LORA, DIGITAL TWIN, MACHINE LEARNING, CNN-LSTM

An Integrated LoRa Fog and Digital Twin Framework for Predictive Maintenance of Telecommunication Shelters

Gabriel Chibuoso Okpara[1], Nwazor Nkolika[2]

[1]Centre for Information and Telecommunication Engineering, Faculty of Engineering, University of Port Harcourt, Port Harcourt, Nigeria
[2]Department of Electrical/Electronic Engineering, University of Port Harcourt, Port Harcourt, Nigeria

Published November 2026

Abstract

Telecommunication shelters depend on environmental control and backup power to protect network equipment, yet fixed-threshold monitoring often identifies faults only after conditions have deteriorated. This paper presents a simulation-based predictive-maintenance framework that combines LoRa telemetry, fog-level analytics, a synchronized digital twin, and a convolutional neural network–long short-term memory (CNN–LSTM) model. The simulated shelter produced multivariate temperature, humidity, power, cooling, battery, and grid-state telemetry at five-minute intervals over a nine-day horizon. Controlled scenarios represented cooling degradation and failure, door opening, load increase, and grid outage. A LoRa link model introduced packet loss and delay; fog processing smoothed signals, reconstructed missing timestamps, and generated local anomaly alerts. The digital twin tracked thermal, humidity, and power states, while a multi-task CNN–LSTM classified health states and estimated remaining useful life (RUL). Under the reported configuration, mean LoRa packet success was 86.3% and mean latency was 238 ms. Fog processing reduced temperature-signal variance by 41.6%, reconstructed 92.1% of missing timestamps, and achieved anomaly-detection precision, recall, and F1-score of 0.91, 0.87, and 0.89. Digital-twin tracking errors were 0.78 °C for temperature and 3.1% relative humidity for humidity. The CNN–LSTM achieved 0.94 accuracy and an RUL RMSE of 2.6 hours on the simulated evaluation data. Fog inference averaged 184 ms compared with 612 ms for the cloud-only comparison. These results indicate the potential of an integrated edge and digital-twin approach for shelter monitoring. Since the data and fault labels were simulation-generated, field validation remains necessary before operational use.

Keywords: CNN–LSTM; digital twin; fog computing; LoRa; predictive maintenance; remaining useful life; telecommunication shelter

1 Introduction

Telecommunication shelters house radio, transmission, power, and cooling equipment whose performance depends on controlled temperature, humidity, and electrical supply. Degradation of an air-conditioning unit, loss of grid power, or abnormal battery discharge can place equipment at risk even when the network itself continues to operate. Conventional monitoring commonly relies on fixed alarm thresholds. Such alarms are useful for detecting limit violations, but they provide less information about gradual deterioration and may trigger after maintenance opportunities have narrowed.

Low-power wide-area networks can support distributed sensing at remote sites where wide coverage and low energy demand matter more than high data rates [1]. Fog computing places data processing

close to the devices and can reduce reliance on cloud connectivity for time-sensitive decisions [2]. Digital twins add a state model that can be synchronized with telemetry and used to examine disturbances and maintenance strategies. Reviews of digital-twin fault detection and predictive maintenance report growing interest in these methods while identifying continuing challenges in data availability, system complexity, and validation [3], [4]. CNN–LSTM models have also been applied to multivariate time-series failure prediction, combining local feature extraction with sequence learning [5].

This study integrates these elements for telecommunication-shelter environmental maintenance. Its contribution is an end-to-end simulated workflow that explicitly includes communication loss, fog preprocessing, digital-twin state tracking, fault classification, and RUL estimation. The work evaluates system behavior under repeatable operational disturbances and compares fog-based inference with a cloud-only latency baseline. All performance values reported here arise from simulation and emulation; they should not be interpreted as results from a live telecom site.

2 Materials and methods

2.1 Study design and simulated telemetry

A systems-engineering design was used to connect physical-state simulation, communication modeling, edge analytics, digital-twin synchronization, and predictive learning. The shelter model generated temperature, relative humidity, equipment load, cooling power, battery state of charge (SOC), battery voltage, grid availability, door state, and cooling status. Time-dependent environmental inputs and subsystem relationships were used to create plausible operating traces. Each simulation covered nine days with a five-minute sampling interval, yielding 2,592 time steps per configured trace. The study did not collect measurements from an operating shelter or human participants.

