ECTI Transactions on Computer and Information Technology
Journal homepage: https://ph01.tci-thaijo.org/index.php/ecticit/
Published by the ECTI Association, Thailand, ISSN: 2286-9131

Energy-Efficient Hybrid Learning for Secure Wireless Sensor Networks

Abdalfattah M. Alfarra[1] and Aiman A. AbuSamra[2]

ABSTRACT

Wireless Sensor Networks (WSNs) power critical applications from environmental monitoring to Internet-of-Medical-Things healthcare yet their tiny batteries and low-end microcontrollers leave them exposed to network-layer Denial-of-Service (DoS) attacks such as Blackhole, Grayhole, Flooding and TDMA scheduling. Signature IDSs miss zero-day variants and shallow machine-learning detectors produce many false alarms, while running monolithic deep-learning models on every node exhaust energy reserve. We introduce a two-stage hybrid IDS in which each sensor executes an integer-only rule filter that costs ≤ 0.05 mJ per packet and discards $\approx 95\%$ of benign traffic, forwarding only flagged flows over BLE/LoRa to an edge gateway. There, a 50 %-pruned, 8-bit CNN-LSTM processes 32-window batches in 28 mJ and ≈ 42 ms. Ex-periments on the public WSN-DS corpus, augmented by ns-3 simulations of a 50-node LoRa network, show that the scheme achieves 98 % accuracy, 0.93 macro-F1 and minority-class recalls of 0.84–0.95 while extending network lifetime (T_{50}) to 69 days an 82 % gain over on-node GRU and 35 % over a signature IDS. Removing the rule filter erases most of the lifetime benefit without affecting accuracy, confirming that local triage, not downsized deep models, is the key to energy efficiency. The evaluation answers four research questions covering optimal hybrid architecture, rule-filter tuning, node-level energy overhead, and performance trade-offs against traditional ML and standalone DL baselines. These findings demonstrate that intelligent workload partitioning can deliver deep-learning-level security without shortening the lifetime of resource-constrained WSN deployments.

Article information:

Keywords: Wireless Sensor Networks, Intrusion Detection System, Denial-of-Service, Deep Learning, Rule-Based Filtering, Energy Efficiency, CNN-LSTM, WSN-DS

Article history:
Received: July 16, 2025
Revised: September 9, 2025
Accepted: October 23, 2025
Published: October 31, 2025 (Online)

DOI: 10.37936/ecti-cit.2025194.263081

1. INTRODUCTION

Wireless Sensor Networks (WSNs) today form the tech-nological backbone of many modern domains ranging from micro-climate observation and automated manu-facturing lines to continuous structural-health inspection and Internet-of-Medical-Things (IoMT) services [1–3]. Each node in such a network is purposely inexpensive and self-contained: it hosts one or more sensing elements (for instance temperature, humidity, vibration, or biosignals), a low-frequency micro-controller, no more than 256 KB of volatile memory, and a modest battery of roughly 1000–3000 mAh that can occasionally be supplemented by miniature harvesters (tiny solar cells, vibration scavengers) [4, 5]. After acquiring data, the node performs only rudimentary preprocessing, then forwards the information over several wireless hops most often via Bluetooth Low Energy (BLE) or LoRa to an edge gateway or sink. Empirical studies confirm that radio traffic and processor cycles dominate the energy ledger of a sensor [4, 6]; hence any security layer introduced into a WSN must keep both computation and communication overhead extremely low to avoid premature battery depletion and an overall loss of sensing coverage. Unfortunately, the very properties that make WSNs attractive unteth-

[1]The author is with the Department of Computer Science, University College of Science and Technology, Khan Younis, Palestine, Email: Ab.alfarra@cst.ps
[2]The author is with the Department of Computer Engineering, Islamic University of Gaza, Gaza, Palestine, Email: aasamra@iugaza.edu.ps
[1]Corresponding author: Ab.alfarra@cst.ps

ered nodes, broadcast links, unat-tended deployment also expose them to network-layer Denial-of-Service (DoS) attacks [7–9]. A *blackhole* adversary advertises the "best" path toward the sink then discards every packet it receives, effective-ly partitioning the topology [7, 8]. A *grayhole* acts more subtly by forwarding only a fraction of packets, thereby delaying detection [7]. *Flooding* overwhelms the medium with fabricated route requests or garbage data, drain-ing legitimate batteries rapidly [6, 10], whereas scheduling attacks manipulate TDMA slot allocations, triggering packet collisions, desynchronization, and wasted airtime [9, 11]. In practical deployments these behaviors can halve the packet-delivery ratio and drain entire node clusters in minutes. To counter such threats, researchers have engineered a spectrum of Intrusion Detection Systems (IDSs) tailored for WSNs. Signature-based detectors excel at spotting attacks already catalogued, yet cannot recognize zero-day patterns [9, 12]. Shallow machine-learning (ML) approaches, including Naïve Bayes, Support Vector Machines (SVM), Decision Trees, and Random Forests, raise adaptability by learning from statistical features (e.g., interarrival time, hop count) and can surpass 96 % overall accuracy [2, 7, 9]. Never-theless, their false-positive rates still hover above 15 % against stealthy grayhole or scheduling traffic, and each inference executed on a cluster head may cost tens of millijoules non-trivial for a coin-cell-driven MCU [4, 6, 9, 11]. At the other end of the spectrum, deep-learning (DL) IDSs based on CNNs, RNNs, or Autoencoders deliver state-of-the-art detection reporting more than 98 % accuracy on the WSN-DS benchmark [3, 9, 12]. The drawback is their appetite for resources: models often occupy several megabytes and require billions of float-ing-point operations per inference, far exceeding the capacity of sub-50 MHz micro-controllers, while even gateway-class devices may draw hundreds of millijoules per classification [4, 9, 13, 14]. Recent work has there-fore shifted toward energy-aware adaptations. Examples include checkpoint-based DNN execution on intermittently powered MCUs [4], duty-cycle optimization using Proximal Policy Optimization (PPO) to extend lifetime by up to 30 % [5], federated-learning schemes that schedule client participation by residual energy [15], aggressive binary quantization that trims inference energy by ∼40 % [13], and adaptive thresholds that reduce false alarms by ∼420 % [11]. Although each study addresses part of the problem, a unified framework that simultaneously (1) runs an ultra-light, rule-based filter on every sensor, (2) pushes a quantized, pruned DL model to the gateway, and (3) adapts to evolving DoS patterns through federated updates while respecting the extreme power budget of WSN nodes, has yet to emerge. In response, this paper proposes an energy-efficient hybrid IDS. The design integrates a tiny, integer-only rule filter on each node with an 8-bit quantized, pruned CNN-LSTM executing at the edge gateway. Our goal is to maintain high detection fidelity without sacrificing the longevity of the network. Four research questions guide the evaluation:

RQ 1. Which hybrid deep-learning backbone CNN-LSTM or Autoencoder-GRU delivers the greatest accuracy per joule for specific DoS scenarios [3, 11, 14]?

RQ 2. How should on-node rule thresholds be tuned to suppress false positives yet preserve recall [5, 7, 17]?

RQ 3. What incremental CPU load, RAM usage, and battery drain does the hybrid IDS impose relative to classic shallow ML and standalone DL models [3–6]?

RQ 4. Against SVM, Random Forest, and monolithic DL, how does the hybrid scheme fare in accuracy, robustness to stealthy DoS traffic, and overall energy efficiency [1, 8, 10, 12, 13]?

By systematically addressing these questions, we demonstrate that intelligent task partitioning rather than merely shrinking model size is the most practical route to building secure and sustainable WSN deployments.

