© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1722957

Comparative Analysis of Machine Learning Techniques for Intrusion Detection in Nigerian IoT Networks: A Systematic Literature Review

OJIMA GABRIELLA OB'LAMA[1], ERU AKWUMA NATHANIEL[2], AHIABA SOLOMON[3], RIDWAN KOLAPO[4], KUREVE STEPHANIE NGUHEMEN[5]

[1,2,3,4] *Department of Information Technology and Systems, Faculty of Computing, Nile University of Nigeria, Abuja - Nigeria*

[5] *Department of Computer Science, Faculty of Natural and Applied Science, Nasarawa State University, Keffi, Nigeria*

Abstract- ***A large part of the increase in cyber threats to the network of Nigeria is the introduction of IoT devices in the areas of healthcare, smart homes, transportation, surveillance, banking, and business. These cyber threats include the likes of DDoS, malware injection, spoofing, unauthorized access. In this paper, we summarize a literature review and a comparative analysis of the techniques used in machine learning for intrusion detection in IoT networks, with a focus on their suitability for resource-constrained environments in Nigeria. PRISMA framework and Parsifal guided the process of reviewing and managing. Out of 850 records discovered on the different online platforms that are IEEE Xplore, Scopus, ScienceDirect, SpringerLink, and Google Scholar, the research documented 30 peer-reviewed studies which were published between 2021 and 2026 and included 30 papers for final synthesis. The most commonly used techniques in the IoT intrusion detection literature are Random Forest, Support Vector Machine, Decision Tree, K-Nearest Neighbors, Artificial Neural Networks, Convolutional Neural Networks (CNN), and Long ShortTerm Memory (LSTM) models. CNN and LSTM models usually had the highest reported detection scores, being about 96-99% in accuracy, precision, recall, and F1-score; however, their high computational and memory requirements limit their practical deployment on low-power IoT devices. Random Forest is evaluated as the best choice because of its appropriate overall balance in terms of reporting high accuracy statistics (95-99%) and also the moderate computational cost with the strong suitability rating for Nigeria's current IoT infrastructure. Decision Tree and KNN were also mentioned as lightweight alternatives, but they could stand lower performance compared to the complex attacks. The study declares that for Nigeria's IoT cybersecurity resilience to be effective, it is imperative to have accurate, lightweight, explainable, and context-aware intrusion detection model.***

Index Terms- ***Internet of Things, Intrusion Detection System, Machine Learning, Deep Learning, PRISMA, Random Forest, Resource-Constrained Environment, Nigeria***

I. INTRODUCTION

The Internet of Things (IoT) is an ecosystem of sensors, controllers, machines, vehicles, healthcare devices, cameras, industrial assets, and software applications that wirelessly intercommunicate data with each other in real time. With the help of this interconnectivity, the processes become automated, the activities are monitored in realtime, and the new digital services are produced, but it also, in turn, increases the advanced attack surface due to heterogeneous hardware, poor credentials, unsecured interfaces, and long-serving devices that often receive less security support [1], [2]. Therefore, distributed denial-of-service, malware injection, spoofing, scanning, and unauthorized access are recurrent fears in an IoT setting. IoT devices are installed, and they typically perform the predefined routine interaction scripts, and they do not get to meet the device cluster during the script making process. where they have limited memory, processing power, and energy capacity. As a result, Machine learning–based intrusion detection systems (IDSs) are widely studied as adaptive defenses that can detect anomalous and malicious network activity, including in IoT, cloud, and 5G environments. Across many datasets and architectures, a recent report shows that the very high detection accuracies with low false alarm rates are often enough for the real-time use or near real-time

use. [2]. Machine learning and deep learning have consequently become central to adaptive IoT security because they can classify traffic, discover anomalies, and learn temporal or structural attack patterns from network flows and device telemetry [3], [4].

