computing systems.	Improved Model Performance: High recall rates were achieved to prioritize malware sample identification.
				Auto encoder	--	--	90	010			
[58]	2024	Log parser analyzer -API monitoring-Extension checker module	CNN and LSTM	Hybrid CNN-LSTM	96	96	--	96	2500 Malware 1000 Benign	The dataset size is relatively small, which may limit the reliability and generalizability of the results.	--
[35]	2023	--	--	MADP-IIME	99.5	--	--	--	IoT Network Intrusion	Performance indicators are based on a certain set of data, thus the findings aren't as useful as they may be.	--
[17]	2023	SMOTE	ACO - GLCM - GLRLM	SVM with PSO	95.56	95.26	94.12	96.43	338 normal images and 512 malicious images.	Using a class with an imbalance in the cost function and only using a single data set.	SVM performs effectively in highdimensional spaces and provides excellent generalization
[59]	2023	Data enhancement techniques such as rotating, flipping, brightening, darkening and adding Gaussian noise	Spatio-temporal feature fusion using CNN and LSTM	CPL-Net: CNN + LSTM	98.7	98.6	--	--	Malimg	--	--
[60]	2023	--	Double-Density Discrete Wavelet Transform (D3WT)	Hybrid CNN + LSTM	99.98	--	--	--	IoT malware	--	--
					96.97	--	--	--	Microsoft BIG-2015		
					99.96	--	--	--	Malimg		
[63]	2023	--	--	XAI-based Hybrid CNN + Bi-GRU (XAI-AMD-DL)	97.98	97.75	97.75	97.76	CICAndMal2019	--	--

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
[36]	2023	--	--	RF	93.3	--	--	--	The dataset was downloaded from VirusTotal, AnyRun, and PolySwarm.	Accuracy was not reported for some models. In addition, machine learning models suffer from insufficient training samples, particularly for fileless malware, which may lead to poor representation of the targeted malware types.	--
				DT	90.9	--	--	--			
				SVM	--	--	--	--			
				LR	86.7	--	--	--			
				KNN	--	--	--	--			
				XGBoost	93.0	--	--	--			
				GB	90.0	--	--	--			
[45]	2022	Bytecode	Chars2Vec	FCN	98.9	98.3	98.9	--	It contains 16,680 available from Google play store and sites such as Amazon for benign dataset collection. and approximately 11,200 malicious apps from virus share	The feature detector may be ineffective against code obfuscation and fails to handle dynamic code loading at runtime.	Handles large-scale datasets effectively without overfitting.
		One-hot encoding of opcodes.		LSTM	98.9	99.0	98.4	--			
[37]	2022	--	EKHO DOBL	DWOML-RWD Model	99.4	99.4	99.4	99.4	420 goodware samples 420 ransomware samples	The database range: it is limited to 840 samples, which may restrict the ability to generalize in the real world and the variables of dynamic ransom programs.	DWO algorithm enhances ELM performance through robust parameter tuning.
[64]	2022	Z-score normalization	Correlation Coefficient Score approach (CCS)	ELBA-IoT (Ensemble Learning)	99.6	97.7	98.4	97.1	N-BaIoT	--	--
[61]	2022	One-hot	--	CNN + LSTM (Hybrid DL model)	99.4	--	--	--	NSL-Botnet	--	--
		Min-Max Scaler			93	--	--	--	UNSW-NB15		
[65]	2022	Cuckoo Sandbox using Static and Dynamic analysis	--	Detection Filters	95	--	--	--	194 ransomware samples (10 types, 46 variants)	The lack of real-time detection for next-generation ransomware variants ,specific preprocessing, and advanced machine learning methods	Reduced latency in ransomware detection (0.025–0.03 ms)

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
[12]	2022	NLP	Automated high-level abstraction extraction using CNN + LSTM	Hybrid CNN-LSTM	99	1	99	99	collected on the website Kaggle	--	--
[66]	2021	--	--	Artificial Neural Network (ANN)	92.8	99	1	99	N-BaIoT	It didn't look at advanced feature engineering or interpretability techniques, the study only provided a limited amount of information on model transparency.	Significant discoveries on affordable learning systems for identifying Mirai Malware strains
[67]	2021	Convert the malware to an binaries RGB image representation	--	CNN-based approach	98.57	98.62	98.79	98.47	Samples for Trojans, worms and backdoors and sizes of these samples range from 30KB to 5000KB	No direct comparison with other methods has been made.	--
[46]	2021	--	Logistic Regression (LR) algorithm	ResNetDDoS-1	98.99	99.04	98.91	99.18	CICIDS19	Focusing on limited attack types restricts coverage, requiring validation on more diverse datasets	Effective feature selection techniques to improve model performance and reduce complexity.
					97.95	95.30	95.66	99.44	CICIDS17		
					99.16	98.88	98.02	99.75	Bot-IoT		
					98.70	97.74	97.53	99.46	Average Result		
				ResNetDDoS-2	70.22	56.51	96.67	39.92	DDoSLab		
					78.41	40.86	63.01	31.03	CICIDS17		
					68.58	27.60	93.56	12.57	Bot-IoT		
					72.40	30.13	84.41	27.84	Average Result		
				ResNetD	64.15	9.79	70.81	5.26	DDoSLab		

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
				DoS-3	49.60	9.91	81.28	5.27	CICIDS19		
					64.15	9.79	70.81	5.26	Bot-IoT		
					58.77	19.36	82.28	11.53	Average Result		
				ResNetDDoS-4	71.75	59.03	99.30	71.75	DDoSLab		
					47.53	0.44	70.31	0.63	CICIDS19		
					78.14	1.29	56.90	6.05	CICIDS17		
					65.81	20.25	75.50	14.29	Average Result		
[15]	2021	Image conversion from binary to grayscale	--	CNN-DMA	99	**--**	--	--	Malimg	Imbalance between malware classes and small image size	Strong against methods that hide things, such encryption and packaging.
[47]	2021	--	--	ANN	92.9	--	--	86.5	consists of 115 features and 49548 sets for each Mirai and Benign	--	--
[16]	2020	--	--	FNN and ANFIS	83	--	--	--	NSL-KDD	Small testing environment and dataset scope	Lightweight design with reduced impact on system latency.
[48]	2020	Bat Algorithm	--	DAIMD Model using CNN	99.28	--	--	--	840 malware images (training) 561 files (testing)	The training dataset's quality and diversity have a big impact on the model's efficacy and how well it generalizes to new malware variants.	Dynamic Analysis technique eliminates the need for subjective intervention and enables the detection of both known and novel IoT malware types.
[49]	2020	--	IG OpCode	MTHAEL	99.98	99.94	99.96	99.97	Collected from sources like Virus-Share, Detux, and IoTPoT	Not all possible IoT architectures are fully covered by the dataset.	Low Overhead
[18]	2020	n-gram	TF-IDF	RNN	98.71	**--**	--	--	The dataset consisted of system calls collected by Authors	The study is limited by a small dataset size and the use of a single hidden layer in the model.	The use of n-gram and TF-IDF techniques allowed for meaningful feature extraction from

