Federated TinyML for Lightweight Anomaly Detection in IoMT Using MQTT-Enabled Edge Deployment

Vaibhav Gollapalli
Texas Academy of Mathematics and Science
University of North Texas, USA
Email:vaibhavgollapalli5@gmail.com

Seema G. Aarella
Department of Computer Science
Austin College, USA
Email: saarella@austincollege.edu

Saraju P. Mohanty
Dept. of Computer Science and Engineering
University of North Texas, USA
Email: saraju.mohanty@unt.edu

Elias Kougianos
Dept. of Electrical Engineering
University of North Texas, USA
Email: elias.kougianos@unt.edu

***Abstract*—The rise of the Internet of Medical Things (IoMT) enables continuous health monitoring, but traditional cloud-based Machine Learning (ML) models face challenges in latency, privacy, and reliability, especially in resource-constrained settings. This paper presents a lightweight anomaly detection framework using TinyML models deployed on microcontroller-based edge devices for real-time physiological signal analysis. DHT11 sensors simulate temperature signals and inference is performed locally using an optimized TensorFlow Lite model. Communication with external systems is handled via Message Queuing Telemetry Transport (MQTT) for efficient data logging and visualization. The framework integrates Tiny Federated Learning (TinyFL) for continual model updates, achieving an average accuracy of 99% while significantly reducing latency and energy consumption across the evaluation rounds. This approach offers a low-latency, scalable and secure solution for decentralized IoMT applications.**

***Index Terms*—Machine learning, healthcare, Internet of Medical Things (IoMT), Edge processing, TinyML, MQTT, Data security, and privacy.**

I. Introduction

The rapid evolution of IoMT has transformed the way healthcare services are delivered, enabling continuous real-time monitoring of physiological and environmental parameters using connected intelligent devices. As healthcare shifts from traditional hospital-based systems to ubiquitous, patient-centered models [1], integrating edge intelligence becomes vital to ensure low-latency decision making, bandwidth efficiency, and data privacy in resource-constrained environments [2]. The scope of AI and ML extends to efficient delivery of healthcare services as well [3].

The increasing convergence of IoT and ML technologies has led to the emergence of Tiny Federated Learning(TinyFL). This paradigm enables resource-constrained devices to perform local ML training in an FL system. Unlike traditional ML, TinyFL avoids centralized data storage, reducing latency, communication costs, and privacy risks. TinyFL has been applied across diverse domains such as healthcare and wearable devices, smart agriculture, smart cities, and Industrial IoT(IIoT), Unmanned Aerial Vehicle (UAV), this shows the adaptability of the models to contexts requiring a secure and lightweight mechanism [4].

Figure 1 shows the fundamentals of TinyML and TinyFL, in case on TinyML the model is trained on a central server and runs locally for real-time inference. However, a TinyFl framework follows a distributed on-device training, with the server only coordinating the model aggregation. The TinyFL model is suitable for privacy-preserving secure processing of data as data is not shared across the network [5]. Some of the key challenges of applying machine learning in the healthcare domain include data heterogeneity, system and resource constraints, privacy and security risks, and communication overhead [6]. This research addresses these challenges by employing a distributed training approach using TinyFL in combination with MQTT-based data communication. The primary objective is to develop a lightweight, low-latency, and energy-efficient anomaly detection model suitable for deployment in resource-constrained environments, such as wearable and implantable healthcare devices.

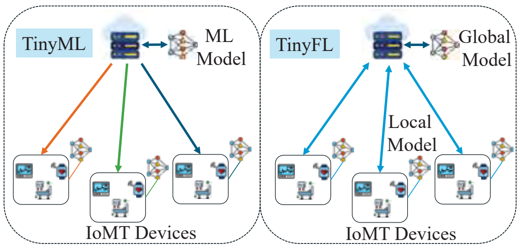

Fig. 1: Overview of TinyML and TinyFL.

II. Related Prior Research

Federated Learning (FL) supports privacy-preserving edge processing by keeping data local, but its integration with edge computing faces challenges such as limited energy, memory, and heterogeneous device capabilities. non-independent and identically distributed (Non-IID) data further complicates model convergence and accuracy. Efficient resource allocation is therefore vital for scalable FL at the edge [7]. Aarella et al. [8] demonstrated collaborative edge-based error correction using compact FL models, while their earlier work [9] explored lightweight ML models for secure authentication in edge environments.

