ASJET

Al-Salam Journal for Engineering and Technology
Journal Homepage:
http://journal.alsalam.edu.iq/index.php/ajest
e-ISSN: 2790-4822 p-ISSN: 2958-0862

Smart and Sustainable Malware Detection in A Resource-Constrained IoT Environment: A Survey of Continuous and Context-Aware AI Methods

Sattar J. J.Yahya[1]* and Baraa I. Farhan[1]

[1]Wasit University-Collage of Education for Pure Sciences, Wasit Governorate - Kut - Al-Rabi'a District, 52001, IRAQ.

*Corresponding Author: Sattar J.J. Yahya

DOI: https://doi.org/10.55145/ajest.2026.05.01.003
Received August 2025; Accepted November 2025; Available online Febuary 2026

ABSTRACT: Due to The rapid expansion of Internet of Things (IoT) technologies in many different fields in our lives has led to critical challenges of service security, also while most notably the risky threat of malware. Despite the many methods for detecting this malicious software, the resource constraints of IoT devices remain a fundamental challenge that reduces the effectiveness and operational capacity of these solutions. This study conducts a comprehensive survey to identify effective detection strategies in resource-constrained environments and aims to evaluate the feasibility of integrating Context Awareness with machine learning and deep learning approaches to establish a detection system that adapts to device resources and network conditions. This study highlights the fact that most of today's AI solutions are fundamentally defective: while highly accurate, they are unable to adjust and continuously learn to adapt to modern malicious patterns over a long scope. In this survey, we introduce our view, the main reason for this weakness is the omission of the context-awareness factor. and we will provide insights and criteria for designing lightweight, context-aware AI models to achieve the perfect balance between detection accuracy and model flexibility and incremental learning in resource-constrained IoT environments, to counter advanced threats.

Keywords: IoT, AI Context-aware, Continual learning, IoT Malware Detection, Lightweight and Resource-Constrained Devices

1. INTRODUCTION

The massive progress in Internet of Things (IoT) devices and their multiple uses in various aspects of our lives has highlighted the importance of IoT systems as an ideal solution for continuous communication between things and people [1]. The scope of IoT systems has evolved to include healthcare systems and smart cities to improve user well-being [2]. Some estimates indicate that the number of IoT devices will double over the next decade [3], reaching tens of billions in industrial and home use [4]. This highlights the need for innovative and effective security solutions to ensure the continuous operation of these devices [5]. Due to their increasing and rapid expansion, IoT devices are now a major target for cyber-attacks, including malware, spyware, and distributed denial of service (DDOS) attacks [6]. For this reason, critical infrastructure systems are vulnerable, making them a prime target for advanced cyber-attacks [7]. Recent studies show that IoT malware detection systems work best when the detection model is accurate and the system can understand the context of the environment and device, adapt to changing resource needs and processing loads, and stay strong against new types of cyberattacks and changing system conditions [8].

This study highlights the gaps that previous studies have suffered from [9-11] in three main areas:

1. Existing models cannot adapt to new threats because to few models that support continuous learning.
2. Lack of contextual awareness or poor application, which is essential to understanding the nature and behavior of each device.
3. the penury of real-world dataset availability because many studies lack data that reflect the diversity of scenarios and devices, which weakens the generalization of results.

This study makes the following contributions:

- We survey the previous studies that support malware detection models and methods in a resource-constrained IoT environment.
- We integrate the principles of continuous learning and contextual awareness to enhance malware detection.
- We proposed a Standardized classification of malware in an IoT environment, and we compared the performance of the IoT software dataset.

2. Systematic Review Methodology

The systematic reviews aim to measure precisely AI tools used for malware detection in the Internet of Things space, especially under conditions where systems learn constantly, maintain awareness of context, and face limited resources. To uphold the high standards of transparency and accuracy essential for systematic reviews of this caliber, we dedicated ourselves to the selection and analysis process guided by the PRISMA 2020 protocols (Preferred Items for Systematic Reviews and Meta-Analyses). This AI model shows potential in various fields, including healthcare, law enforcement, public safety, and education. To maintain the required standards for high-quality systematic reviews, our team committed to the selection and analytical process aligned with the mentioned methodology.

2.1 Sources of Data and Search Methodology

This research focused on contemporary studies and research published between 2021 and 2025 in prominent academic sources (IEEE Xplore, ACM Digital Library, ScienceDirect, etc.). and used integrated keywords phrases employed were: "IoT Malware Detection" AND "Resource-Constrained", "Continuous Learning" AND "IoT Security", and "Context-Aware AI" AND "Lightweight Model".

2.2 Inclusion and Exclusion Criteria

Papers that addressed AI models for malware detection in IoT and explicitly discussed performance challenges in resource-constrained environments, or Continual learning and Context-Aware mechanisms, were included. Papers that did not directly address malware or that did not meet the specified time frame were excluded.

2.3 Paper Selection Process (PRISMA Flow)

The search led to 129 results exclusively from databases. Prior to screening, 44 records were eliminated: 31 duplicates, 0 deemed ineligible by automated techniques, and 13 discarded for alternative reasons. Of the 85 records available for screening, 23 were rejected based on title and abstract evaluation. 62 records were requested for retrieval; 1 report was unobtainable, resulting in 61 reports evaluated for eligibility. No additional reports were deleted at this phase. A total of 61 published research publications and one technical report were incorporated into the review. The PRISMA flow diagram illustrates the entire procedure shown in Figure 1.

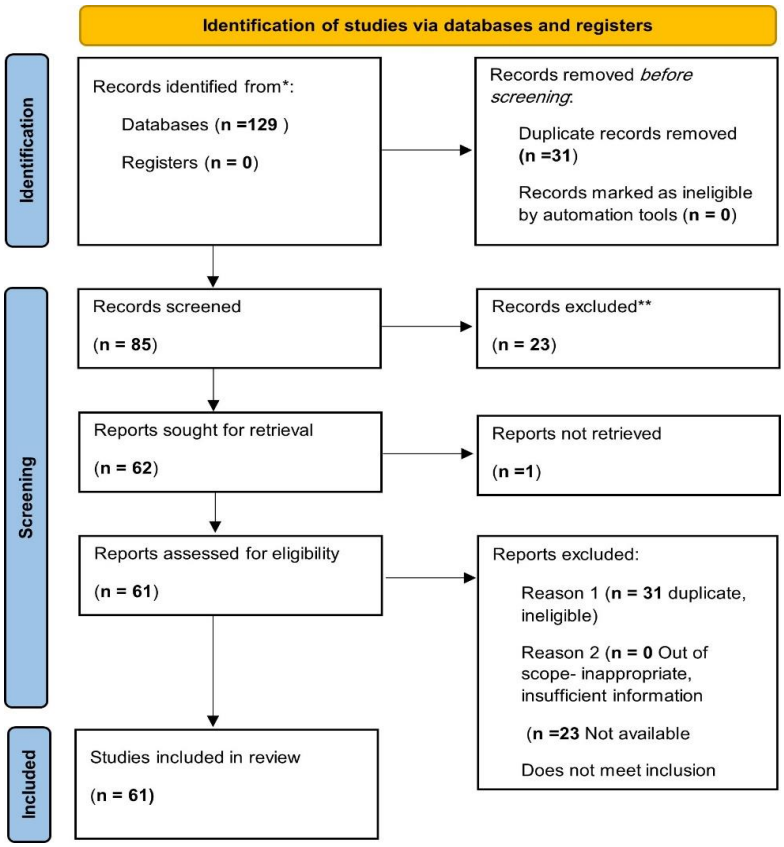


FIGURE 1 PRISMA 2020 flow diagram of our study selection process

3. Background

3.1 The "Internet of Things (IoT)"

The "IoT" is a massive network of inter-connected physical devices with different topologies, which enable the transmission of data among the devices and central servers or users [12]. This evolution is considered one of the main drivers for digital transformation in various fields, such as smart homes, transportation, and smart cities [13], which offers living standards with cost reduction and better performance [14]. It does that by sensing and processing data from its environment and then makes decisions and takes actions either autonomously or semi-autonomously [15]. Examples include autonomous vehicles and smart homes, which have changed our lifestyles and the way we relate to technology [16].

