

An Efficient and Privacy-Preserving Framework for Edge AI Using Federated Learning and Homomorphic Encryption

Sanjivani Zite¹, Prof. P. P. Chavan², Prof. A. J. Shaikh³, Dr. H. B. Jadhav⁴

Student, Computer Engineering¹

Guide, Computer Engineering²

Assistant Professor, Computer Engineering³

HOD, Computer Engineering⁴

Adsul Technical campus, Ahilyanagar, India

Abstract: Edge Artificial Intelligence (Edge AI) supports intelligent processing close to where data is generated. However, training machine-learning models across edge devices is difficult when devices have limited memory, processing power, battery capacity, and network bandwidth. Federated learning (FL) addresses one important concern by allowing devices to train a shared model without sending their raw records to a central repository. Nevertheless, model updates may still expose information, while repeated communication and encryption can increase training time and resource use. This paper reviews an integrated architecture that combines federated learning, homomorphic encryption (HE), client clustering, update pruning, adaptive quantization, and convergence-aware stopping. In the architecture, each client trains locally, removes low-impact update values, compresses the remaining update, and encrypts it before transmission. The server aggregates encrypted updates and coordinates the next training round. The source study reports that its FLHC-QPR approach improves accuracy and reduces latency and memory overhead compared with selected comparison methods on a Human Activity Recognition task. The paper explains the architecture, workflow, potential applications, evaluation measures, limitations, and future research directions. The discussion is based on the supplied research article; performance values are reported from that study and are not presented as newly measured results.

Keywords: Edge AI, Federated Learning, Homomorphic Encryption, Privacy Preservation, Model Pruning, Quantization, Client Clustering.

I. INTRODUCTION

The growth of connected devices has increased the volume of data produced by smartphones, wearable sensors, industrial equipment, and Internet of Things (IoT) systems. Sending all of this information to a central cloud can create network delays, increase bandwidth use, and raise privacy concerns. Edge AI reduces dependence on distant computing resources by bringing parts of the learning and inference process closer to the data source. This is especially useful in applications that require timely responses or handle personal and operational information.

Federated learning is a distributed training approach in which participating devices keep their training data locally and share model updates instead of raw records. A coordinating server combines these updates to improve a global model. Although this reduces the need for centralized data collection, it does not automatically guarantee that the updates are free from privacy risks. Homomorphic encryption can help by allowing selected computations to be performed on encrypted values, so the aggregation server need not inspect the individual updates in plaintext.



A practical edge-learning system must balance several competing goals: predictive quality, privacy, communication cost, computation time, memory use, and participation by devices with different capabilities. This paper describes a combined approach that uses clustering to organize clients, pruning to remove small update values, quantization to reduce numerical precision, and homomorphic encryption to protect updates during transfer and aggregation.

II. PROBLEM STATEMENT AND OBJECTIVES

A. Problem Statement

Conventional centralized training requires data to be collected at a central location, which may be unsuitable for sensitive or bandwidth-limited edge applications. Federated learning reduces raw-data movement, but the repeated exchange of model updates can still consume bandwidth and may reveal information about local training examples. Applying homomorphic encryption strengthens confidentiality, yet encrypted computation often requires additional time, memory, and communication. Consequently, an edge-learning design needs to reduce the cost of exchanging and processing updates without causing an unacceptable decline in model quality.

B. Objectives

- To explain a federated architecture that keeps client training data on the local device.
- To use homomorphic encryption to protect model updates while they are transmitted and aggregated.
- To reduce the size of updates through pruning and adaptive quantization.
- To organize clients into clusters to limit communication pressure on the coordinating server.
- To monitor model convergence and stop unnecessary training rounds when improvement becomes small.
- To assess the design using accuracy, latency, memory overhead, communication cost, and security-related overhead.

C. Scope of the Study

The paper focuses on the architecture and methods described in the supplied source article, including evaluation on the Human Activity Recognition (HAR) dataset. It discusses the relevance of the approach to healthcare, wearable devices, IoT monitoring, and other privacy-sensitive edge settings. The available source does not establish that the same numerical results will hold for every dataset, device type, or deployment environment.

III. EXISTING SYSTEM

In a basic federated-learning setup, a server sends a global model to participating clients. Each client trains the model using its own local data and returns an update. The server aggregates the updates and sends the improved model back to clients for the next round. This process avoids collecting all raw data centrally, but frequent model exchange can be costly when there are many clients or when devices operate over low-bandwidth connections.