Five controlled disturbance types were injected into the simulated traces: gradual cooling degradation, abrupt cooling failure, door opening, equipment-load increase, and grid outage. The injected event schedule supplied the ground-truth labels. Fault conditions were defined by event rules and operational thresholds, including indoor temperature above 34 °C, indoor relative humidity above 75%, or cooling being off. The simulated scenarios were used to test whether the processing chain could recognize the resulting changes and preserve their timing. The main simulation and model settings are listed in Table 1.

Table 1. Main simulation and model settings.

Setting	Value
Sampling interval	5 min
Observation horizon	9 days; 2,592 samples per trace
Sequence window	24 samples (2 h), stride 1 sample
Window count	2,568 per trace before data partitioning
LoRa configuration	868 MHz; 14 dBm; 125 kHz; SF11; 28-byte payload
Telemetry interval	5 min
CNN–LSTM maximum training	100 epochs; batch size 32; dropout 0.3
Learning task	Health-state classification and RUL regression

2.2 LoRa communication and fog analytics

A packet-level LoRa link model transformed generated telemetry into received telemetry. The model used a log-distance path-loss relation with log-normal shadowing and logged received signal strength, signal-to-noise ratio, packet delivery, and latency. For each simulated transmission, successful or lost delivery was recorded rather than assuming perfect communication. The selected 868 MHz setting is an experimental parameter and would require region-specific configuration before any deployment.

At the fog layer, the received signals were smoothed and conditioned, gaps were interpolated, and time windows were assembled for anomaly analysis. Anomaly scores combined normalized signal levels and rates of change. The local alert logic categorized events by severity and duration, allowing routine telemetry to be filtered before uplink. This design follows the fog-computing principle of distributing computation toward the network edge [2].

2.3 Digital twin and predictive model

The digital twin maintained synchronized estimates of indoor temperature, humidity, total power, SOC, battery voltage, and health state. At each update, it received fog-processed telemetry and compared measured behavior with its state representation. Tracking error was summarized using temperature and humidity RMSE and power MAPE; drift correction was recorded as a success proportion. Scenario simulations were used to determine whether the twin reproduced expected changes in cooling, moisture, load, and battery behavior.

For predictive maintenance, the telemetry was divided into overlapping 24-step windows, representing two hours of history. Each window contained approximately 8–12 physical and derived features. The reported workflow used chronological partitioning to preserve the direction of time and reduce leakage between adjacent windows. The CNN component learned local patterns in the multivariate sequences; the LSTM component represented temporal dependencies. A multi-task output produced health-state classes and an RUL estimate. Training used Adam with an initial learning rate of 0.001, batch size 32, up to 100 epochs, dropout of 0.3, and early stopping after a validation plateau. RUL targets were derived from the simulated event schedule, not from observed component failures.

2.4 Evaluation measures

Communication performance was measured by packet success, packet loss, RSSI, SNR, and latency. Fog preprocessing was assessed by signal-variance reduction, missing-timestamp reconstruction, and processing time. Anomaly detection was evaluated using precision, recall, F1-score, false-alarm rate, and detection delay. Twin performance was evaluated with state-tracking errors and update time. Predictive-model metrics included accuracy, precision, recall, F1-score, early-warning time, RUL RMSE, MAPE, and correlation. The cloud-only comparison was used to assess inference latency and uplink bandwidth against the fog-based pathway.

3 Results

3.1 Environmental and communication behavior

The simulated shelter produced a mean indoor temperature of 29.8 °C (standard deviation 2.7 °C; range 24.9–36.7 °C) and mean indoor relative humidity of 56.9% (standard deviation 8.8%; range 33.5–82.4%). Mean equipment load was 1,008 W, mean cooling power was 512 W, and mean total

power was 1,520 W. The mean battery SOC was 0.74. These values describe the configured model and its injected operating scenarios; they are not field measurements.