Detection performance alone does not determine whether a model can be deployed. IoT endpoints and gateways may have limited processing power, memory, energy, storage, and communications capacity. Feature selection, compact ensembles, graph-based representations, edge or fog processing, and lightweight neural architectures have therefore been investigated to preserve detection quality while reducing resource demand [5]-[9]. These trade-offs are particularly important when models are expected to operate continuously rather than only in a laboratory benchmark. Giving the specificity of the geographical area under Nigeria which presents a unique context for the implementation of sensors therefore suitability in terms of sensor deployment in Nigeria is envisaged as an operational design challenge rather than a blanket assertion that sensors shall work the same parameters in Nigeria A sensor is thought possible for a locale if it can continue the strongest signals with the lowest processing and storage of data Thus sensor with GFG and CANBUS can be thought of for a locality storage constraints in terms of GFG and CANBUS.This framing is relevant to home; healthcare, surveillance, banking, agriculture, and business deployments in which gateway-class or edge infrastructure may be available, but continuous access to high-end computing cannot be assumed.

Existing reviews provide broad taxonomies of intrusion detection techniques, datasets, and evaluation practices, but comparatively less attention is given to the combined question of predictive performance, computational efficiency, explainability, scalability, and contextspecific deploy ability [10], [11]. This article therefore identifies the techniques most used for IoT intrusion detection, compares their reported accuracy, precision, recall, F1-score, and computational cost, evaluates their suitability for resource-constrained Nigerian IoT environments, and develops deployment-oriented recommendations from the evidence.

II. LITERATURE REVIEW

Intrusion detection systems are commonly divided into signature-based and anomaly-based approaches. Signature-based systems compare observed activity with a repository of known attack patterns. They are effective for established threats and are usually straightforward to interpret, but they cannot reliably detect a new attack until a corresponding signature has been created. Anomaly Detection Theory is established on the belief that every considerable difference from a system's normal operation can be viewed as a likely sign of malevolent activities. According to this principle in cybersecurity, anomaly-based intrusion detection systems are built wherein after the normal network traffic pattern is first established, any deviation from these patterns is flagged to be checked for further action. What really makes this strategy powerful is the effectiveness of the mechanism in recognizing previously non-existent threats like the zero-day attacks which don't have particular syntaxes. [2].

Classical supervised learning techniques remain prominent in IoT intrusion detection. Random Forest combines multiple decision trees and is valued for robustness, nonlinear classification, tolerance of mixed features, and comparatively transparent featureimportance analysis [12]. Support Vector Machine is the optimal hyperplane by which it shares two classes and it is used in the most probable state in high-dimensional spaces, although its training and inference could become costly with an increasing data volume [13]. Decision Tree and K-Nearest Neighbors are easy to implement, therefore they are suitable for low-powered environments, but single trees are prone to overfitting, and KNN can be affected by feature scaling and the amount of training set it retains.

Neural methods of learning extend beyond just learning complex feature representations. Artificial Neural Networks build model of nonlinear relationships; Convolutional Neural Networks can pick local patterns in altered traffic features; and recurrent architectures, including Long Short-Term Memory networks, capture sequential dependencies. Graph neural networks on top of that include

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1722957

relations between the communicating hosts and flows. [3], [7]. Based on the studies that undergoes through these models, we find the biggest advantages to be their capacity to deal with the most difficult and sequential attacks, while their disadvantages are the training cost, memory consumption, and model opacity as well as the requirement of diligent drift management.

The choice of feature engineering and dataset has a major impact on the results that can be published. Featuresselection analysis indicates that even with the removal of redundant variables, a number of intrusion datasets are able to preserve almost optimal detection, which directly results in the reduction of storage and computation.[5], [6]. Public datasets differ in terms of network design, attack distribution, feature definitions, class imbalance, and realism. Among them, one of the most common ones is used for experiments is UNSW-NB15. [18], while the listed studies likewise brought to our attention numerous IoT-specific benchmark families. Therefore, it can be concluded that getting a high score in a particular benchmark is not enough to guarantee the performance in another network or in a live Nigerian deployment.

The recent development regarding the lightweight CNNBiLSTM and cooperative RPL-oriented intrusion recognition system has been established that the level of detection from high to low is inversely related to the complexity of tasks. [8], [9]. Nevertheless, the literature continues to report a practical gap between benchmark accuracy and sustained operation on constrained devices. The conceptual pipeline used in this review is shown in Figure 1: traffic is collected, cleaned, transformed into relevant features, classified by a machine learning or deep learning model, and converted into a security decision with feedback for model maintenance.