3.2 Types of the "Internet of Things" by Domain

Depending on their usage, the "IoT" is categorized into two broad types: Consumer IoT, including personal devices, smart homes, and wearable devices, and Industrial & Critical IoT, involving applications in the healthcare sector, agriculture, smart cities, and smart vehicles [17]. The categorization of the "Internet of Things (IoT)" is depicted in Fig. 2.

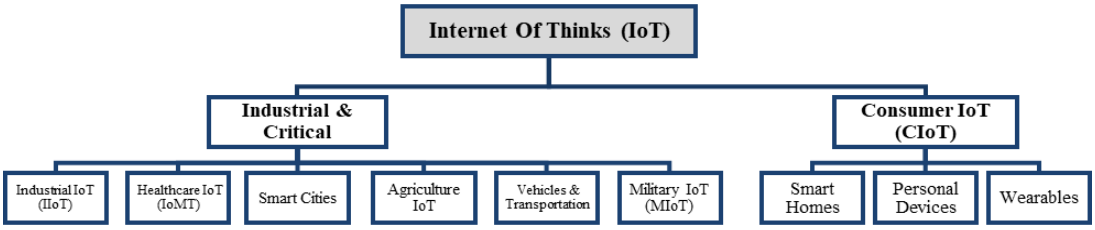


FIGURE 2 illustrate Types of IoT and classification in areas [18]

One such less talked IoT application in academic research or open academic literature is the Military IoT (MIoT), an important extension of intelligent systems in combat and critical environments. This kind is based on state-of-the-art technical equipment, including drones or remote sensing technology, as well as protected military communications systems that require a resilient OS to hostile environments and the corresponding architecture security [19].

Specialized MIoT domains such as Internet of Things on the Battlefield (IoBT) demonstrate capabilities to directly engage in dangerous military scenarios [19]. This involves integrating robots, including ground and aerial vehicles, as well as the Soldiers as Sensor Nodes (SAM) system, which improves operations in the field. It also includes smart logistics systems (SML) [20], command, control, communications, and information systems (C4ISR) [21], advanced cyber defense systems, as well as naval and aerospace applications. These contemporary systems demand high security, advanced technology, and a vicious kind of operational autonomy [22].

4. Architecture of Internet of Things (IoT)

A conventional Internet of Things (IoT) architectural scheme is established based on a multi-layer model, which can help us describe and analyze the parts and relationships inside a system. [23]. The details shown in Table 1 [24]:

Table 1 Internet of Things' architecture [24]

Layers	Sub-Layer	Key Features	Key Technologies
Application Layer	IoT Applications Application Support Layer	Handheld Devices, Terminals, and User Interface	Cloud Computing, Middleware, M2M, Service Support Platform
Transmission Layer	Local & Wide Area Network Core Network Access Network	Connectivity Establishment and Information Transmission	Internet, Wi-Fi, GPRS, ad-hoc Network
Perception Layer	Perception Network Perception Nodes	Sensing, Identification, Actuation, and Communication Technologies	RFID, WSN, GPS, Bluetooth
Network management	**Physical and Information Security Management**		**Trust Management**

4.1 Perception Layer

The perception layer has the duty of collecting information from the environment through sensors and devices. The Internet of Things also encompasses a variety of interactions, including those between devices, called M2M interactions; between users and devices, which are both H2M and M2H interactions; and at times between humans, referred to as H2H interactions, through IoT platforms in general [25]. The described patterns stand crucial in building Internet of Things applications, since many depend on the capability of devices to make their own decisions or alert service users where needed, while also offering a flexible interactive interface to enhance communication among all parties involved [26].

4.2 Network Layer

It forwards data collected from the embedded environment to the processing center for analysis and decision making, which in turn issues instructions accordingly Idi [27].

4.3 Application Layer

The application layer supplies interactive services to end users based on the intended use. It signifies the ultimate output of preceding operations inside the Internet of Things architecture, delivering the user the aggregated and analyzed data outputs. This architecture improves integration, security, and communication efficacy among system components [28]. These devices are categorized depending on their interaction within IoT-based solutions [29]:

1. **Machine-to-Machine (M2M) Interaction**

This type of interaction between devices to direct communication between them. Objects autonomously communicate and interact with their surroundings to gather, analyze, or transmit data to a central system. Examples of this type include environmental sensors (a temperature sensor sends a signal to a cooling device to operate it) [30]. This type is most common in IIoT (Industrial Internet of Things) systems and smart cities.

2. **Machine-to-Person (M2P) Interaction**

This interaction includes devices or objects interacting with the surrounding environment and sending their data to humans. The human interacts with the machine or device via voice commands or a control panel. Examples include a fire alarm system that sends notifications to a phone or adjusts the lighting using a mobile device [31].

3. **Person-to-person (P2P) interaction**

Although this type does not directly represent the “Internet of Things,” it pertains to interactions wherein individuals utilize an Internet of Things platform for communication. It is an indirect interaction via a smart intermediary and is often part of health or social systems [29].

4.4 IoT Malware

Privacy and cybersecurity have become crucial parts of building technological frameworks, hence determining the level of society's trust therein [32]. One of the IoT devices' risks involves malware; it represents a new category of cyberattacks against the smart networks in the IoT environment [33]. It has the intention of delegating control systems and distorting sensor information to avoid the detection of the system by changing and repeating system commands [34]. The intrusion-based threats are mainly in IoT systems since they make use of variable behavior to mask themselves in typical traffic patterns [9].

4.5 IoT Malware Classification

Malware class is key for knowing about attack patterns and how they work, which helps to build strong defense [35]. This class, as seen in Figure 3, covers malware fully for Internet of Things (IoT) with eight main topics. These eight aspects show how varied malware is and list: how it spreads, who it hits, what it does, what gadgets get infected, how it acts and the methods it uses, how it changes, and what stage the attack is at [35-41].

These eight parts create many issues for AI systems, especially where resources are low. For example, studying parts like malware's actions or type of victim gadget (like IoT tools) needs smart and high-cost checking methods. Detecting hidden actions or those that use simple protocols such as MQTT/CoAP is even harder. These limits on resources make big models difficult to use and require strategies that focus on making calculations easier, which is a main security issue. Besides the lack of resources, there is another limit in making sure of strong security that lasts. Factors like how it works and how adjustable show the failures of fixed models. New forms of malware that change or act differently put much pressure on AI systems. Threats are always changing, and it is not feasible to always train the whole system from scratch (because of time and cost), which shows the need for ongoing learning as a key part for fighting the entire chain of IoT threats [42,43].

