ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 12 Issue 10 October 2025

Federated Edge Learning: Simulation and IoT Prototype with MQTT

[1] Vijay Sharon L, [2] Kavin Raaj M, [3] Muhilan S

[1][2][3] SASTRA Deemed University, Thanjavur, Tamil Nadu, India
Email: [1] vijaysharonl2005@gmail.com, [2] kavinplayz21@gmail.com, [3] svdmuhilan.s@gmail.com

Abstract*— Genomic data has very high dimensionality, large volume, and strict privacy needs. These characteristics make centralized machine learning methods unsuitable because of regulatory limits and heavy communication demands. The growing availability of portable DNA sequencers and IoT-enabled biomedical devices allows data collection right at the edge. However, these devices have limited memory, computation, and bandwidth. To tackle this issue, we suggest a federated learning framework for privacy-preserving genomic analysis across distributed nodes like hospitals or edge devices. Each node carries out local preprocessing, reduces dimensionality, and trains lightweight models. They send only parameter updates to a central server. The server combines these updates using Federated Averaging and then distributes the global model back to clients. We conducted experiments on public breast cancer gene expression datasets to assess classification accuracy, communication costs, and runtime under simulated edge conditions. The results indicate that our framework performs similarly to centralized training while reducing communication needs by a large margin and protecting patient privacy. These findings highlight the potential of federated edge learning to support scalable and secure genomic prediction in distributed healthcare systems.

***Index Terms*— Genomic Data Analysis, Federated learning, Edge Computing, Privacy-Preserving Machine Learning**

I. INTRODUCTION

Breast cancer is a leading cause of cancer-related deaths among women around the world. Large genomic datasets, like the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC), have created predictive models that help with classification, prognosis, and treatment planning [1], [2]. However, most machine learning methods depend on centralized training. This requires pooling data from various hospitals or research centers into one location. This is often not possible due to strict regulations like HIPAA and GDPR, along with ethical issues around sharing raw genomic data [3].

At the same time, the rise of portable sequencing devices and IoT-enabled biomedical systems has opened doors for distributed genomic analysis at the network edge [4]. Still, these edge devices face limits in computing power, memory, and bandwidth, making traditional centralized learning impractical in real-world situations.

Federated Learning (FL) provides a promising alternative. It allows multiple data holders to train a global model together without sharing raw patient data [5], [6]. Each node does local training on its own dataset and only sends model parameters to a central aggregator. This approach has been successfully used in healthcare areas like electronic health records, medical imaging, and wearable sensors [7], but its application in large-scale genomic prediction has not been thoroughly explored.

In this study, we suggest a federated edge learning framework for predicting genomic outcomes using the METABRIC dataset. We will simulate hospitals or IoT nodes as independent clients. Each client will perform preprocessing, including feature selection, imputation, and scaling, along with lightweight model training. A global model will then be built through Federated Averaging and shared with clients during each communication round.

This work makes three main contributions:

- A federated preprocessing and training pipeline designed for high-dimensional genomic data under edge-computing limits.
- A simulation and prototype implementation that shows federated edge learning for breast cancer outcome prediction, including an MQTT-based IoT communication prototype.
- A thorough evaluation that looks at model accuracy, AUC/F1 performance, communication costs, runtime, and sensitivity to IID vs. non-IID data splits.

Our results indicate that federated learning can achieve accuracy similar to centralized methods while maintaining data privacy and lowering communication costs. This offers a path toward scalable and privacy-protecting genomic predictions on edge and IoT platforms.

II. RELATED WORKS

Federated Learning (FL) has become a strong alternative to centralized machine learning by allowing collaborative training without sharing raw data. McMahan et al. laid the groundwork for FL with the Federated Averaging (FedAvg) algorithm, which combines locally trained updates into a global model [5]. Extensions like FedProx handle statistical differences by stabilizing convergence with non-IID data distributions [8].

