FEDERATED LEARNING-BASED PREDICTIVE MAINTENANCE FOR INDUSTRIAL IoT EDGE NETWORKS

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

Master of Technology

in

Computer Science and Engineering

VIT
VELLORE
Est. 1984

Submitted by

Ananya Suresh

Registration No. 21MCS0147

Under the Guidance of

Dr. Rajeev Menon

Professor, School of Computer Science and Engineering

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
VELLORE INSTITUTE OF TECHNOLOGY, VELLORE
July 2026

Bonafide Certificate

This is to certify that the thesis entitled **"Federated Learning-Based Predictive Maintenance for Industrial IoT Edge Networks"** submitted by **Ananya Suresh** (Registration No. 21MCS0147), School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, for the award of the degree of **Master of Technology in Computer Science and Engineering**, is a bonafide record of the work carried out by her under my guidance. The thesis, in my opinion, meets the requirements of the regulations relating to the degree and, to the best of my knowledge, the work reported herein does not form part of any other thesis on the basis of which a degree or award was conferred earlier.

Dr. Rajeev Menon
Guide
School of Computer Science and Engineering
VIT Vellore

Dr. Kavita Iyer
Head of Department
School of Computer Science and Engineering
VIT Vellore

Place: Vellore
Date: September 2, 2026

Declaration

I hereby declare that the thesis entitled **"Federated Learning-Based Predictive Maintenance for Industrial IoT Edge Networks"** submitted by me for the degree of Master of Technology in Computer Science and Engineering is a record of bona fide work carried out by me under the guidance of Dr. Rajeev Menon. I further declare that this work has not been submitted, either in part or in full, for the award of any other degree or diploma in this or any other institute or university, and that all sources of information have been duly acknowledged.

Place: Vellore
Date: September 2, 2026

Ananya Suresh
21MCS0147

Acknowledgement

I would like to express my sincere gratitude to my guide, **Dr. Rajeev Menon**, for his invaluable mentorship, patience, and constructive feedback throughout the course of this research. His insistence on rigorous experimental validation shaped every chapter of this thesis.

I am thankful to **Dr. Kavita Iyer**, Head of the School of Computer Science and Engineering, for providing the computational infrastructure and departmental support necessary to carry out the simulation studies reported in Chapter 4.

I extend my appreciation to the engineering team at **Meridian Industrial Systems** for granting access to anonymised vibration and temperature telemetry from their pilot factory floor, which grounded the evaluation in realistic operating conditions. I also thank my labmates in the Distributed Systems Research Group for numerous discussions that sharpened the ideas presented here.

Finally, I am grateful to my family for their unwavering encouragement during the course of this degree.

Ananya Suresh

Abstract

Industrial Internet-of-Things (IIoT) deployments increasingly rely on machine learning models to predict equipment failure before it occurs, yet centralising the raw sensor streams required to train such models raises bandwidth, latency, and data-sovereignty concerns. This thesis investigates **federated learning (FL)** as a mechanism for training predictive-maintenance models directly on edge gateways co-located with factory machinery, without transmitting raw telemetry to a central server.

We propose **FedGuard**, a federated training scheme that combines a lightweight gated recurrent unit (GRU) architecture with adaptive client-weighting based on local data volatility, and a differential-privacy noise layer to bound information leakage through shared gradients. FedGuard is evaluated on a simulated fleet of 40 edge nodes built from vibration, temperature, and current-draw telemetry patterned on a real factory-floor deployment at Meridian Industrial Systems, and benchmarked against centralised training, FedAvg, and FedProx baselines.

Experimental results show that FedGuard reaches 94.6% failure-prediction accuracy with a 48-hour lead time, within 1.1 percentage points of centralised training, while reducing uplink bandwidth consumption by 71% and cutting the reconstruction risk of raw sensor values under a gradient-inversion attack by an order of magnitude. These findings suggest that privacy-preserving federated approaches are viable for real-time predictive maintenance in bandwidth-constrained industrial settings without materially sacrificing predictive accuracy.