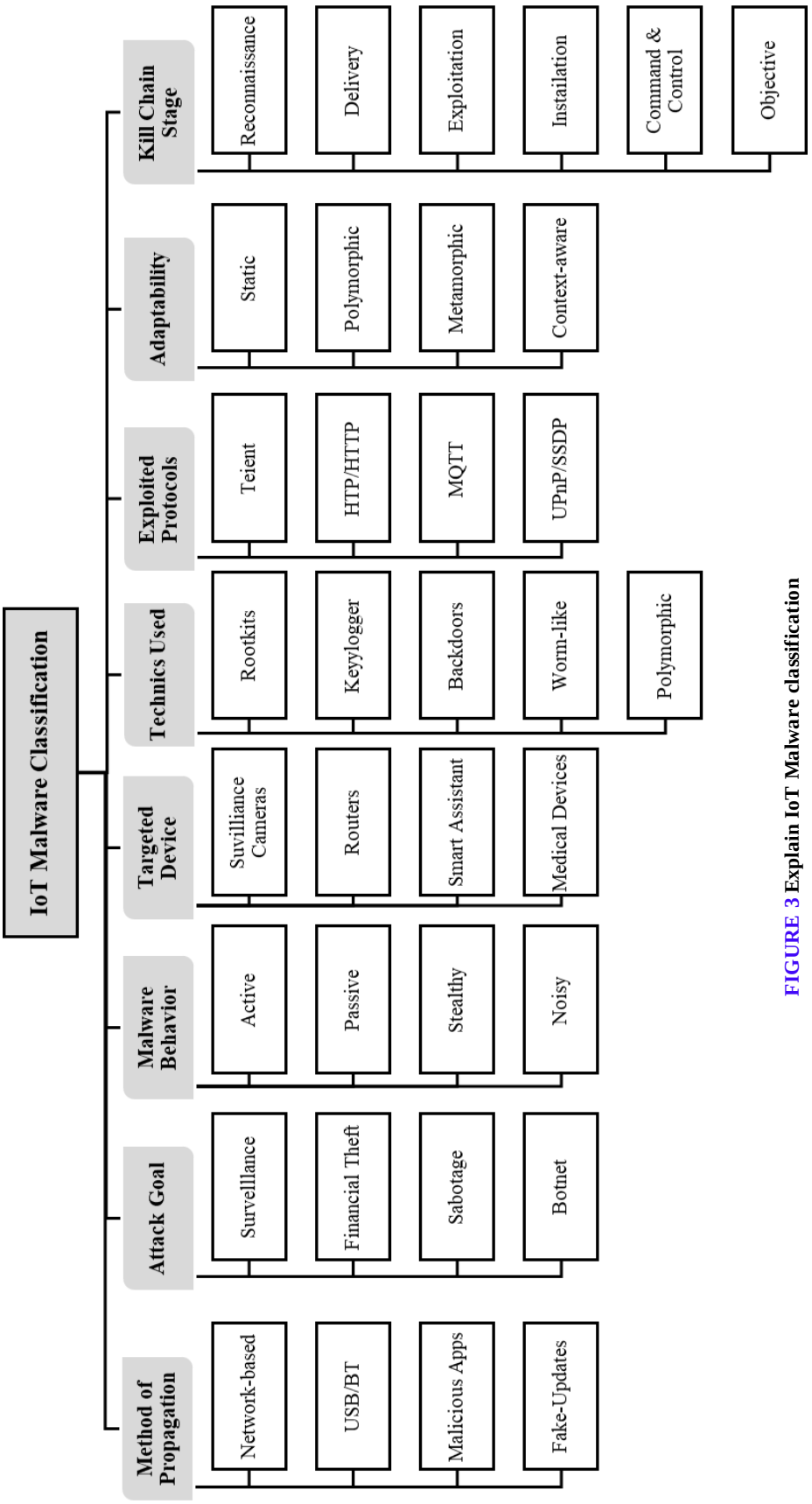


FIGURE 3 Explain IoT Malware classification

5. AI Techniques for IoT malware detection

The Artificial intelligence technologies, especially machine learning, provide an advanced framework for detecting malware in an IoT environment, as traditional methods rely on fixed timings or instructions [6]. A.I. facilitates the examination of end-to-end network data, identification of complex patterns, and automatic adaptation to emerging threats, which is essential for handling zero-day malware and attack mutations. and for AI to achieve sustainability in the diverse and resource-constrained IoT sector, models need to progress and move from static learning to include continuous adaptation and contextual perception [7].

5.1 Machine Learning in Malware Detection

Machine learning, which is an essential component of artificial intelligence, contributes crucially to the identification of modern malware [34]. Traditional machine learning techniques such as "random forests" and "support vector machines" continue to contribute significantly to the settings of resource-constrained IoT because they can provide a good balance between computational efficiency and satisfactory performance [44,45], especially when they are designed as low-power models [20]. Nevertheless, even these traditional algorithms, with their minimal computational requirements, are found lacking. This limitation implies that the models have to be entirely retrained, which would be expensive and not sustainable to address new attack patterns, hence indicating that there is a substantial gap in the achievement of long-term sustainable security, more so when compared with dynamic models like reinforcement learning [46].

5.2 Supervisedlearning

Supervised learning is a major paradigm for artificial intelligence where models are trained on pre-labeled data to classify files as malicious or benign based on certain patterns.[39][46] While efficient in detecting defined malware, the need for large, high-quality, explained datasets is a serious limiting factor in IoT, specifically when it comes to finding and modifying zero-day malware or new malware.[7] In this sense, this approach inherently lacks the ability for true continuous learning required by permanent security, as traditional supervised learning-based detection models face a serious limitation due to their inability to adapt themselves to newly emerging or zero-day threats without full retraining, which is at odds with the strict cost limitations on hardware/energy resources [7]. However, discussion of the algorithms shown in Table 2 is necessary since many of them remain preferable due to the negligible computational cost at inference time, achieving a good trade-off between efficiency and resource utilization in real-time models.

Table 2 Main Supervisedlearning algorithms

No.	Algorithm	Description
1.	"Random Forest (RF)"	Good accuracy with reasonable execution speed.
2.	"Support Vector Machines (SVM)"	Especially effective with high-dimensional data.
3.	"Decision Trees (DT)"	Low power consumption and easy deployment
4.	"K-Nearest Neighbors (KNN)"	Simple and effective, but slower with large datasets.
5.	"Gradient Boosting / XGBoost"	High accuracy but higher resource consumption.

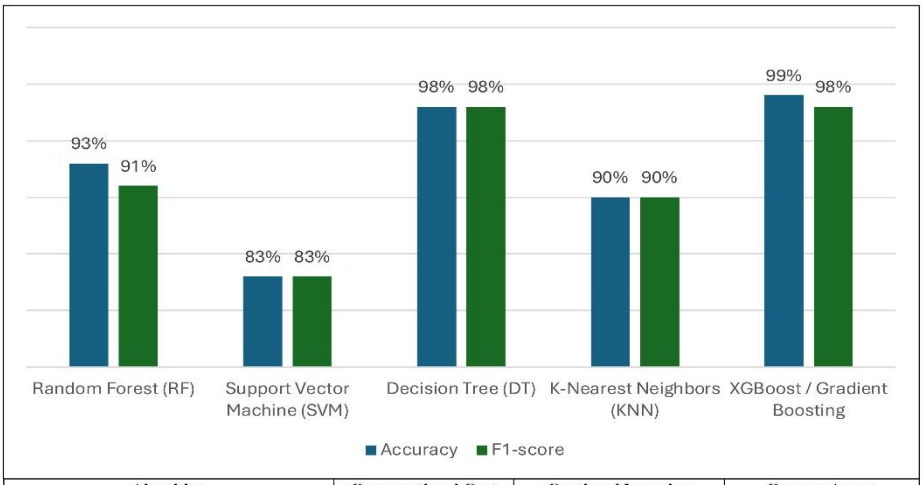
The comparison review of the supervised learning algorithms for malware detection in IoT (Table 3) identify the merits and demerits of these algorithms according to the resource limitation situation. Other algorithms weren't able to achieve such high accuracy, eg, XGBoost and Decision Tree also displayed a good performance across the board but reached only 98–99% [45][47]. Decision Tree algorithm is considered as a relatively well-balanced between performance and efficiency, thereby needing only little computational resources with implementation on resource-constrained "Internet of thinks" devices [45], but models like "Support Vector Machine" or "k-Nearest Neighbor" was found less efficient for several reasons like such as their lower classification accuracy or higher computational cost, its impracticality in real-world IoT environment where its great volume of data produced to meet the time constraint demand [44][48]. The most interesting observation from this figure is the existence of two large glitches for all supervised learning methods:

1. No continuous learning: Nowhere in the literature were any of the reviewed studies [43-47] developed with forms of continuous learning, which means all ML models are static, needing to be retrained from scratch; using significant resources when attempting to adapt for new or evolving threats, thereby compromising their long-term sustainability.

2. Lack of contextual awareness: The algorithms appeared to be unable to take the operational context (e.g., type of device, availability status, and traffic load of the affected network elements) into account. The lack of flexibility limits the capability of the model to improve resource efficiency and perform strongly in various IoT environments. Finally, although supervised learning algorithms are indispensable in certain scenarios in terms of their simplicity and computational efficiency (e.g., decision tree algorithm), we argue that the absence of an updating mechanism and allocation-aware with these traditional ones make them imperative to evolve from towards a more dynamic, sustainable artificial intelligence solutions.

