ISA Journal of Engineering and Technology (ISAJET)

Homepage: https://isapublisher.com/isajet/

Volume 3, Issue 5, Sept-Oct, 2026 ISSN: 3049-1843

Value-Aware RF-Enabled TinyML Embedded Architecture for Intelligent Environmental Monitoring in Resource-Constrained Environments: A Computational Experimental Evaluation

Nnaji Kelechi Timothy[1], Oluwatomisin Isogun[2], Chi Mac-Braine Wanisi[3], Edeh Chukwuemeka Dominic[4]

[1]Department of Electronics and Computer Engineering, Nnamdi Azikiwe University, Awka, Nigeria
[2]Department of Computer Science, Olusegun Agagu University of Science and Technology, Nigeria
[3]Department of Computer Engineering, College of Technology, University of Bamenda, Cameroon;
[4]Department of Mechatronics Engineering, Federal University Otuoke, Bayelsa State, Nigeria;

Received: *10.08.2026* **|** ***Accepted:*** *02.10.2026* **|** ***Published:*** *13.10.2026*

***Corresponding Author:** Nnaji Kelechi Timothy

DOI: *10.5281/zenodo.23337367*

Abstract **Original Research Article**

Resource-constrained environmental sensing systems must simultaneously satisfy conflicting requirements for predictive intelligence, communication reliability, memory efficiency, and energy conservation. Although Tiny Machine Learning (TinyML) enables local inference on microcontroller-class devices, existing environmental monitoring systems frequently treat environmental inference, transmission scheduling, and radio-frequency resource allocation as separate optimization problems. This study proposes a value-aware RF-enabled TinyML architecture in which environmental severity, prediction uncertainty, temporal novelty, information freshness, and RF-link state jointly determine whether information is transmitted and the communication resources assigned to its delivery. A computational experimental evaluation was performed using a real environmental time series containing 2,231 usable temporal sequences. Four candidate models Decision Tree, Random Forest, multilayer perceptron, and compact one-dimensional convolutional neural network were evaluated using chronological training, validation, and independent test partitions. The Random Forest achieved the highest test performance, with 89.85% accuracy, 88.08% macro-F1, and 91.18% recall for the highest-severity class. A Value-of-Information controller subsequently combined model severity, predictive entropy, temporal novelty, and information age. Four communication strategies were compared: periodic fixed-RF monitoring, periodic TinyML telemetry, event-triggered TinyML telemetry, and the proposed value-aware adaptive RF architecture. The proposed architecture reduced transmission count by 72.54% and transmitted bytes by 74.22% relative to periodic monitoring. Across 300 modeled RF-link realizations, mean packet-delivery ratio increased from 70.78% for conventional periodic telemetry to 94.44%, while mean critical-event delivery increased from 76.41% to 87.04%. Mean modeled communication energy decreased by 32.13% relative to periodic transmission, although a simpler event-triggered architecture consumed less energy at the cost of substantially poorer reliability. The results demonstrate that communication efficiency in intelligent environmental sensing should be defined not only by energy minimization but by the reliable delivery of high-value environmental information. Physical microcontroller, LoRa, and power measurements remain necessary for final hardware validation.

Keywords: TinyML; environmental monitoring; edge intelligence; LoRa; adaptive RF telemetry; Value of Information; embedded machine learning; resource-constrained IoT; low-power sensing; Internet of Things.

Copyright © 2026 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

1. Introduction

Environmental monitoring is fundamental to applications ranging from air-quality observation and climate sensing to smart agriculture, industrial safety, water-quality assessment, and ecosystem management. The proliferation of Internet of Things (IoT) technologies has made it possible to deploy geographically distributed sensing nodes at substantially lower cost than conventional centralized monitoring infrastructure. However, the scalability of such deployments remains constrained by energy availability, wireless communication overhead, network coverage, computational limitations, and the need for rapid local decision-making.

Conventional IoT environmental monitoring architectures frequently transmit raw or lightly processed sensor measurements to remote gateways or cloud servers. Such an architecture provides substantial computing capacity at the destination, but continuous communication may impose a disproportionate energy burden on battery-powered or energy-harvesting nodes. It also introduces dependence on communication availability and increases network traffic. These limitations are especially significant in remote, rural, mobile, or infrastructure-constrained environments.

Tiny Machine Learning has emerged as an important approach to addressing these limitations by enabling trained machine-learning models to execute directly on resource-constrained microcontrollers. TinyML seeks to move inference from conventional cloud or high-performance edge platforms to ultra-low-power endpoint devices while respecting severe constraints in Flash memory, SRAM, execution latency, and energy [1], [2]. Frameworks such as TensorFlow Lite Micro have further reduced deployment barriers by providing lightweight inference runtimes for embedded systems [3]. Hardware-software co-design approaches such as MCUNet have demonstrated that neural inference can be substantially optimized for microcontroller memory and latency constraints [4]. Similarly, MLPerf Tiny has established the importance of evaluating TinyML systems using accuracy, latency, and energy jointly rather than considering predictive accuracy in isolation [5].

TinyML has increasingly been applied to environmental sensing. Pereira et al. deployed an embedded water-potability classifier on an ESP32 and demonstrated reductions in memory, energy, and inference time compared with a cloud-oriented solution [6]. Adebayo et al. integrated TinyML with multiple environmental sensors for mushroom-farm monitoring and control [7]. Wiese et al. developed a multimodal environmental IoT node containing eleven sensing modalities and demonstrated energy savings by processing environmental information locally rather than continuously streaming raw data [8]. More recently, Reis proposed a low-power environmental node combining onboard machine-learning inference, adaptive duty cycling, and event-driven LoRaWAN communication [9]. In the Nigerian context, Akanji, Francis, and Akintola [22] analyzed PM2.5, PM10, CO, NO2, SO2, and O3 trends across Abuja, Lagos, Kano, Port Harcourt, and Enugu using satellite-based observations and time-series methods, highlighting pronounced seasonal and city-level variability in urban air pollution.

These studies demonstrate that the simple combination of environmental sensors and TinyML is no longer sufficient to constitute a strong standalone research contribution. Current research is moving toward system-level co-design. Trînc et al. developed and field-tested a solar-powered environmental monitoring platform integrating edge processing with LTE-M communication [10]. Sabovic et al. proposed energy-aware dynamic selection among multiple TinyML models according to available energy and resource constraints [11]. Alanazi et al. introduced a cross-layer TinyML optimization approach for ARM Cortex-M microcontrollers that considered model compression, memory scheduling, and runtime power management [12]. Uko et al. further investigated adaptive energy harvesting and TinyML scheduling for autonomous sensing platforms [13].

An analogous evolution is occurring at the wireless communication layer. LoRa and LoRaWAN have become attractive for low-power, long-range

environmental sensing because their spreading-factor and transmit-power configurations provide multiple reliability-energy trade-offs. Arratia et al. introduced AlLoRa to improve environmental-information transport across long-range LoRa networks [14]. AI*LoRa subsequently demonstrated that machine learning can support dynamic RF configuration at the edge [15]. Lodhi et al. developed adaptive data-rate strategies based on channel-quality estimation to improve LoRaWAN reliability and energy efficiency [16]. More recently, EdgeSense combined TinyML and temporal learning to estimate signal-to-noise ratio and adapt spreading factor and transmission power [17].

Despite these advances, environmental intelligence and communication intelligence are still commonly optimized as separate processes. Environmental inference determines what condition exists; RF adaptation determines how information should be transmitted. These two decisions are not necessarily independent. A routine measurement representing a stable environmental condition should not necessarily consume the same communication resources as a rapidly changing or potentially critical observation. Conversely, a highly important environmental event may justify additional RF energy to increase delivery reliability.