If encryption is added without compression or communication optimization, encrypted updates may require additional storage and processing. Devices with different computing capacities may finish local training at different times, and repeated rounds can increase end-to-end latency. A server that receives individual updates also needs a carefully designed security model to limit exposure of client information.

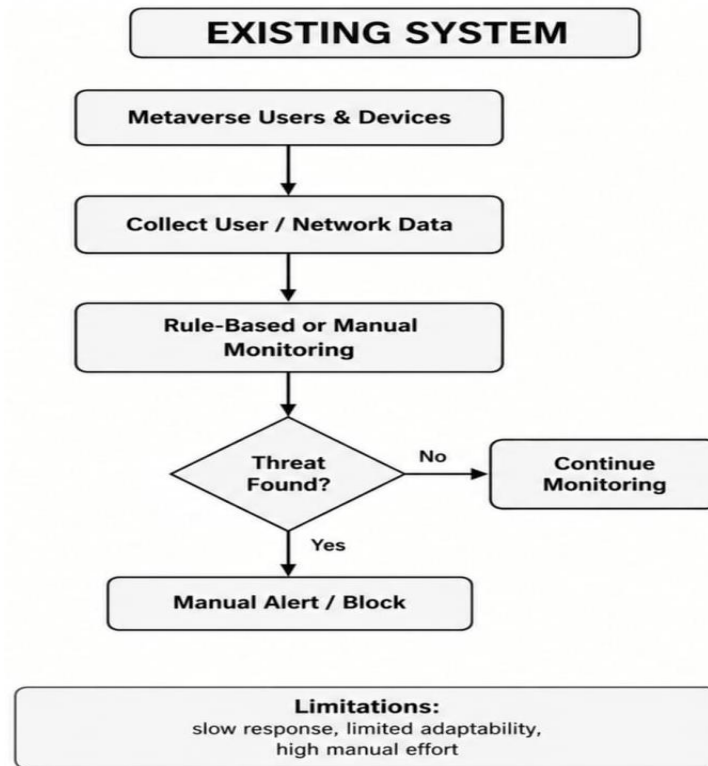
A. Limitations of the Existing Approach

- Frequent client-to-server communication can increase bandwidth consumption.
- Encrypted operations can increase processing time and memory requirements.
- A fixed quantization setting may not suit update values with different distributions.
- Sending many small or low-impact update values can waste communication resources.
- Heterogeneous clients and uneven local data distributions can slow convergence.
- Training may continue for unnecessary rounds if convergence is not monitored effectively.



B. Need for Improvement

These limitations indicate that privacy and efficiency should be addressed together. Encryption alone protects the confidentiality of selected computations but does not necessarily reduce the amount of information transmitted. Likewise, compression alone does not provide cryptographic confidentiality. The design considered here combines both categories of techniques and introduces clustering and early stopping to reduce communication and avoid redundant work.



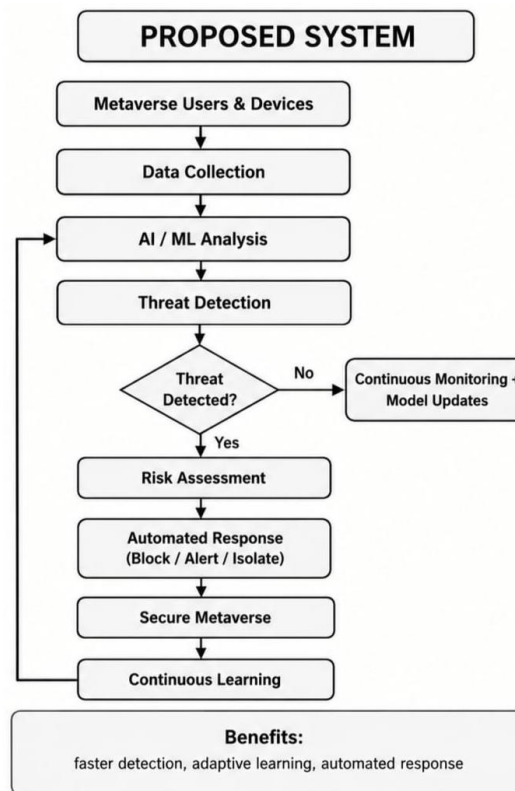
IV. PROPOSED SYSTEM

The proposed architecture follows the FLHC-QPR design described in the source article. It comprises edge clients, a coordinating server, a clustering mechanism, local model optimization, encrypted update aggregation, and convergence monitoring. Client data remain local throughout training. Clients calculate model updates, prune values below a threshold, quantize the remaining values, and encrypt the compressed updates using the CKKS homomorphic-encryption scheme. The server aggregates encrypted updates and coordinates the global model update.