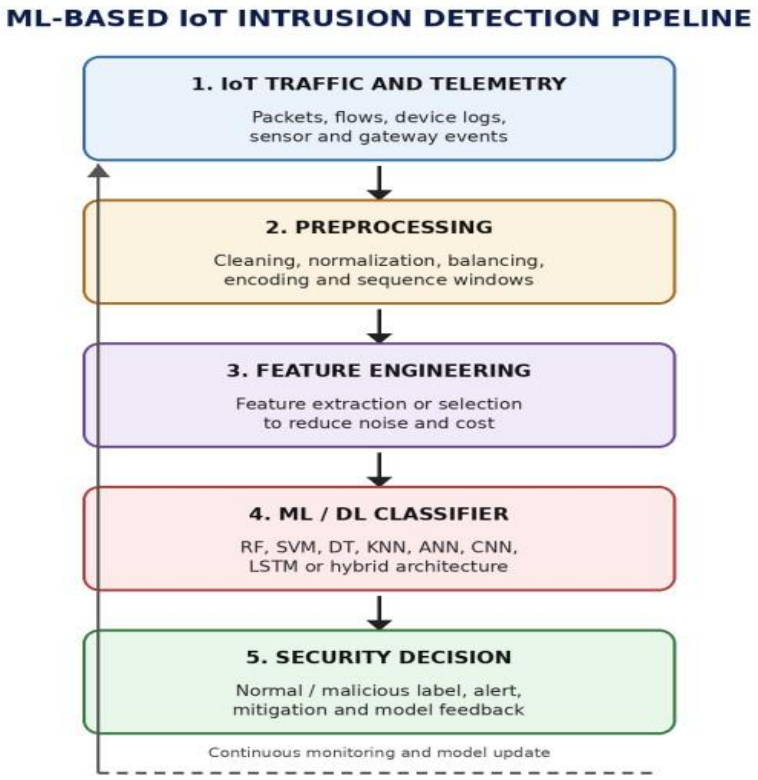

Figure 1: Machine-learning-based IoT intrusion detection pipeline

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1722957

III. METHODOLOGY

The research applied a systematic literary review model. The use of PRISMA 2020 for a transparent reporting of identification, screening, eligibility, and inclusion [15] was followed. The protocol and synthesis procedures also adhered to the rules of thumb for systematic reviews in the software and information systems [16], [17]. The review questions were generated through Parsifal, the playback medium for questions and decisions on screening decisions, the medium cached for managing inclusion and exclusion criteria, and the document storing information on data extraction. The analysis was driven by four review questions: what are the most common machine learning techniques for IoT intrusion detection; how do the selected techniques compare in accuracy, precision, recall, F1-score and computational efficiency; what recommendations are there for deploying the IoT with limited resources in Nigeria; and, what are the context-aware practices that can help deployment? These questions determined the search terms, the fields for extraction, the comparison framework, and the findings of presentation. The search will be on IEEE Xplore, Scopus, ScienceDirect, SpringerLink, and Google Scholar. The search strings contain IoT Activity, hinders description, machine-learning terminology, and technique names. A representative example would be: ("internet of Things" OR IoT) AND ("intrusion detection" OR IDS) AND ("machine learning" OR "deep learning"). The additional terms are Random Forest, Support Vector

Machine, Decision Tree, KNN, CNN, LSTM, cybersecurity, computational efficiency, and resource constrained deployment. The review is planned for the period from 2021 to June 2026.

Table 1: Review Protocol and Eligibility Criteria

Protocol Item	Specification
Review design	Systematic literature review reported with PRISMA 2020 and managed in Parsifal
Coverage	Peer-reviewed English-language studies published from 2021 to June 2026
Sources	IEEE Xplore, Scopus, ScienceDirect, SpringerLink, and Google Scholar
Core query	(IoT OR Internet of Things) AND (intrusion detection OR IDS) AND (machine learning OR deep learning)
Included	IoT-focused ML/DL intrusion detection with empirical metrics or deployment evidence
Excluded	Non-IoT studies, duplicates, editorials, non-peerreviewed items, or studies without adequate evaluation
Synthesis	Descriptive performance ranges and thematic assessment of efficiency, scalability, and suitability