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
		TF-IDF									system calls, enhancing classification performance.
[50]	2020	--	--	ANN	99	--	--	--	N-BaIoT	The performance varies based on classifier choice.	--
				J48 Decision Tree	99	--	--	--			
				NB	99.1	--	--	--			
[51]	2020	--	DCNN	DCNN	97.81	95.13	95.16	95.10	IIOT dataset, namely, Leopard Mobile dataset	Very small dataset size and the absence of comparative analysis with other deep learning models.	--
[52]	2019	CFG	--	CNN	99.66	--	--	--	2,962 malware samples, randomly selected from CyberIOCs	Malware is vulnerable to obfuscation techniques, affecting detection performance at program level, and static analysis may overlook dynamic malware behavior.	High accuracy and low error rates using CNN for IoT malware detection and classification.Use of CFG-based features for distinguishing IoT and Android malware effectively
				SVM	97.65	--	--	--			
		Radare2		RF	98.48	--	--	--			
				LR	97.47	--	--	--			
[38]	2019	packet traffic capture	--	KNN	94.44	96	92	1	Simulated IoT traffic dataset	Difficulty in detecting undiscovered new malware, and lacks transparency in documenting the source of the data, making it difficult to verify or reproduce the results.	Scalable for large networks.
				RF	88.8	--	--	--			
				GNB	77.78	--	--	--			
[68]	2019	Clustering algorithm	--	NB	98	98	98.2	--	Collected from Google Play and the Chinese app stores includes 6192 benign and 5560 malware	The study has problems with hiding features when decompiling APK files using Dex2jar.	--
[69]	2019	VAE-CAE	--	DNN-BN	95.38	94.71	96.99	--	Included observations of system API packages	The preprocessing steps are complex and require significant computational	DNN with batch normalization provided improved

RF	Year	Method			Result %				Dataset	Limitation	Advantage
		Pre-processing	Feature Selection	Detection	Acc	F1-score	Precision	Re-call			
										resources.	generalization, improving the model's ability to adapt to new ransomware behaviors.
[53]	2019	Tokenization Stemming Stopword Removal	TFIDF LogTF	DCNN	97.46	97.44	97.43	97.46	Collected from Google Code Jam (GCJ) and Leopard Mobile	There is no mechanism to detect malware from unknown families; only malware from known families is detected.	---
[54]	2018	--	---	DAE + DFFNN (Deep Autoencoder + DFFNN)	98.6	--	--	--	NSL-KDD	--	--
					92.4	--	--	--	UNSW-NB15		
[39]	2018	--	--	SVM	91.19	88.44	83.33	94.20	The dataset was generated by monitoring the power usage of targeted applications on three different Android	Unpredictable power consumption patterns and less promise from conventional classification methods like SVM, KNN, and NN.	Evaluated multiple classification algorithms and window sizes, providing a thorough analysis of their performance.
				KNN	94.27	92.31	89.19	95.65			
				RF	87.56	82.09	85.94	78.57			
				NN	75.93	67.01	61.68	73.33			
[70]	2018	--	PSI Graph	CNN classifier based on PSI graph	92	94	--	--	10033 ELF files including 4002 IoT botnet samples and 6031 benign files	It does not give performance on multiple attack types.	--
[62]	2018	--	**--**	LSTM-based RNN	98.18	--	--	--	281 malware 270 benign ware	The study is limited by a small dataset size	**--**
[40]	2018	--	--	DT	95.32	--	--	--	UNSW-NB15	Complexity of Modern Attacks Dependence on Feature Quality Generalizability	Enhanced Detection Performance: The use of the AdaBoost ensemble method leads to improved detection rates and reduced false positives.
					96.10	--	--	--	NIMS		
				NB	91.17	--	--	--	UNSW-NB15		
					88.28	--	--	--	NIMS		
				ANN	92.61	--	--	--	UNSW-NB15		
					94.22	--	--	--	NIMS		
				Ensemble Method	99.54	98.93	--	--	UNSW-NB15		
				AdaBoost	98.29	97.38	--	--	NIMS		

A broad variety of ML and DL techniques have been investigated by researchers for the purpose of detecting malware in Internet of Things environments, as is demonstrated in Table 2. Deep learning-based methods, particularly CNN, RNN and LSTM architectures, have a tendency to outperform classical machine learning models in terms of accuracy and detection rate. This is something that is readily apparent. However, many of the systems that are already in use continue to struggle with issues with the imbalance of datasets, scalability, and the ability to detect events in real time.

5 |Discussion and Analysis

In this section, we present a thorough analysis of the reviewed studies to identify the primary research directions, methodologies, and performance results in IoT malware detection. The discussion emphasizes the progression of the field from conventional machine learning (ML) methods to sophisticated deep learning (DL) and hybrid architectures that integrate the advantages of various techniques. Furthermore, it analyzes the variety of benchmark datasets including UNSW-NB15, NSL-KDD, N-BaIoT, BoT-IoT, CICIoT2023, CICAndMal2019, and CSE-CIC-ID2018, which have been instrumental in the experimental assessment of detection frameworks. By analyzing the reported results and methodologies, this section seeks to identify the strengths, shortcomings, and research gaps that persist in shaping the field of IoT malware detection.

5.1 | Descriptive Analysis

The literature research shows that there are many different ways to find IoT malware, from traditional machine learning models (such Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB), and Decision Tree (DT)) to deep learning architectures (like CNN, LSTM, and RNN), hybrid models, and image-based or opcode-level analysis.

Traditional machine learning techniques generally work well when they have well-designed features. For instance, ensemble-based machine learning methods worked effectively on a number of datasets, especially when they were used alongside feature selection methods as shown in studies [4], [40], [36], [39], and [50].

These models benefit from simplicity, efficiency, and suitability for low-resource environments, with accuracies reaching up to 99.98% in [4]. However, ML models repeatedly demonstrated sensitivity to dataset quality, imbalance, and restricted feature variety, especially in research concerning fileless malware or datasets lacking adequate representativeness. For example, the AdaBoost ensemble in [40] performs exceptionally well on UNSW-NB15 but shows instability when faced with more complex or heterogeneous traffic patterns.