With the selected LoRa configuration, average RSSI was −108.4 dBm and average SNR was −7.2 dB. Packet success was 86.3% (13.7% loss), while mean and 95th-percentile latency were 238 ms and 612 ms, respectively. The observed loss created missing data for downstream processing and therefore allowed the fog gap-handling stage to be assessed.

3.2 Fog processing and anomaly detection

Fog preprocessing reduced temperature-signal variance by 41.6% and humidity-signal variance by 34.2%. The temperature trace before and after fog conditioning is compared in Figure 1. The processing stage reconstructed 92.1% of missing timestamps. Mean processing latency was 18.7 ms, with a 95th-percentile value of 41.3 ms. The anomaly detector achieved precision 0.91, recall 0.87, and F1-score 0.89, with a reported false-alarm rate of 6.8% and detection delay of one sample interval (five minutes). The system recorded 27 warning, 9 critical, and 14 deferred alerts. Local filtering reduced reported uplink bandwidth use by 52.6%.

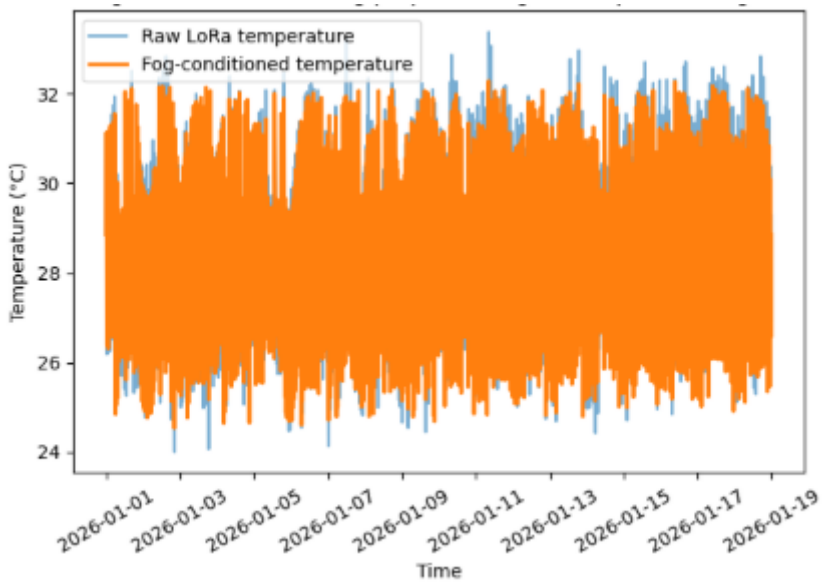


Figure 1. Raw and fog-conditioned temperature signal in the simulated shelter.

3.3 Digital-twin tracking and scenario response

The twin tracked indoor temperature with RMSE 0.78 °C and humidity with RMSE 3.1 percentage points. Figure 2 compares physical-model temperature with the digital-twin estimate. Power tracking error was 6.4% MAPE; drift correction succeeded in 93.2% of reported cases. The mean state-update cycle took 0.43 s. Under scenario injection, cooling degradation produced a temperature rise of 0.38 °C/h, while cooling failure increased temperature by 5.6 °C over two hours. Door opening raised indoor humidity by 13.4 percentage points within 20 minutes. A load spike increased total power by 410 W, and a six-hour grid outage produced an SOC decrease of 0.19.