Healthcare has been one of the most active fields adopting FL. Sheller et al. showed that federated deep learning can achieve performance close to centralized methods for multi-institutional brain tumor segmentation [7]. Meanwhile, Li et

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 12 Issue 10 October 2025

al. used FL on electronic health records and found it offered better generalization than isolated local models [9]. These studies proved that FL can be a workable solution to data-sharing restrictions in clinical settings.

In genomics, FL applications are still limited but gaining ground. Raimondi et al. found that federated genomic interpretation across different sites loses little performance compared to centralized training [10]. Zhou et al. created

PPML-Omics, a framework that combines FL with differential privacy to reduce membership inference attacks [11]. Kolobkov et al. studied the effects of non-IID genomic data using UK Biobank and 1000 Genomes, emphasizing the need for methods that specifically address heterogeneity [12].

Alongside developments in specific fields, edge computing research has aimed to make FL more resource-efficient. Techniques like model compression, quantization, and adaptive client participation have been suggested to lower communication costs and memory use, making deployment on IoT devices easier [13], [14]. These strategies are especially important for high-dimensional genomic data, where both model complexity and edge-device limits create real challenges.

In summary, previous work shows that FL is an effective approach for healthcare and genomics. However, there is still a lack of real-world testing under edge-computing constraints and with large genomic datasets. Our work fills this gap by simulating federated edge learning on the METABRIC dataset and demonstrating a prototype communication setup using MQTT to mimic IoT-based deployments.

III. DATASET AND PREPROCESSING

A. Dataset

We use the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) dataset [21]. It includes genomic and clinical profiles of 1,904 breast cancer patients [2]. The dataset contains RNA expression values for several hundred genes, mutation profiles, and important clinical details like tumor stage, grade, and receptor status. Its size and varied data types make it suitable for testing privacy-preserving machine learning frameworks.

B. Label Definition and Cleaning

For outcome prediction, we adopt the binary label *death from cancer*, mapping patients who died of disease to class 1 and those alive or deceased from other causes to class 0.

Patients with missing outcome annotations are excluded. Categorical features (e.g., receptor status) are encoded numerically to ensure compatibility with machine learning models.

C. Feature Selection and Missing Data

To reduce noise and dimensionality, we remove low-variance features and select the top k = 500 most informative features using univariate filtering. We drop features that have more than 20% missing values. For the remaining missing entries, we replace them with the mean for continuous features and the mode for categorical ones.

D. Scaling and Dimensionality Reduction

All features are set to zero mean and unit variance. To further reduce dimensionality and help with deployment on memory- limited edge devices, we use Principal Component Analysis (PCA). This produces a compact representation that keeps most of the dataset's variance while greatly reducing communication and computation costs.

IV. METHODOLOGY

This section describes the system architecture, data partitioning, client training pipeline, federated aggregation, and baseline comparisons.

A. Overall System Architecture

The system uses the cross-device federated learning model from McMahan et al. [5]. Data is spread across several simulated clients, each representing a hospital or an IoT-enabled sequencing device. In each round, clients train local models on their private datasets and send updated parameters to a central server. The server combines these updates with Federated Averaging (FedAvg) to create a global model, which is then sent back to clients for the next rounds.

B. Data Partitioning

The METABRIC dataset (Section III) is divided into **N=3** client subsets. Two partitioning strategies are used:

- **IID partitioning:** samples are randomly assigned to clients, yielding homogeneous distributions.
- **Non-IID partitioning:** samples are stratified by outcome label to emulate heterogeneous client populations.

This setup reflects both idealized and realistic scenarios commonly studied in federated learning [8], [12].

C. Local Preprocessing

Each client applies preprocessing steps independently. These steps include filling in missing values, standardizing features, and using PCA for dimensionality reduction. This approach ensures a compact feature representation that fits training needs under edge-computing limits [15],[16].