Deep learning–based approaches exhibit stronger generalization and consistently outperform traditional ML. CNN-based architectures [15], [42], [52], image-based methods [48], [67], sequence-modeling techniques using RNN, LSTM, or Bi-GRU [45], [18], [62], and combined CNN–LSTM pipelines [56], [12], [59] achieve detection accuracies exceeding 98% across a variety of datasets including CIC-IoT 2023, Malimg, and BoT-IoT. These models automatically learn hierarchical behavioral patterns and are more resilient to noise and variations in malware behavior. However, their computational cost makes them less suitable for deployment on constrained IoT devices. Static CNN-based approaches (e.g., [52], [15]) also face limitations against code obfuscation and dynamic malware behavior.

In the literature, hybrid deep learning models have demonstrated strong performance in several studies [57], the multi-transform hybrid MTHAEL model in [49], the D3WT-enhanced CNN–LSTM in [60], and the XAI-supported CNN-BiGRU architecture in [63] frequently outperform ML and single-model DL methods, with accuracies nearing or exceeding 99.9%. These models combine spatial and temporal variables, which makes it easier to find advanced malware families and makes it easier to use across different datasets. The MTHAEL [49], the D3WT hybrid system [60], the DL ensemble in [57], and the XAI-AMD-

DL approach in [63] have demonstrated high performance, robustness, and methodological improvements in their respective studies.

Even with these improvements, the field still has problems that won't go away. One of the biggest problems is with datasets: many studies use limited datasets like [68], [37], [62], [39], and [58], samples that are very unbalanced like ransomware detection, or collections that focusing on certain types of malware, like Mirai or Bashlite. Also, some models are tested on only one dataset, which makes it harder to prove that they are better than others as shown in [15], [52] and [62].

Research employing simulated datasets [38] or controlled contexts encounters restricted real-world applicability. Static-analysis–based solutions are still open to obfuscation and polymorphism [52], and dynamic approaches are quite slow [48], [58]. Only a few research look at explainability, the most important of which is [63]. This shows that there is a big gap in the use of XAI.

Current trends reveal a shift toward multi-modal learning, representation-based feature extraction (e.g., opcode embedding in [31], grayscale/RGB transformation in [15], [48], [67], the adoption of NLP-based techniques using opcodes and system calls [18], [45] ,wavelets in [60], PSI graphs in [54]), and hybrid DL architectures that integrate CNNs with LSTM, RNN, or transformer-inspired modules. There is also an increasing focus on using multiple datasets for validation as shown in [57] and [60] to enhance reliability. However, challenges such as computational efficiency, model interpretability, adversarial robustness, and real-time detection remain largely unaddressed.

Research gaps identified include the need for large-scale, real-world IoT datasets covering diverse architectures and obfuscation techniques; lightweight DL models for edge deployment; explainable detection frameworks; adversarially robust architectures; and standardized evaluation protocols incorporating cross-dataset and cross-device validation. Existing studies rarely explore zero-day detection or adversarial attack resilience, despite their importance in practical IoT environments.

To get efficient on-device detection, it is important to create lightweight deep learning models like TinyML or quantized architectures. A further significant objective is to generate huge, realistic IoT malware datasets that show a wide range of devices, behaviors, and ways to cover up malware. Although lightweight edge-oriented models reduce computational complexity and enable real-time malware detection on resource-constrained devices, they often involve a trade-off between model size and detection performance. Techniques such as model pruning, quantization, and knowledge distillation can significantly reduce memory usage and inference latency; however, maintaining high detection accuracy while minimizing resource consumption remains a major implementation challenge for edge-based IoT systems.

To make deep learning pipelines more open and trustworthy, they need to include explainable AI (XAI). To address this gap, implementing Explainable AI (XAI) algorithms into the process of IoT malware detection is crucial to ensure transparency, post-hoc interpretability, and trustworthiness. The application of post-hoc interpretability approaches, such as SHapley Additive exPlanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME), is very important for malware detection because these approaches determine the contribution of every feature to the classification result whether it is an individual API system call, network traffic characteristic, or even a certain set of binary codes. For security experts, these explanations make deep learning models' predictions understandable by showing what exact malicious actions or patterns cause the alarm. For example, in study [49], the application of the explainability algorithms within the hybrid CNN-BiGRU model proves how the interpretation framework can verify the decisions of the machine learning algorithm using threat intelligence and, thus, can prove that the model does not use any random associations and biased data. Therefore, the transition from mentioning of XAI approaches to their actual implementation becomes a vital necessity in developing future security systems.

Finally, future systems should be thoroughly tested against attacks from outside sources and situations in other domains to make sure they will work well in the real world. These improvements will help bridge the gap between the carefully controlled academic outcomes and the real-world needs of IoT security systems.

This review highlights that while ML models provide efficiency and interpretability, DL and hybrid approaches deliver superior accuracy and robustness.

The best-performing systems in recent literature particularly MTHAEL [49], Hybrid CNN–LSTM+D3WT [60], CNN–RNN–LSTM ensembles [57], and XAI-AMD-DL in [63] have demonstrated advanced and effective approaches for IoT malware detection in their respective studies.

5.2 | Comparative Critical Analysis

The above-reviewed literature indicates that the latest advancements made in detecting the IoT malware via ML, DL, and hybrid intelligent systems have led to higher accuracy in their application. Nevertheless, a comparative analysis shows that just having high performance of classifiers is not enough to evaluate the practical efficiency of these methods used in IoT. Other important evaluation criteria include computation cost, scalability, generalization ability, and applicability. Figure 1 summarizes the evolution of IoT malware detection research from traditional machine learning approaches toward hybrid, explainable, and deployment-oriented AI frameworks, highlighting the major research trends identified throughout this review.

5.2.1 | Comparative Analysis of Detection Approaches

Nevertheless, machine learning techniques such as Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), Naïve Bayes (NB) and K-Nearest Neighbors (KNN) are attractive due to low computational demand and quick inference. Several works show the ability to achieve a competitive detection performance while providing lightweight models suitable for IoT devices [4], [40], [36], [39], and [50]. At the same time, the performance of these approaches depends strongly on manual feature engineering and feature selection, making them adaptable to emerging malware.