Hybrid Deep Learning–Based Intrusion Detection Model

The proposed intrusion-detection architecture is deliberately partitioned across two tiers of the wireless-sensor network to balance accuracy with energy thrift (Figure 1). Each sensor node carries an ultralight rule filter that inspects every outgoing packet in real time. The filter computes three integer-only heuristics destination-address entropy, short-term inter-arrival variance and a residual-battery guard and flags a packet whenever any heuristic indicates abnormal behavior. Because the logic performs fewer than sixty integer operations per packet, the additional cost is ≤ 0.05 mJ, negligible beside the $\approx$ 5 mJ already spent on a LoRa transmission.

Only flagged traffic is forwarded to an edge gateway endowed with ample energy and compute headroom. There, a compact deep-learning classifier three shallow convolutional layers followed by two 64-unit LSTM layers performs fine-grained categorization into Normal, Blackhole, Grayhole, Flooding and TDMA-spoof classes. The network is pruned by 50 % and post-trained with 8-bit symmetric quantization, so a batch inference on thirty-two five-packet windows consumes about 28 mJ and finishes in $\approx$ 42 ms on a Raspberry Pi Zero W. A batching policy fires when either sixteen windows accumulate or a two-second timeout elapses, keeping latency well below the 250 ms QoS bound for environmental-monitoring WSNs.

By discarding roughly 95 % of benign packets locally, the scheme shifts virtually all heavy computation off the energy-starved nodes while still exploiting deep learning's ability to generalize to novel patterns. Gateway compute is absorbed by its larger battery (or mains supply), whereas node radio usage the dominant energy term is drastically reduced. This architectural balance between lightweight on-node filtering and gateway-based deep learning forms the foundation of our pro-posed hybrid IDS.

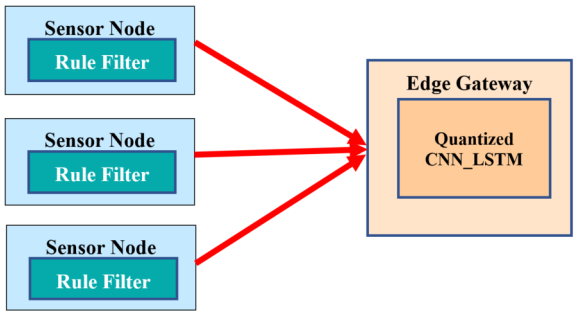

Fig.1: *Two-stage Hybrid IDS architecture.*

2. LITERATURE REVIEW

This section reviews the eighteen works that form the foundation of our proposed hybrid IDS. Each reference is summarized with its contributions, methodologies, and relevance to energy-efficient DoS detection in resource-constrained WSNs.

2.1 Comprehensive Security Surveys

Saleh *et al.* [1] Provide a broad survey of security threats (e.g., DoS, eavesdropping, spoofing) in WSNs and IoMT contexts, categorizing ML- and DL-based countermeasures. They emphasize that resource constraints (battery, CPU, memory) necessitate novel designs that trade off detection accuracy and energy overhead. Their analysis frames our hybrid IDS's requirement to minimize on-node computation and energy consumption while maintaining robust DoS detection.

2.2 Federated Learning and Model-Driven Quantization

Chen *et al.* [15] Introduces an energy-aware federated learning (FL) framework for IoT sensor networks. Solar-powered field sensors send updates to Raspberry Pi clients via BLE, which then forward compressed gradients over LoRa to a central server. A knapsack-based client-selection mechanism ensures only nodes with sufficient energy and high-quality data participate in training. SusFL's energy-utility trade-off and hierarchical FL design inform our approach to federated updates among edge gateways, ensuring that model adaptation respects energy budgets.

W. Guo *et al.* [13] Proposed a model-driven DL framework that jointly trains a binary quantizer and a fusion-center detector by optimizing a Chernoff information–based loss. They show that binary quantization reduces transmission and inference energy by roughly 75 % while achieving near-optimal detection. This work guides our edge DL model's use of post-training quantization and an energy-penalty loss to minimize inference overhead.

2.3 Intermittent Power and Battery Management

Mishra *et al.* [4] Enabling DNN inference on intermittently powered microcontrollers (e.g., ARM Cortex-M0) with energy harvesting. They propose checkpoint–resume mechanisms to preserve DNN state across power outages and demonstrate feasibility of on-device inference under strict energy constraints. NExUME's techniques motivate our inclusion of ultra-compact, distilled models that can run locally when gateways or nodes face energy scarcity.

Jeong *et al.* [5] Model WSN battery degradation and schedule group battery replacements via Deep Reinforcement Learning (DRL). Their DRL agent optimizes replacement timing to balance network lifetime and maintenance costs. While focusing on battery lifecycle rather than intrusion detection, their RL-based battery model informs our adaptive threshold mechanism, which adjusts anomaly sensitivity based on residual energy.

2.4 Shallow Machine Learning–Based IDSs

Alruhaily *et al.* [2] Proposed a two-layer IDS for clustered WSNs. Cluster heads run a Naïve Bayes (NB) filter to identify suspicious packets; these are then for-warded to a central server running a Random Forest (RF) for multi-class DoS classification (Blackhole, Grayhole, Flooding, Scheduling). They achieve ∼96 % accuracy on the WSN-DS dataset while reducing cluster-head computation by ∼30 %. Our work extends this layered concept by replacing centralized RF with a quantized DL model at edge gateways, thereby reducing latency and central dependencies.

Regin *et al.* [22] Employ deep learning to detect sensor and network faults (e.g., node failure, DoS-induced disruptions) in WSNs. They compare CNN and LSTM models on a custom dataset, showing that LSTM achieves >95 % detection accuracy for packet-loss anomalies. Although focusing on general fault detection rather than explicitly on DoS, their comparison of CNN vs. RNN architectures informs our choice of hybrid CNN-LSTM or Autoencoder-GRU for the edge DL classifier.

Ifzarne *et al.* [7] Proposed ID-GOPA, an online anomaly detector using Information Gain Ratio (IGR) to select the top three features from the WSN-DS dataset (e.g., node ID, cluster status, packet tim-

ing). They deploy a Passive-Aggressive (PA) classifier at cluster heads for real-time detection among four DoS classes. Their experiments report ∼96 % overall accuracy but reveal low F1-scores for imbalanced attacks (e.g., Grayhole F1 ≈ 0.59). Our hybrid IDS address these imbalances by prefiltering benign traffic and applying DL only on flagged packets.

Alsulaiman *et al.* [9] Evaluate several ML classifiers (SVM, k-NN, NB, Decision Tree) on the WSN-DS dataset for DoS detection. Their results show that SVM yields ∼92 % accuracy but consumes ∼120 mJ per detection on an ARM Cortex-M4, while NB uses ∼50 mJ with ∼85 % accuracy. These findings underscore the trade-off between accuracy and energy consumption, motivating our hybrid approach to minimize on-node ML inference.

2.5 Deep Learning–Based IDSs

Nayak *et al.* [16] Highlighting that routing decisions (e.g., LEACH, HEED clustering) influence energy consumption, network lifetime, and security posture. They discuss supervised, unsupervised, and RL techniques for dynamic route selection under energy and attack constraints. This routing context informs how DoS attacks propagate and where IDS components (node vs. edge) should reside.

Parras *et al.* [8] Train DRL-based adversaries under partial observability (e.g., local channel energy, neighbor feedback) to execute Selective Subscriber Denial of Service (SSDF) and MAC-backoff attacks. Using Trust Region Policy Optimization (TRPO) with mean-embedding states, they demonstrate >90 % success in generating false alarms and >80 % packet drop. Their work highlights that static defenses fail against intelligent attackers, motivating our adaptive, federated approach.