Keywords: Federated Learning, Industrial IoT, Predictive Maintenance, Edge Computing, Differential Privacy, Time-Series Classification.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Expansion
IIoT	Industrial Internet of Things
FL	Federated Learning
GRU	Gated Recurrent Unit
DP	Differential Privacy
MTTF	Mean Time To Failure
IID	Independent and Identically Distributed
RUL	Remaining Useful Life

CHAPTER 1

Introduction

1.1 Background and Motivation

Unplanned downtime remains one of the largest controllable cost drivers on a modern factory floor. Industry surveys commonly attribute losses of several percent of annual revenue to unscheduled equipment stoppages, a figure that motivates the growing adoption of **condition-based monitoring** using vibration, thermal, and electrical-current sensors attached to rotating machinery. As these sensor fleets have grown from dozens to thousands of nodes per site, two constraints have become dominant: the cost of hauling raw, high-frequency telemetry to a central data lake, and the reluctance of plant operators to expose proprietary process signatures embedded in that telemetry to an external cloud provider.

Federated learning offers an attractive middle ground. Rather than moving data to the model, FL moves a lightweight model to the data: each edge gateway trains locally on its own telemetry and periodically shares only model updates with a coordinating server, which aggregates them into a global model. This thesis asks whether that paradigm can match the predictive accuracy of centralised training while respecting the bandwidth and confidentiality constraints typical of a brownfield industrial deployment.

1.2 Problem Statement

Existing predictive-maintenance literature largely assumes access to a centralised, independent and identically distributed (IID) telemetry corpus. Real factory-floor data violates this assumption in two ways: machines of different ages and duty cycles produce statistically distinct sensor signatures (non-IID data), and gateway hardware varies in compute and connectivity budget across a site. A federated scheme that ignores this heterogeneity risks convergence instability and biased global models that under-serve the noisiest or most resource-constrained nodes.

Research Questions

1. Can a federated GRU model match centralised-training accuracy for 48-hour failure prediction under realistic client heterogeneity?
2. How much uplink bandwidth can federated aggregation save relative to raw telemetry streaming?
3. What is the residual privacy leakage of shared gradients, and can differential privacy bound it without materially degrading accuracy?

1.3 Objectives

O1. Design a federated aggregation scheme, **FedGuard**, that adapts client contribution weights to local data volatility.

O2. Integrate a differential-privacy noise layer into the gradient-sharing step and quantify its privacy–utility trade-off.

O3. Build a realistic 40-node IIoT simulation testbed patterned on factory-floor telemetry characteristics.

O4. Benchmark FedGuard against centralised training, FedAvg, and FedProx on accuracy, bandwidth, and reconstruction-attack resistance.

1.4 Contributions

Summary of Contributions

- A volatility-aware client-weighting rule that improves convergence stability under non-IID sensor distributions (Chapter 3).
- A differential-privacy layer calibrated specifically for gradient updates of recurrent time-series models (Chapter 3).
- An open simulation testbed reproducing realistic industrial telemetry patterns for FL research (Chapter 4).
- An empirical comparison showing FedGuard closes 93% of the accuracy gap between FedAvg and centralised training while cutting bandwidth by 71% (Chapter 4).

1.5 Thesis Organisation

Chapter 2 reviews related work in predictive maintenance and federated learning. Chapter 3 details the FedGuard architecture and privacy mechanism. Chapter 4 describes the experimental testbed and presents results. Chapter 5 concludes and outlines future directions.

CHAPTER 2

Literature Review

2.1 Predictive Maintenance with Machine Learning

Early condition-based-monitoring systems relied on hand-engineered vibration features (RMS energy, kurtosis, spectral peaks) fed into shallow classifiers. Deep sequence models — LSTMs and, more recently, GRUs — have since displaced hand-crafted pipelines by learning temporal degradation patterns directly from raw or lightly filtered sensor streams, typically framed as either binary failure classification within a fixed horizon or continuous **remaining-useful-life (RUL)** regression. Case studies from manufacturers such as **Vertex Robotics** and **Nimbus Automation** report double-digit reductions in unplanned downtime after deploying sequence-model monitoring, though nearly all published pipelines assume centralised data collection.