Deep learning methods eliminate most of these issues since they automatically learn hierarchical features from malware data. Methods based on CNN, RNN, LSTM, and Bi-GRU showed excellent detection performance on various benchmarks [15, 56] [42], [38], [63] , [42], and [62]. However, according to the literature review, these models are characterized by high computational complexity, long training time, and significant memory consumption. As a result, several authors noted the challenges of deploying such models due to their processing overhead [57], [58], and [48].

The hybrid architecture appears to be one of the most promising directions as it allows to utilize the advantages of various learning paradig and complement them. For example, the combination of CNN, RNN and LSTM showed the highest level of detection performance[57]; similarly, Hybrid CNN-LSTM[60], MTHAEL framework [49] and XAI-AMD-DL architecture [63] reached some of the best results shown in the literature. Still, these achievements are paid off with increased complexity, higher computational costs and more advanced hardware requirements, making the implementation challenging.

In addition, complex hybrid architectures often require greater memory resources, longer inference time, and higher energy consumption, which may hinder their deployment on resource-constrained IoT and edge devices. Therefore, future hybrid malware detection frameworks should balance detection performance with computational efficiency to ensure practical real-time deployment in IoT environments[71].

The more recent works in 2025 have provided additional evidence to this claim by showing the evident trend of IoT malware detection framework moving towards the deployment approach. Thus, the I-MCM framework stresses lightweight cross-architecture malware detection based on Tiny ANN, and the CUDA-based CNN-DNN architecture is concerned about computational efficiency by making use of GPU acceleration. The hybrid CNN-LSTM architecture utilizes both spatial and temporal features to improve

detection capability. Overall, these works imply that the current research efforts are not only concentrating on improving detection performance but also creating the practical IoT malware detection framework.

5.2.2 | Challenges in Model Evaluation and Generalization

One significant point to note about the examined literature is that even though high detection accuracies have been observed by most studies, performance measures such as false positive rate (FPR), false negative rate (FNR), inference time, memory consumption, and energy efficiency have not been systematically evaluated or have been ignored completely. Thus, the assessment of malware detection models only on the basis of accuracy will not be a complete evaluation of the model as far as its usability in actual edge-based IoT environments is concerned.

In view of the constraints imposed on IoT devices due to limitations in computation power, memory storage, and energy, such factors need to be included along with the accuracy factor for the evaluation of malware detection approaches. Currently, only a minimal subset of the reviewed literature evaluates these practical dimensions. Specifically, in [46] addressed inference latency by achieving an optimized, rapid response window of 0.025–0.03 ms for ransomware prevention.

Furthermore, in [61] systematically pivoted from traditional behavioral logging to monitoring device energy consumption footprints to capture ransomware signatures, highlighting the practical necessity of energy efficiency metrics. Future architectures must transition toward holistic benchmarking frameworks that weigh detection capabilities equally against on-device resource constraints.

One of the most important observations made based on the literature review is that several models for malware detection recently designed and trained tend to achieve extremely high accuracies, surpassing 99% in some cases. Even though the success of such models shows how much has been accomplished within machine learning and hybrid approaches to malware detection, the performance alone cannot be taken as an indication of the robustness and effectiveness of a model. The high performance can be achieved due to various reasons like overfitting, biases in the data sets used, class imbalance problems, and evaluation on only one benchmark dataset. Moreover, models that have been tested only within experiments might not perform similarly in the heterogeneous real world of IoT where there are different types of devices and evolving malware threats.

Another interesting issue that should be mentioned in the course of this work is the generalization ability of the studied malware detection models. Many papers demonstrate the excellent performance in experiments, but quite a number of the reviewed studies evaluates the proposed methods using only a single benchmark dataset or very limited datasets[42] , [58], [37] , [52] , and[62]. Such an approach makes it impossible to evaluate how well the model performs in heterogeneous IoT environments, different devices architectures and unseen malware families.

Additionally, depending on one dataset could inadvertently cause bias towards the particular dataset, thus making the model overly specific in terms of the dataset's features rather than being general with regard to malware characteristics[27]. Hence, the reported performance might be an artifact that fails to provide an accurate picture of how the model would perform in real-life conditions with various IoT devices, networks, and malware types. For future research, cross-validation with the use of several publicly available benchmarks should be considered.

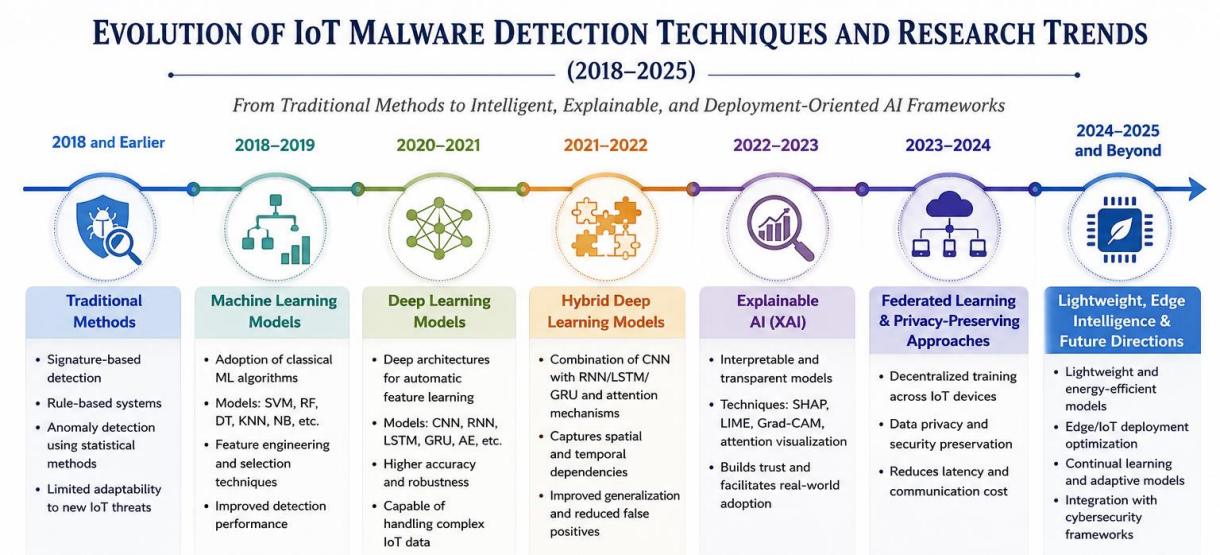

Figure 1. Evolution of AI-Based IoT Malware Detection Techniques and Emerging Research Trends (2018–2025).

The figure provides a visual overview of the progression from conventional machine learning methods to recent deployment-oriented, explainable, and lightweight AI-based approaches, thereby improving the overall readability of the survey.