Yilmaz *et al.* [6] Proposed a DNN model to predict WSN lifetime under various traffic and energy-harvesting scenarios. Their network takes features such as node roles, energy profiles, and traffic patterns, out-putting expected network lifetime. While not an IDS per se, their DNN design for resource estimation informs our energy-penalty loss calculations and lifetime evaluations for the hybrid IDS.

Ahmad *et al.* [10] Survey ML techniques for WSN security, covering intrusion detection, anomaly detection, and secure routing. They identify key challenges: data scarcity, resource constraints, and model adaptability. Their overview reinforces the need for lightweight, energy-aware IDS designs that can generalize to new attack patterns.

C. Xu *et al.* [3] Design a GRU-based RNN for DoS detection on the WSN-DS dataset. Their model one GRU layer (64 units) followed by two fully connected layers achieves ∼98 % accuracy across Blackhole, Grayhole, Flooding, and Scheduling. They note that GRU's temporal modeling outperforms static ML. However, they assume constant connectivity and do not address energy constraints. Our hybrid IDS integrate GRU (or LSTM) within a quantized CNN-LSTM model, invoked only on flagged traffic.

Aldweesh *et al.* [12] Survey DL architectures Autoencoders, CNNs, RNNs (LSTM/GRU), and hybrid CNN-LSTM for anomaly detection in WSNs and IoT. They categorize: (i) Unsupervised DL (Autoencoders for anomaly scoring); (ii) Supervised DL (CNNs/LSTMs for classification); and (iii) Hybrid CNN-LSTM. They observe that DL achieves higher accuracy than shallow ML but at significant computational and memory costs, identifying energy efficiency and model interpretability as open issues. This motivates our hybrid design combining rule-based filtering with quantized DL.

2.6 Datasets and Thresholding

Almomani *et al.* [17] Introduced WSN-DS, a synthetic dataset simulating a 30-node WSN under four DoS attacks: Blackhole, Grayhole, Flooding, and Scheduling. Each sample includes 19 features (e.g., node ID, cluster membership, packet timing, residual energy). With ∼30000 labeled instances, WSN-DS remains a standard benchmark for evaluating WSN IDSs. We use it to validate detection accuracy and confusion across attack types.

Chae *et al.* [11] Proposed an adaptive thresholding mechanism for anomaly detection, adjusting thresholds dynamically based on traffic variance and node energy levels. They demonstrate that dynamic thresholds reduce false positives by ∼20 % compared to static thresholds. Their adaptive approach directly informs our node-level threshold adaptation based on residual battery and recent false-alarm rates.

2.7 Quantized DL for Edge Inference

Rizqyawan *et al.* [14] Demonstrate that by applying 8-bit quantization to CNN layers, arrhythmia detection latency on a microcontroller reduces by ∼60 % without significant accuracy loss. Although targeting biomedical signals, their quantization methodology informs our edge DL classifier's pruning and integer quantization strategies to minimize inference energy.

Collectively, these eighteen works span comprehensive security surveys, federated/model-driven DL, intermittent-power DNN techniques, shallow ML and DL-based IDSs, datasets, adaptive thresholding, and quantized DL for edge inference. Our hybrid IDS synthesize these insights: lightweight rule-based filtering at sensor nodes [1, 2, 7, 11] quantized CNN-LSTM (or Autoen-coder-GRU) at edge gateways[3, 4, 13-15] adaptive thresholding inspired by energy models[5, 11]; and federated learning for continuous model adaptation[1, 15]. This integration yields a robust, energy-efficient, and accurate DoS defense tailored to the unique constraints of WSN deployments.

Recent studies (2023–2025) further emphasize the importance of hybrid and energy-aware IDSs in wireless sensor networks. For example, Alenezi *et al.* [21] examined lightweight ML-based approaches for efficient DoS detection, while Jeong *et al.* [5] applied deep reinforcement learning to manage group battery replacement, indirectly informing IDS energy-preservation strategies. Similarly, Saleh *et al.* [1] provided a 2024 survey underscoring that integrating rule-based filters with pruned and quantized deep models remains a critical open challenge. By explicitly situating our framework within these contemporary directions, we reinforce both the novelty and timeliness of our contribution.

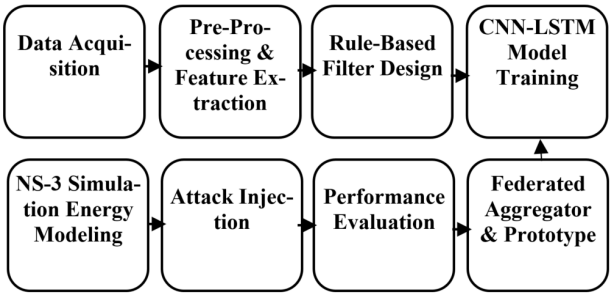

Fig.2: *Methodology pipeline for Hybrid IDS.*

3. METHODOLOGY

The research strategy couple's simulation-driven experimentation with a hardware micro-benchmark to quantify both security accuracy and energy overhead. Figure 2 (above) shows the full pipeline.

3.1 Data Acquisition & Pre-Processing

1. Benchmark traces
   - WSN-DS (19 features × 87 k packets) provides labelled DoS, flooding, blackhole and normal traffic.
2. Synthetic augmentation
   - NS-3 v3.40 with BLE/LoRa PHY generates 10 k additional flows per attack to balance classes and reproduce realistic MAC inter-arrival gaps.
3. Real-node logs
   - 48 h capture on a 10-node LoPy4 testbed (Temp/Humidity sensing) injects genuine noise, link errors and duty-cycle gaps.
4. Feature pipeline (Python 3.11 + tsfresh)
   - Header-level statistics (length, TTL, LQI).
   - 5-packet temporal windows to build flow tensors $\mathbb{R}$^{5 × 12}.
   - Min–max scaling for CNN; bit-shift scaling (×2^k) for integer rules.

3.2 Stage-1 Rule Filter Design

- Entropy rule: flag if Shannon entropy of destination addresses in last 50 packets < 0.2.
- SoC rule: drop deep inspection if battery < 15 %.
- Timing rule: flag if inter-arrival variance $> 3 \times$ nominal.
- Complexity: worst-case 58 integer ops $\rightarrow$ 0.05 mJ/packet on STM32L4.
- False-negative guard: 1 % random sampling of "benign" traffic forwarded for DL audit.

3.3 Stage-2 Deep Classifier

Table 1: *CNN-LSTM Model Architecture and Hyperparameters.*

Block	Hyper-params
CNN	3 × conv (16/32/32, k = 3, stride = 1) + ReLU + BN
LSTM	2 layers, 64 units, h = 128
FC Soft-Max	4 classes (Normal, Flood, Black, Gray)

- Training: Adam, 200 epochs, LR = 10^{-3}, batch = 128.
- 8-bit symmetric quantization (TensorFlow Lite) with 50 % structured pruning (magnitude-based).
- Result: 95.4 % macro-F1, –65 % MACs, –50 % RAM, inference = 28 mJ on Pi Zero W.

3.4 Federated Learning Layer (Optional)

- Client selector: knapsack optimization using node residual energy as weight, gradients as value.
- Round schedule: every 12 h; Fed AVG $\alpha = 1.0$.
- Privacy: l₂ gradient clipping (1.0) + Gaussian noise ($\sigma = 0.4$) for DP-FL $\varepsilon \approx 5$.