Value-of-Information (VoI) approaches offer a useful theoretical framework for this problem. VoI explicitly estimates the usefulness of information relative to a specific operational objective and enables constrained sensing systems to prioritize information according to its value rather than merely its existence [18].

The principal research gap addressed in this study is therefore the absence of an explicit closed-loop mechanism linking TinyML-derived environmental information value with adaptive RF-resource expenditure. Instead of asking only whether an event has occurred or whether a link is weak, the proposed system asks whether the current information is sufficiently valuable to justify transmission and, if so, what reliability and payload resources should be allocated to it.

The principal contributions of this work are:

- Development of a value-aware TinyML environmental-monitoring architecture integrating local environmental inference, information-value estimation, and adaptive RF telemetry.
- Formulation of an Environmental Value-of-Information score combining predicted severity, machine-learning uncertainty, temporal novelty, and information freshness.
- Development of a cross-layer controller that determines transmission necessity, payload size, spreading factor, and transmit power using environmental information value and RF-link condition.
- Comparative evaluation against periodic, TinyML-periodic, and conventional event-triggered telemetry baselines.
- An ablation study identifying the influence of severity, uncertainty, novelty, information age, and RF-link adaptation.
- Monte Carlo evaluation across 300 modeled RF-link realizations.
- Explicit examination of the trade-off between minimum energy expenditure and reliable delivery of environmentally important information.

The remainder of the paper is organized as follows. Section 2 reviews the relevant literature. Section 3 presents the system model and proposed architecture. Section 4 describes the materials and methods. Section 5 presents the results. Section 6 discusses the findings, limitations, and implications. Section 7 outlines future hardware validation. Section 8 concludes the paper.

2. Related Work

2.1 TinyML on Resource-Constrained Devices

TinyML represents the intersection of embedded systems, low-power computing, and machine learning. Ray [1] identified limited processing capability, memory, energy availability, and deployment tooling as central challenges in TinyML

implementation. Schizas et al. [2] similarly emphasized the importance of lightweight models, efficient inference engines, and low-energy hardware for large-scale TinyML deployments.

TensorFlow Lite Micro was specifically designed to execute machine-learning inference under embedded-system resource constraints [3]. MCUNet subsequently demonstrated that neural-network architecture and inference-engine design can be co-optimized for microcontrollers, reducing memory use and improving execution efficiency [4]. MLPerf Tiny formalized the need to measure accuracy, latency, and energy together when evaluating TinyML implementations [5]. These developments are particularly relevant to environmental monitoring because battery-powered sensing nodes may need to operate unattended for extended periods.

2.2 TinyML for Environmental Monitoring

Pereira et al. [6] demonstrated that a water-potability classifier could be adapted and embedded on an ESP32, showing the feasibility of local environmental classification. Adebayo et al. [7] combined sensors measuring temperature, humidity, CO_2, light, and soil characteristics with TinyML-based local environmental control.

Wiese et al. [8] advanced this concept through a multimodal node integrating eleven environmental sensing modalities and a low-power edge-AI processor. Their results demonstrated that local AI processing could save energy compared with raw-data streaming. Reis [9] extended the concept further by combining quantized environmental inference, adaptive duty cycling, and event-triggered LoRaWAN communication. Akanji, Francis, and Akintola [22] provide an important regional air-quality context for the present study. Using satellite-based remote-sensing observations from 2021 to 2023, they examined PM2.5, PM10, CO, NO2, SO2, and O3 across five major Nigerian cities and applied ARIMA/SARIMA time-series modeling, seasonal decomposition, and correlation analysis. Their findings showed substantial spatial and seasonal variation, including Harmattan-related particulate pollution, reinforcing the need for adaptive, distributed, and resource-efficient monitoring architectures in rapidly urbanizing environments.

The present research differs from simple event-triggered communication by introducing a continuous information-value score that affects both whether information is transmitted and how much RF reliability is allocated to its transmission.

2.3 Energy- and Resource-Aware TinyML

Energy-aware TinyML systems are increasingly moving beyond static model deployment. Sabovic et al. [11] demonstrated dynamic selection among multiple TinyML models according to changing energy availability. Their work showed that smaller models can provide execution advantages while larger models may provide higher prediction accuracy, motivating dynamic rather than fixed model selection.

Alanazi et al. [12] similarly showed that model compression, memory scheduling, and runtime power-management policies should be optimized jointly. Uko et al. [13] integrated adaptive hybrid energy harvesting with reinforcement-learning-based TinyML scheduling for autonomous IoT sensing. These studies support the broader principle that TinyML performance must be treated as a multidimensional resource-allocation problem rather than an accuracy-only problem.

2.4 LoRa and Adaptive RF Communication

LoRa provides configurable spreading factor, bandwidth, coding rate, and transmit power, each influencing time-on-air, energy consumption, range, and packet reliability. AlLoRa was designed to improve content transfer for environmental sensing in challenging long-range settings [14]. AI*LoRa employed machine learning at the edge to select RF configurations dynamically and was evaluated through real-world long-range testing [15]. Lodhi et al. [16] proposed improved adaptive-data-rate strategies using filtered SNR estimates to assign spreading factor and transmission power. EdgeSense [17] further combined TinyML and temporal learning for SNR-aware LoRaWAN adaptation.

These approaches primarily optimize the communication channel. The present study instead adds an environmental-semantic dimension: the desired reliability of a transmission changes according to the value of the information being communicated.

2.5 Value of Information in IoT Systems

VoI provides a mechanism for determining how useful sensed information is relative to a specific application objective. Alawad and Kraemer [18] reviewed VoI methods in wireless sensor networks and IoT and emphasized their potential for selecting which information deserves limited sensing, communication, and energy resources. The proposed study extends this principle to TinyML by making environmental VoI dependent on prediction severity, classifier uncertainty, temporal novelty, and information freshness.

2.6 Research Gap

The literature demonstrates substantial progress in TinyML environmental inference [6]–[10], energy-aware TinyML execution [11]–[13], intelligent LoRa/RF configuration [14]–[17], and VoI-aware resource allocation [18]. However, these areas have largely evolved independently. The present study therefore investigates whether environmental information value can serve as a cross-layer control variable linking local TinyML inference to RF transmission decisions.

3. Proposed System Architecture

3.1 System Overview

The proposed architecture contains six logical components: environmental sensing; preprocessing and temporal feature extraction; TinyML inference; Environmental Value-of-Information estimation; joint resource/RF decision control; and adaptive RF telemetry. The overall architecture is illustrated in Figure 1.

At sampling time t, the environmental observation vector is

$$\mathbf{x}_t = \left[x_{t,1}, x_{t,2}, \dots, x_{t,d}\right]^{\mathsf{T}} \in \mathbb{R}^d \qquad (1)$$

A temporal window of length W is represented as

$$\mathbf{X}_t = [\mathbf{x}_{t-W+1}, \dots, \mathbf{x}_t]^{\mathsf{T}} \in \mathbb{R}^{W \times d} \qquad (2)$$

A TinyML classifier parameterized by θ produces the class-probability vector

$$\mathbf{p}_t = f_\theta(\mathbf{X}_t) = \left[p_{t,1}, \dots, p_{t,K}\right]^{\mathsf{T}}, \quad p_{t,k} \geq 0, \quad \sum_{k=1}^{K} p_{t,k} = 1. \qquad (3)$$