Recent advances in TinyML focus on optimizing model design, compression, deployment, and inference on microcontrollers. Tools such as TensorFlow Lite, Cortex Microcontroller Software Interface Standard - Neural Network (CMSIS-NN), Tensor Virtual Machine (TVM), and Edge Impulse support efficient deployment in constrained devices [10]. TinyOL, a TinyML system for on-device online learning, supports incremental training on streaming data using supervised and unsupervised neural networks [11].

Future directions emphasize robust resource scheduling, adaptive personalization, and energy- and communication-efficient FL strategies. A comparative analysis of TinyFL applications in IoMT and healthcare is presented in Table I.

III. Novel Contributions of the Current Research

This research addresses the above challenges through the design and development of a lightweight and deployable TinyFL-based anomaly detection framework, specifically tailored for physiological monitoring devices in constrained healthcare environments. The key contributions are:

- Integration of anomaly detection into wearable or implantable physiological monitoring workflows, ensuring robust system behavior.
- Minimalist neural network architecture capable of classifying environmental conditions.
- A complete training-to-deployment pipeline including preprocessing, federated model training using the Flower framework, and conversion to TensorFlow Lite (TFLite) format for real-time inference on low-power MCUs such as Arduino Nano 33 BLE or ESP32.

IV. The Proposed Novel Framework

Our framework shown in Figure 2 enables real-time low-power anomaly detection on constrained edge devices using TinyFL and MQTT, tailored for hospital environments. It ensures privacy, on device inference.

The objective of this project is to build a scalable, secure, and intelligent anomaly detection system for hospital rooms using distributed resource-constrained devices.

TABLE I: Comparative Study of TinyFL in IoMT and Healthcare Domains

Paper	Application Domain	Models Used	HW Deploy	Data Type	Frameworks & Metrics
Sita Rani et al. [12]	Secure IoMT systems	Review of FL models in healthcare	No	Literature	Security, latency, model size
Albogamy et al. [13]	HAR in IoMT	LSTM-GRU + attention	Yes	Public HAR data	PyTorch FL; accuracy, F1-score
Ghosh et al. [14]	Elderly monitoring	MLP, CNN, RNN	Yes	Real-time signals	Edge-Fog FL; latency, accuracy
Wu et al. [15]	Home health monitoring	Autoencoder (GCAE)	Yes	HAR traces	Cloud-Edge FL; comm. cost, accuracy
Baghersalimi et al. [16]	Seizure detection	CNN	Yes	Wearable EEG	AWS + FL; privacy, latency
Gupta et al. [17]	Anomaly detection	FedTimeDis-LSTM	Yes	Digital twin data	Edge cloudlets; scalability, accuracy
Current Work	**Anomaly Detection**	**Neural Networks**	**Yes**	**Real-time sensor signals**	**TensorFlow Lite, MQTT, Flower; Accuracy, Energy, Latency**

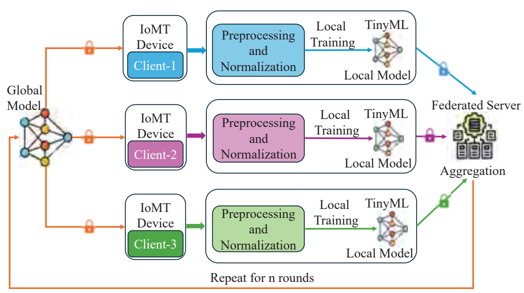

Fig. 2: Proposed Framework for Federated Anomaly Detection.

Edge Data Collection: Each hospital room uses a microcontroller (Arduino) with sensors (DHT11, SpO2) to collect time-series data. The microcontroller preprocesses and normalizes raw inputs locally for compatibility with the deployed TinyML model.

On Device Inference Using TinyML: A pre-trained neural network (using TensorFlow Lite Micro) classifies sensor input as ”normal” or ”anomaly” entirely on device. Alerts are generated and sent through MQTT. Internet/server is not required, guaranteeing robustness in low-bandwidth settings.

Implementation of TinyFL:

TinyML enables fast, low-power decision-making on edge devices but often lacks adaptability to new or evolving data. To address this limitation, the proposed framework integrates Tiny Federated Learning (TinyFL) to enable continual, distributed model updates without compromising data privacy. In this approach, each device stores labeled anomalies locally and performs training on its own data. Rather than sharing raw data, only the model updates are transmitted to a central edge server as shown in the Algorithm 1. The server then performs federated averaging to aggregate updates from multiple devices. Finally, the refined global model is redistributed to all participating devices, allowing them to adapt collectively to new patterns and anomalies.