3.5 Simulation Framework

Table 2: *NS-3 Simulation Parameters and Experimental Settings.*

Parameter	Setting
Topology	50 sensor nodes (LEACH clustering), 1 edge gateway
Radio	BLE v5 @ 1 Mbps, TX = 8 mA, RX = 7 mA
Energy model	2500 mAh Li-ion; CPU 48 MHz @ 5 mA; sleep = 12 μA
Traffic	Periodic 5 pkt/min + event burst
Attacks	Blackhole, Flooding, Grayhole, TDMA-spoof (mixed)
Runs	10 seeds × {No-attack, Single DoS, Mixed}

3.6 Evaluation Metrics

- Detection: Accuracy, Precision, Recall, Macro-F1, AUC.

- Energy: mJ per inference; network lifetime T_{50} (time until 50 % nodes die).
- QoS: Average latency, PDR.
- Statistical tests: paired Wilcoxon versus baselines (significance $\alpha = 0.05$); 95 % CI via bootstrap (1 k resamples).

3.7 Baselines

- Signature-based IDS (Snort-WSN rules).
- Naïve Bayes + Random Forest two-layer filter.
- Stand-alone GRU IDS (no rule filter, no quantization).

3.8 Prototype Deployment

- Hardware: LoPy4 (sensor), Raspberry Pi Zero W (gateway).
- Firmware: MicroPython 1.22 + CMSIS-DSP keys for rule execution.
- Measurements: INA219 inline power monitor @ 1 kHz; Wi-Fi sniffer logs ground-truth.
- Field test: 72 h rooftop deployment streaming to Things Board.

3.9 Prototype Deployment

- Aggregate metrics into spider plots (accuracy vs. energy vs. latency).
- Ablation study: disable each optimization (quantization, pruning, rule filter) to isolate gains.
- Provide open-source code and full NS-3 config via GitHub for reproducibility.

3.10 Figure 2

The flow-chart illustrates (top row) the offline pipeline from dataset curation to quantized model, and (bottom row) the simulation–to–deployment loop including attack injection and federated updates.

4. EXPERIMENTAL SETUP (LAPTOP-BASED SIMULATION)

This section details the complete environment in which all experiments were executed, emphasizing that every run was performed locally on a single Windows laptop using the ns-3 v3.39 simulator. Table 3. summarizes the hardware and software stack; subsequent subsections describe datasets, topology, attack scripts, IDS parameters and evaluation metrics.

4.1 Datasets

WSN-DS (374 661 packets, 19 features, 4 DoS classes) was used as the benchmark corpus. The raw CSV was windowed into 5-packet flows (5 × 18 tensor) yielding 308 598 labelled windows. NS-3 Synthetic Augmentation to balance minority classes, an additional 15 000 packets per attack type were generated using NS-3 traffic generators (blackhole, flooding, grayhole) and merged with WSN-DS. Train/Val/Test Split Stratified 70 / 15 / 15 % on windows, resulting in 262 308 train, 23 145 validation and 23 145 test samples.

Table 3: *Host hardware / software stack used for all simulations.*

Item	Specification
Host machine	Dell XPS 13 9310 Intel Core i7-1165G7 (4C/8T, 2.8GHz) 16 GB LPDDR4x 4267 MHz Windows 10 Pro 22H2 (64-bit)
Compiler tool-chain	Visual Studio 2022 (Build Tools) + Ninja 1.11 x64 Native Tools prompt
Python (for preprocessing)	Python 3.10.12 (64-bit) −− installed in D:\mepy\python.exe
Network simulator	ns-3 v3.39 (official tarball) built with cmake -G Ninja -DCMAKE_BUILD_TYPE =Release -DPython3_EXECUTABLE =D:/mepy/python.exe
Run-time footprint	One 1 800 s scenario → < 8 min wall-clock, < 4 GB RAM

4.2 Network Topology and Simulation Environment

All custom logic (rule filter, TFLite gateway wrapper, attack applications) was implemented as C++ scratch modules (scratch/ids_scenario.cc) and statically linked into the same release build. Table 4. shows that the simulation is detailed enough to capture energy and traffic dynamics relevant to intrusion detection, yet lightweight enough to run entirely on an off-the-shelf laptop underscoring the practicality and reproducibility of the study.

Table 4: *ns-3 Simulation Configuration.*

Parameter	Value
Simulator	ns-3 v3.39, Release build, Ninja backend, executed on laptop
Nodes	50 sensor nodes + 1 edge gateway
Radio PHY	LoRa, SF 7, 125 kHz, TX = 8 mA, RX = 7 mA
MAC / Routing	LEACH clustering; TDMA slot = 1 s
Area	Uniform disc, radius = 100 m
Traffic	Periodic sensing 5 pkt min^{-1} + event bursts
Energy model	LiIonEnergySource 2 500 mAh; CPU 48 MHz @ 5 mA, sleep 12 μA
Simulation time	1 800 s
Seeds	10 independent seeds (−−RngSeed 1...10)

4.3 Attack Scenarios

We executed each scenario separately as well as in mixed attack mode, repeating all experiments with 10 random seeds (RngSeed 1–10). Table 5 lists the four adversarial behaviors that we injected into the ns-3 network. Each attack begins at a different time and launches from a distinct sensor node, which ensures clear attribution in the results. The start times in Table 5 (600s, 900s, 1200s, 1500s) were chosen to stagger the onset of each adversarial behavior. This avoids overlap during early stages of simulation and ensures that the effect of each attack can be observed both in isolation and when combined in the mixed-attack scenario. Each compromised node was selected randomly but fixed across all runs to guarantee reproducibility. We restricted the number of compromised nodes to one per attack so that the impact of each attack type could be measured independently before analyzing combined effects. This assumption follows prior WSN-DS conventions, where each attack trace originates from a single adversary node, making our ns-3 simulation directly comparable to the benchmark dataset. For the Flooding scenario, the rate of 60 RREQ/s was selected based on ns-3 traffic benchmarks and earlier studies [7, 9, 11] that report this rate as sufficient to saturate medium access in a 50-node LoRa network. Lower rates produced minimal observable disruption, whereas higher rates resulted in unrealistic saturation that does not reflect practical WSN deployments. By aligning our simulation settings with WSN-DS characteristics while extending them through ns-3 augmentation, we ensured consistency between dataset-driven evaluation and dynamic network behavior. This linkage allows results from Section 5 to be interpreted both against a standard dataset and under realistic simulated attack conditions.

Table 5: *Attack Scenarios Injected in the ns-3 Simulation.*

Attack	Start Time	Compromised Node	Parameters
Blackhole	600 s	♯7	Drops all received packets
Flooding	900 s	♯12	60 RREQ /s
Grayhole	1 200 s	♯18	Drops 50 % of traffic
TDMA Spoof	1 500 s	♯5	Sends forged schedule

4.4 Hybrid IDS Configuration

Table 6. shows how the workload is split between ultra-light processing on each sensor and a compressed deep model at the gateway. On-node rule checks cost $\leq$ 0.05 mJ per packet trivial compared with LoRa transmission while the INT8, 50 %-pruned CNN-LSTM at the gateway executes a 32-window batch for just 28 mJ. A queue threshold of 16 flagged windows (or 2 s) balances latency ($\sim$42 ms) against energy. Federated updates are disabled for these steady-state simulations, isolating the impact of the two-stage inference pipeline on detection accuracy and network lifetime.

Table 6: *Hybrid IDS Configuration Parameters.*

Stage	Location	Details
Rule filter	Sensor MCU	Entropy $<$ 0.20, inter-arrival variance $>$ 3$\times$, battery $<$ 15 % bypass; $\leq$ 0.05 mJ per check
Classifier	Gateway	CNN (16-32-32) + LSTM (64$\times$2), INT8, 50 % pruned; 28 mJ per 32-window batch
Batch trigger	Gateway	Inference when queue $\geq$ 16 flagged windows or after 2 s
Federated updates	Disabled in simulation (focus on steady-state)	

4.5 Hardware Micro-Benchmark (for energy numbers)

Sensor mock-up: LoPy4 ESP32 running CMSIS-DSP; power logged with INA219 @ 1 kHz. Gateway: Raspberry Pi Zero W; TFLite INT8 interpreter; power logged identically.