The predicted state is

$$\hat{y}_t = \underset{k \in \{1,\dots,K\}}{\arg\max}\ p_{t,k} \qquad (4)$$

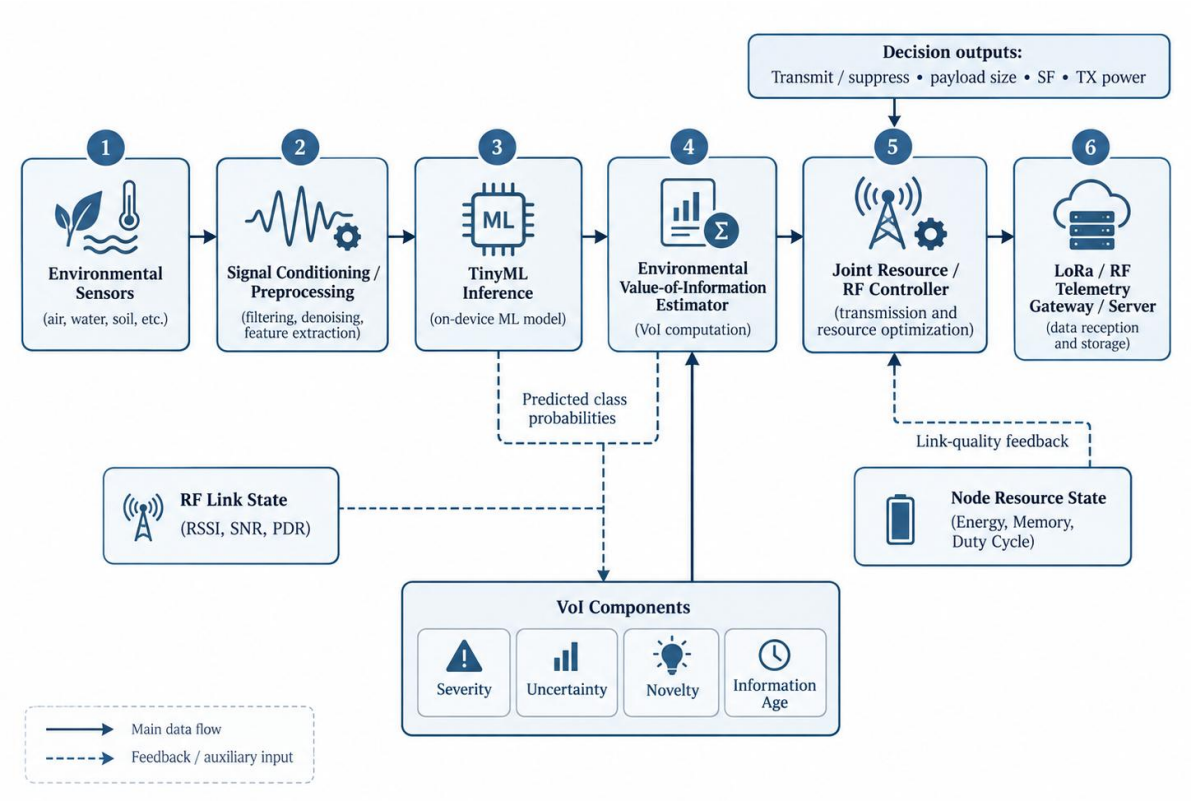


Figure 1. Proposed value-aware RF-enabled TinyML architecture for intelligent environmental monitoring.

3.2 Prediction Uncertainty

Prediction uncertainty was estimated using normalized entropy:

$$U_t = -\frac{1}{\ln K}\sum_{k=1}^{K} p_{t,k}\ln\left(p_{t,k}+\varepsilon\right), \qquad 0 \le U_t \le 1 \tag{5}$$

A low value represents high classifier confidence, while values approaching one indicate ambiguous class probabilities.

3.3 Environmental Severity

Severity was calculated from class probabilities as

$$S_t = \sum_{k=1}^{K} w_k^{(s)}\, p_{t,k}, \qquad 0 \le S_t \le 1 \tag{6}$$

where w_k^(s) denotes the normalized severity weight assigned to class k. For the three-class experiment, the severity-weight vector is

$$\mathbf{w}^{(s)} = [0,\ 0.5,\ 1]^{\mathsf{T}} \tag{7}$$

3.4 Temporal Novelty

Repeated measurements that closely resemble recent observations contain less new information than rapidly changing measurements. Temporal novelty was therefore represented as

$$D_t = \min\left\{1, \frac{|x_t - \bar{x}_{t-r:t-1}|}{\tau_D}\right\} \tag{8}$$

where τ_D denotes the novelty-normalization constant estimated exclusively from the training data.

3.5 Information Age

To prevent long-term communication suppression during stable environmental periods, information freshness was incorporated through

$$A_t = \min\left\{1, \frac{t - t_{\text{last}}}{T_{\max}}\right\} \tag{9}$$

where t_last denotes the time index of the most

recent transmitted update.

3.6 Environmental Value of Information

The complete Environmental Value-of-Information score is

$$V_t = \alpha_s S_t + \alpha_u U_t + \alpha_d D_t + \alpha_a A_t \quad (10)$$

where the weights satisfy

$$\alpha_s + \alpha_u + \alpha_d + \alpha_a = 1, \qquad \alpha_i \geq 0 \quad (11)$$

In the experiment, severity was assigned weight 0.40, while uncertainty, novelty, and information age were each assigned weight 0.20. Thus,

$$V_t = 0.40S_t + 0.20U_t + 0.20D_t + 0.20A_t \quad (12)$$

3.7 Adaptive Telemetry Action

The controller selects an action

$$\mathbf{a}_t = [z_t, \mathrm{SF}_t, P_t^{\mathrm{TX}}, L_t]^{\mathrm{T}} \quad (13)$$

where z_t is the transmit/suppress decision, SF_t is the LoRa spreading factor, P_t^TX is the transmit power, and L_t is the payload length. High-value observations are assigned a higher target delivery reliability than low-value routine information. The underlying design principle is

$$V_t \uparrow \quad \Rightarrow \quad R_t^{\mathrm{req}} \uparrow \quad (14)$$

4. Materials and Methods

4.1 Experimental Design

A computational experimental design was adopted to evaluate both environmental classification and adaptive communication behavior. The study consisted of environmental time-series preprocessing, temporal feature construction, model training, independent model evaluation, VoI-controller calibration, baseline communication comparison, Monte Carlo RF evaluation, and ablation analysis. A fixed random seed of 42 was used where stochastic computational procedures required reproducibility. The overall computational workflow is summarized in Figure 2.

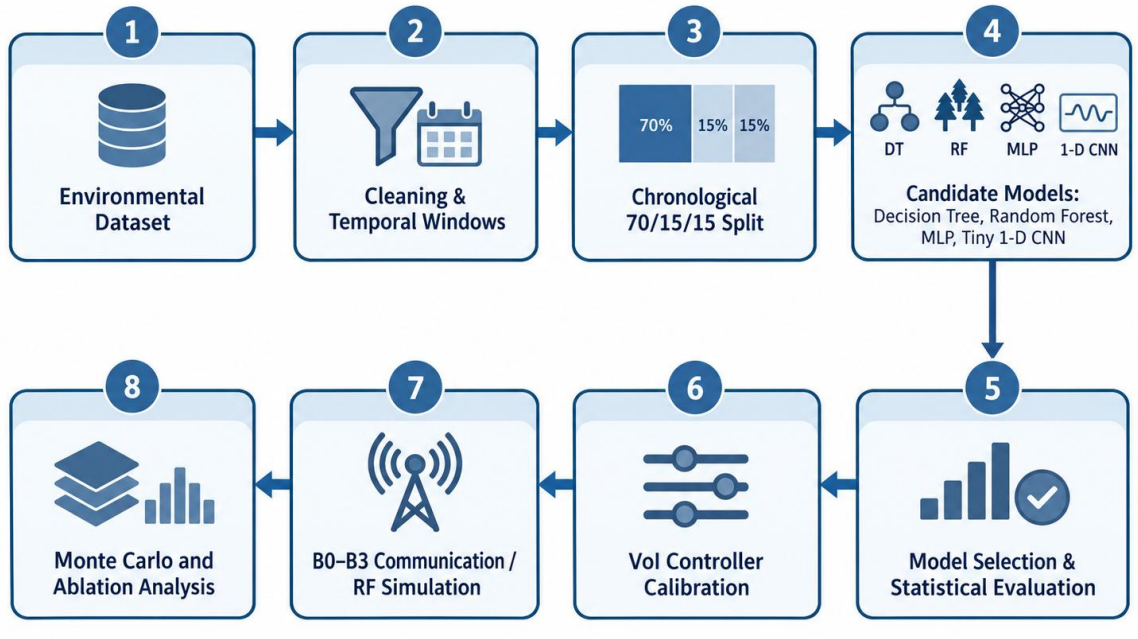


Figure 2. Computational experimental workflow from environmental dataset preprocessing to Monte Carlo and ablation analysis.