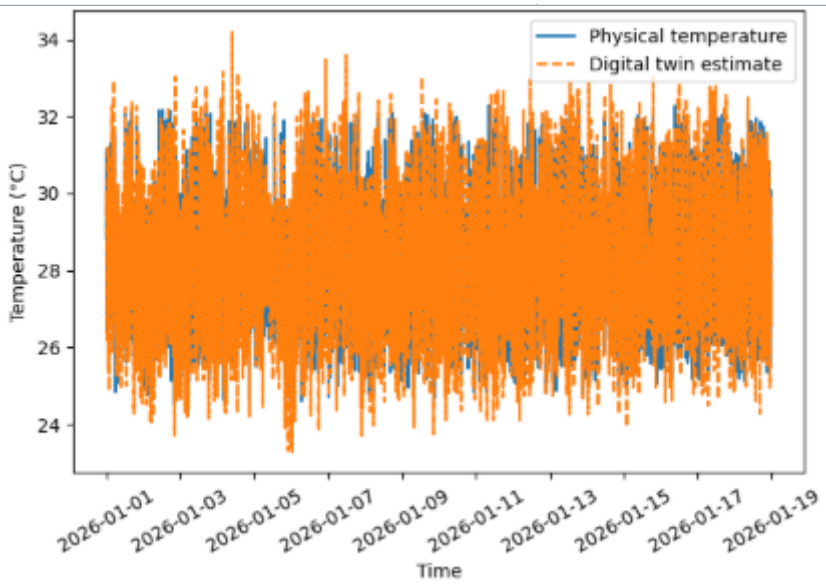


Figure 2. Physical-model temperature and digital-twin estimate over the simulated period.

3.4 Predictive maintenance and system comparison

The CNN–LSTM classification task achieved accuracy 0.94, precision 0.93, recall 0.90, and F1-score 0.92. Figure 3 presents the corresponding confusion matrix. The reported mean early-warning time was 1.7 hours. For RUL regression, RMSE was 2.6 hours, MAPE was 11.8%, and correlation with the simulated target was 0.89. These scores apply to the simulated, schedule-derived targets and should be read as proof of concept rather than as a field reliability estimate. The selected performance results of all subsystems are summarized in Table 2.

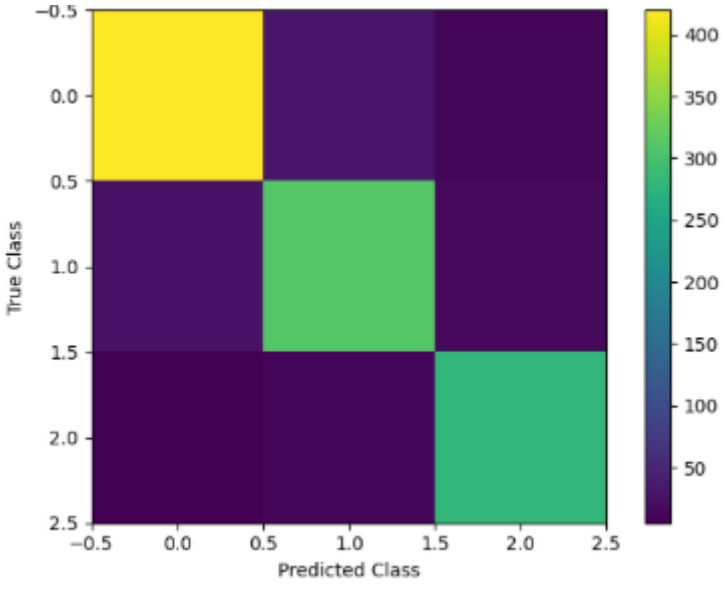


Figure 3. Confusion matrix for CNN–LSTM fault classification on simulated evaluation data.

Table 2. Selected performance results reported for the simulated framework.

Subsystem / task	Measure	Reported result
LoRa link	Packet success; mean latency	86.3%; 238 ms

Subsystem / task	Measure	Reported result
Fog preprocessing	Temperature variance reduction; gap reconstruction	41.6%; 92.1%
Fog anomaly detection	Precision; recall; F1; false-alarm rate	0.91; 0.87; 0.89; 6.8%
Digital twin	Temperature RMSE; humidity RMSE	0.78 °C; 3.1% RH
CNN–LSTM classification	Accuracy; F1; early warning	0.94; 0.92; 1.7 h
CNN–LSTM RUL	RMSE; MAPE; correlation	2.6 h; 11.8%; 0.89
Inference comparison	Fog mean latency; cloud-only mean latency	184 ms; 612 ms
System-level	Latency reduction; bandwidth reduction	69.9%; 52.6%

Fog-based model inference averaged 184 ms compared with 612 ms for cloud-only inference, a 69.9% decrease under the reported comparison. The proposed framework also used 47.4% of the reference uplink bandwidth, corresponding to a 52.6% reduction. The separate anomaly-detection comparison reported F1-scores of 0.89 for the integrated framework and 0.80 for cloud-only processing. These comparisons show the simulated benefit of local processing; they do not establish behavior under production network loading or hardware constraints.

