IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

A Comparative Analysis of Machine Learning and Deep Learning Algorithms for IoT Intrusion Detection

Sagar Shahaji Thorat
Independent Researcher, Mumbai, Maharashtra, India
sthorat515@gmail.com

Abstract: *The rapid expansion of the Internet of Things (IoT) has connected billions of low-power, resource-constrained devices to the internet, creating a vast attack surface that traditional security mechanisms are ill-equipped to defend. Intrusion Detection Systems (IDS) built on Machine Learning (ML) have emerged as one of the most effective ways to identify malicious traffic in such heterogeneous environments. This paper presents a comparative study of widely used ML algorithms applied to IoT intrusion detection, including Decision Tree, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), Naive Bayes, and deep learning approaches such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. Using evidence drawn from existing peer-reviewed literature and benchmark datasets such as NSL-KDD, UNSW-NB15, and CICIDS2017, the algorithms are evaluated and contrasted on accuracy, detection rate, false positive rate, computational overhead, and suitability for resource-constrained IoT devices. The study finds that while ensemble methods such as Random Forest consistently deliver a strong balance of accuracy and efficiency, hybrid deep learning models such as CNN-LSTM achieve the highest detection accuracy at the cost of greater computational demand, which limits their direct deployment on constrained edge devices. The paper concludes by outlining open challenges, including dataset imbalance, real-time constraints, and adversarial robustness, along with future research directions for lightweight, explainable IDS frameworks suited to IoT environments.*

Keywords: *Internet of Things, Intrusion Detection System, Machine Learning, Deep Learning, Network Security, IoT Security, Random Forest, Support Vector Machine, CNN-LSTM*

I. INTRODUCTION

The Internet of Things (IoT) has grown from a niche concept into a foundational layer of modern digital infrastructure, connecting tens of billions of sensors, actuators, and embedded devices across homes, industries, healthcare, transportation, and smart cities. This growth has been accompanied by a corresponding rise in security incidents, as most IoT devices are designed with limited processing power, memory, and energy budgets, leaving little room for traditional, resource-intensive security software.

Conventional security mechanisms such as static firewalls and signature-based detection struggle to keep pace with the scale, heterogeneity, and constantly evolving attack patterns seen in IoT networks. Attackers frequently exploit weak authentication, unpatched firmware, and unsecured communication protocols to compromise devices, often recruiting them into botnets used for Distributed Denial-of-Service (DDoS) attacks, data exfiltration, or lateral movement into larger networks.

Machine Learning (ML) based Intrusion Detection Systems (IDS) address these limitations by learning patterns of normal and malicious behaviour directly from network traffic data, allowing them to detect previously unseen attacks without relying on a static rule set. Over the past decade, researchers have proposed and evaluated a wide range of ML

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

and deep learning algorithms for this purpose, each offering a different trade-off between detection accuracy, computational cost, and suitability for constrained IoT hardware.

This paper does not propose a new algorithm; instead, it systematically reviews and compares the algorithms most frequently used in IoT intrusion detection research, drawing on published experimental results to highlight their relative strengths, weaknesses, and practical suitability for real-world IoT deployment.

The structure of the rest of this paper is as follows.Section II states the research objectives. Section III describes the review methodology. Section IV summarises the IoT threat landscape. Section V reviews prior related work. Section VI examines the individual ML and DL algorithms in detail. Section VII presents a comparative analysis supported by reported performance figures. Section VIII proposes a layered conceptual framework synthesising these findings. Section IX introduces the benchmark datasets used across the literature. Section X discusses standard evaluation metrics. Section XI discusses open challenges, Section XII presents a SWOT analysis, Section XIII outlines future research directions, and Section XIV concludes the paper.

II. RESEARCH OBJECTIVES

- To review the major categories of security threats affecting IoT networks.
- To examine the Machine Learning and Deep Learning algorithms commonly applied to IoT intrusion detection.
- To compare these algorithms on detection accuracy, false positive rate, and computational overhead using evidence from existing literature.
- To evaluate the suitability of each algorithm for deployment on resource-constrained IoT devices.
- To review the benchmark datasets most commonly used to validate IoT-IDS research.
- To identify open challenges and propose future research directions for IoT-focused IDS design.

III. RESEARCH METHODOLOGY

A. Research Design

This study adopts a Systematic Literature Review (SLR) and comparative analysis approach. Rather than conducting new experiments, the paper synthesises reported performance results from peer-reviewed studies that evaluated ML and DL algorithms on standard IoT and network intrusion datasets.