At the virtual IoT systems, these studies were only covered in machine-learning-based or deep-learningbased as intrusion detection if they determined at least one relevant metric like accuracy, precision, recall, F1score, false-positive rate, computational efficiency, scalability, or deployment feasibility. Papers focusing on sectors other than non-IoT networks, items that did not have an empirical or systematic evaluation, duplicate records, editorials, opinion pieces, books, dissertations, and non-peer-reviewed material were excluded from the review. The criteria were first applied to titles and abstracts and later to the full texts of the papers. The selection counts provided uniformly in the project abstract and results table formed on the basis of the journal manuscript. The primary investigation brought out 850 records. Following the removal of 150 duplicates, 700 records were qualitatively evaluated. Screening of titles and abstracts eliminated 580 records and provided 120 full texts for eligibility assessment. Out of these, 90 full-text articles were rejected whereas 30 were included for final synthesis. The resultant workflow is depicted in Figure 2.

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1722957

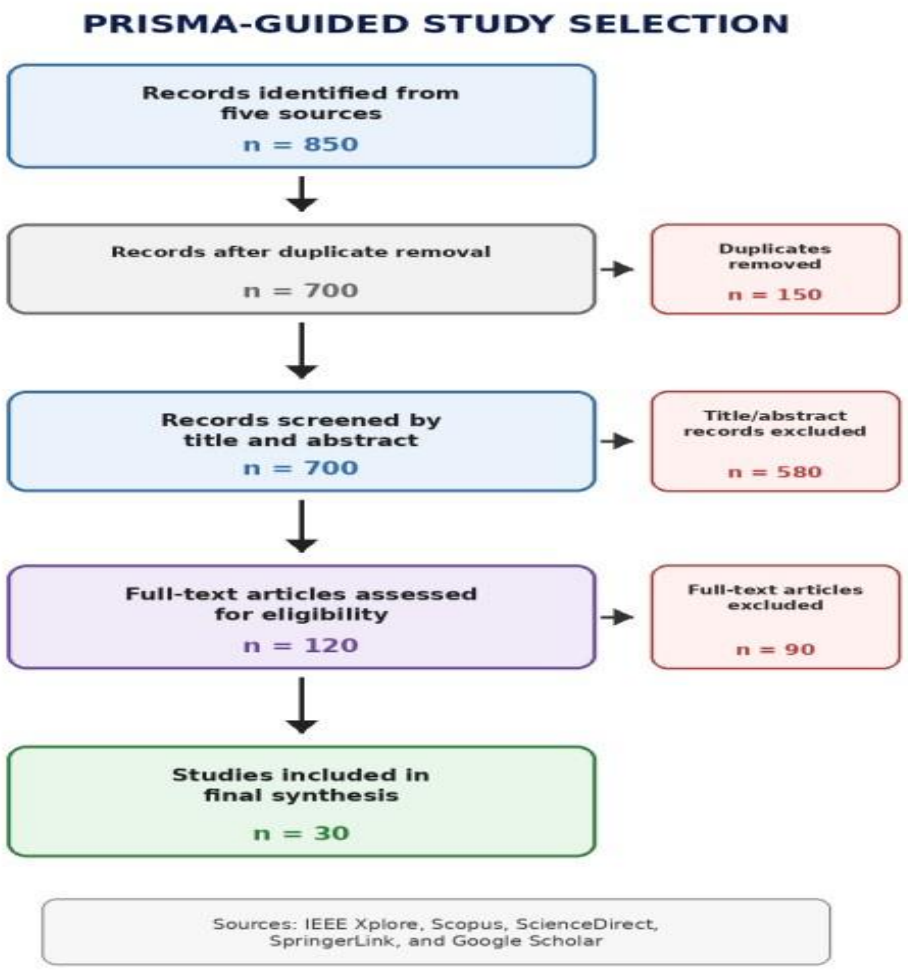

Figure 2: PRISMA-guided study-selection workflow

Each of the studies included in this research had the extraction form containing details like bibliographic information, machine learning technique, dataset, attack class, evaluation metrics, key findings, limitations, computational requirements, scalability, and deployment implications. The interpretation of accuracy, precision, recall, and F1-score was based on their conventional meanings in standard classification's forms.[14]. The review could not undergo a statistical meta-analysis because the studies were based on different datasets, class distributions, preprocessing pipelines, and experimental designs. It, therefore, only reported descriptive ranges and a thematic comparison. What is shown is a midpoint composite which is not a pooled effect estimate but a visualization of the ranges in the project table.