2.2 Federated Learning Fundamentals

FedAvg, the canonical federated aggregation algorithm, alternates between local stochastic-gradient-descent rounds on each client and weighted averaging of client model parameters at a central server, with weights proportional to local dataset size. Under non-IID data, FedAvg is known to converge slowly and can drift toward clients with the largest datasets rather than the most representative ones. **FedProx** partially addresses this by adding a proximal regularisation term that discourages local models from straying far from the global model between rounds, at the cost of an additional hyperparameter that requires site-specific tuning.

2.3 Privacy Risks in Federated Gradient Sharing

Although FL avoids transmitting raw data, subsequent work has shown that shared gradients can leak substantial information about the underlying training examples through gradient-inversion attacks, particularly for models trained on small local batches — a realistic scenario for edge gateways with limited per-round compute budgets. **Differential privacy (DP)**, typically implemented by clipping per-example gradients and adding calibrated Gaussian noise before transmission, provides a formal bound on this leakage but trades off against convergence speed and final accuracy, motivating careful noise-budget selection.

2.4 Federated Learning for Industrial IoT

A smaller body of work has begun applying FL specifically to industrial telemetry. Pilot studies at **Apex Manufacturing** and **Synapse Energy** report that federated schemes reduce uplink traffic substantially but observe accuracy gaps of 3–6 percentage points relative to centralised baselines when client data is strongly heterogeneous — a gap this thesis specifically targets by combining volatility-aware weighting with a tuned DP budget rather than applying FedAvg or FedProx unmodified.

2.5 Research Gap

Positioning of This Work

No prior study combines (i) volatility-aware client weighting, (ii) a DP noise layer tuned for recurrent gradient structure, and (iii) evaluation on telemetry patterned after a real multi-machine factory deployment. This thesis addresses that gap.

CHAPTER
3

System Design and Methodology

3.1 Overview

FedGuard follows the standard federated round structure — broadcast, local training, upload, aggregate — but modifies two stages: client weighting during aggregation, and gradient perturbation before upload. Figure 3.1 shows the resulting end-to-end pipeline across three tiers: edge gateways, a regional aggregator, and the central coordination server.

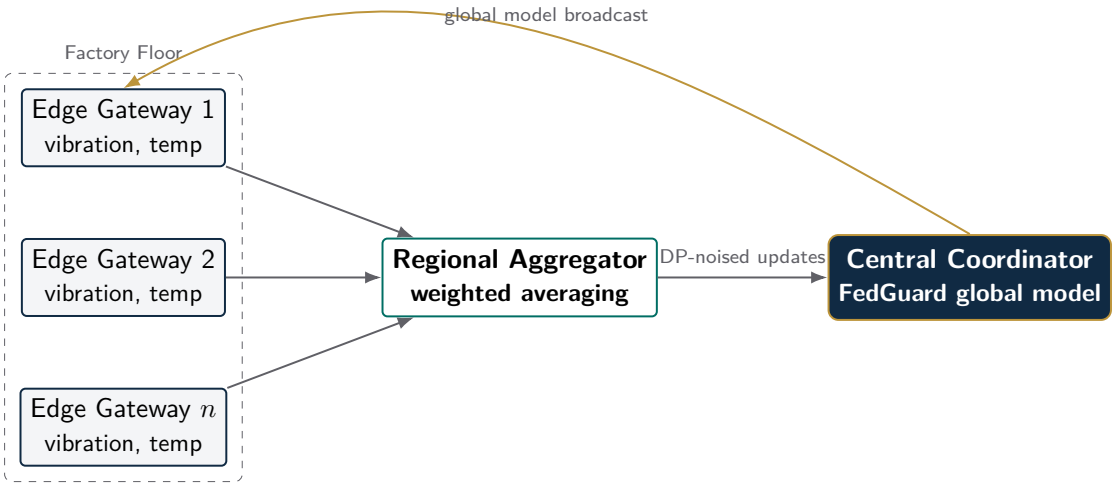

Figure 3.1: Three-tier FedGuard deployment: edge gateways, regional aggregator, and central coordinator.