Table 3 Comparative of "Supervised Machine Learning" Algorithms for "Internet of Things" Malware Detection

Ref.	Algorithm	Strength	Limitation	Best Use Case	Accuracy	F1-score	Computational Cost	Continual Learning Support	Context-Aware Capability
43	**Random Forest (RF)**	High detection accuracy on IoT malware datasets; robust to imbalance	Slower prediction on large datasets; limited optimization for IoT devices	Detecting malware in synthetic & real IoT datasets	93%	91%	Medium	No	No
44	**Support Vector Machine (SVM)**	Good precision on small/clean IoT-23 subsets; effective binary classifier	High computational cost; not scalable for large IoT flows	Binary IoT malware vs benign traffic classification	83%	83%	High	No	No
45	**Decision Tree (DT)**	Simple, interpretable, fast training	Prone to overfitting; slightly less accurate than RF	Lightweight IoT malware detection on constrained devices	98%	98%	Low	No	No
46	**K-Nearest Neighbors (KNN)**	Easy to implement, intuitive	Very high inference cost; memory-intensive	Small-scale or lab-controlled IoT malware classification	90%	90%	Very High	No	No
47	**XGBoost / Gradient Boosting**	Very high accuracy; robust against overfitting; strong for multi-class IoT malware	Training is slow; it requires heavy parameter tuning	Multi-pattern IoT malware detection (CIC-IoT2023, IoT-23)	99%	98%	High	No	No



Algorithm	Computational Cost	Continual Learning Support	Context-Aware Capability
Random Forest (RF)	Medium	No	No
Support Vector Machine (SVM)	High	No	No
Decision Tree (DT)	Low	No	No
K-Nearest Neighbors (KNN)	Very High	No	No
XGBoost / Gradient Boosting	High	No	No

FIGURE 4 Metrics Evaluation for Main Supervised Learning Algorithms in IoT Malware

5.3 Unsupervised learning

In contrast to supervised learning, unsupervised learning does not depend on labelled data; it examines the intrinsic structure of data to uncover concealed patterns and anomalies. [42]. This is an important necessity for identifying zero-day attacks, and to identify unfamiliar malicious activities that differ from typical data behavior and without the need for prior awareness of specific attack signatures [6,7]. More importantly, the one we will discuss in this section is unsupervised learning algorithms, where models specifically enable Internet malware to detect things that are not previously trained on [46]. And because unsupervised learning focuses on detecting anomalies, it can aggregate similar unknown threats [46], thus providing a superior basis for continuous adaptation compared to supervised learning models that are static. but the effectiveness of unsupervised learning in IoT depends entirely on its computational cost and ability to maintain performance with minimal resources.

Table 4 main unsupervised learning algorithms

No.	Algorithm	Description
1.	K-Means Clustering	for grouping similar patterns
2.	DBSCAN	for detecting irregularly shaped clusters
3.	Self-Organizing Maps (SOM)	for drawing low-dimensional maps that help uncover patterns
4.	Isolation Forest	very effective at detecting anomalies

In an effort to address the dynamic security challenges posed by IoT environments, unsupervised learning methodology is emerging to detect malware and emerging threats. However, the choice of the optimal algorithm in this context is governed by a critical trade-off [47].

A relative search of unsupervised learning methodologies for identifying IoT malware (Table 5 and Figure 5) reveals a clear trade-off between detection accuracy and computational cost.

Table 5 Comparative Analysis of Unsupervised Learning Algorithms for Continual and Context-Aware IoT Malware Detection

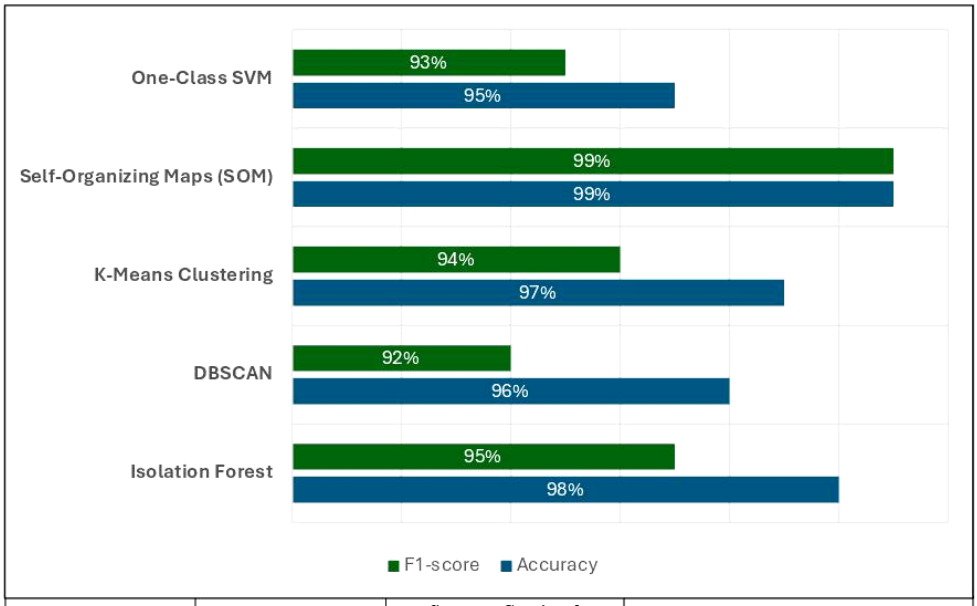
Ref.	Algorithm	Strength	Limitation	Best Use Case	Accuracy	F1-score	Computational Cost	Supports Continual Learning?	Context-Aware Capability
48	**Isolation Forest**	Optimized for heterogeneous and streaming IoT traffic; robust anomaly detection	Sensitive to contamination ratio; performance depends on parameter tuning	Intrusion/malware detection in IIoT streaming data	Reported 96–98% on IIoT datasets	95% (reported)	Moderate; tree-based ensemble optimized for streaming	No	No
49	**DBSCAN**	Effective in detecting arbitrary-shaped attack clusters; works well on the IoT-23 dataset	Requires careful tuning of ε and MinPts; scalability issues on very large data	Malware anomaly detection in IoT-23 traffic	94–96%	92%	Low to moderate (density-based clustering)	No	No
50	**K-Means Clustering**	Simple, scalable, fast; achieved good detection on IoT malware traffic	Struggles with noisy or non-spherical clusters	Large-scale IoT malware detection and traffic clustering	95–97%	94%	Low computational cost	No	No
51	**Self-Organizing Maps (SOM)**	Strong nonlinear mapping; effective when combined with deep learning	High training cost; requires careful initialization	Hybrid IoT attack detection with deep learning (CICIoT2023, NSL-KDD, UNSW-NB15)	99.99%	>99%	High (neural training overhead)	No	Partially (pattern-aware)
52	**One-Class SVM**	Good for zero-day and one-class anomaly detection in IoT; handles unseen threats	Sensitive to kernel parameters; computationally heavy on large data	IoT malware detection with limited labeled data	93–95%	91–93%	High (quadratic complexity)	No	No

Hybrid models combining Self-Organizing Map (SOM) algorithms and deep learning techniques attained optimal performance, with an accuracy of 99.99% [51], thus demonstrating their effectiveness in analyzing complex and non-linear data traffic. Nonetheless, this high execution comes at the expense of substantial training complexity and

sensitivity, posing significant challenges for deployment on resource-constrained devices. Conversely, algorithms such as Isolation Forest and K-means Clustering offer a better balance, achieving acceptable accuracy (95% to 97%) with low to moderate computational cost [48][50]. This makes them more suitable for simple anomaly detection scenarios. However, even robust algorithms like DBSCAN [49] and one-class SVM [52] suffer from parameter tuning sensitivity and high computational complexity with large datasets, limiting their practical applicability. It is essential to acknowledge that, notwithstanding the advantages of unsupervised learning for detecting novel threats, this analysis confirms that all current unsupervised learning techniques suffer from an inherent flaw:

1. Static adaptation: None of the studied algorithms were designed to support continuous learning. They are static models that require manual updates or complete retraining, rendering them ineffective against the ever-evolving nature of IoT malware.