4.2 Environmental Dataset

The Phase-1 computational benchmark used a real weekly atmospheric CO_2 time series derived from the long-running Mauna Loa atmospheric record. The Mauna Loa record represents one of the longest direct atmospheric CO_2 measurement series, beginning in 1958 [19]. The computational dataset contained 2,284 weekly calendar positions. Missing values were reconstructed by linear interpolation before sequence construction.

The dataset was used to validate the proposed computational architecture rather than to establish health-related air-quality categories. A second external validation dataset has been identified for subsequent work: the Beijing Multi-Site Air Quality dataset from the UCI Machine Learning Repository. It contains 420,768 hourly measurements from 12 monitoring stations and is distributed under CC BY 4.0 [20].

4.3 Sequence Generation

A temporal context window of W = 24 weeks was used. For each observation, the input representation included 24 historical CO_2 values, 24 first-order differences, current standardized anomaly, sine of week-of-year, and cosine of week-of-year. This yielded 51 features for the conventional machine-learning models. After temporal sequence generation, 2,231 usable observations remained.

4.4 Future-State Label Construction

To prevent target leakage, the target represented the next environmental observation rather than the same observation used as model input. A rolling 52-week mean and standard deviation were calculated:

$$\mu_t = \frac{1}{52}\sum_{i=t-51}^{t} x_i \tag{15}$$

$$\sigma_t = \sqrt{\frac{1}{51}\sum_{i=t-51}^{t}(x_i - \mu_t)^2} \tag{16}$$

The standardized future anomaly was

$$z_{t+1} = \frac{x_{t+1}-\mu_t}{\sigma_t} \tag{17}$$

Classification thresholds were estimated from the training target distribution only. The resulting thresholds were:

$$q_{0.50} = 0.3716, \qquad q_{0.80} = 1.3221 \tag{18}$$

The three relative environmental states were then defined as:

$$y_t = \begin{cases} 0, & z_{t+1} < q_{0.50}, \\ 1, & q_{0.50} \le z_{t+1} < q_{0.80}, \\ 2, & z_{t+1} \ge q_{0.80}. \end{cases} \tag{19}$$

These classes represent relative statistical environmental states rather than regulatory exposure categories.

4.5 Data Partitioning

The data were partitioned chronologically. The test partition was excluded from model fitting, normalization estimation, class-threshold generation, novelty normalization, and telemetry-threshold selection.

Table 1. Chronological dataset partition.

Partition	Samples	Percentage
Training	1,561	70.0%
Validation	335	15.0%

Partition	Samples	Percentage
Test	335	15.0%
Total	2,231	100%

The independent test set contained 164 Class-0 observations, 103 Class-1 observations, and 68 Class-2 observations.

4.6 Candidate Machine-Learning Models

4.6.1 Decision Tree

The Decision Tree employed a maximum depth of 10, a minimum leaf size of six observations, and balanced class weighting.

4.6.2 Random Forest

The Random Forest contained 32 trees, a maximum tree depth of nine, a minimum leaf size of four observations, and balanced subsample weighting.

4.6.3 Multilayer Perceptron

The multilayer perceptron contained hidden layers of 24 and 12 neurons with ReLU activations. The regularization coefficient was 3×10^{-4}, with early stopping and a maximum of 300 iterations.

4.6.4 Tiny One-Dimensional Convolutional Neural Network

The compact temporal CNN used two temporal input channels, two convolutional layers, eight filters per convolutional layer, a kernel size of three, adaptive average pooling, and a three-neuron output layer. Training used Adam with learning rate 2×10^{-3}. The maximum training duration was 100 epochs with early stopping.

4.7 Performance Metrics

The models were evaluated using accuracy, macro-precision, macro-recall, macro-F1, class-specific precision, class-specific recall, class-specific F1, confusion matrix, and serialized model size. The multi-metric approach is consistent with the broader TinyML benchmarking principle that deployment suitability should not be determined using predictive accuracy alone [5], [21]. Bootstrap confidence intervals were estimated using 2,000 test-set resamples. Exact McNemar tests were used for paired comparison of classifier errors.

4.8 VoI Controller Calibration

Temporal novelty normalization was fixed using the 95th percentile of the training novelty distribution, and the two communication thresholds were selected from the validation set as:

$$\tau_D = 0.90, \quad T_1 = 0.575, \quad T_2 = 0.675 \qquad (20)$$

When the VoI score was below the lower threshold, immediate transmission could be suppressed. At or above the lower threshold, an update was transmitted. A larger 32-byte packet was selected at or above the upper threshold, when the environmental state was high severity, or when predictive uncertainty exceeded the selected entropy criterion. Otherwise, a compact 12-byte packet was transmitted. In the Phase-1 implementation, the entropy criterion was U_t >= 0.72 and the information-age horizon T_max was 24 sampling intervals (24 weeks). The validation grid search selected T1 and T2 to minimize modeled RF energy subject to a critical-event delivery rate of at least 0.90 and a packet-delivery ratio of at least 0.82.

4.9 Communication Baselines

Four configurations were evaluated: B0 periodic fixed-RF monitoring, in which all 335 test observations generated a 32-byte transmission; B1 periodic TinyML telemetry, in which TinyML was executed but every observation generated a compact 12-byte transmission; B2 event-triggered TinyML, in which environmental events triggered transmission but the radio configuration remained fixed; and B3, the proposed value-aware adaptive RF system, in which VoI determined transmission requirement and payload length while modeled RF-link quality influenced radio configuration.

4.10 Modeled RF Environment

The communication experiment considered spreading factors SF ∈ {7, 8, 9, 10, 11, 12} and transmit-power levels P^TX ∈ {14, 17, 20, 22} dBm. LoRa time-on-air was estimated from payload length, spreading factor, bandwidth, coding rate, and preamble assumptions. Communication energy was calculated as

$$E_t^{\mathrm{TX}} = V_s\, I_t^{\mathrm{TX}}\, T_t^{\mathrm{air}} \tag{21}$$

A supply voltage of 3.3 V was assumed. For numerical reproducibility, the Phase-1 time-on-air model used 125 kHz bandwidth, coding rate 4/5, an eight-symbol preamble, CRC enabled, an explicit header, and low-data-rate optimization for SF11-SF12. Modeled transmit currents were 45, 58, 84, and 118 mA at 14, 17, 20, and 22 dBm, respectively. The modeled SNR requirements for SF7-SF12 were -7.5, -10, -12.5, -15, -17.5, and -20 dB. Link-success probability used a logistic model with slope 0.72 applied to the margin SNR - SNR_req + 0.11(P_TX - 14). The adaptive controller targeted modeled delivery probabilities of 0.94 for upper-threshold or Class-2 transmissions and 0.82 otherwise. Among configurations satisfying the target, it selected the minimum modeled transmit energy; when no configuration satisfied the target, it maximized p_success - 0.06 ln(1 + E_TX). These RF results are simulation/model-based results, not direct radio power measurements.