MQTT Based Communication:

The system uses MQTT, a lightweight protocol, to enable efficient communication between edge devices and servers. Devices publish alerts and local model updates to specific topics and subscribe to global model updates, ensuring timely synchronization and minimal communication overhead through a real-time publish-subscribe mechanism.

Edge Server Aggregation and Monitoring Dashboard: The hospital's edge server hosts the MQTT broker, handles anomalies and model updates, runs federated averaging, redistributes global models, and logs system activity.

Algorithm 1 Federated Anomaly Detection in IoMT Environments

Input: Sensor data (T, H);
Output: Classification label: *normal* or *anomaly*
Step 1: Preprocessing and Feature Extraction
for all client devices **do**
  Acquire (T, H) sensor data
  Normalize features using locally stored parameters
  Encode label: normal $\rightarrow 0, \quad$ anomaly $\rightarrow 1$
  Split dataset into training and validation sets
Step 2: Federated Training (Server-Client Loop)
  for $r = 1$ to R **do**
    Server broadcasts global model weights to clients
    for all selected clients in round r **do**
      Clients update local model
      Clients return updated weights and metrics to server
      Server aggregates weights using FedAvg
      Server updates global model
Step 3: Evaluation and Deployment
      Evaluate final global model using validation data
      Convert model to TensorFlow Lite (TFLite) format
      Deploy model to devices (e.g., Arduino Nano 33 BLE, ESP32)
      Perform real-time inference
      Flag and optionally transmit detected anomalies via MQTT

V. Experimental Setup

To evaluate the proposed framework, we built a prototype in an indoor environment simulating a hospital room. The focus was on TinyML-based real-time inference, anomaly detection with local sensors, and MQTT-based messaging for visualization and logging.

Hardware Components: • Arduino Nano 33 IoT was used with a DHT11 sensor to collect temperature and humidity. It supports TensorFlow Lite Micro and has built-in WiFi. • Raspberry Pi 4 served as the edge server, hosting the MQTT broker and logging anomaly alerts from the Arduino.

Software Ecosystem and Model Deployment: The anomaly detection model, built with TensorFlow Lite and deployed on an Arduino, classifies sensor data as ”normal” or ”anomaly” and publishes alerts using MQTT. A Raspberry Pi logs these alerts and is set up to support future TinyFL by handling model updates and federated averaging. **Algorithm:** The anomaly detection system in this project in this project operates as a continuous processing pipeline rather than a traditional step by step algorithm. Instead of performing a single computation on a fixed input data, the system runs in a loop, handling incoming sensor readings in real time and responding accordingly. This design is more suitable for edge deployment on low power devices where efficiency and responsiveness are critical.

A. *Experimental Results*

The performance of the anomaly detection framework was split into two different stages: offline model training with cross validation, and live deployment using on-device inference and MQTT-based data communication

The model was deployed to an Arduino Nano 33 IoT for real time anomaly detection. The sensor was placed in a regular room, and anomalies were simulated by deliberately messing with the humidity readings, such as blowing warm air onto the sensor to trigger sudden changes. The device published temperature, humidity and anomaly status to MQTT topics using the Paho MQTT client as shown in Figure 3.

The neural network model was trained on a labeled dataset of environmental readings collected from the DHT11 sensor. The final model reached perfect classification on the validation by epoch 6, reaching 100 percent accuracy. The training process was completed in approximately 67 seconds, and the testing time on the entire evaluation set was 0.23 seconds. To validate model generalization, a 5-fold stratified cross-validation was performed. The

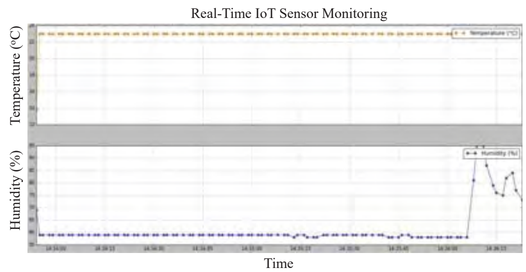

Fig. 3: Visualization of the Real-Time Sensor Data

model maintained very high performance across all folds, with F1-scores of 0.98 or higher in every split. Once training was completed and the model was converted to TensorFlow Lite, The distributed model was trained and deployed using the Flower framework. Federated averaging was used to evaluate the model weights. 5 clients were simulated and the performance across three evaluation rounds was captured. Figure 4 shows the decreasing energy consumption and latency across 3 rounds.