4 Discussion

The results indicate that communication, preprocessing, state synchronization, and learning should be evaluated as a single chain. A packet success rate of 86.3% left a material proportion of samples unavailable at the receiver, yet the fog layer recovered most missing timestamps and kept processing delay below the five-minute sampling interval. This supports the design choice to include communication effects in predictive evaluation rather than training and assessing models only on complete telemetry. The measured advantage is specific to the configured link model and sampling rate.

The digital twin provided a compact representation of thermal, moisture, and power behavior under controlled disturbances. Its tracking errors and response to injected events show that the selected state variables were sufficient to distinguish several modeled failure patterns. However, because both the reference traces and the twin were generated within the simulation workflow, the error figures quantify internal consistency under those assumptions. Independent field data are needed to establish fidelity to actual shelters.

The CNN–LSTM results suggest that sequence-based learning can use the modeled telemetry to identify degradation and estimate time to a scheduled fault. The broader literature similarly considers hybrid temporal models for predictive maintenance [5]. In this study, however, RUL labels were derived from known simulation schedules. The 2.6-hour RMSE therefore does not measure prediction against naturally occurring asset failure. It measures agreement with the generated target and should not be translated directly into maintenance lead time without external validation.

At the system level, the comparison between fog and cloud-only inference highlights a practical trade-off: local processing can reduce round-trip delay and data transmission at the cost of relying on an edge node with limited resources. Prior work on fog systems identifies the edge placement of computation as a means of supporting IoT services while introducing its own deployment and coordination requirements [2]. A field implementation would need to test processor load, power consumption, cybersecurity, link adaptation, and the stability of alert thresholds in changing climatic and network conditions.

5 Limitations

The study is simulation-driven and does not use live telemetry, real device failure histories, or a deployed LoRa/fog testbed. The fault scenarios and thresholds were specified in advance, and the RUL targets were generated from those scenarios. The reported predictive scores may therefore overstate performance on unmodeled events or site-to-site variation. The number and diversity of independently simulated site profiles are not sufficient to support broad generalization claims. The radio configuration, including 868 MHz, is a modeling choice and must be aligned with local regulations and deployment conditions. Finally, resource contention and hardware limitations at an operational fog node were not fully characterized.

6 Conclusion

This paper presented a simulation-based framework integrating LoRa telemetry, fog analytics, digital-twin state tracking, and CNN–LSTM predictive maintenance for telecommunication shelters. In the reported experiments, fog processing handled packet gaps, detected anomalies within one sampling interval, and reduced the simulated inference latency and uplink bandwidth relative to cloud-only processing. The twin reproduced modeled cooling, moisture, load, and outage responses, while the CNN–LSTM achieved strong classification and RUL scores against schedule-derived targets. The evidence supports the framework as a proof of concept. Before operational adoption, the next step is validation with time-stamped data from real shelter sensors, independent fault events, and representative fog hardware, followed by recalibration of the communication and alarm settings for the target region.

References

[1] Raza U, Kulkarni P, Sooriyabandara M. Low power wide area networks: an overview. *IEEE Communications Surveys & Tutorials*. 2017;19(2):855–873. https://doi.org/10.1109/COMST.2017.2652320

[2] Chiang M, Zhang T. Fog and IoT: an overview of research opportunities. *IEEE Internet of Things Journal*. 2016;3(6):854–864. https://doi.org/10.1109/JIOT.2016.2584538

[3] Hosamo HH, Nielsen HK, Alnmr AN, Svennevig PR, Svidt K. A review of the digital twin technology for fault detection in buildings. *Frontiers in Built Environment*. 2022;8:1013196. https://doi.org/10.3389/fbuil.2022.1013196

[4] van Dinter R, Tekinerdogan B, Catal C. Predictive maintenance using digital twins: a systematic literature review. *Information and Software Technology*. 2022;151:107008. https://doi.org/10.1016/j.infsof.2022.107008

[5] Wahid A, Breslin JG, Ali MI. Prediction of machine failure in Industry 4.0: a hybrid CNN-LSTM framework. *Applied Sciences*. 2022;12(9):4221. https://doi.org/10.3390/app12094221