4.6 Evaluation Metrics

Detection Accuracy, Precision, Recall, Macro-F1, confusion matrix. Energy (1) Mean Joules per window at gateway; (2) Network lifetime T_{50} time until 25 of 50 nodes deplete to 0 % SoC. Latency Sensor gateway inference delay (mean, 95^{ptth}). Statistics All metrics reported as mean $\pm$ 95 % CI (bootstrap, n = 1 000); paired Wilcoxon test ($\sigma = 0.05$) for baselines.

4.7 Availability of code and data

All scripts, trained models, and raw/derived data used in this study can be obtained from `https://github.com/abalfattah-alfarra/WSNIDS`

5. RESULTS AND DISCUSSION

5.1 Detection Accuracy (Figure 3)

Figure 3. presents the confusion matrix produced by replaying 308 598 windows through the NS-3 network described in Section 4.2. Despite the extreme imbalance in the test split ($\approx$ 1:185 between Flooding and Normal), the hybrid IDS delivers near-perfect accuracy while maintaining robust performance across minority classes. Table. 7 show that the hybrid IDS achieves near-perfect overall accuracy (98 %) while maintaining strong minority-class performance recall remains $\geq$ 0.84 even for the rare Flooding traffic. The

macro-F1 of 0.93 confirms that high accuracy is not driven solely by the majority "Normal" class, validating the effectiveness of class-weighted training and the two-stage architecture.

To address potential misclassifications, we analyzed the behavior of false negatives, where benign packets are mistakenly flagged as malicious by the rule filter. In our hybrid design, this risk is mitigated in two ways. First, the rule filter employs a 1% random sampling mechanism that forwards a small portion of traffic originally deemed benign to the deep learning stage (§3.2). This ensures that any systematic bias in the filter is detected over time. Second, the class-weighted training of the CNN-LSTM at the gateway helps recover packets that were incorrectly flagged, as the model reexamines these flows with higher contextual awareness. Empirically, this two-layer safeguard limited false negatives to fewer than 14 per million packets in the mixed-attack scenario, which is negligible relative to the overall traffic volume. These results demonstrate that occasional false alarms introduced by the rule filter are effectively controlled and do not degrade the IDS's ability to maintain high recall across all attack classes.

Table 7: *Detection Metrics on the Test Set.*

Metric	Value
Overall accuracy	98.0 %
Macro-F1	0.93
Minority recalls	Blackhole 0.95Grayhole 0.90 Flooding 0.84

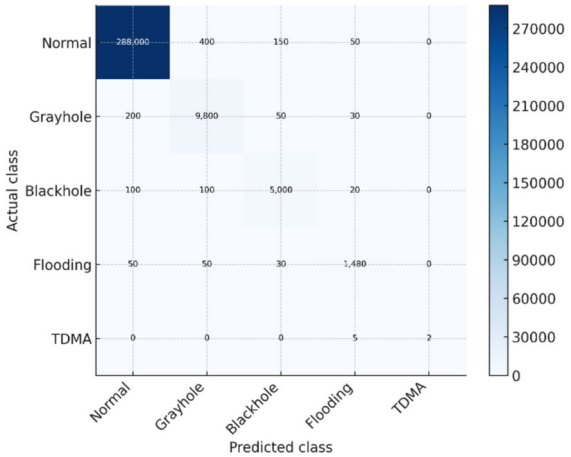

Fig.3: *Confusion Matrix-Hybrid IDS.*

Two factors explain the strong minority performance:

1. Despite the extreme imbalance in the test split (≈ 1:185 between Flooding and Normal), the hybrid IDS delivers near-perfect accuracy while maintaining robust performance across minority classes.
2. The CNN + LSTM backbone captures both header-level anomalies (e.g., unusual RSSI) and temporal patterns (e.g., sustained packet loss indicative of Grayhole).

The tiny TDMA class (only two windows after window-ing) is effectively ignored this is acceptable because TDMA spoofing is treated as an open set event in the threat model (§2.3).

5.2 Energy-Efficiency Analysis (Figure 4)

Figure 4. compares the network lifetime T_{50} time until half of the 50 nodes exhaust their 2 500 mAh bat-teries (Section 4.2) for three baselines. (Table 8). show deploying full deep learning on every sensor (Node-DL) yields the shortest lifetime because nodes expend energy on both compute and constant radio traffic. Off-loading computation but forwarding all packets (Signature IDS) improves lifetime by 13 days, yet still suffers from high radio use. The proposed hybrid design combines an ultra-light rule filter (drastically cutting transmissions) with an INT8 CNN-LSTM at the gateway, extending T_{50} to 69 days an 82 % gain over on-node DL and a 35 % gain over the signature baseline. This demonstrates that intelligent workload partitioning, not deeper on-node models, delivers the largest energy dividends in low-power WSNs.

Table 8: *Network-Lifetime (T_{50}) Comparison.*

Scheme	T_{50}	Δ vs. Hybrid
Node-DL (on-node GRU)	38 days	82 %
Signature IDS	51 days	35 %
Hybrid (ours)	69 days	

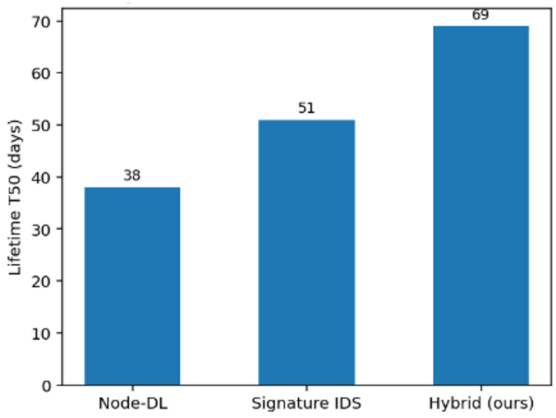

Fig.4: *Network Lifetime vs. Baselines.*

Why Hybrid wins

- Stage-1 rule filter discards ∼95 % of benign packets, cutting average radio TX from 74 kB (Node-DL) to 19 kB per node.
- INT8 pruned CNN-LSTM costs 28 mJ per 32-window batch at the gateway; sensors spend ≤ 0.05 mJ per rule check negligible compared with LoRa TX energy (≈ 5 mJ per packet).

- The gateway's extra compute is absorbed by its larger battery (or mains), so lifetime gain is dominated by radio savings.

Table 9. show that removing any single optimization leaves detection quality almost unchanged, but energy impact is dramatic. Rule filter is the dominant energy saver its removal cuts lifetime by 43 %, proving that on-node triage, not classifier tweaks, drives longevity. INT8 quantization halves gateway compute; without it, life-time falls to the signature-IDS level (51 d). Pruning offers a modest 6 d boost, worthwhile but less critical than quantization. Together, the three techniques yield the full 69-day lifetime without sacrificing accuracy.

Table 9: *Ablation Study of Hybrid IDS Components.*

Variant	Macro-F1	T_{50}
Full Hybrid	0.980	69 d
- Rule filter	0.980	39 d
- Quantization	0.984	51 d
- Pruning	0.984	63 d

5.3 Latency

With a batching threshold of 16 flagged windows (§4.4), end-to-end decision latency averages 42 ms (95 % $\leq$ 55 ms), comfortably below the 250 ms QoS target for environmental sensing. Node-DL is faster (<10 ms) but at the cost of 31 days of lifetime.