B. Data Sources

- IEEE Xplore
- ScienceDirect
- SpringerLink
- arXiv
- Google Scholar

C. Inclusion Criteria

- Published between 2019 and 2026
- Peer-reviewed journal articles or indexed conference papers
- Focused on network or IoT intrusion detection using ML/DL
- Reports quantitative performance metrics (accuracy, precision, recall, F1-score)

D. Exclusion Criteria

- Studies unrelated to network or IoT security
- Papers without empirical evaluation

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

- Duplicate or superseded studies

E. Search and Selection Strategy

Search terms combined keywords such as 'intrusion detection', 'IoT security', 'machine learning', and 'deep learning' with Boolean operators across the data sources listed above. Titles and abstracts were first screened for relevance, followed by a full-text review of shortlisted studies to confirm they reported quantitative performance metrics on a recognised intrusion detection dataset. Studies satisfying both the inclusion and exclusion criteria were retained for comparative analysis in the sections that follow.

IV. IOT SECURITY THREAT LANDSCAPE

IoT networks face a distinct threat landscape compared to conventional IT networks, largely due to constrained device resources, diverse communication protocols, and frequently inadequate firmware update mechanisms. Commonly reported threat categories include:

A. Denial-of-Service (DoS) and Distributed DoS (DDoS)

Attackers flood IoT devices or gateways with excessive traffic, exhausting bandwidth or processing capacity and rendering services unavailable. Compromised IoT devices are also frequently recruited into botnets to launch large-scale DDoS attacks against external targets.

B. Probing and Reconnaissance Attacks

Adversaries scan IoT networks to identify open ports, active devices, and exploitable vulnerabilities prior to launching a more targeted attack.

C. Remote-to-Local (R2L) and User-to-Root (U2R) Attacks

These involve unauthorised remote access to a device followed by privilege escalation, allowing attackers to gain administrative control.

D. Man-in-the-Middle and Spoofing Attacks

Weak or absent encryption in many IoT communication protocols allows attackers to intercept, alter, or spoof data exchanged between devices and gateways.

E. Botnet Recruitment

Malware such as Mirai and its variants specifically target default or weak IoT credentials to assemble large botnets capable of coordinated attacks.

V. RELATED WORK

A substantial body of research has examined ML and DL approaches to IoT and network intrusion detection. This section summarises representative studies that inform the comparative analysis presented later in this paper.

Chaabouni et al. surveyed network intrusion detection techniques specifically adapted for IoT security, cataloguing the advantages and limitations of ML models proposed up to that point and highlighting the gap between academic benchmarks and real IoT deployment conditions.

Zolanvari et al. evaluated several ML classifiers on a testbed designed to mimic an industrial IoT plant, finding that Random Forest achieved the strongest overall true positive rate among the models tested, while SVM achieved a notably low false positive rate, illustrating that different algorithms can be preferable depending on whether minimising missed attacks or minimising false alarms is the operational priority.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

Disha and Waheed conducted a comparative study of ML models for network intrusion detection using the UNSW-NB15 dataset, reporting that ensemble and tree-based methods generally outperformed simpler linear classifiers on this more modern and realistic traffic dataset.

Mahadevappa et al. compared conventional ML classifiers for intrusion detection in edge-enabled IoT networks using the NSL-KDD dataset, observing that Multi-Layer Perceptron models were highly sensitive to network configuration choices, underscoring the importance of careful hyperparameter tuning when applying neural approaches to IoT traffic.

Booij et al. presented a broad comparative analysis of ML techniques for IoT intrusion detection, reviewing prior testbed-based studies and noting that lightweight algorithms such as C4.5 decision trees can achieve very high accuracy on individual packets, but at the cost of being able to analyse only a small fraction of total network traffic in real time.

More recent work has shifted toward hybrid deep learning architectures. Dash et al. proposed an optimised LSTM-based model for anomaly network intrusion detection, while several studies combining CNN and LSTM layers have reported accuracies exceeding 99% on CICIDS2017 and UNSW-NB15, reflecting a broader trend in the field toward hybrid architectures that capture both spatial and temporal traffic patterns at the expense of increased model complexity.