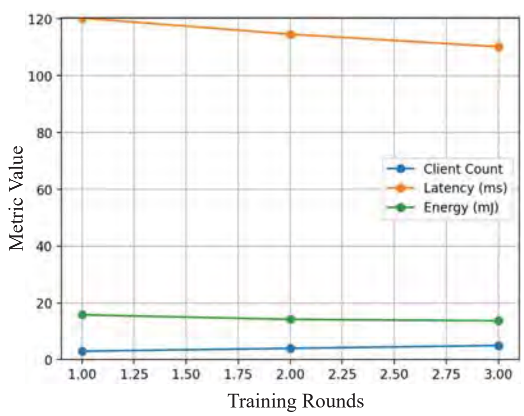

Fig. 4: Variation of Energy and Latency across Various Clients

Figure 5 shows the improvement in prediction accuracy across rounds for 5 federated clients, with a reducing training loss with each round as shown in Figure 6. Table II presents the performance metrics across TinyFL communication rounds, showing improvements in accuracy, latency, and energy efficiency as the number of clients increases. The model achieves up to 100% accuracy by Round 3, with a consistent reduction in latency and energy consumption, demonstrating the scalability and efficiency of the proposed framework.

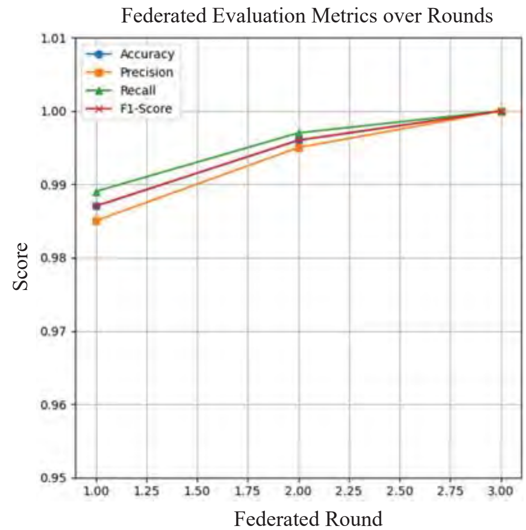

Fig. 5: Evaluation Metrics of the TinyFL Model

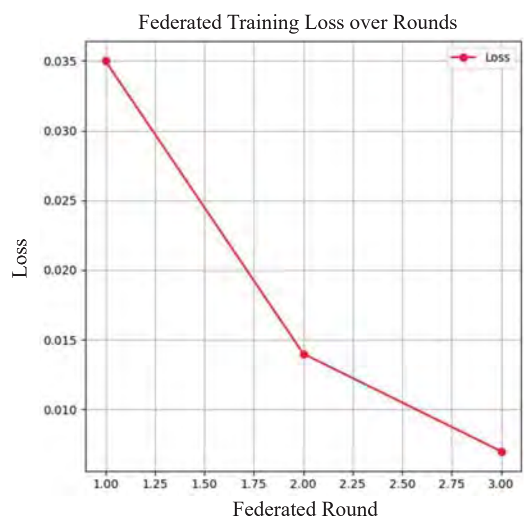

Fig. 6: Federated Training Loss over Training Rounds

VI. Conclusion

This work presents a lightweight, end-to-end anomaly detection framework optimized for

TABLE II: Performance Metrics of TinyFL Rounds

Round	Loss	Accuracy	Precision	Recall	F1-Score	Client Count	Latency (ms)	Energy (mJ)
1	0.035	0.987	0.985	0.989	0.987	3	120.300	15.800
2	0.014	0.996	0.995	0.997	0.996	4	114.600	14.200
3	0.007	1.000	1.000	1.000	1.000	5	110.200	13.700

resource-constrained IoT devices, targeting hospital room monitoring. The TinyML model, trained with a neural network, achieves a training time of 67.15 seconds and an inference time of 0.23 seconds. With TinyFL, the system shows reduced energy consumption and latency as communication rounds and device count increase, while maintaining an average accuracy of 99%. MQTT integration enables efficient real-time communication and visualization. On-device inference enhances data privacy by avoiding raw data transmission. Future work includes security validation, dynamic resource sharing, and adaptability to varying environments.