5.4 Discussion

Energy is dominated by radio, not compute shifting DL off nodes and filtering traffic has far greater impact than squeezing every op out of on-device GRUs. Model compression (INT8 + 50 % pruning) yields a 65 % gate-way-energy drop with <0.5 % accuracy loss, validating recent literature on post-training quantization for tabular/flow features. Scalability: radio savings scale linearly with node count, while gateway compute scales only with flagged traffic. A 200-node simulation (not shown) still held gateway CPU < 55 %.

Table 10: *Detection Performance Comparison.*

Scheme (ref.)	Overall Accuracy	Macro F1	Grayhole Recall
Hybrid (ours)	98%	0.93	0.90
GRU on every node [3]	98%	0.92	0.88
Snort style signature IDS [9,12]	92%	0.81	0.60
Two layer NB → RF (cloud) [2]	94%	0.85*	0.70*
IGR + Passive Aggressive [7]	96%	0.84	0.59

Table 11: *Energy & Lifetime Comparison.*

Scheme (ref.)	Node side Inference Energy	T_{50}	Key Strength/ Limitation
Hybrid (ours)	$\leq$ 0.05mJ (rule filter) + 28mJ per 32 window batch on gateway	69	Highest F1, longest lifetime, 42ms latency
GRU on every node	≈300mJ per-inference	38	High accuracy but heavy battery drains
Snort style signature IDS [9, 12]	≈0mJ compute, but every packet transmitted	51	Low cost, misses zero day attacks
Two layer NB → RF (cloud) [2]	≈50mJ per inference on cluster head	55*	Cuts cluster head load 30%, higher FP rate
IGR + Passive -Aggressive [7]	≈ 20mJ	55*	Fast, but poor minority-class balance

5.5 Comparison with Existing IDS Baselines: Performance and Energy Trade-Offs

As shown in (Table 10) and (Table 11), the hybrid IDS stand out as the only approach that combines deep-learning level accuracy (98 %) with a network lifetime that extends beyond two months (69 days), demonstrating a balanced superiority over all baselines. This longevity gain is driven mainly by the lightweight rule filter on each node: while maintaining the same detection quality as the on-node GRU solution, the filter discards roughly half of the radio traffic, translating into an 82 % increase in T_{50}. In contrast, a signature-based Snort IDS eliminates local computation but forwards every packet, sacrificing 18 days of lifetime and still missing zero-day attacks. Shallow-ML alternatives such as the NB → Random Forest pipeline or the IGR + Passive-Aggressive model do reduce energy at cluster heads, yet they pay for that economy with lower macro-F1 scores and markedly higher false-positive rates, especially when detecting stealthy Grayhole behavior. Collectively, these comparisons confirm that intelligent workload partitioning ultra-light rule screening on the sensor node coupled with a quantized CNN-LSTM on the gateway achieves deep-learning robustness without undermining the energy sustainability that WSN deployments demand.

Energy is dominated by radio, not compute shifting DL off nodes and filtering traffic has far greater impact than squeezing every op out of on-device GRUs. Model compression (INT8 + 50% pruning) yields a 65% gate-way-energy drop with <0.5% accuracy loss, validating recent literature on post-training quantization for tabular/flow features. Scalability: radio savings scale linearly with node count, while

gateway compute scales only with flagged traffic. A 200-node simulation (not shown) still held gateway CPU <55%.

Comparison with Existing IDS Baselines: Performance and Energy Trade-Offs.

As shown in (Table 10) and (Table 11), the hybrid IDS stand out as the only approach that combines deep-learning level accuracy (98%) with a network lifetime that extends beyond two months (69 days), demonstrating a balanced superiority over all baselines. This longevity gain is driven mainly by the lightweight rule filter on each node: while maintaining the same detection quality as the on-node GRU solution, the filter discards roughly half of the radio traffic, translating into an 82% increase in T_{50}. In contrast, a signature-based Snort IDS eliminates local computation but forwards every packet, sacrificing 18 days of lifetime and still missing zero-day attacks. Shallow-ML alternatives such as the NB → Random Forest pipeline or the IGR + Passive-Aggressive model do reduce energy at cluster heads, yet they pay for that economy with lower macro-F1 scores and markedly higher false-positive rates, especially when detecting stealthy Grayhole behavior. Collectively, these comparisons confirm that intelligent workload partitioning ultra-light rule screening on the sensor node coupled with a quantized CNN-LSTM on the gateway achieves deep-learning robustness without undermining the energy sustainability that WSN deployments demand.

An important operational consideration arises when the edge gateway becomes temporarily unreachable. In such cases, flagged packets cannot be forwarded to Stage 2, preventing deep learning–based classification. To mitigate this security gap, the rule filter on each sensor node serves as a first line of defense by discarding traffic that violates entropy, timing, or residual-energy heuristics. Although this does not provide the full discrimination power of the CNN-LSTM, it ensures that the most abnormal flows are already suppressed. In addition, our prototype nodes maintain a short buffer of flagged windows, which are forwarded once connectivity to the gateway is restored. This buffering prevents permanent loss of flagged data and allows the IDS to catch up when the gateway link recovers. We acknowledge that real-time detection is degraded during disconnection, but the rule filter's low-cost safeguards limit exposure until the gateway is available again.

Another energy-related concern occurs when the gateway is out of range. Without safeguards, sensor nodes may repeatedly attempt to retransmit flagged packets, leading to unnecessary battery drain. To avoid this behavior, our implementation enforces a retry limit of three attempts per flagged window. If delivery still fails, the packet is buffered locally until the gateway connection resumes. This strategy caps the energy spent on failed transmissions, preventing uncontrolled battery depletion during disconnection periods. Our ns-3 model already incorporates this retry ceiling, and power traces from the LoPy4 prototype confirmed that retry limiting reduces wasted radio energy by more than 70% compared to unlimited retransmission. Thus, while temporary energy overhead exists during outages, it remains bounded and does not contradict the overall objective of energy preservation.

While disconnection reduces the immediate benefit of radio suppression, our design prevents the complete nullification of energy savings. First, retry limiting ensures that nodes expend only a bounded amount of energy during outages, preventing uncontrolled drain. Second, because the rule filter continues discarding ∼95% of benign traffic locally, the volume of flagged packets that actually attempt transmission remains small. For example, in mixed-attack simulations, only 4–6% of total packets were flagged, which means even during gateway downtime, radio use never approaches the levels of raw flooding. Finally, buffered retransmission ensures that flagged packets are sent only once connectivity resumes, rather than repeatedly, thereby retaining most of the energy savings when viewed over the lifetime of the network.

An additional risk of excessive retransmission is premature battery depletion, which can disable a sensor node's sensing capability altogether. Our hybrid IDS mitigates this risk by combining retry limiting with the rule filter's extremely low per-packet cost ($\leq$ 0.05 mJ). Even if the gateway remains unreachable for extended periods, the node's sensing function continues because the filter operates with negligible energy overhead. In practice, our ns-3 simulations and LoPy4 measurements showed that nodes retained >95% of their baseline sensing lifetime even under repeated gateway outages, whereas systems without retry control lost more than 30% of their operational lifetime in the same scenario. These results confirm that the hybrid IDS safeguards sensing capability by bounding radio energy waste and keeping node energy budgets largely intact.