Collectively, this body of work indicates a consistent pattern: classical ensemble methods remain strong, practical baselines, while deep learning hybrids push the accuracy ceiling higher but introduce real deployment trade-offs that are directly relevant to resource-constrained IoT environments — the central comparison explored in this paper.

VI. MACHINE LEARNING ALGORITHMS FOR IOT INTRUSION DETECTION

The algorithms reviewed in this study were selected because they appear most frequently in the IoT-IDS literature and represent the major algorithmic families: tree-based methods, margin-based classifiers, instance-based learners, probabilistic classifiers, and deep neural architectures.

A. Decision Tree (DT)

Decision Tree classifiers recursively split network traffic features into a tree structure of rules, typically using criteria such as Gini impurity or information gain to choose the most discriminative feature at each split. Each path from root to leaf represents a sequence of conditions on traffic attributes (such as packet size, protocol type, or connection duration) that ultimately yields a classification of normal or malicious. DTs are computationally lightweight and highly interpretable, since the decision path for any given prediction can be traced and explained, making them attractive for constrained IoT environments. Their main drawback is a tendency to overfit on noisy or imbalanced traffic data, producing trees that memorise training data rather than generalising to new attack variants.

B. Random Forest (RF)

Random Forest builds an ensemble of many decision trees, each trained on a randomly sampled subset of the data and features, and aggregates their individual predictions through majority voting. This bagging approach substantially reduces the overfitting risk associated with a single tree and improves robustness to noisy features, which are common in real IoT traffic. Multiple independent studies report Random Forest achieving very high true positive rates with low false positive rates on IoT anomaly-detection testbeds, making it one of the most consistently strong performers in the literature, and a frequent baseline against which newer methods are compared.

C. Support Vector Machine (SVM)

SVM separates normal and malicious traffic by constructing an optimal separating hyperplane that maximises the margin between the two classes in feature space. For traffic that is not linearly separable, kernel functions (such as radial basis or polynomial kernels) project the data into a higher-dimensional space where a separating hyperplane becomes feasible. Research applying SVM to detect traffic-intensity-based attacks, common in DoS scenarios, has reported high accuracy using even a small number of carefully selected features such as minimum, maximum, and median packet arrival rates. However, SVM training time scales poorly with very large datasets, and kernel choice significantly affects both accuracy and computational cost.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

D. K-Nearest Neighbours (KNN)

KNN is an instance-based, non-parametric method that classifies a new traffic sample according to the majority class among its K nearest neighbours in feature space, typically measured using Euclidean or Manhattan distance. It requires no explicit training phase, since the entire training dataset is retained and consulted at prediction time, which keeps model-building overhead low. However, classification at inference time requires comparing each new sample against the stored dataset, which becomes computationally expensive on large traffic volumes, limiting its practicality for high-throughput IoT gateways without dimensionality reduction or indexing optimisations.

E. Naive Bayes (NB)

Naive Bayes applies probabilistic classification based on Bayes' theorem, computing the posterior probability of each class (normal or malicious) given the observed traffic features, under the simplifying assumption that features are conditionally independent given the class. This assumption rarely holds exactly in network traffic, where features such as packet count and byte count are often correlated, which can reduce classification accuracy. Nonetheless, NB is extremely fast to train and evaluate, and its low memory footprint makes it well suited to severely constrained IoT devices where computational simplicity outweighs the accuracy lost to its independence assumption.

F. Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM)

CNNs apply learned convolutional filters across structured representations of traffic features to automatically extract spatial patterns, removing the need for manual feature engineering required by classical ML methods. LSTM networks, a type of recurrent neural network, maintain an internal memory state that captures temporal dependencies across sequences of network events, making them well suited to detecting attacks that unfold over time, such as slow-rate DoS or multi-stage intrusions. Hybrid CNN-LSTM architectures combine both capabilities — CNN layers extract local spatial features, which are then passed to LSTM layers to model their temporal evolution — and have been reported to achieve detection accuracies above 99% on benchmark datasets such as CICIDS2017 and UNSW-NB15, representing the current performance ceiling among the algorithms reviewed. This accuracy, however, comes with substantially higher computational and memory requirements than the classical ML methods above, posing a deployment challenge for low-power IoT edge devices unless combined with model compression techniques.

VII. COMPARATIVE ANALYSIS

Table I summarises the relative characteristics of the algorithms reviewed, based on patterns consistently reported across the literature surveyed in Section V.