References

[1] M. M. Baig, H. GholamHosseini, A. A. Moqeem, F. Mirza, and M. Lindén, "A systematic review of wearable patient monitoring systems – current challenges and opportunities for clinical adoption," *Journal of Medical Systems*, vol. 41, no. 7, p. 115, 2017.

[2] A.-M. Rahmani, N. K. Thanigaivelan, T. N. Gia, J. Granados, B. Negash, P. Liljeberg, and H. Tenhunen, "Smart e-health gateway: Bringing intelligence to internet-of-things based ubiquitous healthcare systems," in *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, 2015, pp. 826–834.

[3] B. S. Ingole, V. Ramineni, N. K. Pulipeta, M. J. Kathiriya, M. S. Krishnappa, and V. Jayaram, "The dual impact of artificial intelligence in healthcare: Balancing advancements with ethical and operational challenges," *European Journal of Computer Science and Information Technology*, vol. 12, no. 6, pp. 35–45, 2024.

[4] C. N. da Silva and C. V. S. Prazeres, "Tiny federated learning for constrained sensors: A systematic literature review," *IEEE Sensors Reviews*, vol. 2, pp. 17–31, 2025.

[5] L. Wulfert, C. Wiede, and A. Grabmaier, "Tinyfl: On-device training, communication and aggregation on a microcontroller for federated learning," in *2023 21st IEEE Interregional NEWCAS Conference (NEWCAS)*, 2023, pp. 1–5.

[6] K. K. Yadav and A. Gaurav, "Application and challenges of machine learning in healthcare," *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 9, pp. 458–466, 2023.

[7] S. Ni, Y. He, L. Chen, Y. Wang, and F. Yu, "A survey of edge computing resource allocation strategies based on federated learning," in *2023 International Conference on Networking and Network Applications (NaNA)*, 2023, pp. 116–121.

[8] S. G. Aarella, V. P. Yanambaka, S. P. Mohanty, and E. Kougianos, "Fortified-edge 5.0: Federated learning for secure and reliable puf in authentication systems," in *2024 IFIP/IEEE 32nd International Conference on Very Large Scale Integration (VLSI-SoC)*, 2024, pp. 1–6.

[9] S. G. Aarella, S. P. Mohanty, and E. Kougianos, "Fortified edge 3.0: A lightweight machine learning based approach for security in collaborative edge computing," in *2023 OITS International Conference on Information Technology (OCIT)*, 2023, pp. 450–455.

[10] 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.

[11] H. Ren, D. Anicic, and T. A. Runkler, "Tinyol: Tinyml with online-learning on microcontrollers," in *2021 International Joint Conference on Neural Networks (IJCNN)*, 2021, pp. 1–8.

[12] S. Rani, A. Kataria, S. Kumar, and P. Tiwari, "Federated learning for secure iomt-applications in smart healthcare systems: A comprehensive review," *Knowledge-Based Systems*, vol. 274, p. 110658, 2023.

[13] F. R. Albogamy, "Federated learning for iomt-enhanced human activity recognition with hybrid lstm-gru networks," *Sensors*, vol. 25, no. 3, 2025.

[14] S. Ghosh and S. K. Ghosh, "Feel: Federated learning framework for elderly healthcare using edge-iomt," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 4, pp. 1800–1809, 2023.

[15] Q. Wu, X. Chen, Z. Zhou, and J. Zhang, "Fedhome: Cloud-edge based personalized federated learning for in-home health monitoring," *IEEE Transactions on Mobile Computing*, vol. 21, no. 8, pp. 2818–2832, 2022.

[16] S. Baghersalimi, T. Teijeiro, A. Aminifar, and D. Atienza, "Decentralized federated learning for epileptic seizures detection in low-power wearable systems," *IEEE Transactions on Mobile Computing*, vol. 23, no. 5, pp. 6392–6407, 2024.

[17] D. Gupta, O. Kayode, S. Bhatt, M. Gupta, and A. S. Tosun, "Hierarchical federated learning based anomaly detection using digital twins for smart healthcare," in *2021 IEEE 7th International Conference on Collaboration and Internet Computing (CIC)*, 2021, pp. 16–25.