5.2.3 | Practical Deployment Challenges and Future Directions

In the future, work should be performed to create multi-modal hybrid detection frameworks that combine static, dynamic, and network-level data to make them more reliable. Federated learning seems as well like a good way to allow collaborative model training while keeping data private in dispersed IoT environments. However, deploying federated learning in distributed IoT environments presents several practical challenges. IoT devices often differ significantly in their computational capabilities, communication bandwidth, storage capacity, and data distribution. These heterogeneous conditions increase communication overhead, complicate model synchronization, and may slow the convergence of the global model. Furthermore, the non-IID nature of the data generated through the use of the IoT sensors makes the task of training even more difficult and may impact negatively on the convergence of the global model as well as its ability to generalize. Moreover, even though federated learning ensures the privacy of the data by storing the data locally, security mechanisms are required for the data transfer between models[72, 73]. Consequently, designing communication efficient and resource-aware federated learning frameworks remains an important research direction for practical IoT malware detection.

Considering the deployment issues, few works discuss the practical issues of implementing malware detection models on edge IoT devices, such as the inference latency, energy consumption, memory usage or other efficiency-related metrics. Almost all the frameworks described in the literature were examined in laboratory conditions without taking into account the real-world implementation. Thus, the development of lightweight architectures, model compression techniques, federated learning, explainable artificial intelligence (XAI) and efficient edge computing is the direction of the future work.

Apart from the existing lightweight neural network designs, contemporary studies have paid increasing attention to the potential of TinyML in implementing AI solutions on IoT devices with restricted resources. The use of TinyML provides the possibility of performing deep learning models in the device itself using small amounts of memory, low computing cost, and less energy consumption, which helps reduce communication delay and enhance data confidentiality[74]. In addition, various model compression methods like pruning, quantization, knowledge distillation, and neural network architecture engineering

have become the effective tools for making AI models smaller and less costly in inference at the same time retaining competitive detection capacity. This is especially important for IoT devices running on battery power with limited computing resources, and thus making lightweight AI models crucial for further research in IoT malware detection.

Another limitation observed across the reviewed literature is that model evaluation still relies predominantly on overall detection accuracy, whereas considerably less attention is paid to evaluation issues that directly affect practical deployment. Although several studies report excellent classification performance, important factors such as class imbalance may significantly influence the interpretation of these results[75]. In highly imbalanced IoT malware datasets, high accuracy does not necessarily indicate reliable detection of minority attack classes. Therefore, complementary evaluation measures such as precision, recall, F1-score, and false positive rate should be considered together with standardized evaluation protocols to provide a more comprehensive assessment of malware detection systems.

Another critical problem that has been discovered from the reviewed literature is the low reproducibility and lack of experimental transparency of many existing IoT malware detection techniques. While many techniques have shown promising results, very few have provided the code and exact settings of their experiments so that researchers can independently validate their findings. Moreover, while there are publicly available benchmarking datasets such as UNSW-NB15 and CICIDS2017, different data preprocessing steps, feature extraction process, and train-test splits make it hard to compare these techniques directly. Hence, improvement of experimental transparency is highly needed to ensure reliable benchmarking and further development of the field.

Overall, the reviewed literature indicates that, according to the reviewed literature, none of the approaches fulfills all the requirements of modern IoT malware detection system. While the machine learning methods are characterized by low computational demands and easy implementation, deep learning models demonstrate better feature learning and detection capabilities. The hybrid models further improve the classification performance, but increase the implementation complexity. Thus, the malware detection frameworks of the future need to strike a balance between these criteria. To provide a clearer cross-study comparison of the reviewed approaches, Table 3 summarizes the major findings of the literature in terms of detection accuracy, computational complexity, generalization capability, scalability, and deployment feasibility.

Table 3. Comparative Analysis of IoT Malware Detection Approaches Based on the Reviewed Literature.

Evaluation Criterion	Main Findings from Reviewed Studies	Representative References
Detection Accuracy	Hybrid and DL models consistently reported the highest detection accuracy.	[57], [60] , [63] , [49]
Computational Cost	DL and Hybrid models generally require higher computational resources.	[57], [58] , [48]
Generalization	Most studies relied on a single benchmark dataset, limiting model generalization.	[42], [58] , [37] , [62]
Scalability	Few studies evaluated scalability in real IoT environments.	[57], [49], [40]
Deployment Feasibility	Practical deployment on resource-constrained IoT devices remains insufficiently addressed.	[57], [58] , [48]

5.2.4 | Open Challenges in Real-Time IoT Malware Deployment

While significant progress has been made towards the development of effective IoT malware detection systems, there are still a number of practical obstacles that prevent the real-time application of such systems. First, many IoT devices are resource-constrained with limited computing capabilities, memory, and energy reserves, which makes it difficult to apply computationally costly models based on deep learning algorithms. Furthermore, one of the key issues that needs to be solved is the minimization of inference latency and

maximization of detection accuracy, especially in the context of mission-critical IoT applications. Additional practical challenges, include communication costs in distributed and federated settings, device heterogeneity, the lack of model explainability, and the absence of benchmarking procedures.

6 | Conclusion

This review paper provides a comprehensive and critical analysis of IoT malware detection methodologies developed in recent years, elucidating the evolution of machine learning, deep learning, hybrid analytics, and non-AI techniques in response to the growing complexity of IoT threats. The comparative analysis shows that classical ML models still operate well when they are backed up by strong feature engineering. However, deep learning and hybrid frameworks always perform better since they use automated representation learning and multi-modal behavioral analysis. However, the literature also indicates several persistent challenges that remain unresolved. For example, many studies rely on small or unbalanced datasets, don't test their findings in a variety of IoT settings, require a lot of computing power that makes real-time deployment difficult, and deep models often aren't easy to understand.Key difficulties include detecting zero-day malware, not being able to be manipulated by adversaries, protecting data privacy, and making on-device inference more energy-efficient are still mostly unsolved. Future research should concentrate on creating lightweight deep learning architectures tailored for edge devices, amalgamating federated and distributed learning frameworks, augmenting publicly accessible IoT malware datasets, enhancing robustness via adversarial training, and integrating explainable AI to foster trust and operational transparency. In a practical sense, this review offers valuable suggestions for researchers and industry professionals concerned with the cybersecurity of IoT devices. To the researcher, the comparison analysis discussed in this article shows the gaps in current research, emerging trends, and future direction for developing resilient, explainable, and efficient malware detection systems. To industry professionals, the conclusions drawn from the review show that choosing detection models that consider both prediction accuracy and computing efficiency is vital. The suggestions provided in this article may help in developing reliable AI-based malware detection systems for IoT devices.