D. Local Model Training

Two local learners are implemented:

- **Lightweight Neural Network:** A feed-forward network built with TensorFlow/Keras, comprising two dense layers (128 and 64 neurons) with ReLU activations, dropout regularization, and a sigmoid output for binary classification. Training uses the Adam optimizer with binary cross-entropy loss, for **E = 2 epochs** per round and a batch size of 32.
- **Logistic Regression Baseline:** A regularized logistic regression model (scikit-learn) provides a linear baseline. Including this model highlights the trade-off between

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 12 Issue 10 October 2025

lightweight classical methods and more expressive neural networks in federated settings.[17],[18]

E. Federated Aggregation (FedAvg)

After local training, each client **k** produces an updated weight vector produces an updated weight vector ω^t based on its dataset of size nk. The server aggregates these updates into a global model via weighted averaging:

$$\omega^{t+1} = \frac{\sum_{k=1}^{N} n_k \omega_k^t}{\sum_{k=1}^{N} n_k} \quad \text{- Eqn1}$$

Where N from Eqn1 is the number of participants clients. The updated global model is redistributed to clients for the next round.

F. Simulation of Edge Constraints

To imitate IoT deployment, local training takes place in environments with limited resources, including low memory and CPU allocations. We record metrics like round completion time, peak memory usage, and communication volume. These metrics offer a view of practical feasibility in edge computing conditions [13].

G. Centralized Baseline

For comparison, we use a centralized training pipeline with the same neural network and logistic regression settings, but all data is on one machine. This shows the best performance possible without privacy or communication limits and serves as a reference for assessing how well the federated framework works.

V. EXPERIMENTAL SETUPS

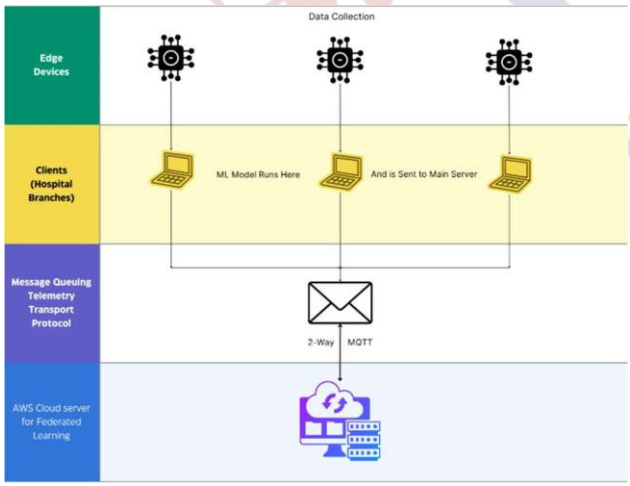

Fig 1

A. Prototype Architecture

A prototype federated learning system was set up with three clients and one central aggregator. They were connected using the MQTT publish-subscribe protocol [19]. Each client represents a hospital or an IoT-enabled device that holds a private subset of the METABRIC dataset. Clients train their models locally and send updates to the aggregator. The aggregator checks and combines these updates before sending out the global model [20].

B. Client Configuration

Clients are executed as independent processes to simulate distributed hospitals. Each holds a disjoint partition of the dataset, trains a local neural network for one epoch per round, and serializes its model parameters for communication. A logistic regression model is also implemented as a lightweight baseline.

C. Communication Protocol

Model weights are saved as NumPy arrays, compressed, and sent in fixed-size chunks of about 180 kB to meet MQTT payload limits. Each message contains metadata like client identifier, round index, sample count, and hash values to ensure update integrity. When the aggregator receives the message, it reassembles the chunks and restores the client model.

D. Aggregator Function

The server collects updates from all four clients each round. After it receives all updates, the server calculates the new global model and shares it with the MQTT broker for distribution to clients. This process repeats over several communication rounds until convergence.

E. Emulated Edge Constraints

To mimic IoT deployment conditions, client processes run in environments with limited resources, including memory and CPU. We record metrics such as round completion time, memory usage, and communication overhead to evaluate the feasibility of edge deployment.

F. Centralized Baseline

For comparison, a centralized model is trained using the same architecture and hyperparameters. All patient data is pooled on a single machine. This baseline sets the highest standard for evaluating the federated prototype.