The review only considered secondary data that had been publicly reported. Ethical practices were respected by way of citing and managing references. The main factors that questioned the validity were the absence of uniformity in the reporting of results from different studies, the possible restrictions on database access, the great variation of benchmark datasets, and the limited number of studies that had been carried out in Nigeria, which were not specific to the region. These factors were mentioned as limitations that affected the results, in which these threats can be found.

IV. RESULTS AND DISCUSSION

The final sample consisted of 30 peer-reviewed studies that were released in the review window. Random Forest was the most frequently assessed

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1722957

classical method while SVM and Decision Tree were also met regularly. KNN and ANN were the less frequently used methods. Deeplearning articles mainly featured CNN, LSTM, and assorted frameworks, particularly in the later publications. Across the literature, classical models were usually valued for lower complexity and easier interpretation, whereas deep models were selected for representation learning and the detection of complex or sequential attacks [5]-[9].

Table 2 consolidates the performance ranges reported in the project and combines them with the review-based computational-cost and Nigerian-suitability ratings. The ranges are descriptive values reported by heterogeneous studies; they should not be read as estimates from a common dataset or experimental protocol.

Table 2: Comparative Performance and Deployment Suitability

Technique	Reported Ranges (%)	Cost	Nigeria Fit
Random Forest	Acc 95-99 P 94-99 R 94-98 F1 94-98	Moderate	Very High
SVM	Acc 90-97 P 89-96 R 88-95 F1 89-95	Moderate	Moderate
Decision Tree	Acc 88-95 P 87-94 R 86-93 F1 87-93	Low	High
KNN	Acc 85-94 P 84-93 R 83-92 F1 84-92	Low	High
CNN	Acc 96-99 P 95-99 R 95-99 F1 95-99	High	Low
Technique	Reported Ranges (%)	Cost	Nigeria Fit
LSTM	Acc 96-99 P 95-99 R 95-99 F1 95-99	Very High	Low

Note: Acc = accuracy; P = precision; R = recall. Values are descriptive author-reported ranges from heterogeneous studies and are not pooled estimates.

CNN and LSTM scored the most fantastic forecast ranges in the reporting of 96-99 which were dedicated to the partnering of four fundamental metrics. Their positive link is the same as local feature patterns where they learn and temporal dependencies that they model are of CNNs and LSTMs, respectively. Nevertheless, the same architectures were allocated high computational costs due to the requirements of larger memory, more training time, and more inference capacity. The bad quality of the model does not necessarily mean that it fails, and we can make it with uncompressed models

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1722957

on powerful gateways, edge servers, fog nodes, or any cloud infrastructure but not on low-energy end devices. Random Forest reported an accuracy rate of 95-99%, precision of 94-99%, recall of 94-98%, and F1-score of 94-98% with moderate computational cost. This configuration produced the strongest overall suitability rating. The model is relatively impervious to the impact of noisy and nonlinear data; it supports the analysis of importance of feature, and it can be installed or updated on the hardware of the gateway class. However, its major drawbacks are the memory growth that arise from large ensembles and the requirement of calibrating the class weights and thresholds for targeted class imbalances. SVM does pretty good; however, Random Forest runs the show, and it is ideal for situations where the feature spaces are very high and the control training samples are enough. Despite its technical features, big data traffic through nonlinear kernels is often cited as the explanation for the increased cost of training and inference. Decision Tree and KNN were lower on the performance scale, but they were rated the highest on the suitability scale. A small tree is both understandable and cheap to run, while KNN is easy to implement; yet KNN keeps the training examples, and its prediction cost increases with the dataset size. Hence, they are best seen as lightweight baselines, edge filters, or hierarchical construction parts rather than the first choice of detectors against complex attacks. Figure 3 depicts a descriptive midpoint composite created from the ranges in Table 2. The visualization brings out the central tradeoff: CNN and LSTM have the regular highest midpoint; Random Forest follows closely, and the Decision Tree and KNN, which are lighter weight methods, offer some performance in return for a lower resource demand. Although the composite is only for comparative purposes, it does not serve as a substitute for the dataset-specific evaluation.