4.11 RF Metrics

$$\mathrm{PDR} = \frac{N_{\mathrm{received}}}{N_{\mathrm{transmitted}}} \tag{22}$$

$$R_{\mathrm{TX}} = \left(1 - \frac{N_{\mathrm{TX},B3}}{N_{\mathrm{TX},B0}}\right) \times 100\% \tag{23}$$

$$R_E = \left(1 - \frac{E_{B3}}{E_{B0}}\right) \times 100\% \tag{24}$$

$$\mathrm{CEDR} = \frac{N_{\mathrm{critical,\ delivered}}}{N_{\mathrm{critical,\ true}}} \tag{25}$$

4.12 Monte Carlo Analysis

Three hundred independently generated RF-link realizations were used to assess robustness. For each realization, environmental predictions remained unchanged, simulated RF-link quality varied, packet reception was evaluated independently, and adaptive RF settings were recalculated. Mean performance and empirical 2.5th and 97.5th percentiles were calculated.

4.13 Ablation Study

Five ablated B3 systems were evaluated: without uncertainty; without temporal novelty; without information age; without environmental severity; and without RF-link adaptation. The ablation experiment was used to establish which architecture components were responsible for observed system behavior. For reproducible component diagnosis, Table 6 reports the test-set ablation under the fixed RF-link trace generated with link-trace seed 202, while the machine-learning components use the study-wide training seed 42. Tables 4 and 5 report aggregate RF metrics across 300 independently generated link realizations. Consequently, realization-level RF ratios in Table 6 are not expected to equal the Monte Carlo means.

5. Results

5.1 Machine-Learning Performance

Table 2 presents the predictive performance of the four candidate models. Figure 3 provides a visual comparison of the candidate models.

Table 2. Test-set classification performance.

Model	Accuracy (%)	Macro-Precision (%)	Macro-Recall (%)	Macro-F1 (%)	Artifact Size (KB)
Decision Tree	84.48	81.61	82.50	81.97	11.36
Random Forest	89.85	87.52	88.96	88.08	421.27
MLP	83.58	83.50	77.67	78.68	27.29
Tiny 1-D CNN	82.69	82.62	77.00	78.59	4.03

The Random Forest achieved the highest test accuracy and macro-F1. Its bootstrap 95% confidence interval for accuracy was 86.57%–93.13%. The corresponding macro-F1 interval was 84.20%–91.72%. The Decision Tree achieved lower predictive performance but required a substantially smaller serialized representation. The Tiny CNN produced the smallest model artifact.

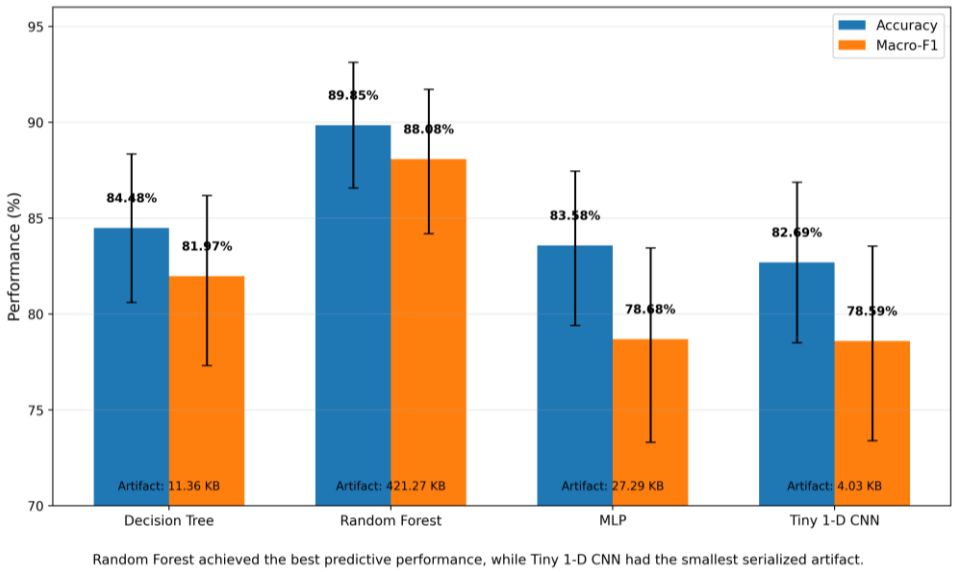


Figure 3. Comparative predictive performance of the candidate machine-learning models in terms of accuracy and macro-F1 score.

5.2 Pairwise Statistical Comparison

Exact McNemar tests comparing Random Forest predictions against the competing models produced $p = 0.0039$ for Random Forest versus Decision Tree, $p = 0.0038$ for Random Forest versus MLP, and $p = 0.0027$ for Random Forest versus Tiny CNN. The Random Forest therefore exhibited significantly different and superior paired error behavior at the conventional 0.05 significance level.

5.3 Class-Specific Performance

Table 3. Random Forest class-specific performance.

Class	Precision (%)	Recall (%)	F1 (%)	Support
Class 0	97.50	95.12	96.30	164
Class 1	85.57	80.58	83.00	103
Class 2	79.49	91.18	84.93	68

The highest-severity Class 2 achieved a recall of 91.18%. The confusion matrix is shown in Figure 4 and was

$$\mathbf{C} = \begin{bmatrix} 156 & 8 & 0 \\ 4 & 83 & 16 \\ 0 & 6 & 62 \end{bmatrix} \quad (26)$$

Of the 68 true Class-2 observations, 62 were correctly classified and six were classified as Class 1. None was directly misclassified as Class 0.

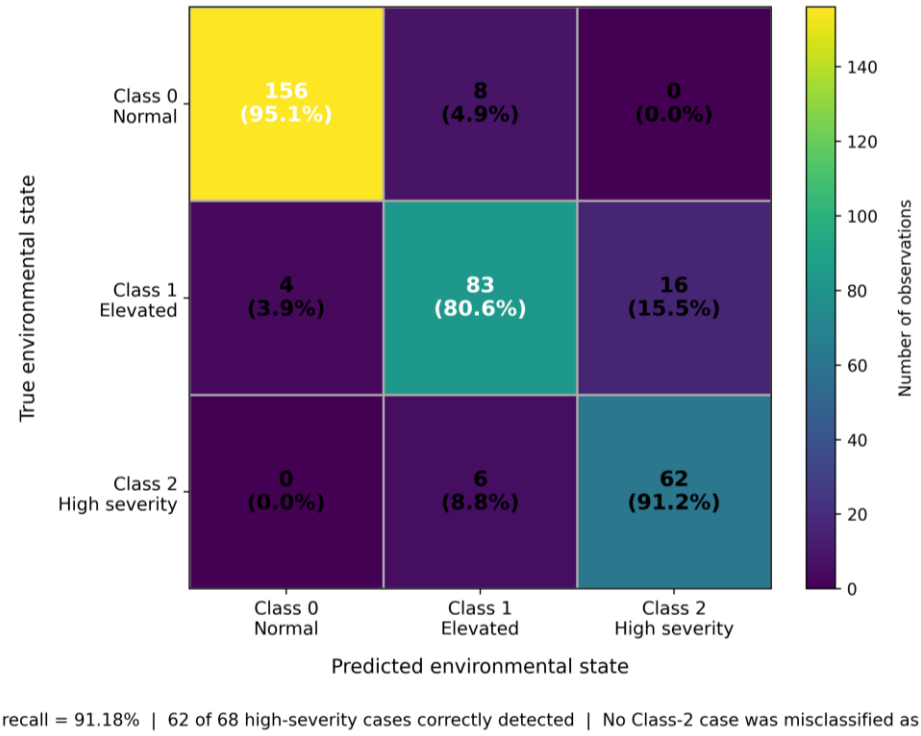


Figure 4. Confusion matrix of the selected Random Forest classifier on the independent test set.

5.4 Communication Activity

Table 4. Communication and modeled RF performance.