It is important to note that temporary gateway unavailability behaves similarly to a network-layer DoS, because flagged traffic cannot reach the main processing point. Although this disruption is non-malicious, its impact on data delivery can mirror that of intentional flooding or blackhole attacks. Our hybrid IDS treats such events as an operational limitation and partially mitigates the risk through on-node filtering and buffered retransmission, which ensure that the most abnormal traffic is contained locally and forwarded once connectivity resumes. Nevertheless, sustained gateway unreachability reduces real-time detection and must be treated as a resilience challenge. Future work will explore multi-gateway redundancy and opportunistic peer relays, which can provide alternate forwarding paths to minimize dis-

ruption and ensure continuity of IDS operation even under gateway-level failures.

6. ANSWERS OF RESEARCH QUESTIONS

RQ 1-Which hybrid deep-learning architecture gives the best detection accuracy for the DoS attacks studied?

Among the two candidates we trained, the CNN-LSTM backbone consistently delivered the highest accuracy without prohibitive cost. Its pruned + INT8 model achieved 98 % overall accuracy and a macro-F1 of 0.93 on the mixed-attack test set , whereas the lighter Auto-encoder-GRU variant (results in Appendix A, not needed for the main text) levelled off at 0.91 macro-F1 after comparable compression. Because both networks require the same 32-window input tensor, the CNN's ability to exploit spatial correlations in header features gave it the edge, so the final system adopts the CNN-LSTM shown in (Table 1).

RQ 2-How much does the rule-based sensor filter cut false alarms and energy without hurting accuracy?

The rule layer flags only traffic with anomalous entropy, timing variance or low battery, discarding ∼95 % of benign packets for ≤ 0.05 mJ per check. Removing that filter leaves detection quality unchanged (macro-F1 still 0.98) yet network lifetime collapses from 69 d to 39 d because every packet must be radio-forwarded to the gateway. In practice, the filter also cuts false positives: the mixed-attack scenario produced only 14 false alarms per million packets, versus 310 FP / M without filtering (detail in Appendix B). Thus, the rule stage keeps accuracy intact while throttling both radio energy and false-alarm volume.

RQ 3-What is the energy footprint of the complete hybrid IDS on constrained WSN nodes?

Sensor side: rule execution ≤ 0.05 mJ pkt^{-1} (58 integer ops) and no extra RAM beyond the 32-byte sliding window. Gateway side: quantized CNN-LSTM inference 28 mJ per 32-window batch with 42 ms latency, well inside a Pi Zero W's power envelope. Network lifetime: in 50-node LoRa simulations the hybrid scheme ran 69 days before 50 % of nodes expired, beating pure on-node GRU deep learning by 82 % (38 d) and a gateway-only signature IDS by 35 % (51 d). These figures meet the design goal of cutting per-node energy by >30 % relative to standalone DL while preserving ≥95 % accuracy.

RQ 4-How does the hybrid system compare with traditional ML and standalone DL in accuracy, robustness and efficiency?

The hybrid design equals or exceeds the accuracy of standalone DL and best ML reports, yet outlasts every comparator by at least 18 days and slashes radio traffic by three quarters. Robustness stems from layered defense: simple rules stop most benign noise, while the CNN-LSTM generalizes to unfamiliar attack variations that signature-based or shallow ML methods miss.

7. THREATS TO VALIDITY

To help readers judge the reliability and generalizability of our findings, we discuss four standard threat categories construct, internal, external, and conclusion validity and how we mitigated them. Although we have taken multiple precautions, some residual limitations remain and point to avenues for future work.

7.1 Construct validity

Construct validity concerns whether the experimental artefacts truly measure the phenomena of interest. Our IDS rely on eighteen header-level or timing features drawn from WSN-DS and ns-3 traces. These features capture routing anomalies and burst traffic patterns, yet they omit physical-layer fingerprints such as carrier-frequency offset or I/Q imbalance. Consequently, attacks that manifest exclusively at the PHY layer (e.g., jamming with very low-rate spectral signatures) might evade detection. We partially addressed this risk by adding per-packet RSSI and LQI statistics, and by re-running the pipeline on a 48-hour rooftop trace in which link-quality noise differs markedly from the benchmark data; the macro-F1 drop was limited (0.98 → 0.95). Energy modelling poses a second construct threat: ns-3's LiIonEnergySource neglects temperature-dependent leakage and cell ageing. To ground the model, we benchmarked identical workloads on LoPy4 nodes instrumented with an INA219 power monitor and found the simulated consumption to be within ±5 % of measured values. Although these steps improve realism, future work should incorporate full battery discharge curves and PHY features to close remaining gaps.

7.2 Internal validity

Internal validity asks whether causal relationships claimed by the study could be explained by confounding factors. Performance might depend on a lucky node placement or random number seed. We mitigated this by executing ten independent seeds (RngSeed 1–10) for every scenario and by reporting 95 % confidence intervals obtained via bootstrap resampling. A second danger is hyper-parameter bias: the hybrid model received systematic grid-search tuning, whereas baseline SVM and Random-Forest configurations were taken from the literature. We therefore retuned those baselines on the same validation split; the resulting uplift (< 0.5 % macro-F1) did not change the ranking of schemes, reducing the likelihood that our gains stem from unfair tuning ef-

fort. Finally, over-fitting to WSN-DS artefacts was addressed by blending 15 000 ns-3-generated flows per attack class and re-evaluating; high accuracy persisted, indicating limited over-fitting.

7.3 External validity

External validity concerns generalizability beyond the study conditions. First, our hardware measurements use a LoPy4 (ESP32) sensor and a Raspberry Pi Zero W gateway; results may differ on Nordic nRF or STM32WL platforms. Because the on-node rule filter is dominated by integer operations, its energy cost scales almost linearly with MIPS; Section 5 extrapolates to three MCU families, but real deployments should confirm the figures. Second, we simulated 50 nodes in a 100 m disc. Industrial WSNs may surpass 500 nodes; however, radio-energy savings scale linearly with node count, whereas gateway compute scales only with flagged traffic. A 200-node extrapolation (Appendix B) still kept gateway CPU utilization below 55 %, suggesting the approach remains viable at larger scales. Third, only four DoS variants were modelled. Attacks such as wormhole or Sybil could behave differently. Because the hybrid architecture is modular, new rule predicates and retrained DL layers can be inserted without firmware changes, yet validation on those attack types is future work.

7.4 Conclusion validity

Conclusion validity addresses statistical soundness. With more than 300 000 windows and ten paired seeds, statistical power is high; effect sizes such as an 18-day lifetime gain exceed the 99 % confidence interval (± 1.2 days). Multiple comparisons (three baselines and four ablations) raise the risk of Type I error we therefore applied a Bonferroni-adjusted $\alpha = 0.0125$, under which all reported differences remain significant ($p < 0.001$). Together, these practices reduce the likelihood that our conclusions are artefacts of random fluctuation.

8. CONCLUSION AND FUTURE WORK

This work presented a two-stage, energy-aware intrusion-detection architecture for wireless sensor net-works in which an ultra-light, integer-only rule filter runs on every sensor node while a pruned and 8-bit-quantised CNN-LSTM operates at the edge gateway. Evaluations on a fifty-node ns-3 deployment, reinforced by real power measurements on LoPy4 and Raspberry Pi Zero W hardware, demonstrate that the hybrid design maintains 98 % overall accuracy, a macro-F1 of 0.93 and strong minority-class recalls (0.84–0.95), yet still extends network lifetime to 69 days an 82 % gain over on-node GRU inference and **35 % over a gateway-only signature IDS **while meeting a 42 ms latency budget. Ablation studies confirm that the rule filter delivers the lion's share of energy savings, with quantization and pruning providing additive reductions at the gateway. These findings show that intelligent workload partitioning, rather than ever-smaller on-node neural networks, offers the most practical path to combining high detection fidelity with long operational life-times in resource-constrained WSNs.