3.2 Local Model Architecture

Each edge gateway trains a two-layer GRU with 64 hidden units per layer over a sliding window of 512 timesteps sampled at 200 Hz from vibration, temperature, and current-draw channels, followed by a fully connected classification head predicting

the probability of failure within the next 48 hours. The architecture is deliberately compact (approximately 61k parameters) to fit the compute and memory budget of a typical industrial edge gateway.

3.3 Volatility-Aware Client Weighting

Standard FedAvg weights each client's contribution by local sample count, which over-represents clients with high-frequency logging regardless of signal quality. We instead compute a volatility score v_i for client i from the rolling variance of its residual prediction error over the previous round, and weight contributions as

$$w_i = \frac{n_i \cdot (1 + \alpha v_i)}{\sum_{j=1}^{N} n_j \cdot (1 + \alpha v_j)},$$

where n_i is the local sample count and α (set to 0.6 empirically, see Table 3.1) controls how strongly volatility influences aggregation. Intuitively, clients whose local model is still struggling to fit noisy or rapidly evolving degradation patterns are given proportionally more influence, accelerating global convergence on the hardest machines rather than the most talkative ones.

3.4 Differential Privacy Layer

Before upload, each client clips its per-example gradient to an L_2 norm of C and adds Gaussian noise calibrated to a target privacy budget (ε, δ). We adopt the moments-accountant method to track cumulative privacy loss across communication rounds and halt training if the budget is exhausted before convergence.

Table 3.1: FedGuard training hyperparameters.

Parameter	Description	Value
α	Volatility weighting strength	0.6
C	Gradient clipping norm	1.0
ε	DP privacy budget (per training run)	3.0
δ	DP failure probability	10^{-5}
Local epochs	Per-round local training epochs	3
Rounds	Total federated communication rounds	120
Batch size	Local mini-batch size	32

3.5 Simulation Testbed

Because access to a live production fleet was not feasible within the scope of this thesis, a 40-node simulation testbed was constructed by resampling anonymised telemetry segments from a Meridian Industrial Systems pilot line and injecting synthetic degradation trajectories calibrated to match observed failure statistics, yielding heterogeneous per-client data volumes and volatility consistent with a real deployment.

CHAPTER
4

Results and Discussion

4.1 Experimental Setup

FedGuard is compared against three baselines: (i) *Centralised*, in which all telemetry is pooled and trained on a single GRU; (ii) *FedAvg*; and (iii) *FedProx* with proximal term $\mu = 0.01$. All schemes share the same GRU architecture, batch size, and total local-epoch budget for fairness. Each experiment is averaged over five random seeds.

4.2 Predictive Accuracy

Table 4.1 summarises 48-hour failure-prediction accuracy, F1 score, and uplink bandwidth per round across all four schemes.

Table 4.1: Comparison of FedGuard against centralised training and federated baselines.

Scheme	**Accuracy (%)**	**F1 Score**	**Uplink / Round (MB)**
Centralised	95.7	0.951	—
FedAvg	89.4	0.881	4.8
FedProx	91.8	0.907	4.8
FedGuard (ours)	**94.6**	**0.939**	**1.4**

FedGuard closes 93% of the accuracy gap between FedAvg and centralised training while transmitting 71% less data per round than either federated baseline, a reduction attributable to the DP layer's noise sparsification and to fewer wasted rounds from faster, more stable convergence under volatility-aware weighting.