A. Main Components

- Edge clients: Devices that hold local datasets and perform local model training.
- Client clustering: Groups clients to support cluster-level aggregation and reduce communication pressure.
- Pruning and quantization: Reduce the number and precision of transmitted update values.
- Homomorphic encryption: Protects updates so aggregation can be carried out on encrypted values.
- Aggregation server: Combines available updates, evaluates the global model, and distributes the next model.
- Convergence monitor: Tracks loss or improvement across rounds and can trigger early stopping.





V. CHALLENGES AND LIMITATIONS

Although the combined architecture is promising, it does not remove every difficulty associated with federated learning or privacy-preserving computation. The following considerations should be addressed before deployment:

- Encryption cost: Homomorphic encryption can require substantial processing time and memory, particularly on small edge devices.
- Parameter sensitivity: Pruning thresholds and quantization bit widths must be tuned; aggressive settings can damage model accuracy.
- Client heterogeneity: Devices may differ in processing capability, connectivity, battery level, and local data distribution.
- Non-identically distributed data: Different clients may observe different activity patterns or populations, making convergence more difficult.
- Communication interruptions: Unreliable connections can delay participation and complicate synchronization.
- Key management: Secure distribution, storage, rotation, and recovery of cryptographic keys are necessary for a dependable system.
- Threat coverage: Encryption of updates does not by itself prevent poisoning, malicious participation, model inference, denial of service, or compromised endpoints.
- Evaluation scope: Results from one dataset and selected configurations do not prove the same performance in every real-world application.
- Energy consumption: The source article notes that energy use and intermittent connectivity are outside the detailed scope of its evaluation.



- Implementation complexity: Integrating compression, clustering, encryption, and convergence monitoring increases system design and testing effort.

VI. CONCLUSION

This paper has described an edge-learning architecture that combines federated learning and homomorphic encryption with client clustering, update pruning, adaptive quantization, and early stopping. The design aims to preserve local data boundaries while reducing the communication and resource costs associated with repeated model exchange. The supplied source article reports improvements in accuracy, latency, memory use, and security overhead for its FLHC-QPR approach on the tested Human Activity Recognition settings. These findings support further investigation, but they should not be treated as universal guarantees. Real-world deployment requires evaluation on target devices, careful cryptographic configuration, robust handling of heterogeneous clients, and testing against a broader set of security threats.

REFERENCES

- [1] D. Rahbari, M. Danesh talab, and M. Jenehan, "An Efficient Architecture for Edge AI Federated Learning With Homomorphic Encryption," *IEEE Access*, vol. 13, pp. 97919-97929, 2025, doi: 10.1109/ACCESS.2025.3576689.
- [2] N. M. Hijazi, M. Alogaily, M. Guizani, B. Ouni, and F. Karray, "Secure federated learning with fully homomorphic encryption for IoT communications," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 4289-4300, Feb. 2024.
- [3] Q. Xie et al., "Efficient optimization techniques in privacy-preserving federated learning with homomorphic encryption: A brief survey," *IEEE Internet of Things Journal*, vol. 11, no. 14, pp. 24569-24580, Jul. 2024.
- [4] C. Hu and B. Li, "MaskCrypt: Federated learning with selective homomorphic encryption," *IEEE Transactions on Dependable and Secure Computing*, vol. 22, no. 1, pp. 221-233, Jan. 2025.
- [5] Y. Guo et al., "Efficient and privacy-preserving federated learning based on full homomorphic encryption," *arXiv:2403.11519*, 2024.
- [6] L. Zhang, J. Xu, P. Vijayakumar, P. K. Sharma, and U. Ghosh, "Homomorphic encryption-based privacy-preserving federated learning in IoT-enabled healthcare system," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 5, pp. 2864-2880, May 2022.
- [7] Q. Li, R. Cai, and Y. Zhu, "GHPPFL: A privacy-preserving federated learning based on gradient compression and homomorphic encryption in consumer app security," *IEEE Transactions on Consumer Electronics*, early access, Apr. 21, 2025, doi: 10.1109/TCE.2025.3562767.
- [8] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *Proc. International Conference on the Theory and Application of Cryptology and Information Security*, 2017, pp. 409-437.
- [9] Y. Cao, J. Ma, Z. He, and Y. LI, "AP-CFL: Clustered federated learning through dynamic clustering and adaptive participation in heterogeneous IoT," *IEEE Internet of Things Journal*, vol. 12, no. 10, pp. 13671-13682, May 2025.