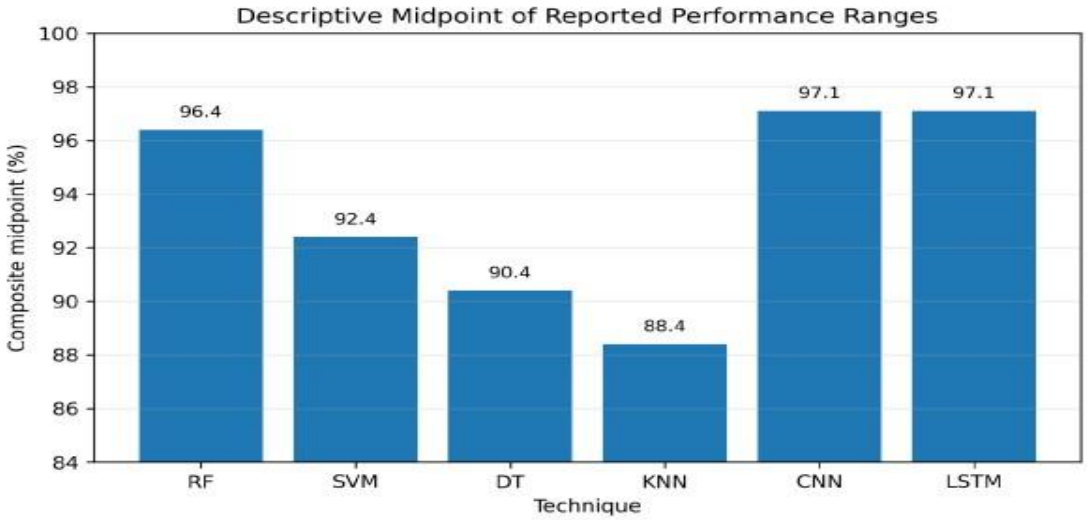

Figure 3: Descriptive midpoint composite of reported performance ranges

Evidence points towards a tiered architecture for Nigeria deployment. Only critical sensors should be deployed on low power devices, which rely on basic preprocessing. To begin, an edge or gateway node can conduct feature extraction and implement either Random Forest or a super-small Decision Tree algorithm for consistent performance monitoring. Unlike cheaper models, more advanced CNN or LSTM can be deployed based on whether the network traffic is unclear; the attacks are complex, or the analysis is done periodically. Through this fragmentation, the device accesses a deep detector only when needed while saving energy and communication. The choice of the model should always be associated with operational controls. The feature set should be trimmed down as much as possible while maintaining recall; the models should be tested with realistic class imbalance; false positives and detection latency should be included in the report alongside the accuracy; and the threshold

should be configured to the environment. Other things are the explainable outputs, the versioned models, the drift monitoring, the secure update channels, and the analyst feedback needed to build the trust and the performance. If data cannot be centralized, federated or privacy-preserving learning can be looked at, albeit its communications cost must be assessed and its susceptibility to poisoned updates evaluated before deployment. There are four main flaws in the review. Firstly, the review depended upon third-party data and did not carry out experiments on its own. Secondly, the dataset's differences, feature engineering, binary or multiclass framing, and metric calculation were causes of the restrictions of the direct comparison. Thirdly, no realtime deployment was run on a Nigerian IoT testbed, the kind of field assessment that's usually carried out was not done, which leaves the appropriateness ratings as reasoned assessments against defined resource criteria. Fourthly, the little number of Nigeria-specific studies has undermined the country's case for an extensive adoption of the research work done in other countries. Due to these constraints, local data collection followed by operational validation is an imperative and should be the topmost priority for further research.

V. CONCLUSION

This systematic literature review compared machine learning techniques for intrusion detection in IoT networks and evaluated their practicality for resourceconstrained environments relevant to Nigeria. The evidence showed that CNN and LSTM models generally produced the highest reported detection ranges, but their resource demands limit direct use on many low-power devices. When it comes to the overall performance of a given algorithm in terms of accuracy, robustness, interpretability, and moderate computational cost, Random Forest is the one that stands out. Decision Tree and KNN, as expected, were the lightweight alternatives, but SVM brought to the table strong classification at a cost of more variable scalability.