System	Packets Sent	Bytes Sent	Mean Modeled RF Energy (mJ)	Mean PDR (%)	Mean CEDR (%)
B0 Periodic fixed RF	335	10,720	15,823.54	70.78	76.41
B1 TinyML periodic	335	4,020	9,257.76	70.78	76.41
B2 Event TinyML fixed RF	177	3,684	6,420.16	70.59	76.41
B3 Proposed VoI + adaptive RF	92	2,764	10,738.93	94.44	87.04

B3 transmitted 92 packets compared with 335 under periodic monitoring. Thus, transmission reduction was 72.54%. The transmitted payload decreased from 10,720 bytes to 2,764 bytes, equivalent to 74.22% fewer transmitted bytes. In Table 4, packet and byte counts are deterministic for the fixed 335-observation test sequence, whereas modeled RF energy, PDR, and CEDR are means across the 300 RF-link realizations. A visual comparison of the communication architectures is provided in Figure 5.

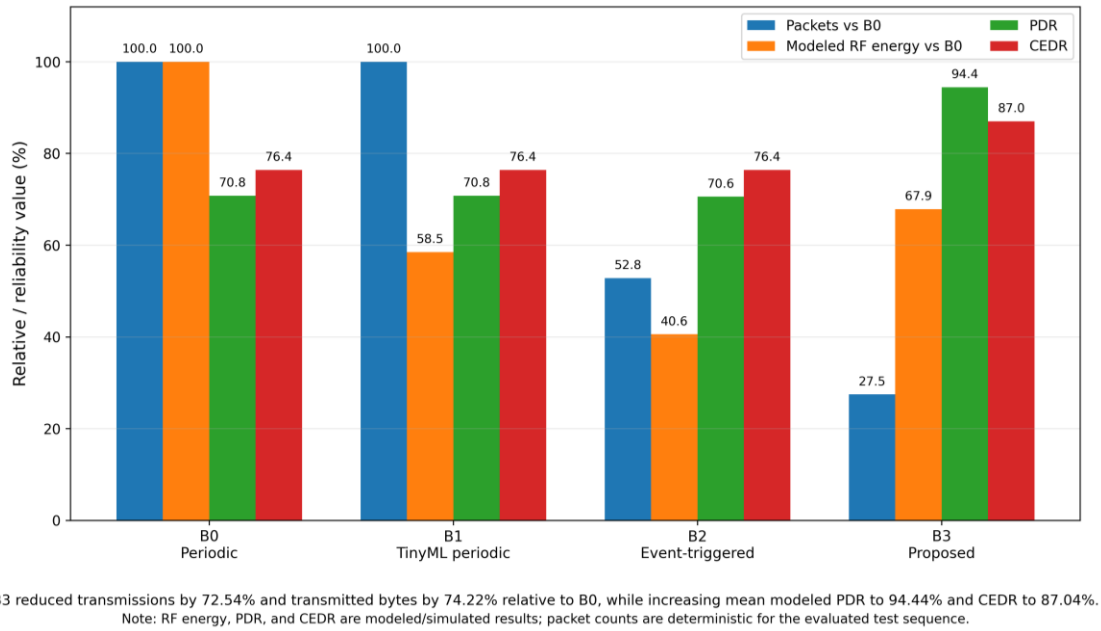


Figure 5. Comparison of the four communication architectures in terms of packets sent, normalized modeled RF energy, packet-delivery ratio, and critical-event delivery rate. RF-related metrics are modeled/simulated.

5.5 Modeled Communication Energy

B0 required 15,823.54 mJ of modeled RF energy, whereas B3 required 10,738.93 mJ. The modeled reduction was 32.13%. However, B2 required only 6,420.16 mJ and therefore consumed less modeled RF energy than B3. This result demonstrates that B3 was not designed to minimize energy at any cost. It deliberately selected more robust RF configurations when environmental information value or modeled link conditions justified additional communication expenditure.

5.6 Packet-Delivery Reliability

Across 300 modeled RF-link realizations, the mean B3 PDR was 94.44%. The empirical 95% interval was 90.22%–97.83%. B0 achieved a mean PDR of 70.78%, with an empirical interval of approximately 66.57%–74.33%. The absolute mean PDR improvement was therefore 23.66 percentage points.

5.7 Critical-Event Delivery

The mean critical-event delivery rate increased from 76.41% under the fixed-RF baseline to 87.04% under B3, an absolute improvement of 10.63 percentage points. The result demonstrates that reduced communication volume did not automatically imply lower delivery of high-value events.

5.8 Monte Carlo Results

Table 5. Monte Carlo communication results.

System	Mean RF Energy (mJ)	Mean PDR (%)	Mean CEDR (%)
B0	15,823.54	70.78	76.41
B1	9,257.76	70.78	76.41
B2	6,420.16	70.59	76.41
B3	10,738.93	94.44	87.04

For B3, modeled RF energy varied across link realizations, with an empirical 95% range of approximately 8,775.85–12,916.25 mJ. This variation reflects adaptive changes in radio configuration.

5.9 Ablation Analysis

Table 6. Fixed RF-link-trace component ablation results (link seed 202; ML seed 42).

Variant	Packets	RF Energy (mJ)	PDR (%)	CEDR (%)
Full B3	92	10,483.82	94.57	86.76
Without uncertainty	90	10,453.00	95.56	86.76
Without novelty	92	10,549.48	94.57	86.76

Variant	Packets	RF Energy (mJ)	PDR (%)	CEDR (%)
Without information age	105	11,993.40	93.33	86.76
Without severity	98	10,956.15	94.90	86.76
Without link adaptation	92	4,169.18	72.83	63.24

Note: Table 6 reports the reproducible test-set RF-link trace generated with seed 202, whereas Tables 4 and 5 report 300-realization aggregate RF metrics. In the Full B3 trace, 87 of 92 transmitted packets were delivered (94.57% PDR), and 59 of 68 true critical events were delivered (86.76% CEDR). Full B3 and the four non-link ablations each delivered the same 59 of 68 critical events, which explains the repeated 86.76% CEDR. Without link adaptation, 67 of 92 transmitted packets and 43 of 68 critical events were delivered, corresponding to 72.83% PDR and 63.24% CEDR. For comparison, the 300-realization B3 means in Tables 4 and 5 are 10,738.93 mJ, 94.44% PDR, and 87.04% CEDR.

Within this fixed RF-link-trace diagnostic, removal of RF-link adaptation produced the largest decline in reliability. PDR decreased from 94.57% to 72.83%, while critical-event delivery decreased from 86.76% (59 of 68) to 63.24% (43 of 68). Modeled RF energy also decreased substantially. This confirms that the reliability improvement obtained through adaptive RF operation carries an explicit energy cost. The ablation effects are summarized visually in Figure 6.