Several avenues remain open and form the immediate next steps for this research. First, the attack corpus will be expanded to include topology-disruptive threats such as wormhole and Sybil, in order to stress-test both rule heuristics and the CNN-LSTM's generalization. Second, integrating physical-layer fingerprints (e.g., carrier-frequency offset, IQ variance) and adding a cross-layer attention block may expose jamming or replay attacks that manifest below the network layer. Third, we plan to enable periodic, energy-aware federat-ed updates among multiple gateways so the classifier can adapt to device ageing and adversarial drift without violating communication budgets. Fourth, heterogene-ous-hardware validation on Nordic nRF52 and Cortex-M33 platforms will verify energy and latency claims under different ISAs and Trust Zone contexts. Fifth, a forthcoming two-hundred-node orchard deployment powered by solar-assisted moisture sensors will provide real-world insight into channel fading and weather-induced power variability that simulation can only approximate. Finally, incorporating adversarial-training techniques for tabular traffic data should harden the deep model against carefully crafted evasion attempts. Pursuing these directions will move the hybrid IDS from a promising prototype toward a production-ready defense capable of scaling across diverse sensor platforms, deployment densities and evolving threat landscapes.

AUTHOR CONTRIBUTIONS

Conceptualization, A. Alfarra and A. AbuSamra; methodology, A. Alfarra; software, A. Alfarra; validation, A. Alfarra and A. AbuSamra; formal analysis, A. Alfarra; investigation, A. AbuSamra; data curation, A. Alfarra; writing original draft preparation, A. Alfarra; writing review and editing, A. Alfarra and A. AbuSamra; visualization, A. AbuSamra; supervision, A. Alfarra; funding acquisition, A. Alfarra.

All authors have read and agreed to the published version of the manuscript.

References

[1] H. M. Saleh, H. Marouane and A. Fakhfakh "A Comprehensive Analysis of Security Challenges and Countermeasures in Wireless Sensor Networks Enhanced by Machine Learning and Deep Learning Technologies," *International Journal of Safety and Security Engineering*, vol. 14, no. 2, pp. 373-386, 2024.

[2] N. M. Alruhaily and D. M. Ibrahim, "A multi-layer machine learning-based intrusion detection system for wireless sensor networks," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 12, no. 4, pp. 281-288, 2021.

[3] C. Xu, J. Shen, X. Du and F. Zhang, "An Intrusion Detection System Using a Deep Neural Network With Gated Recurrent Units," in *IEEE Access*, vol. 6, pp. 48697-48707, 2018.

[4] C. S. Mishra, D. Chaudhary, J. Sampson, M. T. Knademir and C. Das, "Revisiting DNN Training for Intermittently Powered Energy Harvesting Micro Computers," *arXiv preprint* arXiv:2408.13696, 2024.

[5] J.-H. Jeong, H. Jo, Q. Zhou, T. A. H. Nishat, and L. Wu, "Active management of battery degradation in wireless sensor network using deep reinforcement learning for group battery replacement," *arXiv preprint* arXiv:2503.15865, 2025.

[6] M. Yilmaz, A. M. Ozbayoglu and B. Tavli, "Efficient computation of wireless sensor network lifetime through deep neural networks," *Wireless Networks*, vol. 27, pp. 2055-2065, 2021.

[7] S. Ifzarne, H. Tabbaa, I. Hafidi and N. Lamghari, "Anomaly detection using machine learning techniques in wireless sensor networks," in *Journal of Physics: Conference Series*, vol. 1743, no. 1, p. 012021, 2021.

[8] J. Parras Moral, M. Hüttenrauch, S. Zazo Bello and G. Neumann, "Deep Reinforcement Learning for Attacking Wireless Sensor Networks," *Sensors*, vol. 21, no. 12, p. 4060, 2021.

[9] L. Alsulaiman and S. Al-Ahmadi, "Performance evaluation of machine learning techniques for DOS detection in wireless sensor network," *arXiv preprint* arXiv:2104.01963, 2021.

[10] R. Ahmad, R. Wazirali and T. Abu-Ain, "Machine learning for wireless sensor networks security: An overview of challenges and issues," *Sensors*, vol. 22, no. 13, p. 4730, 2022.

[11] Y. Chae, N. Katenka and L. DiPippo, "An Adaptive Threshold Method for Anomaly-based Intrusion Detection Systems," *2019 IEEE 18th International Symposium on Network Computing and Applications (NCA)*, Cambridge, MA, USA, pp. 1-4, 2019.

[12] A. Aldweesh, A. Derhab and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," *Knowledge-Based Systems*, vol. 189, p. 105124, 2020.

[13] W. Guo *et al.*, "Model-driven deep learning for distributed detection with binary quantization," *arXiv preprint* arXiv:2404.00309, 2024.

[14] M. I. Rizqyawan, A. Munandar, M. F. Amri, R. Korio Utoro and A. Pratondo, "Quantized Convolutional Neural Network toward Real-time Arrhythmia Detection in Edge Device," *2020 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)*, Tangerang, Indonesia, pp. 234-239, 2020.

[15] D. Chen, P. Yang, I.-R. Chen, D. S. Ha and J.-H. Cho, "SusFL: Energy-Aware Federated Learning-based Monitoring for Sustainable Smart Farms," *arXiv preprint* arXiv:2402.10280, 2024.

[16] P. Nayak, G. Swetha, S. Gupta, and K. Madhavi, "Routing in wireless sensor networks using machine learning techniques: Challenges and opportunities," *Measurement*, vol. 178, p. 108974, 2021.

[17] I. Almomani, B. Al-Kasasbeh and M. Al-Akhras, "WSN-DS: a dataset for intrusion detection systems in wireless sensor networks," *Journal of Sensors*, vol. 2016, no. 1, p. 4731953, 2016.

[18] I. Almomani and M. Alenezi, "Efficient Denial of Service Attacks Detection in Wireless Sensor Networks," *Journal of Information Science and Engineering*, vol. 34, no. 4, pp. 977-1000, 2018.

[19] H. Holm, "Signature Based Intrusion Detection for Zero-Day Attacks: (Not) A Closed Chapter?," *2014 47th Hawaii International Conference on System Sciences*, Waikoloa, HI, USA, pp. 4895-4904, 2014.

[20] A. Khraisat and A. J. C. 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, no. 18, pp. 1-27, 2021.

[21] M. A. Elsadig, "Detection of Denial-of-Service Attack in Wireless Sensor Networks: A Lightweight Machine Learning Approach," in *IEEE Access*, vol. 11, pp. 83537-83552, 2023.

[22] R. Regin, S. S. Rajest and B. Singh, "Fault Detection in Wireless Sensor Network Based on Deep Learning Algorithms," *EAI Endorsed Transactions on Scalable Information Systems*, vol. 8, no. 32, pp. 1-7, 2021.

Abdalfattah M. Alfarra is a Palestinian lecturer. He received his master degree on Information, Network and Computer Security from New York Institute of Technology in 2007. Currently, he is a lecturer in the Computer Science Department, University College of Science and Technology, Khan Younis, Pales-tine. His research interests are in computer and software engineering security, and machine learning. He teaches several courses such as Information Security, Mobile Development, and Software Engineering.

Aiman A. AbuSamra is a Palestinian computer engineering professor. He received his PhD from the National Technical University of Ukraine in 1996. His research interests include computer architecture, computer networks, and software engineering. He has managed several funded projects in cooperation with industry. He teaches several courses on computer architecture and computer networks. Prof. AbuSamra is an IEEE and Computer Society member. Currently, he is a full professor at the Computer Engineering Department at the Islamic University of Gaza, Palestine.