The most straightforward and efficient way to deploy the models is not just to use a single algorithm but to have a proper hierarchical context-aware structure in place. While optimized deep models are set aside for capable gateways or cloud-driven analysis, lightweight preprocessing, and classical models continuously operate on the edge. Using IoT, Nigerian companies must consider models that reflect the actual local traffic, the prevalence of realistic attacks, the required latency, memory, energy consumption, and false-positive rates rather than just the benchmark accuracy results.

Future studies must revolve around the creation of genuine Nigerian IoT datasets; conducting validation of models on actual devices and gateways; developing the most compact hybrid architectures; exploring explainable artificial intelligence; and studying federated and edgecomputing approaches. Advancement in these sections may help tread IoT from the current high benchmark scores towards reasonably priced, scalable, and dependable cybersecurity protection in real operational environments.

REFERENCES

[1] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, “A survey of machine and deep learning methods for Internet of Things (IoT) security,” *Ad Hoc Networks*, Art. no. 101792, 2020, doi: 10.1016/j.adhoc.2019.101792.

[2] 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, Art. no. 18, 2021, doi: 10.1186/s42400-021-00077-7.

[3] M. Almiani, A. AbuGhazleh, A. Al-Rahayfeh, S. Atiewi, and A. Razaque, “Deep recurrent neural network for IoT intrusion detection system,” *Simulation Modelling Practice and Theory*, vol. 101, Art. no. 102031, 2020, doi: 10.1016/j.simpat.2020.102031.

[4] E. Gyamfi and A. Jurcut, “Intrusion detection in Internet of Things systems: A review on design approaches leveraging multi-access edge computing, machine learning, and datasets,” *Sensors*, vol. 22, no. 10, Art. no. 3744, 2022, doi: 10.3390/s22103744.

[5] M. Sarhan, S. Layeghy, and M. Portmann, "Feature analysis for machine learning-based IoT intrusion detection," *arXiv:2108.12732*, 2021.

[6] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, and M. Portmann, "Feature extraction for machine learning-based intrusion detection in IoT networks," *arXiv:2108.12722*, 2021.

[7] W. W. Lo, S. Layeghy, M. Sarhan, M. Gallagher, and M. Portmann, "E-GraphSAGE: A graph neural network based intrusion detection system for IoT," *arXiv:2103.16329*, 2021.

[8] M. Jouhari and M. Guizani, "Lightweight CNN-BiLSTM based intrusion detection systems for resource-constrained IoT devices," in *Proc. Int. Wireless Communications and Mobile Computing Conf. (IWCMC)*, 2024, pp. 1558–1563, doi: 10.1109/IWCMC61514.2024.10592352.

[9] H. Azzaoui, A. Boukhamla, P. Perazzo, M. Alazab, and V. Ravi, "A lightweight cooperative intrusion detection system for RPL-based IoT," *Wireless Personal Communications*, vol. 134, pp. 2235–2258, 2024, doi: 10.1007/s11277-024-11009-2.

A. H. Ali et al., "Unveiling machine learning strategies and considerations in intrusion detection systems: A comprehensive survey," *Frontiers in Computer Science*, vol. 6, Art. no. 1387354, 2024, doi: 10.3389/fcomp.2024.1387354.

[10] Pinto, L. C. Herrera, Y. Donoso, and J. A. Gutierrez, "Survey on intrusion detection systems based on machine learning techniques for the protection of critical infrastructure," *Sensors*, vol. 23, no. 5, Art. no. 2415, 2023, doi: 10.3390/s23052415.

[11] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[12] Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.

[13] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing and Management*, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

[14] M. J. Page et al., "The PRISMA 2020 statement: An updated guideline for reporting systematic reviews," *BMJ*, vol. 372, Art. no. n71, 2021, doi: 10.1136/bmj.n71.

[15] Kitchenham and S. Charters, *Guidelines for Performing Systematic Literature Reviews in Software Engineering*, EBSE Technical Report EBSE-2007-01, Keele University and Durham University, 2007.

[16] Okoli, "A guide to conducting a standalone systematic literature review," *Communications of the Association for Information Systems*, vol. 37, pp. 879–910, 2015, doi: 10.17705/1CAIS.03743.

[17] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in *Proc. Military Communications and Information Systems Conf. (MilCIS)*, 2015, pp. 1–6, doi: 10.1109/MilCIS.2015.7348942.