Author Contribution

All authors contributed equally to this work.

Funding

This research received no funding.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] G. Kołaczek, "Internet of Things (IoT) Technologies in Cybersecurity: Challenges and Opportunities," *Applied Sciences,* vol. 15, p. 2935, 2025.

[2] S. S. Sefati, B. Arasteh, S. Halunga, and O. Fratu, "A comprehensive survey of cybersecurity techniques based on quality of service (QoS) on the Internet of Things (IoT)," *Cluster Computing,* vol. 28, p. 792, 2025.

[3] D. Canavese, L. Mannella, L. Regano, and C. Basile, "Security at the edge for resource-limited IoT devices," *Sensors,* vol. 24, p. 590, 2024.

[4] M. Azeem, D. Khan, S. Iftikhar, S. Bawazeer, and M. Alzahrani, "Analyzing and comparing the effectiveness of malware detection: A study of machine learning approaches," *Heliyon,* vol. 10, 2024.

[5] Ö. A. Aslan and R. Samet, "A comprehensive review on malware detection approaches," *IEEE access,* vol. 8, pp. 6249-6271, 2020.

[6] E. S. Alomari, R. R. Nuiaa, Z. A. A. Alyasseri, H. J. Mohammed, N. S. Sani, M. I. Esa*, et al.*, "Malware detection using deep learning and correlation-based feature selection," *Symmetry,* vol. 15, p. 123, 2023.

[7] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, and S. Venkatraman, "Robust intelligent malware detection using deep learning," *IEEE access,* vol. 7, pp. 46717-46738, 2019.

[8] H. Darabian, S. Homayounoot, A. Dehghantanha, S. Hashemi, H. Karimipour, R. M. Parizi*, et al.*, "Detecting cryptomining malware: a deep learning approach for static and dynamic analysis," *Journal of Grid Computing,* vol. 18, pp. 293-303, 2020.

[9] A. Yazdinejad, H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, G. Srivastava, and M.-Y. Chen, "Cryptocurrency malware hunting: A deep recurrent neural network approach," *Applied Soft Computing,* vol. 96, p. 106630, 2020.

[10] S. Jeon and J. Moon, "Malware-detection method with a convolutional recurrent neural network using opcode sequences," *Information Sciences,* vol. 535, pp. 1-15, 2020.

[11] P. Thakur, V. Kansal, and V. Rishiwal, "Hybrid deep learning approach based on lstm and cnn for malware detection," *Wireless Personal Communications,* vol. 136, pp. 1879-1901, 2024.

[12] M. S. Akhtar and T. Feng, "Detection of malware by deep learning as CNN-LSTM machine learning techniques in real time," *Symmetry,* vol. 14, p. 2308, 2022.

[13] Q.-D. Ngo, H.-T. Nguyen, V.-H. Le, and D.-H. Nguyen, "A survey of IoT malware and detection methods based on static features," *ICT Express,* vol. 6, pp. 280-286, 2020/12/01/ 2020.

[14] M. M. Rahman, S. A. Shakil, and M. R. Mustakim, "A survey on intrusion detection system in IoT networks," *Cyber Security and Applications,* vol. 3, p. 100082, 2025/12/01/ 2025.

[15] A. Anand, S. Rani, D. Anand, H. M. Aljahdali, and D. Kerr, "An Efficient CNN-Based Deep Learning Model to Detect Malware Attacks (CNN-DMA) in 5G-IoT Healthcare Applications," *Sensors,* vol. 21, p. 6346, 2021.

[16] F. Farhin, I. Sultana, N. Islam, M. S. Kaiser, M. S. Rahman, and M. Mahmud, "Attack Detection in Internet of Things using Software Defined Network and Fuzzy Neural Network," in *2020 Joint 9th International Conference on Informatics, Electronics & Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, 2020, pp. 1-6.

[17] A. El-Ghamry, T. Gaber, K. K. Mohammed, and A. E. Hassanien, "Optimized and Efficient Image-Based IoT Malware Detection Method," *Electronics,* vol. 12, p. 708, 2023.

[18] M. Shobana and S. Poonkuzhali, "A novel approach to detect IoT malware by system calls using Deep learning techniques," in *2020 International Conference on Innovative Trends in Information Technology (ICITIIT)*, 2020, pp. 1-5.

[19] E. Safeer, S. Tahir, A. Nawaz, M. Humayun, M. Shaheen, and M. Khan, "Advanced hybrid malware identification framework for the Internet of Medical Things, driven by deep learning," *Security and Privacy,* vol. 8, p. e454, 2025.

[20] H. Alkahtani and T. H. Aldhyani, "Botnet attack detection by using CNN-LSTM model for Internet of Things applications," *Security and Communication Networks,* vol. 2021, p. 3806459, 2021.

[21] S. Abijah Roseline and S. Geetha, "A comprehensive survey of tools and techniques mitigating computer and mobile malware attacks," *Computers & Electrical Engineering,* vol. 92, p. 107143, 2021/06/01/ 2021.

[22] S. Riaz, S. Latif, S. M. Usman, S. S. Ullah, A. D. Algarni, A. Yasin*, et al.*, "Malware Detection in Internet of Things (IoT) Devices Using Deep Learning," *Sensors,* vol. 22, p. 9305, 2022.

[23] Y. Song, D. Zhang, J. Wang, Y. Wang, Y. Wang, and P. Ding, "Application of deep learning in malware detection: a review," *Journal of Big Data,* vol. 12, p. 99, 2025/04/22 2025.

[24] S. Sasikala and S. Janakiraman, "A Review on Machine Learning-based Malware Detection Techniques for Internet of Things (IoT) Environments," *Wireless Personal Communications,* vol. 132, pp. 1961-1974, 2023/10/01 2023.

[25] W. Almobaideen, O. Abu Alghanam, M. Abdullah, S. B. Hussain, and U. Alam, "Comprehensive review on machine learning and deep learning techniques for malware detection in android and IoT devices," *International Journal of Information Security,* vol. 24, p. 110, 2025/04/09 2025.

[26] S. Rathore and J. H. Park, "Semi-supervised learning based distributed attack detection framework for IoT," *Applied Soft Computing,* vol. 72, pp. 79-89, 2018/11/01/ 2018.

[27] A. Khraisat and A. Alazab, "A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges," *Cybersecurity,* vol. 4, p. 18, 2021/03/08 2021.