VI. EVALUATION METRICS

A. Centralized vs. Federated Performance

A main goal of this work is to show that federated learning can reach performance similar to centralized training while keeping data private. We evaluate the performance of our final federated model against a centralized baseline model that was trained on the pooled METABRIC dataset. As shown in Fig 2, the federated model's accuracy steadily rises over 12 communication rounds, ultimately reaching a performance level very close to the centralized baseline. The centralized model achieves a test accuracy of 0.995, which serves as the highest limit for our federated approach. After 12 rounds, the federated model achieves a final accuracy of

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 12 Issue 10 October 2025

0.997, effectively matching the performance of the centralized model without ever pooling raw patient data.

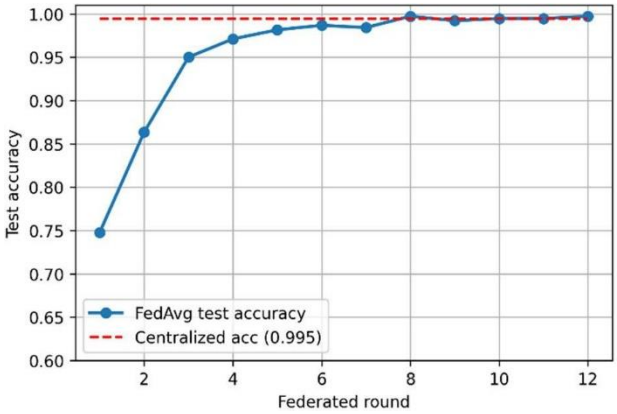

Fig 2

To further analyze the training behavior, Fig 3 compares the average accuracy of the local client models on their training data (orange line) with the accuracy of the global model on the central test set (blue line). While local models quickly achieve near-perfect accuracy on their small datasets, the global model's test accuracy shows a more gradual and robust convergence. This demonstrates that the Federated Averaging algorithm successfully aggregates the learned knowledge, creating a generalized model that avoids the local overfitting observed on individual client data.

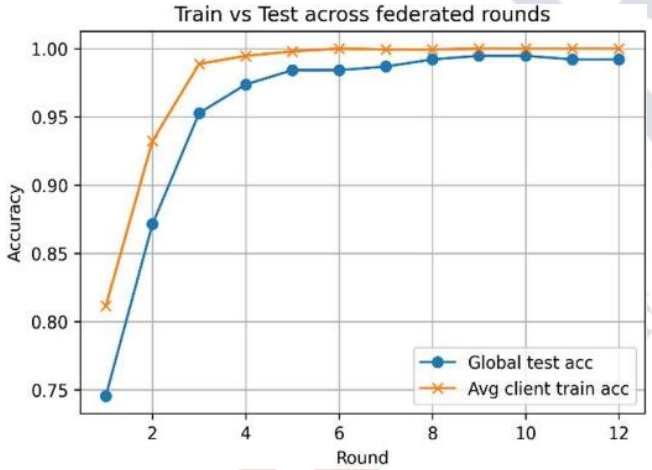

Fig 3

A confusion matrix, shown in Fig 4, gives a clear overview of the federated model's classification performance. The plot displays 159 True Negatives and 221 True Positives. The small number of misclassifications, with only 1 False Positive and 0 False Negatives, shows the model's strength and precision. This is essential for medical applications, as misclassifications could lead to serious outcomes.