2. Missing contextual integration: These models do not support the integration of contextual data (such as device status or power level) to dynamically optimize resource consumption, which reduces their long-term effectiveness in resource-constrained environments.



Algorithm	Computational Cost	Supports Continual Learning?	Context-Aware Capability
Isolation Forest	Medium	No	No
DBSCAN	Low	No	No
K-Means Clustering	Low	No	No
Self-Organizing Maps (SOM)	High	No	Partially (pattern-aware)
One-Class SVM	High	No	No

FIGURE 5 Metrics Evaluation for Main Unsupervised Learning Algorithms in IoT Malware Detection

5.4 "Reinforcement learning"

"Reinforcement learning" denotes a highly dynamic type of artificial intelligence, in which a model, known as an agent, interacts with its environment to learn optimal actions through rewards and penalties [53,54]. This iterative methodology enables the model to continually enhance its defense strategy over time [55]. thereby increasing its capacity to prevent or alleviate security breaches predicated on its past interactions [56]. Due to its unique characteristics, reinforcement learning is considered the most promising approach for achieving sustainable security, as its core principle supports both continuous learning and context awareness. The agent can adapt its actions based on evolving threat behaviors and the real-time operational state of the IoT device. Despite challenges such as the difficulty in defining the learning environment and high computational costs [57], its potential for achieving long-term resilience led us to categorize its algorithms into the main learning categories, as shown in Table 6.

Table 6 Reinforcement Learning Category and Algorithms

Main Category	Algorithm	Description / Principle	Environment / Use Case	Reference
"Value-Based Methods"	"Q-Learning"	Updates Q(s,a) to approximate the expected return; does not necessitate an environmental model (model-free).	General RL problems.	[56]
	SARSA	Similar to Q-learning, but updates based on the current action taken; more conservative and accurate in estimation.	Safer learning, where exploration must be controlled.	[56]
	DQN	Integrates deep neural networks with Q-learning to estimate the value function; handles high-dimensional states.	Complex/high-dimensional environments (e.g., video games).	[56]
"Policy-Based Methods"	REINFORCE	Monte Carlo Policy Gradient relies on full episode sampling to estimate gradients.	Foundation for policy gradient methods.	[56]
	Actor-Critic	Integrates value-based and policy-based approaches: the actor modifies the policy while the critic assesses value..	General RL tasks: bridge between value and policy learning.	[56]
	PPO	Simplifies and stabilizes policy optimization using a clipped objective.	Robotics, complex control, and gaming.	[56]
	TRPO	Ensures stable updates by constraining policy changes within a trust region.	Safe training in unstable environments.	[56]
Advanced Actor-Critic	A3C	Asynchronous agents running in parallel update a shared global network, improving speed and stability.	Large-scale tasks, faster training.	[58]
	A2C	A synchronous version of A3C; waits for all agents to collect experience before updating.	Simpler implementation, stable but slower.	[58]
	DDPG	Uses deterministic strategies within continuous action spaces with actor–critic networks.	Continuous control (robotics, robotic arm).	[58]
	TD3	Improves DDPG employing dual critics and deferred updates to reduce overestimation.	Continuous control with more stability.	[58]
	SAC	Introduces the maximum entropy principle, encouraging exploration and avoiding premature convergence.	High-sample efficiency tasks; stable training.	[58]
Model-Based Methods	Dyna-Q	Combines model-free learning (trial-and-error) with model-based planning by simulating experiences.	Hybrid environments; efficient planning.	[59]
	MBPO (Model-Based Policy Optimization)	Optimizes policy using simulated rollouts from learned environment models, combined with model-free methods (e.g., SAC).	Balances efficiency and model error; robotics and control.	[59]

5.5 Deep Learning in Malware Detection

Deep learning changed how we check network traffic and find bad patterns in IoT networks. It can pull out big data features on its own, so no need for humans to look for them [13]. We can mix types of neural networks, such as CNN (best for spatial details) and LSTM (best for time-based data), to make finding bad patterns better. This was shown in the IoT-23 list [39]. But using these models can cost a lot of resources. This is a real problem in IoT networks, where resources are tight [11]. It raises key questions: Can these models keep up with the need to save resources? Can they change their methods as needs change? These questions are key to saving resources. Table 7 shows the top 'deep learning algorithms'.

Table 7 The "Deep Learning Algorithms"

Algorithm	Description / Principle	Environment / Use Case	Reference
CNNs (Convolutional Neural Networks)	Use convolution and pooling layers to extract spatial features from data.	Image and video recognition; visual pattern analysis.	[5][11]
RNNs (Recurrent Neural Networks)	Process sequential/temporal data while retaining previous context.	Natural language processing, sequential data tasks.	[11][60]

Algorithm	Description / Principle	Environment / Use Case	Reference
LSTM (Long Short-Term Memory)	Specialized RNNs using gates to retain long-term dependencies and mitigate forgetting.	Time-series forecasting, text processing.	[11][61]
GRUs ("Gated Recurrent Units")	A simplified version of LSTM with fewer parameters and lower computational cost.	Sequential data tasks with limited resources.	[11]
GANs (Generative Adversarial Networks)	Consists of a generator and a discriminator engaged in competition to generate authentic synthetic data.	Image generation, data augmentation, creative applications.	[11][62]
Autoencoders	Compress input into a latent representation and reconstruct it; used for unsupervised tasks.	Dimensionality reduction, denoising, anomaly detection.	[11][62]
Transformers	Use self-attention to model relationships in sequential data efficiently; the foundation of modern language models.	NLP, large-scale models (e.g., BERT, GPT).	[39]
GANs	Generator–discriminator setup, adversarial modeling.	Data augmentation and adversarial IoT malware detection.	[13], [37]
Multimodal Deep Learning Models	Integrate multiple modalities (text, image, audio) into one model to learn shared representations.	Cross-modal analysis, multimedia understanding.	[37]

The comparative analysis of deep learning algorithms (as shown in Table 8 and Figure 5) confirms their superior ability to extract complex patterns from IoT data traffic. Models such as RNN and LSTM demonstrated exceptional capabilities in capturing sequential relationships, achieving high accuracy and F1 scores exceeding 98% [60,61]. Transformer-based models showed the best results ever, with near-perfect accuracy (around 99.4%) [63]. Also unsupervised machine learning techniques, such as autoencoders, have excelled at detecting anomalies while consuming few computing resources [62].

However, our analysis shows a clear trade-off that cannot be overcome in real-time, which is performance versus sustainability. Complex models have high accuracy, but algorithms like RNNs, LSTMs, and transformers require very high training cost and very large memory, which makes its application completely impractical on resource-limited IoT devices [60][63]. Even the most suitable models, like CNNs, for restricted devices after dimension reduction [5]. still cannot manage to sustainably optimize the performance. The most essential and obvious conclusion from this analysis is the inherent static nature of current deep learning models, as proven in two important points:

1. Lack of continuous learning: Despite the sophisticated architecture of these models, none of the analyzed models supported continuous learning. This means that they need a complete retraining to adapt to any new threat that emerges, which is not practically possible for devices that are constantly running and have low power.