[28] S. Arisdakessian, O. Wahab, A. Mourad, H. Otrok, and M. Guizani, "A Survey on IoT Intrusion Detection: Federated Learning, Game Theory, Social Psychology and Explainable AI as Future Directions," *IEEE Internet of Things Journal,* vol. PP, pp. 1-1, 01/01 2022.

[29] Y. Ali, H. U. Khan, and M. Khalid, "Engineering the advances of the artificial neural networks (ANNs) for the security requirements of Internet of Things: a systematic review," *Journal of Big Data,* vol. 10, p. 128, 2023/08/14 2023.

[30] A. Nazir, J. He, N. Zhu, A. Wajahat, X. Ma, F. Ullah*, et al.*, "Advancing IoT security: A systematic review of machine learning approaches for the detection of IoT botnets," *Journal of King Saud University - Computer and Information Sciences,* vol. 35, p. 101820, 2023/12/01/ 2023.

[31] A. Ghaffari, N. Jelodari, S. pouralish, N. derakhshanfard, and B. Arasteh, "Securing internet of things using machine and deep learning methods: a survey," *Cluster Computing,* vol. 27, pp. 9065-9089, 2024/10/01 2024.

[32] S. U. Qureshi, J. He, S. Tunio, N. Zhu, A. Nazir, A. Wajahat*, et al.*, "Systematic review of deep learning solutions for malware detection and forensic analysis in IoT," *Journal of King Saud University - Computer and Information Sciences,* vol. 36, p. 102164, 2024/10/01/ 2024.

[33] O. Alshamsi, K. Shaalan, and U. Butt, "Towards Securing Smart Homes: A Systematic Literature Review of Malware Detection Techniques and Recommended Prevention Approach," *Information,* vol. 15, p. 631, 2024.

[34] V. Pai, B. H. Karthik Pai, G. S. Sudhiksha, V. Kamath, K. Varsha, and S. Manjunatha, "Systematic Approach for Malware Detection in IoT Devices: Enhancing Security and Performance," *International Journal of Computational Intelligence Systems,* vol. 18, p. 196, 2025/07/29 2025.

[35] S. Pundir, M. S. Obaidat, M. Wazid, A. K. Das, D. P. Singh, and J. J. P. C. Rodrigues, "MADP-IIME: malware attack detection protocol in IoT-enabled industrial multimedia environment using machine learning approach," *Multimedia Systems,* vol. 29, pp. 1785-1797, 2023/06/01 2023.

[36] O. Khalid, S. Ullah, T. Ahmad, S. Saeed, D. A. Alabbad, M. Aslam*, et al.*, "An Insight into the Machine-Learning-Based Fileless Malware Detection," *Sensors,* vol. 23, p. 612, 2023.

[37] K. A. Alissa, D. H. Elkamchouchi, K. Tarmissi, A. Yafoz, R. Alsini, O. Alghushairy*, et al.*, "Dwarf Mongoose Optimization with Machine-Learning-Driven Ransomware Detection in Internet of Things Environment," *Applied Sciences,* vol. 12, p. 9513, 2022.

[38] A. Kumar and T. J. Lim, "EDIMA: Early Detection of IoT Malware Network Activity Using Machine Learning Techniques," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, 2019, pp. 289-294.

[39] A. Azmoodeh, A. Dehghantanha, M. Conti, and K.-K. R. Choo, "Detecting crypto-ransomware in IoT networks based on energy consumption footprint," *Journal of Ambient Intelligence and Humanized Computing,* vol. 9, pp. 1141-1152, 2018/08/01 2018.

[40] N. Moustafa, B. Turnbull, and K.-K. R. Choo, "An Ensemble Intrusion Detection Technique Based on Proposed Statistical Flow Features for Protecting Network Traffic of Internet of Things," *IEEE Internet of Things Journal,* vol. PP, pp. 1-1, 09/24 2018.

[41] I. Gulatas, H. H. Kilinc, A. H. Zaim, and M. A. Aydin, "I-MCM: IoT Malware Counter Measures for Cross-Architecture IoT Malware Detection," *IEEE Access,* vol. 13, pp. 95524-95534, 2025.

[42] B. Taşcı, "Deep-Learning-Based Approach for IoT Attack and Malware Detection," *Applied Sciences,* vol. 14, p. 8505, 2024.

[43] M. Ali, M. Shahroz, M. F. Mushtaq, S. Alfarhood, M. Safran, and I. Ashraf, "Hybrid Machine Learning Model for Efficient Botnet Attack Detection in IoT Environment," *IEEE Access,* vol. 12, pp. 40682-40699, 2024.

[44] T. Shi, R. A. McCann, Y. Huang, W. Wang, and J. Kong, "Malware Detection for Internet of Things Using One-Class Classification," *Sensors,* vol. 24, p. 4122, 2024.

[45] M. Amin, D. Shehwar, A. Ullah, T. Guarda, T. A. Tanveer, and S. Anwar, "A deep learning system for health care IoT and smartphone malware detection," *Neural Computing and Applications,* vol. 34, pp. 11283-11294, 2022/07/01 2022.

[46] F. Hussain, S. G. Abbas, I. M. Pires, S. Tanveer, U. U. Fayyaz, N. M. Garcia*, et al.*, "A Two-Fold Machine Learning Approach to Prevent and Detect IoT Botnet Attacks," *IEEE Access,* vol. 9, pp. 163412-163430, 2021.

[47] T. G. Palla and S. Tayeb, "Intelligent Mirai Malware Detection in IoT Devices," in *2021 IEEE World AI IoT Congress (AIIoT)*, 2021, pp. 0420-0426.

[48] J. Jeon, J. H. Park, and Y. S. Jeong, "Dynamic Analysis for IoT Malware Detection With Convolution Neural Network Model," *IEEE Access,* vol. 8, pp. 96899-96911, 2020.

[49] D. Vasan, M. Alazab, S. Venkatraman, J. Akram, and Z. Qin, "MTHAEL: Cross-Architecture IoT Malware Detection Based on Neural Network Advanced Ensemble Learning," *IEEE Transactions on Computers,* vol. 69, pp. 1654-1667, 2020.

[50] Y. N. Soe, Y. Feng, P. I. Santosa, R. Hartanto, and K. Sakurai, "Machine Learning-Based IoT-Botnet Attack Detection with Sequential Architecture," *Sensors,* vol. 20, p. 4372, 2020.

[51] H. Naeem, F. Ullah, M. R. Naeem, S. Khalid, D. Vasan, S. Jabbar*, et al.*, "Malware detection in industrial internet of things based on hybrid image visualization and deep learning model," *Ad Hoc Networks,* vol. 105, p. 102154, 2020/08/01/ 2020.