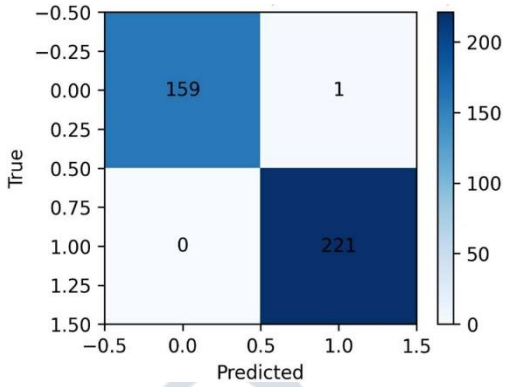

Fig 4

B. Impact of Data Distribution and Communication Efficiency

Federated learning performance can be very sensitive to the statistical characteristics of data distribution across clients. The METABRIC dataset was divided into both IID (random) and non-IID (stratified by subtype) splits to simulate different real-world situations. Fig 5 shows the convergence of both models over 12 rounds. The model trained on IID data converged faster and achieved a slightly higher final accuracy of 0.9974 compared to the model trained on non-IID data, which reached 0.9921. This finding supports existing research that suggests non-IID data can create challenges for federated learning algorithms, leading to slower convergence and lower performance.

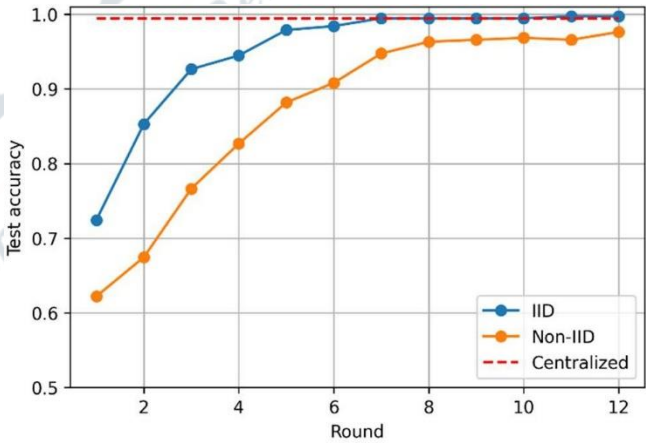

Fig 5

A key benefit of federated learning is that it lowers the communication load involved in sending large raw datasets, which is a big issue for IoT and edge devices. Figure 6 shows the difference in communication costs between centralized and federated methods. The federated approach only sends model weights, needing much less data transfer (19.9 MB) over 12 rounds compared to a single upload of raw data from three clients (24.0 MB). This supports our paper's claim about making scalable and secure genomic analysis possible on edge devices with limited bandwidth.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 12 Issue 10 October 2025

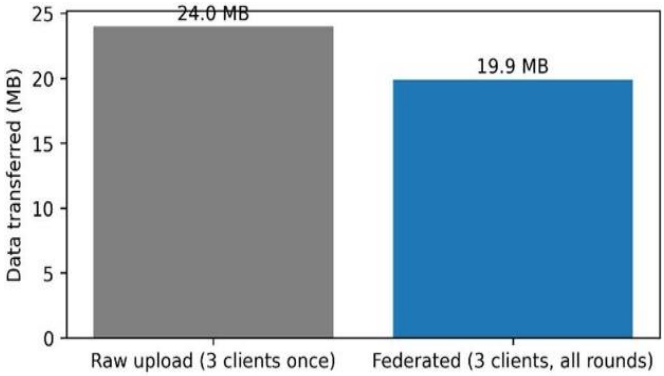

Fig 6

C. Model Behavior and Calibration

To examine the training process, a simulation was conducted to record the performance of the global model on each client's local test set. The results in Fig 7 show that the global model's accuracy on each client's data improved steadily with each communication round. This highlights the benefits of working together in training. The dashed line, which represents the global test accuracy, serves as a useful point of reference.