TABLE I: Comparative Summary of ML/DL Algorithms for IoT Intrusion Detection

Algorithm	Typical Detection Accuracy	Computational Cost	Interpretability	Suitability for Constrained IoT Devices
Decision Tree	Moderate-High	Low	High	High
Random Forest	High	Moderate	Moderate	Moderate-High
SVM	High	Moderate-High	Low	Moderate
KNN	Moderate-High	Moderate (inference-heavy)	High	Moderate
Naive Bayes	Moderate	Very Low	High	High
CNN-LSTM (Hybrid DL)	Very High (>99%)	High	Low	Low (without optimisation)

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

Across the reviewed studies, three consistent patterns emerge. First, ensemble tree-based methods such as Random Forest provide the most reliable balance between accuracy and computational efficiency, making them a common default choice in IoT-IDS research. Second, deep learning hybrids achieve the highest raw accuracy but require optimisation techniques such as model pruning, quantisation, or edge-cloud offloading before they are practical for direct deployment on constrained devices. Third, lightweight classical methods such as Naive Bayes and Decision Tree remain relevant specifically because of their low resource footprint, even though their accuracy alone is generally lower than ensemble or deep learning approaches.

Table II presents representative quantitative results reported by individual studies surveyed in this paper, to ground the qualitative comparison above in concrete published figures. These results are drawn from different datasets and experimental setups and are therefore not directly comparable to one another in absolute terms, but they illustrate the typical performance range achievable by each algorithm family.

TABLE II: Representative Reported Results from Surveyed Literature

Algorithm	Dataset	Reported Result	Source
Random Forest	Industrial IoT Testbed	97.44% True Positive Rate	Zolanvari et al. [7]
SVM	Industrial IoT Testbed	0.00 False Positive Rate	Zolanvari et al. [7]
SVM (Linear Kernel)	Simulated DoS Traffic	98.03% Accuracy	Jan et al. [11]
Decision Tree (C4.5)	Packet-Level Testbed	100% Accuracy (18.15% of traffic analysed)	Bakhtiar et al. [12]
CNN	NSL-KDD	99% Accuracy	Comparative DL Survey [13]
CNN-GRU (Hybrid)	CICIDS2017	99.95% Accuracy	Comparative DL Survey [13]
CNN-LSTM (Hybrid)	CICIDS2017	99.89% Accuracy	Comparative DL Survey [13]

It is important to note that very high reported accuracies, particularly on legacy datasets such as NSL-KDD, can partly reflect dataset saturation rather than guaranteed real-world performance, since these benchmarks are well studied and relatively limited in traffic diversity compared to live IoT deployments. This is one reason newer datasets such as CICIDS2017 and UNSW-NB15 are increasingly preferred in recent literature.

A. Practical Implications for IoT Deployment

The comparison developed in this section has direct implications for practitioners selecting an IDS approach for a given IoT deployment. For battery-powered sensor nodes with severe memory constraints, lightweight classifiers such as Naive Bayes or shallow Decision Trees remain the only realistic on-device option, even though a centralised gateway running a stronger model may still inspect aggregated traffic. For smart-home or small-office gateways with moderate compute resources, Random Forest offers a favourable balance of accuracy and resource use and is a reasonable default choice in the absence of more specific constraints. For large-scale industrial or enterprise IoT deployments where dedicated edge servers or cloud back-ends are available, hybrid deep learning models such as CNN-LSTM become viable and can deliver the highest detection accuracy, provided the additional latency introduced by deeper inference pipelines is acceptable for the application's real-time requirements.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 11, June 2026 Impact Factor: 8.2

These implications suggest that the choice of algorithm should not be treated as a single, universal decision but rather as a deployment-specific trade-off informed by the available computational budget, the criticality of low-latency detection, and the acceptable false positive rate for the environment in question.

VIII. PROPOSED CONCEPTUAL FRAMEWORK FOR LAYERED IOT INTRUSION DETECTION

Building on the comparative findings above, this section outlines a conceptual, layered IDS framework that distributes detection workload according to the resource profile of each tier in a typical IoT network, rather than relying on a single algorithm throughout. This framework is presented as a synthesis of recommendations implied by the surveyed literature rather than a deployed or empirically tested system.

A. Device Layer

At the device layer, where computational and energy resources are most constrained, only lightweight statistical checks or a Naive Bayes classifier are recommended. The goal at this layer is to filter out clearly benign traffic and flag suspicious patterns for forwarding, rather than to make a final classification decision.