[52] H. Alasmary, A. Khormali, A. Anwar, J. Park, J. Choi, A. Abusnaina*, et al.*, "Analyzing and Detecting Emerging Internet of Things Malware: A Graph-Based Approach," *IEEE Internet of Things Journal,* vol. 6, pp. 8977-8988, 2019.

[53] F. Ullah, H. Naeem, S. Jabbar, S. Khalid, M. A. Latif, F. Al-turjman*, et al.*, "Cyber Security Threats Detection in Internet of Things Using Deep Learning Approach," *IEEE Access,* vol. 7, pp. 124379-124389, 2019.

[54] M. Al-Hawawreh, N. Moustafa, and E. Sitnikova, "Identification of malicious activities in industrial internet of things based on deep learning models," *Journal of Information Security and Applications,* vol. 41, pp. 1-11, 2018/08/01/ 2018.

[55] H. Nazari, A. H. Farooqi, B. Raza, S. Kamal, W. Nawaz, and W. Abbass, "A CUDA-Accelerated Hybrid CNN-DNN Approach for Multi-Class Malware Detection in IoT Networks," *IEEE Access,* pp. 1-1, 2025.

[56] O. Polat, A. A. Ahmad, S. Oyucu, E. Algül, F. Doğan, and A. Aksöz, "Temporal-Spatial Feature Extraction in IoT-Based SCADA System Security: Hybrid CNN-LSTM and Attention-Based Architectures for Malware Classification and Attack Detection," *IEEE Access,* vol. 13, pp. 102109-102132, 2025.

[57] Y. A. Maz, M. Anbar, S. Manickam, S. D. A. Rihan, B. A. Alabsi, and O. M. Dorgham, "Majority Voting Ensemble Classifier for Detecting Keylogging Attack on Internet of Things," *IEEE Access,* vol. 12, pp. 19860-19871, 2024.

[58] G. Karat, J. M. Kannimoola, N. Nair, A. Vazhayil, S. V. G, and P. Poornachandran, "CNN-LSTM Hybrid Model for Enhanced Malware Analysis and Detection," *Procedia Computer Science,* vol. 233, pp. 492-503, 2024/01/01/ 2024.

[59] J. Lu, X. Ren, J. Zhang, and T. Wang, "CPL-Net: A Malware Detection Network Based on Parallel CNN and LSTM Feature Fusion," *Electronics,* vol. 12, p. 4025, 2023.

[60] S. K. Smmarwar, G. P. Gupta, and S. Kumar, "AI-empowered malware detection system for industrial internet of things," *Computers and Electrical Engineering,* vol. 108, p. 108731, 2023/05/01/ 2023.

[61] S. Jain, P. M. Pawar, and R. Muthalagu, "Hybrid intelligent intrusion detection system for internet of things," *Telematics and Informatics Reports,* vol. 8, p. 100030, 2022/12/01/ 2022.

[62] H. HaddadPajouh, A. Dehghantanha, R. Khayami, and K.-K. R. Choo, "A deep Recurrent Neural Network based approach for Internet of Things malware threat hunting," *Future Generation Computer Systems,* vol. 85, pp. 88-96, 2018/08/01/ 2018.

[63] S. K. Smmarwar, G. P. Gupta, and S. Kumar, "XAI-AMD-DL: An Explainable AI Approach for Android Malware Detection System Using Deep Learning," in *2023 IEEE World Conference on Applied Intelligence and Computing (AIC)*, 2023, pp. 423-428.

[64] Q. Abu Al-Haija and M. a. Al-Dala'ien, "ELBA-IoT: An Ensemble Learning Model for Botnet Attack Detection in IoT Networks," *Journal of Sensor and Actuator Networks,* vol. 11, p. 18, 2022.

[65] U. Tariq, I. Ullah, M. Yousuf Uddin, and S. J. Kwon, "An Effective Self-Configurable Ransomware Prevention Technique for IoMT," *Sensors,* vol. 22, p. 8516, 2022.

[66] T. G. Palla and S. Tayeb, "Intelligent Mirai Malware Detection for IoT Nodes," *Electronics,* vol. 10, p. 1241, 2021.

[67] Q. Li, J. Mi, W. Li, J. Wang, and M. Cheng, "CNN-Based Malware Variants Detection Method for Internet of Things," *IEEE Internet of Things Journal,* vol. 8, pp. 16946-16962, 2021.

[68] R. Kumar, X. Zhang, W. Wang, R. U. Khan, J. Kumar, and A. Sharif, "A Multimodal Malware Detection Technique for Android IoT Devices Using Various Features," *IEEE Access,* vol. 7, pp. 64411-64430, 2019.

[69] M. Al-Hawawreh and E. Sitnikova, "Leveraging Deep Learning Models for Ransomware Detection in the Industrial Internet of Things Environment," in *2019 Military Communications and Information Systems Conference (MilCIS)*, 2019, pp. 1-6.

[70] H. T. Nguyen, Q. D. Ngo, and V. H. Le, "IoT Botnet Detection Approach Based on PSI graph and DGCNN classifier," in *2018 IEEE International Conference on Information Communication and Signal Processing (ICICSP)*, 2018, pp. 118-122.

[71] M. M. H. Shuvo, S. K. Islam, J. Cheng, and B. I. Morshed, "Efficient acceleration of deep learning inference on resource-constrained edge devices: A review," *Proceedings of the IEEE,* vol. 111, pp. 42-91, 2022.

[72] H. Li, L. Ge, and L. Tian, "Survey: federated learning data security and privacy-preserving in edge-Internet of Things," *Artificial Intelligence Review,* vol. 57, p. 130, 2024/04/29 2024.

[73] B. S. Guendouzi, S. Ouchani, H. El Assaad, and M. El Zaher, "A systematic review of federated learning: Challenges, aggregation methods, and development tools," *Journal of Network and Computer Applications,* vol. 220, p. 103714, 2023/11/01/ 2023.

[74] P. Warden and D. Situnayake, *Tinyml: Machine learning with tensorflow lite on arduino and ultra-low-power microcontrollers*: O'Reilly Media, 2019.

[75] J.-I. Iturbe-Araya and H. Rifà-Pous, "Hyperparameter Optimization and Evaluation Metrics for Unsupervised Anomaly-Based Cyberattack Detection in Imbalanced Smart Home Datasets," *Journal of Network and Systems Management,* vol. 33, p. 99, 2025/09/12 2025.