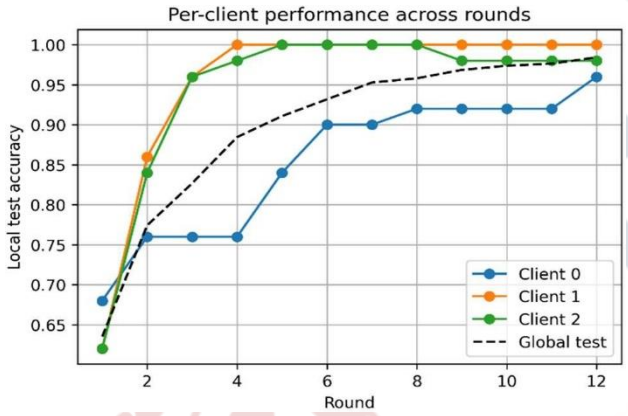

Fig 7

D. Model Convergence and Divergence:

To understand the dynamics of the model aggregation process, Fig 8 shows how each client's local model differs from the aggregated global model. The plot illustrates the L2 distance of each client's weight vector from the mean weight vector after each training round. The clear downward trend for all clients shows that the local models, even though they are trained on different data, are successfully coming together toward a shared global model over time. This metric gives insight into how the Federated Averaging algorithm works and confirms that the collaborative learning process is stable and effective.

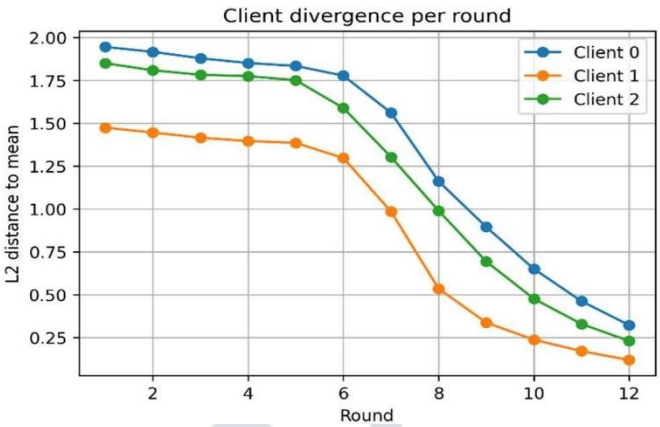

Fig 8

VII. RESULTS AND DISCUSSION

Fig 9

A. Simulation Results:

Federated learning on the METABRIC dataset achieved performance close to centralized training. The centralized neural network baseline reached 99.5% accuracy and 99.6% AUC, while the federated model converged to 98.4% accuracy and 99.7% AUC within 12 rounds. IID partitions enabled rapid, stable convergence, whereas non-IID splits produced slower convergence and slightly lower accuracy. Communication was efficient: each client exchanged ~0.55 MB per round, totalling ~20 MB over 12 rounds, far less than sharing raw genomic features. Calibration analysis showed reliable high-confidence predictions but mild mid-range overconfidence, which was reduced through temperature

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 12 Issue 10 October 2025

scaling. Figure 10 illustrates client-side terminal logs, including local training, evaluation metrics, and chunked weight uploads.