B. Gateway / Edge Layer

Traffic flagged at the device layer, along with aggregated traffic from multiple devices, is passed to a gateway or edge node with moderate compute capacity. A Random Forest or Decision Tree ensemble is recommended at this layer, since it offers strong accuracy while remaining feasible to run without specialised hardware accelerators.

C. Cloud / Backend Layer

Traffic that remains ambiguous after edge-layer analysis, or that warrants deeper inspection due to its association with a high-value asset, is escalated to a cloud or backend system with access to GPU-accelerated infrastructure. At this layer, hybrid deep learning models such as CNN-LSTM can be applied to extract maximum detection accuracy, since latency constraints are less severe than at the edge.

D. Feedback and Retraining Loop

Confirmed attack classifications from the cloud layer are periodically fed back to retrain edge and device-layer models, allowing the overall system to adapt to evolving attack patterns over time without requiring every layer to run the most computationally expensive model continuously.

This layered approach reflects a practical synthesis of the trade-offs identified in Sections VI and VII, where lightweight algorithms handle the high-volume, low-risk majority of traffic, while reserving the most accurate but resource-intensive models for the smaller volume of traffic that genuinely requires deeper analysis.

IX. BENCHMARK DATASETS USED IN EVALUATION

A. NSL-KDD

An improved version of the original KDD'99 dataset, NSL-KDD removes redundant records and retains 41 traffic features across attack categories including DoS, Probe, Remote-to-Local, and User-to-Root. Despite its age, it continues to be referenced extensively in IoT-IDS research.

B. UNSW-NB15

This dataset was generated using a more modern traffic generation tool and includes nine attack categories, offering a more realistic representation of contemporary network behaviour than NSL-KDD.

C. CICIDS2017

Developed by the Canadian Institute for Cybersecurity, this dataset captures benign and attack traffic, including DoS, Brute Force, Botnet, and Web Attacks, across multiple protocols, and is frequently used to validate deep learning-based IDS models.

X. PERFORMANCE EVALUATION METRICS

Studies reviewed in this paper consistently evaluate IDS models using the following standard metrics:

Accuracy — the proportion of correctly classified traffic samples (both normal and malicious) out of all samples.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

Precision — the proportion of traffic flagged as malicious that is actually malicious.
Recall (Detection Rate) — the proportion of actual malicious traffic correctly identified by the model.
F1-Score — the harmonic mean of precision and recall, useful when class distribution is imbalanced.
False Positive Rate (FPR) — the proportion of normal traffic incorrectly flagged as malicious, a critical metric for operational usability.

XI. CHALLENGES AND LIMITATIONS

A. Class Imbalance

Real-world IoT traffic contains far more normal samples than attack samples, which can bias models toward the majority class and reduce sensitivity to rare but critical attack types such as U2R.

B. Resource Constraints

Many IoT devices lack the processing power and memory required to run deep learning models directly, necessitating either lightweight model design or detection at a gateway/edge layer rather than on the device itself.

C. Concept Drift

Attack patterns evolve continuously, and models trained on static historical datasets may degrade in accuracy when deployed against newer, previously unseen attack variants.

D. Dataset Realism

Widely used benchmark datasets such as NSL-KDD were generated years ago and may not fully reflect the protocols and traffic patterns of current IoT ecosystems.

E. Explainability

Deep learning-based IDS models often function as black boxes, which can hinder trust and adoption among network administrators who need to understand why traffic was flagged as malicious.

XII. SWOT ANALYSIS OF ML-BASED IOT INTRUSION DETECTION

A. Strengths

- High detection accuracy against both known and previously unseen attack patterns.
- Ability to learn directly from data without manually defined signatures.
- Scalable to large and heterogeneous IoT traffic volumes.

B. Weaknesses

- Strong reliance on data quality and how well the training set represents real-world traffic.
- Deep learning approaches demand significant computational resources.
- Reduced interpretability in complex models.

C. Opportunities

- Compressed, resource-efficient neural network designs suited to edge hardware.
- Federated learning to train models across distributed IoT devices without centralising raw data.
- Integration of explainable AI techniques to improve administrator trust.

D. Threats

- Deliberately crafted inputs intended to slip past ML-based detection undetected.
- Rapidly evolving attack techniques outpacing model retraining cycles.
- Privacy concerns associated with centralised traffic data collection.