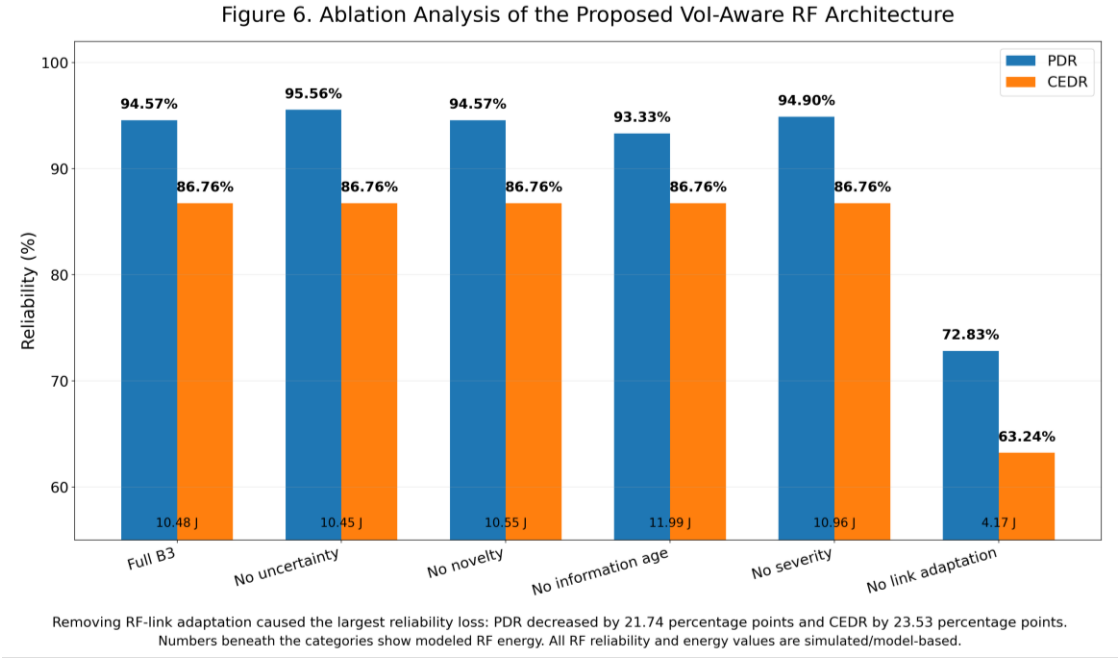


Figure 6. Fixed RF-link-trace (link seed 202; ML seed 42) ablation diagnostic showing the effect of removing individual components of the proposed architecture on modeled RF reliability metrics. RF-related metrics are modeled/simulated.

6. Discussion

6.1 Predictive Model Selection

The Random Forest achieved the strongest classification performance in the Phase-1 experiment, outperforming the Decision Tree, MLP, and compact CNN. The finding illustrates an important principle in TinyML research: deep-learning models should not automatically be assumed to provide the strongest solution for low-dimensional or engineered sensor time-series data.

The Decision Tree nevertheless retained practical interest because its serialized size was approximately 11.36 KB compared with 421.27 KB for the Random Forest. The Tiny CNN was even smaller at approximately 4.03 KB. This illustrates the accuracy-resource trade-off emphasized in TinyML benchmarking and energy-aware model-selection research [5], [11], [12], [21]. Importantly, these serialized file sizes do not represent final embedded Flash or peak SRAM requirements. Actual deployment measurements will be required before selecting the final hardware model.

6.2 Importance of High-Severity Recall

Environmental monitoring systems may have asymmetric error costs. Failing to communicate a routine environmental observation may have little operational effect, whereas failing to detect a high-severity condition may be unacceptable. The Random Forest correctly identified 62 of 68 highest-severity observations, corresponding to 91.18% recall. Furthermore, no true Class-2 sample was directly assigned to Class 0. This characteristic is valuable because the output of the environmental classifier influences the communication policy.

6.3 Communication Suppression

B3 reduced transmission count by 72.54% relative to periodic reporting. This observation is consistent with the general finding that local environmental inference can substantially reduce unnecessary communication. Reis [9], for example, reported major transmission reductions using adaptive/event-driven environmental monitoring. Wiese et al. [8] similarly demonstrated energy savings from local environmental processing relative to raw-data streaming. The contribution of the present architecture is therefore not the discovery that local inference reduces network traffic. Rather, it is the use of a continuous information-value signal to determine both whether communication should occur and how reliable the radio configuration should be.

6.4 Energy versus Reliability

One of the most important findings was that B2 consumed less modeled RF energy than B3. If energy minimization alone were used as the objective, B2 would therefore appear superior. However, B2's mean PDR remained approximately 70.6%, whereas B3 achieved approximately 94.4%. This demonstrates the limitation of reporting energy reduction without considering whether important information is actually delivered. The result is consistent with the broader LoRa literature showing that spreading factor and transmit-power adaptation influence both energy and reliability [16], [17]. The appropriate engineering objective is therefore not necessarily minimum energy alone, but rather maximizing successfully delivered valuable information per unit energy consumed. This motivates the proposed value-aware architecture.

6.5 Why RF-Link Adaptation Dominated the Ablation Study

In the fixed RF-link-trace component-ablation diagnostic, removing link adaptation caused the largest deterioration in PDR and critical-event delivery. The finding indicates that information prioritization alone is insufficient if communication reliability remains poor. An intelligent sensing node must determine not only which observations deserve transmission but also how much communication robustness those observations require. This principle complements AI*LoRa and EdgeSense, which demonstrated the value of intelligent RF

configuration [15], [17]. The distinction in the present study is that the reliability target itself is influenced by environmental information value.

6.6 Limited Contribution of Uncertainty in the Pilot Dataset

Within the fixed RF-link-trace diagnostic, removing the uncertainty term produced only a small change in transmission behavior and no change in CEDR: both Full B3 and the uncertainty-ablated variant delivered 59 of 68 critical events (86.76%). This should not be concealed. The most plausible explanation is that the Phase-1 benchmark contains only one principal environmental variable. Consequently, there are relatively few complex multivariate states in which classifier uncertainty independently adds substantial semantic value. The planned Beijing dataset contains six pollutants and multiple meteorological variables [20], making it more suitable for evaluating whether uncertainty contributes meaningfully in multivariate environmental classification.

6.7 Comparison with Recent Environmental TinyML Systems

Recent environmental-monitoring research has moved rapidly toward edge intelligence. Pereira et al. demonstrated embedded environmental classification [6]. Adebayo et al. integrated TinyML with environmental sensing and control [7]. Wiese et al. demonstrated multimodal edge-AI environmental sensing [8]. Reis combined environmental inference with adaptive duty cycling and LoRaWAN [9]. Trînc et al. demonstrated field-deployed autonomous environmental monitoring with low-power wide-area communication [10]. The present architecture extends this trajectory through explicit environmental information valuation. Regional evidence also strengthens the practical motivation: Akanji, Francis, and Akintola [22] documented multi-pollutant, seasonal air-quality variability across five Nigerian cities, indicating that monitoring nodes may need to respond differently across locations and pollution regimes rather than relying exclusively on fixed-rate telemetry.

Rather than using only an event-to-transmit rule, the proposed system combines environmental state, uncertainty, novelty, freshness, and link state to determine the communication action.

7. Future Hardware Validation

The next stage will deploy the architecture on a physical MCU and LoRa test platform. The recommended sensing configuration includes temperature, relative humidity, CO_2, $PM_{2.5}$, PM_{10}, and optional VOC measurements.

The physical experiments should measure Flash utilization, peak SRAM, inference latency, energy per inference, sensor-acquisition energy, RF transmit current, RSSI, SNR, physical PDR, time-on-air, retransmission rate, end-to-end event latency, and projected battery lifetime. TinyML power benchmarking should separate preprocessing, inference, and post-processing whenever possible. Recent benchmarking research has emphasized repeated hardware measurements and separate analysis of these execution stages [21]. The Beijing Multi-Site Air Quality dataset will additionally be used for multivariate external algorithm validation before physical field deployment [20].

8. Conclusion

This paper proposed a value-aware RF-enabled TinyML architecture for intelligent environmental monitoring under resource constraints. Unlike conventional environmental IoT systems that transmit periodically, and unlike event-triggered TinyML architectures that simply transmit when a classification threshold is crossed, the proposed approach links environmental information value directly to communication-resource allocation.

The Environmental Value-of-Information metric incorporated environmental severity, machine-learning uncertainty, temporal novelty, and information age. This score was combined with RF-link state to determine whether data should be transmitted, the appropriate payload size, and the radio reliability required.

The computational experiment showed that Random Forest achieved 89.85% accuracy, 88.08% macro-F1, and 91.18% recall for the highest-severity class. The proposed communication architecture reduced packet transmissions by 72.54% and transmitted data volume by 74.22% relative to periodic telemetry. Across 300 modeled RF-link realizations, mean PDR improved from 70.78% to 94.44%, while critical-event delivery increased from 76.41% to 87.04%.