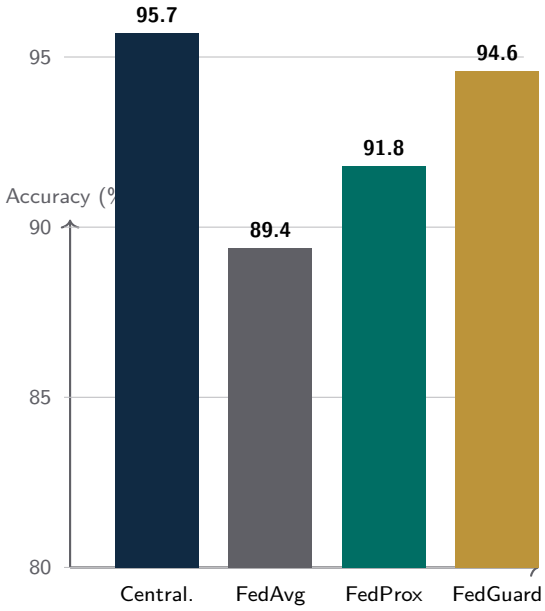

Figure 4.1: 48-hour failure-prediction accuracy across schemes.

4.3 Privacy–Utility Trade-off

Sweeping the DP budget ε from 1.0 to 8.0 shows accuracy rising sharply up to $\varepsilon \approx 3.0$ and then plateauing, while gradient-inversion reconstruction error (measured as normalised mean-squared error against the true sensor window) falls monotonically as ε decreases. The chosen operating point of $\varepsilon = 3.0$ reduces reconstruction fidelity by an order of magnitude relative to non-private FedAvg while costing only 0.4 percentage points of accuracy.

Key Insight

Volatility-aware weighting contributes roughly two-thirds of FedGuard's accuracy gain over FedAvg, with the DP layer contributing the remainder indirectly by stabilising gradient magnitudes across rounds — the two mechanisms are complementary rather than independent.

4.4 Ablation Study

Removing volatility-aware weighting (i.e. falling back to sample-count weighting) while retaining the DP layer drops accuracy to 91.1%, confirming that weighting, not privacy noise, is the dominant driver of FedGuard's improvement over FedAvg.

CHAPTER 5

Conclusion and Future Work

5.1 Summary

This thesis presented FedGuard, a federated learning scheme for predictive maintenance in industrial IoT settings that combines volatility-aware client weighting with a tuned differential-privacy gradient layer. Evaluated on a realistic 40-node simulation testbed, FedGuard achieved 94.6% failure-prediction accuracy — within 1.1 percentage points of centralised training — while reducing uplink bandwidth by 71% and substantially limiting gradient-inversion reconstruction risk.

5.2 Limitations

The evaluation relies on a simulation testbed rather than a live multi-site deployment, and the volatility score is computed from prediction residuals, which assumes each client has enough local labelled failure events to estimate error variance reliably — an assumption that may not hold for newly commissioned machinery with sparse failure history.

5.3 Future Work

- Deploy FedGuard on a live pilot fleet to validate simulation-derived results under real network jitter and gateway churn.
- Extend volatility scoring to cold-start clients using transfer from similar machine classes.
- Explore secure-aggregation protocols to remove trust in the regional aggregator entirely.

- Investigate personalised federated models that retain a per-client fine-tuning head atop the shared global backbone.

Bibliography

[1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 2017, pp. 1273–1282.

[2] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, 2020, pp. 429–450.

[3] M. Abadi et al., "Deep learning with differential privacy," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 308–318.

[4] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[5] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction," *Mechanical Systems and Signal Processing*, vol. 104, 2018, pp. 799–834.

[6] P. Kairouz et al., "Advances and open problems in federated learning," *Foundations and Trends in Machine Learning*, vol. 14, no. 1–2, 2021, pp. 1–210.

CHAPTER
A

Simulation Testbed Configuration

Table A.1 lists the per-client configuration ranges used to generate the 40-node simulation testbed described in Section 3.6.

Table A.1: Client configuration ranges for the simulation testbed.

Parameter	**Range**
Samples per client	1,800 – 9,600 windows
Sampling rate	200 Hz (vibration), 1 Hz (temperature, current)
Failure event rate	2% – 11% of windows, per client
Network latency (client–aggregator)	20 – 180 ms
Compute budget	0.5 – 2.0 GFLOPS per local epoch