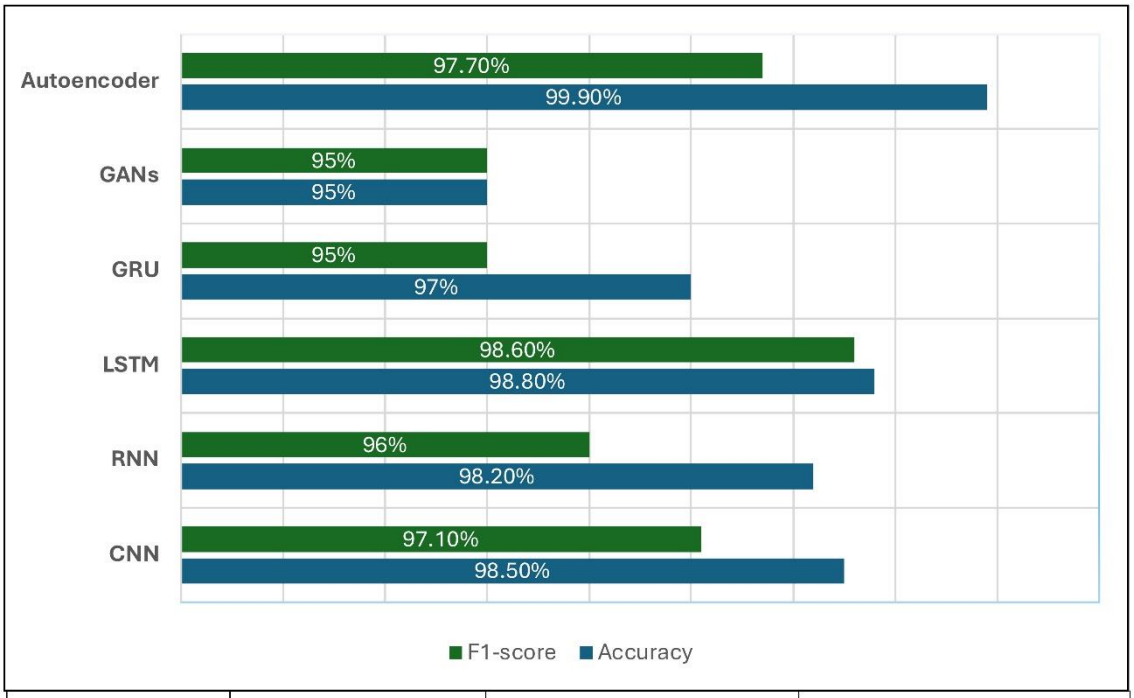
2. Highly contextual cost: Although transformed models may exhibit promising results concerning contextual learning [63], featuring their 'attention' mechanism that is considered to foster contextual awareness, huge computational requirements make leveraging this feature difficult to implement in resource-constrained IoT environments.

Our analysis clearly indicates that, for sustainable security in IoT, there is a clear need to move away from traditional deep learning models and toward light and adaptable models that can manage system resources with high efficiency while considering the operational context.

5.6 Hybrid Malware Detection Models

Hybrid tools are key efforts to fill these past gaps by blending the strengths of detection of some tools versus others and compensating for their weakness. The blend of signature and anomaly-based detection with machine learning tools shows a successful method for fighting back against both old and new attacks [9]. Context-aware models, often blended with deep learning structures such as CNNs and rules of statistics [10]. appear to perform well in balancing between adapting and the higher levels of performance needed to work in the open IoT world. Still, it must be said: while these

hybrid tools improve their detection coverage and contextual awareness, most still lack Continuous Learning built in, which leaves their long-termability to fight the constantly shifting world of IoT malware vulnerable.



Algorithm	Computational Cost	Supports Continual Learning?	Context-Aware Capability
CNN	Medium	No	Low
RNN	Medium	No	Medium
LSTM	Medium	No	High
GRU	Medium	No	Medium
GANs	High	No	Medium
Autoencoder	Low	No	Medium

FIGURE 6 Metrics Evaluation for Main Deep Learning Algorithms in IoTMalware Detection

Table 8 Comparative Analysis of "Deep Learning Algorithms" for Continual and "Context-Aware" IoT Malware Detection

Ref.	Algorithm	Strengths	Limitations	Best Use Case	Accuracy	F1-score	Computational Cost	Supports Continual Learning?	Context-Aware Capability
[5], [11]	CNN	Strong spatial feature extraction; effective in traffic-based malware classification; robust with ensemble methods.	High training cost; limited temporal modeling.	IoT-23 dataset malware detection (Mirai/Gafgyt botnets).	98.50%	97.10%	Medium	No	Low
[60], [11]	RNN	Captures sequential dependencies; adaptable to evolving IoT attack patterns.	Gradient vanishing; slower training; higher complexity.	IoT malware traffic analysis (time-series).	98.20%	~96%	Medium–High	No	Medium
[61], [11]	LSTM	Handles long-term dependencies; proven in IoT malware detection.	More parameters than GRU; higher computation.	CIC-IoT2023 botnet detection.	98.80%	98.60%	Medium–High	No	High

Ref.	Algorithm	Strengths	Limitations	Best Use Case	Accuracy	F1-score	Computational Cost	Supports Continual Learning?	Context-Aware Capability
[11]	GRU	Lightweight compared to LSTM; faster convergence; fewer resources.	May underperform vs. LSTM in very long sequences.	Sequential IoT traffic in constrained devices.	95–97%	~95%	Medium	No	Medium
[13], [39]	GANs	Synthetic sample generation; robust against adversarial malware.	Training instability; high computational cost.	Data augmentation and adversarial IoT detection.	>95%	>95%	High	No	Medium
[62], [11]	Autoencoder	Efficient unsupervised anomaly detection; low overhead; real-time capability.	Sensitive to threshold tuning; may fail with concept drift.	Real-time anomaly/malware detection in IoT networks (BoT-IoT).	99.90%	97.70%	Low–Medium	No	Medium

5.7 Context awareness with lightweight models

Machine learning, deep learning, and hybrid methods seek to find malware threats in a fixed time frame [9]. This has shown good growth in the world of IoT, where power, processor, and memory are very tight, and this is not enough, and this is where the need for a good model that brings the best micromodels and deep model intelligence arises [8].

To maintain safety, there must be a complete plan [22]:

1. Models should be small and fast based on better algorithms, compact models, and a correct way to extract features [8] (as in study [2], where an accuracy of 91.9% was achieved with very little memory use).

2. It requires both contextual awareness and continuous learning as a key to improving threat finding by looking at changing matters such as time and how the device works [10] and being contextually aware to keep up with new risks as they arise [22].

6. Conclusions

In this survey, we provide a comprehensive evaluation of AI-based malware detection techniques in the IoT environments and context-aware. We focused on reviewing the effectiveness of deploying models in resource-constrained environments and assessing their adherence to the principles of incremental (continuous), informed, and context-aware learning. While AI models, particularly deep learning models, have achieved significant efficiency in malware detection, our study reveals a fundamental flaw: most current solutions are inflexible and non-dynamically changing. This limits their ability to adapt to emerging challenges without requiring extensive and costly retraining, which weakens the long-term viability of these models. Furthermore, a severe disregard for contextual information, such as device conditions, environment, and operating patterns, weakens the detection accuracy and resource optimization of AI models. Therefore, it is essential to address this fundamental flaw. Developing an innovative AI model that facilitates continuous (incremental) learning and is context-aware of IoT devices is critical, and an optimal balance between detection accuracy and model efficiency is essential in resource-constrained IoT environments.

FUNDING

None

ACKNOWLEDGEMENT

The authors extend their gratitude to the anonymous reviewers for their valuable efforts.

CONFLICTS OF INTEREST

The authors declare no conflict of interest

REFERENCES

[1] H. Zorgati, R. B. Djemaa, and I. A. B. Amor, "Finding Internet of Things Resources: A State-of-the-Art Study," Data & Knowledge Engineering, vol. 140, p. 102025, 2022, doi: https://doi.org/10.1016/j.datak.2022.102025.

[2] N. U. Huda, I. Ahmed, M. Adnan, M. Ali, and F. Naeem, "Experts and Intelligent Systems for Smart Homes Transformation to Sustainable Smart Cities: A Comprehensive Review," Expert Systems with Applications, vol. 238, p. 122380, 2024, doi: https://doi.org/10.1016/j.eswa.2023.122380.

[3] M. Arnott, "Global IoT Forecast Report, 2024–2034," Transforma Insights, 2025. [Online]. Available: https://transformainsights.com/research/forecast/highlights.

[4] T. Zhukabayeva, L. Zholshiyeva, N. Karabayev, S. Khan, and N. Alnazzawi, "Cybersecurity Solutions for Industrial Internet of Things–Edge Computing Integration: Challenges, Threats, and Future Directions," Sensors, vol. 25, no. 1, p. 213, 2025, doi: https://doi.org/10.3390/s25010213.

[5] S. H. Khan et al., "A new deep boosted CNN and ensemble learning based IoT malware detection," Computers & Security, vol. 133, p. 103385, 2023, doi: https://doi.org/10.48550/arXiv.2212.08008.