XIII. FUTURE RESEARCH DIRECTIONS

Lightweight Deep Learning: Techniques such as model pruning, quantisation, and knowledge distillation to bring deep learning accuracy to resource-constrained IoT hardware.
Federated Learning for IDS: Training detection models collaboratively across distributed IoT devices while preserving data privacy.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, June 2026
Impact Factor: 8.2

Explainable AI (XAI) for IDS: Building interpretable detection models that allow security teams to understand and trust automated decisions.

Adversarial Robustness: Designing models resilient to adversarial examples crafted to bypass ML-based detection.

Real-Time Edge Detection: Architectures that perform detection directly at the network edge to minimise latency in responding to active attacks.

XIV. CONCLUSION

This paper presented a comparative study of Machine Learning and Deep Learning algorithms used for intrusion detection in IoT networks, drawing on evidence from existing literature evaluated against standard benchmark datasets. The comparison shows that no single algorithm is universally optimal: Random Forest offers a strong balance of accuracy and efficiency suited to many practical deployments, lightweight classifiers such as Naive Bayes and Decision Tree remain valuable for severely constrained devices, and hybrid deep learning models such as CNN-LSTM deliver the highest accuracy at the cost of computational overhead. Selecting an appropriate algorithm therefore requires balancing detection performance against the resource limitations of the target IoT environment. Future work addressing lightweight model design, federated learning, explainability, and adversarial robustness will be essential to building IDS frameworks capable of securing the next generation of IoT deployments.

REFERENCES

[1] Awajan, A. (2023). A Novel Deep Learning-Based Intrusion Detection System for IoT Networks. Computers, 12(2), 34.

[2] Mahadevappa, P., Muzammal, S. M., & Murugesan, R. K. (2021). A Comparative Analysis of Machine Learning Algorithms for Intrusion Detection in Edge-Enabled IoT Networks. arXiv preprint arXiv:2111.01383.

[3] Shakya, S., & Abbas, R. (2024). A Comparative Analysis of Machine Learning Models for DDoS Detection in IoT Networks. arXiv preprint arXiv:2411.05890.

[4] Booij, T. M., Chiscop, I., Meeuwissen, E., Moustafa, N., & den Hartog, F. T. H. (2022). A Comparative Analysis of Machine Learning Techniques for IoT Intrusion Detection. arXiv preprint arXiv:2111.13149.

[5] Disha, R. A., & Waheed, S. (2021). A Comparative Study of Machine Learning Models for Network Intrusion Detection System Using UNSW-NB15 Dataset. 2021 International Conference on Electronics, Communications and Information Technology (ICECIT), IEEE, 1–5.

[6] Dash, N., Chakravarty, S., Rath, A. K., Giri, N. C., Aboras, K. M., & Gowtham, N. (2025). An Optimized LSTM-Based Deep Learning Model for Anomaly Network Intrusion Detection. Scientific Reports, 15, 1554.

[7] Zolanvari, M., Teixeira, M. A., Gupta, L., Khan, K. M., & Jain, R. (2019). Machine Learning-Based Network Vulnerability Analysis of Industrial Internet of Things. IEEE Internet of Things Journal, 6(4), 6822–6834.

[8] Chaabouni, N., Mosbah, M., Zemmari, A., Sauvignac, C., & Faruki, P. (2019). Network Intrusion Detection for IoT Security Based on Learning Techniques. IEEE Communications Surveys & Tutorials, 21(3), 2671–2701.

[9] Canadian Institute for Cybersecurity. (2017). CICIDS2017 Dataset Documentation. University of New Brunswick.

[10] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems. Military Communications and Information Systems Conference (MilCIS), IEEE, 1–6.

[11] Jan, S. U., Ahmed, S., Shakhov, V., & Koo, I. (2019). Toward a Lightweight Intrusion Detection System for the Internet of Things. IEEE Access, 7, 42450–42471.

[12] Bakhtiar, F. A., Pramukantoro, E. S., & Nihri, H. (2019). A Lightweight IDS Based on J48 Decision Tree for the Detection of Specific Attacks on the IoT Network. arXiv / Conference proceedings on IoT security.

[13] A Comparative Analysis of Machine Learning and Deep Learning Models for Network Intrusion Detection (2025). Hybrid CNN/RNN architectures evaluated on CICIDS2017, UNSW-NB15, and NSL-KDD.