The proposed architecture did not minimize RF energy absolutely. A simpler event-triggered system consumed less modeled energy. However, that system exhibited substantially poorer delivery reliability. This finding is central to the study: the most energy-efficient environmental sensing system is not necessarily the system that consumes the fewest joules, but the system that spends its constrained energy on reliably delivering the information that matters most.

The results support further investigation of Value-of-Information as a cross-layer control mechanism connecting TinyML environmental inference and adaptive wireless communication. Physical hardware measurements and multivariate air-quality validation constitute the next stage required before making claims regarding measured MCU energy, real-world LoRa reliability, or field deployment performance.

References

[1] P. P. Ray, "A review on TinyML: State-of-the-art and prospects," Journal of King Saud University Computer and Information Sciences, vol. 34, no. 4, pp. 1595–1623, 2022. doi: 10.1016/j.jksuci.2021.11.019.

[2] N. Schizas, A. Karras, C. Karras, and S. Sioutas, "TinyML for Ultra-Low Power AI and Large Scale IoT Deployments: A Systematic Review," Future Internet, vol. 14, no. 12, Art. no. 363, 2022. doi: 10.3390/fi14120363.

[3] R. David, J. Duke, A. Jain, V. J. Reddi, N. Jeffries, J. Li, N. Kreeger, I. Nappier, M. Natraj, T. Wang, P. Warden, and R. Rhodes, "TensorFlow Lite Micro: Embedded Machine Learning for TinyML Systems," Proceedings of Machine Learning and Systems, vol. 3, 2021.

[4] J. Lin, W.-M. Chen, Y. Lin, J. Cohn, C. Gan, and S. Han, "MCUNet: Tiny Deep Learning on IoT Devices," in Advances in Neural Information Processing Systems, vol. 33, 2020.

[5] C. Banbury, V. J. Reddi, P. Torelli, J. Holleman, N. Jeffries, C. Kiraly, P. Montino, D. Kanter, S. Ahmed, D. Pau, U. Thakker, A. Torrini, P. Warden, J. Cordaro, G. Di Guglielmo, J. Duarte, S. Gibellini, V. Parekh, H. Tran, N. Tran, N. Wenxu, and X. Xuesong, "MLPerf Tiny Benchmark," Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks, 2021.

[6] E. A. M. Pereira, J. F. da Silva Santos, and E. de Andrade Barboza, "An energy efficient TinyML model for a water potability classification problem," Sustainable Computing: Informatics and Systems, vol. 43, Art. no. 101010, 2024. doi: 10.1016/j.suscom.2024.101010.

[7] S. Adebayo, H. O. Aworinde, O. O. Olufemi, C. O. Osueke, A. E. Adeniyi, and O. J. Aroba, "Understanding mushroom farm environment using TinyML-based monitoring devices," Environmental Research Communications, vol. 7, Art. no. 045014, 2025. doi: 10.1088/2515-7620/adc5cd.

[8] P. Wiese, V. Kartsch, M. Guermandi, and L. Benini, "A Multi-Modal IoT Node for Energy-Efficient Environmental Monitoring with Edge AI Processing," in 2025 IEEE International Conference on Omni-layer Intelligent Systems (COINS), Madison, WI, USA, 2025. doi: 10.1109/COINS65080.2025.11125738.

[9] M. J. C. S. Reis, "Low-Power Embedded Sensor Node for Real-Time Environmental Monitoring with On-Board Machine-Learning Inference," Sensors, vol. 26, no. 2, Art. no. 703, 2026. doi: 10.3390/s26020703.

[10] E.-C. Trînc, V. Niţă, C. Stolojescu-Crisan, C. Ancuţi, R. M. Mihai, and C. P. Sultănoiu, "Cost-Effective TinyML-Ready Design and Field Deployment of a Solar-Powered Environmental

Monitoring Data Collector Using LTE-M Communication," Applied Sciences, vol. 16, no. 7, Art. no. 3237, 2026. doi: 10.3390/app16073237.

[11] A. Sabovic, J. Fontaine, E. De Poorter, and J. Famaey, "Energy-aware tinyML model selection on zero energy devices," Internet of Things, vol. 30, Art. no. 101488, 2025. doi: 10.1016/j.iot.2025.101488.

[12] A. G. Alanazi, H. A. Alanazi, and N. S. Albalawi, "Cross-Layer Resource Optimization for Ultra-Low-Power TinyML Inference on ARM Cortex-M Microcontrollers," Electronics, vol. 15, no. 13, Art. no. 2918, 2026. doi: 10.3390/electronics15132918.

[13] M. Uko, B. Bako, S. Ekpo, et al., "Perpetual edge intelligence: adaptive hybrid energy harvesting and reinforcement-learning-based TinyML for autonomous IoT sensors," Discover Electronics, vol. 3, Art. no. 33, 2026. doi: 10.1007/s44291-026-00186-y.

[14] B. Arratia, E. Rosas, C. T. Calafate, J.-C. Cano, J. M. Cecilia, and P. Manzoni, "AlLoRa: Empowering environmental intelligence through an advanced LoRa-based IoT solution," Computer Communications, vol. 218, pp. 44–58, 2024. doi: 10.1016/j.comcom.2024.02.014.

[15] B. Arratia-Uribe, E. Rosas-Olivos, E. Pietrosemoli, M. Zennaro, J. M. Cecilia-Canales, and P. Manzoni, "AI*LoRa: Enabling Efficient Long-Range Communication with Machine Learning at the Edge," in Proceedings of the 25th International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing, pp. 458–463, 2024. doi: 10.1145/3641512.3690040.

[16] M. A. Lodhi, L. Wang, K. Mahmood, A. Farhad, J. Chen, and S. Kumari, "Enhanced adaptive data rate strategies for energy-efficient Internet of Things communication in LoRaWAN," International Journal of Communication Systems, vol. 38, no. 2, e5966, 2025. doi: 10.1002/dac.5966.

[17] M. A. Lodhi, X. Sun, K. Mahmood, and A. Lodhi, "EdgeSense: A Hybrid TinyML and Deep Learning Framework for SNR-Aware Adaptive Data Rate in LoRaWAN," IEEE Transactions on Mobile Computing, vol. 25, no. 10, pp. 17535–17546, 2026. doi: 10.1109/TMC.2026.3695916.

[18] F. Alawad and F. A. Kraemer, "Value of Information in Wireless Sensor Network Applications and the IoT: A Review," IEEE Sensors Journal, vol. 22, no. 10, pp. 9228–9245, 2022. doi: 10.1109/JSEN.2022.3165946.

[19] C. D. Keeling, R. B. Bacastow, A. E. Bainbridge, C. A. Ekdahl Jr., P. R. Guenther, and L. S. Waterman, "Atmospheric carbon dioxide variations at Mauna Loa Observatory, Hawaii," Tellus, vol. 28, pp. 538–551, 1976.

[20] S. Chen, Beijing Multi-Site Air Quality [Dataset], UCI Machine Learning Repository, 2017. doi: 10.24432/C5RK5G.

[21] P. Bartoli, C. Veronesi, A. Giudici, D. Siorpaes, D. Trojaniello, and F. Zappa, "Benchmarking Energy and Latency in TinyML: A Novel Method for Resource-Constrained AI," in 2025 International Joint Conference on Neural Networks (IJCNN), 2025. doi: 10.1109/IJCNN64981.2025.11228997.

[22] A. R. Akanji, M. O. Francis, and A. F. Akintola, "Air Quality Trends and Pollution Analysis in Nigerian Cities Using Time Series Methods," International Journal of Advanced Statistics and Probability, vol. 11, no. 2, pp. 108–123, 2024. doi: 10.14419/w5rj1f64.