[6] G. Kołaczek, "Internet of Things (IoT) Technologies in Cybersecurity: Challenges and Opportunities," Applied Sciences, vol. 15, no. 6, p. 2935, 2025, doi: https://doi.org/10.3390/app15062935.

[7] Z. Zhao, A. K. Pathan, and A. Ullah, "ZeroDefender: A Resource Aware IoT Malware Detection Framework Against Zero Day Threats," ACM Trans. Design Autom. Electron. Syst., 2024, doi: https://doi.org/10.1145/3687482.

[8] S. Kasarapu, S. Shukla, and S. M. P. Dinakarrao, "Enhancing IoT Malware Detection through Adaptive Model Parallelism and Resource Optimization," arXiv preprint, 2024, doi: https://doi.org/10.48550/arXiv.2404.08808.

[9] S. Berrios, D. Leiva, B. Olivares, H. Allende-Cid, and P. Hermosilla, "Systematic Review: Malware Detection and Classification in Cybersecurity," Applied Sciences, vol. 15, no. 14, p. 7747, 2025, doi: https://doi.org/10.3390/app15147747.

[10] S. Kasarapu, S. Shukla, and S. M. P. Dinakarrao, "Optimizing malware detection in IoT networks: Leveraging resource-aware distributed computing for enhanced security," arXiv preprint arXiv:2404.10012, 2024. doi: https://doi.org/10.48550/arXiv.2404.10012.

[11] R. Chinnasamy, M. Subramanian, S. V. Easwaramoorthy, and J. Cho, "Deep learning-driven methods for network-based intrusion detection systems: A systematic review," *ICT Express*, 2025. doi: https://doi.org/10.1016/j.icte.2025.01.005.

[12] R. Prakash, J. Neeli, and S. Manjunatha, "A survey of security challenges attacks in IoT," E3S Web of Conferences, vol. 491, 2024, doi: https://doi.org/10.1051/e3sconf/202449104018.

[13] R. Darwish, M. Abdelsalam, and S. Khorsandroo, "Deep learning based XIoT malware analysis: A comprehensive survey, taxonomy, and research challenges," Journal of Network and Computer Applications, p. 104258, 2025, doi: https://doi.org/10.48550/arXiv.2410.13894.

[14] R. Kufakunesu, H. Myburgh, and A. De Freitas, "The internet of battle things: a survey on communication challenges and recent solutions," Discover Internet Things, vol. 5, p. 3, 2025, doi: https://doi.org/10.1007/s43926-025-00093-w.

[15] N. Krishnan, "AI agents: Evolution, architecture, and real-world applications," *arXiv preprint* arXiv:2503.12687, 2025. doi: https://doi.org/10.48550/arXiv.2503.12687.

[16] A. Chandra, "Privacy-Preserving Data Sharing in Cloud Computing Environments," Eduzone: Int. Peer Reviewed Multidisciplinary Journal, vol. 13, no. 1, pp. 104–111, 2024. [Online]. Available: https://www.eduzonejournal.com/index.php/eiprmj/article/view/557.

[17] A. Choudhary, "Internet of Things: a comprehensive overview, architectures, applications, simulation tools, challenges and future directions," Discover Internet Things, vol. 4, p. 31, 2024, doi: https://doi.org/10.1007/s43926-024-00084-3.

[18] A. Dauda, O. Flauzac, and F. Nolot, "A Survey on IoT Application Architectures," Sensors, vol. 24, no. 16, p. 5320, 2024, doi: https://doi.org/10.3390/s24165320.

[19] P. Fraga-Lamas, T. M. Fernández-Caramés, M. Suárez-Albela, L. Castedo, and M. González-López, "A Review on Internet of Things for Defense and Public Safety," arXiv preprint arXiv:2402.03599, 2024, doi: https://doi.org/10.48550/arXiv.2402.03599.

[20] A. Gkelias, P. J. Baker, K. K. Leung, O. Worthington, and C. R. Melville, "Digital Twins for Internet of Battlespace Things (IoBT) Coalitions," arXiv preprint arXiv:2504.02561, 2025, doi: https://doi.org/10.48550/arXiv.2504.02561.

[21] V. M. Baeza, R. Parada, L. C. Salor, and C. Monzo, "AI-Driven Tactical Communications and Networking for Defense: A Survey and Emerging Trends," arXiv preprint arXiv:2504.05071, 2025, doi: https://doi.org/10.48550/arXiv.2504.05071.

[22] I. Coston, E. Plotnizky, and M. Nojoumian, "Comprehensive Study of IoT Vulnerabilities and Countermeasures," Applied Sciences, vol. 15, no. 6, p. 3036, 2025, doi: https://doi.org/10.3390/app15063036.

[23] D. Ameyed, F. Jaafar, and R. Petrillo *et al.*, "Quality and security frameworks for IoT-architecture models evaluation," *SN Computer Science*, vol. 4, p. 394, 2023. doi: https://doi.org/10.1007/s42979-023-01815-z.

[24] N. T. Y. Huan and Z. A. Zukarnain, "A Survey on Addressing IoT Security Issues by Embedding Blockchain Technology Solutions: Review, Attacks, Current Trends, and Applications," IEEE Access, 2024, doi: https://doi.org/10.1109/ACCESS.2024.3378592.

[25] N. Ali Hassan, N. Nizam-Uddin, A. Quddus, S. R. Hassan, A. U. Rehman, and S. Bharany, "Navigating IoT Security: Insights into Architecture, Key Security Features, Attacks, Current Challenges and AI-Driven Solutions Shaping the Future of Connectivity," Computers, Materials & Continua, vol. 81, no. 3, pp. 3499–3559, 2024, doi: https://doi.org/10.32604/cmc.2024.057877.

[26] Y. Jiang *et al.*, "Blockchained federated learning for Internet of Things: A comprehensive survey," *ACM Computing Surveys*, vol. 56, no. 10, art. 258, pp. 1–37, Oct. 2024. doi: https://doi.org/ 10.1145/3659099.

[27] V. Bhardwaj, A. Anooja, L. S. Vermani, et al., "Smart cities and the IoT: an in-depth analysis of global research trends and future directions," Discover Internet Things, vol. 4, p. 19, 2024, doi: https://doi.org/10.1007/s43926-024-00076-3.

[28] X. Yin, W. Shi, and K. K. R. Choo, "A Survey on Federated Learning Applications, Attacks and Defenses in Internet of Things," arXiv preprint arXiv:2309.12344, 2023, doi: https://doi.org/10.48550/arXiv.2309.12344.

[29] S. Rani, D. Jining, K. Shoukat, M. U. Shoukat, and S. A. Nawaz, "A Human–Machine Interaction Mechanism: Additive Manufacturing for Industry 5.0—Design and Management," Sustainability, vol. 16, no. 10, p. 4158, 2024, doi: https://doi.org/10.3390/su16104158.

[30] B. K. Chebolu et al., "AI Powered Cryptography for Secure Machine-to-Machine (M2M) Communication in IoT Networks," in Proc. Int. Conf. Pervasive Computational Technologies (ICPCT), Feb. 2025, doi: https://doi.org/10.1109/ICPCT64145.2025.10941066.

[31] T. Nguyen, H. Vu, L. H. Nguyen, D. T. Hoang, and E. Dutkiewicz, "FLAIR: Federated Learning with Attentive and Immune Regularization for Edge Intrusion Detection," in Proc. ACM Workshop Secure & Trustworthy Cyber-Physical Systems (SaT-CPS), 2021, pp. 3–13, doi: https://doi.org/10.1145/3494322.3494348.

[32] K. Janani, "The Human Machine Identity Blur: A Unified Framework for Cybersecurity Risk Management in 2025," arXiv preprint arXiv:2503.18255, 2025, doi: https://doi.org/10.48550/arXiv.2503.18255.

[33] M. Rupasri and T. Pavani, "A Review on Cybersecurity and Privacy Concern," Int. J. Sci. Innovation Eng., vol. 2, no. 7, 2025. [Online]. Available: https://ijsci.com/index.php/home/article/view/416.

[34] M. Alsumaidaee, M. Yahya, and A. Yaseen, "Optimizing Malware Detection and Classification in Real-Time Using Hybrid Deep Learning Approaches," Int. J. Safety Security Eng., vol. 15, pp. 141–150, 2025, doi: https://doi.org/10.18280/ijsse.150115.

[35] D. Verma, M. Kumar, and A. Kumar, "Cybersecurity in IoT: A Survey of Modern Attacks and Defense Mechanisms," in Proc. 13th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT), 2024, doi: https://doi.org/10.1109/ICCCNT59181.2024.11028034.

[36] S. Akhtar, M. Hanif, M. W. Arshad, and F. Farooq, "Cutting-Edge Malware Detection in Healthcare: Leveraging Cascaded-AlexNet Model," in Proc. ICCSA 2025 Workshops, Springer, 2025, doi: https://doi.org/10.1007/978-3-031-97663-6_7.

[37] X. Wang *et al.*, "Entropy-regulated cross-modal generative fusion for multimodal network intrusion detection," *Information Fusion*, vol. 126, p. 103581, 2025. doi: https://doi.org/10.1016/j.inffus.2025.103581.

[38] C. A. Anser Pasha et al., "Detection Of Malware Families and Classification Using Machine Learning,". 2025. Proc. Int. Conf. Knowledge Eng. Commun. Syst. (ICKECS), 2025, pp. 1–5, doi: https://doi.org/10.1109/ICKECS65700.2025.11035552.

[39] A. Gupta and D. C. Misra, "Hybrid IoT security model with integration of LSTM, BERT, ROBERTA and transform learning for attack classification," Int. J. Inf. Tecnol., 2025, doi: https://doi.org/10.1007/s41870-025-02672-0.

[40] K. Shaukat, "Pattern recognition and machine learning techniques for cyber security," Ph.D. dissertation, Open Research Newcastle, 2025. [Online]. Available: https://hdl.handle.net/1959.13/1514233.

[41] A. Hussain, A. Saadia, and F. M. Alserhani, "Ransomware Detection and Family Classification Using Fine-Tuned BERT and RoBERTa Models," Egyptian Informatics Journal, vol. 30, p. 100645, 2025, doi: https://doi.org/10.1016/j.eij.2025.100645.

[42] P. Victor, A. H. Lashkari, R. Lu, et al., "IoT malware: An attribute-based taxonomy, detection mechanisms and challenges," Peer-to-Peer Networking and Applications, vol. 16, pp. 1380–1431, 2023, doi: https://doi.org/10.1007/s12083-023-01478-w.

[43] K. M. Matsobane and M. Mokwena, "Malware Detection Using a Random Forest Method Trained on a Balanced Synthetic Dataset," ResearchGate preprint, 2025, doi: https://doi.org/10.54327/set2025/v5.i1.167.

[44] J. Strecker, R. Sadre, and B. Stiller, "On the Performance of Machine Learning Techniques for IoT Malware Detection," arXiv preprint arXiv:2110.07832, 2021. [Online]. Available: https://arxiv.org/abs/2110.07832.

[45] M. Omar, "Harnessing the Power of Decision Trees to Detect IoT Malware," arXiv preprint arXiv:2301.12039, 2023, doi: https://doi.org/10.48550/arXiv.2301.12039.

[46] N. Mohamed, "Artificial intelligence and machine learning in cybersecurity: a deep dive into state-of-the-art techniques and future paradigms," Knowledge and Information Systems, vol. 67, pp. 6969–7055, 2025, doi: https://doi.org/10.1007/s10115-025-02429-y.

[47] K. Adewole, A. Jolfaei, and M. Alazab, "Intrusion Detection Framework for Internet of Things with Explanation," Sensors, vol. 25, no. 6, p. 1845, 2025, doi: https://doi.org/10.3390/s25061845.

[48] S. A. Elsaid and A. Binbusayyis, "An Optimized Isolation Forest-Based Intrusion Detection System for Heterogeneous and Streaming Data in the Industrial Internet of Things (IIoT)," Discover Applied Sciences, 2024, doi: https://doi.org/10.1007/s42452-024-06165-w.

[49] P. Sam et al., "Malware Detection in IoT Network using DBSCAN and Ensemble Method," Library Progress International, vol. 44, no. 3, pp. 27444–27450, 2024, doi: https://doi.org/10.48165/bapas.2024.44.2.1.

[50] A. Dayoub and M. Omar, "Advancing IoT Security Posture: K-Means Clustering for Malware Detection," 2024, doi: https://doi.org/10.4018/979-8-3693-1906-2.ch012.

[51] A. Bensaoud and J. Kalita, "Optimized detection of cyber-attacks on IoT networks via hybrid deep learning models," Ad Hoc Networks, vol. 170, p. 103770, 2025, doi: https://doi.org/10.48550/arXiv.2502.11470.

[52] 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, no. 13, p. 4122, 2024, doi: https://doi.org/10.3390/s24134122.

[53] A. K. Shakya, G. Pillai, and S. Chakrabarty, "Reinforcement learning algorithms: A brief survey," Expert Systems with Applications, vol. 231, p. 120495, 2023. doi: https://doi.org/10.1016/j.eswa.2023.120495

[54] J. Terven, "Deep reinforcement learning: A chronological overview and methods," AI, vol. 6, no. 3, p. 46, 2025. doi: https://doi.org/10.3390/ai6030046.

[55] M. He, X. Wang, P. Wei, L. Yang, Y. Teng and R. Lyu, ."Reinforcement Learning Meets Network Intrusion Detection: A Transferable and Adaptable Framework for Anomaly Behavior Identification," . 2024. IEEE Transactions on Network and Service Management, vol. 21, no. 2, pp. 2477-2492, April 2024, doi: https://doi.org/10.1109/TNSM.2024.3352586.

[56] X. Li, Y. Wen, and Z. Qin, "A Comprehensive Survey of Reinforcement Learning: From Algorithms to Practical Challenges," arXiv preprint arXiv:2411.18892, 2024. [Online]. Available: https://arxiv.org/abs/2411.18892.

[57] S. K. Jagatheesaperumal, M. Rahouti, M. Aledhari, A. Hafid, D. Oliveira, H. Drid, and R. Amin, "Distributed Reinforcement Learning for IoT Security in Heterogeneous and Distributed Networks," Computing & AI Connect, vol. 1, p. 0008, 2024, doi: https://doi.org/10.69709/CAIC.2024.100109.

[58] Y. Huang et al., "Revealing the Challenges of Sim-to-Real Transfer in Model-Based Reinforcement Learning," arXiv preprint arXiv:2506.12735, 2025. [Online]. Available: https://arxiv.org/html/2506.12735v1.

[59] M. Cui et al., "DR-SAC: Distributionally Robust Soft Actor-Critic for Reinforcement Learning under Uncertainty," arXiv preprint arXiv:2506.12622, 2025. [Online]. Available: https://arxiv.org/abs/2506.12622.

[60] A. A. Alsadhan, H. Alyami, A. Alotaibi, and A. Alshahrani, "Malware Attacks Detection in IoT Using Recurrent Neural Network (RNN)," Intelligent Automation & Soft Computing, vol. 39, no. 2, pp. 135–155, 2024, doi: https://doi.org/10.32604/iasc.2023.041130.

[61] A. I. Jony and S. Amob, "A Long Short-Term Memory Based Approach for Detecting Cyber Attacks in IoT Using CIC-IoT2023 Dataset," Journal of Engineering & Computing, 2024, doi: https://doi.org/10.55056/jec.648.

[62] N. Cassavia, L. Caviglione, M. Guarascio, A. Liguori, and M. Zuppelli, "Learning autoencoder ensembles for detecting malware hidden communications in IoT ecosystems," Journal of Intelligent Information Systems, vol. 62, no. 4, pp. 925–949, 2023. doi: https://doi.org/10.1007/s10844-023-00819-8.

[63] D. Natsos and A. L. Symeonidis, "Transformer-based malware detection using process resource utilization metrics," Results in Engineering, vol. 25, p. 104250, 2025. doi: https://doi.org/ 10.1016/j.rineng.2025.104250.