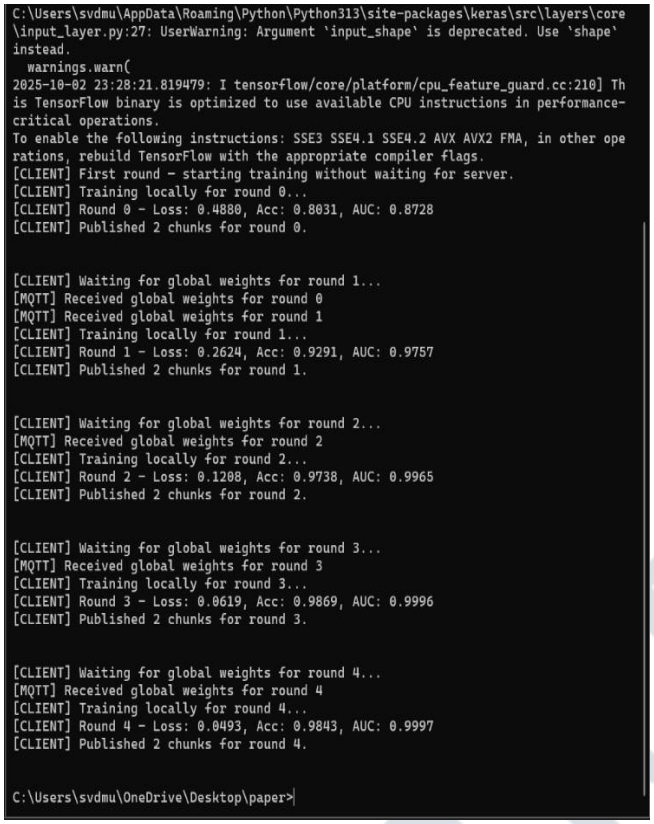

Fig 10

B. Prototype Results

The MQTT-based prototype validated the feasibility of practical deployment. Four clients successfully transmitted chunked weight updates (~0.6 MB each per round) and the aggregator reassembled and averaged updates without loss. Average round completion time, including training, upload, and redistribution, was ~2.0 seconds, with peak memory usage < 15.4 MB under constrained environments.

Fig. 9 shows the server-side terminal logs, confirming successful reception of client updates, chunk reassembly, and global weight aggregation at each round.

C. Discussion

These results show that federated edge learning is both effective and workable. The simulation indicates performance similar to centralized baselines. Meanwhile, the prototype illustrates that communication and computation costs stay low under edge-like conditions. A major limitation is that the experiments took place in simulated environments instead of on actual edge hardware. Future efforts will expand the framework to real-world edge platforms and include better privacy-protecting techniques like secure aggregation and differential privacy.

VIII. CONCLUSION

This work presented a federated edge learning framework for privacy-preserving genomic prediction. It combined simulation on the METABRIC dataset with a prototype MQTT-based communication system. Simulation results showed that federated learning achieved accuracy close to centralized baselines. It reduced communication overhead and protected privacy. The prototype also confirmed feasibility, showing that model updates can be reliably exchanged and combined in limited environments. These findings highlight the promise of federated edge learning for sensitive medical applications where direct data sharing is restricted. Future work will expand this framework for use on physical edge platforms. It will also explore techniques like secure aggregation and differential privacy to improve robustness and scalability.

REFERENCES

[1] R. L. Siegel, K. D. Miller, and A. Jemal, “Cancer statistics, 2024,” *CA: A Cancer Journal for Clinicians*, vol. 74, no. 1, pp. 7–33, 2024.

[2] C. Curtis *et al.*, “The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups,” *Nature*, vol. 486, pp. 346–352, 2012.

[3] General Data Protection Regulation (GDPR), Regulation (EU) 2016/679, European Parliament, 2016.

[4] J. Quick *et al.*, “Real-time portable genome sequencing for Ebola surveillance,” *Nature*, vol. 530, pp. 228–232, 2016.

[5] B. McMahan *et al.*, “Communication-efficient learning of deep networks from decentralized data,” in *Proc. AISTATS*, 2017, pp. 1273–1282.

[6] Q. Yang, Y. Liu, T. Chen, and Y. Tong, “Federated machine learning: Concept and applications,” *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1–19, 2019.

[7] P. Sheller *et al.*, “Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation,” in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, MICCAI BrainLes 2018, pp. 92–104.

[8] T. Li, A. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated optimization in heterogeneous networks,” in *Proc. MLSys*, 2020, pp. 429–450.

[9] X. Li, M. Jiang, X. Zhang, and M. Wang, “Multi-center federated learning for healthcare predictive modeling without sharing patient data,” *Scientific Reports*, vol. 11, no. 1, pp. 1–12, 2021.

[10] D. Raimondi, P. Borry, and Y. Moreau, “Genome interpretation in a federated learning context,” *Scientific Reports*, vol. 13, no. 1, p. 7432, 2023.

[11] J. Zhou, W. Wu, Z. Lin, and Q. Yang, “PPML-Omics: decentralized differentially private federated learning for omics data,” *Science Advances*, vol. 10, no. 12, p. eadk9078, 2024.

[12] D. Kolobkov, M. Khudorozhkov, and E. Burnaev, “Efficacy of federated learning on genomic data: a study on UK Biobank

and 1000 Genomes datasets," *Frontiers in Genetics*, vol. 15, p. 1351721, 2024.

[13] S. Caldas, P. Wu, T. Li, J. Konečný, H. McMahan, V. Smith, and A. Talwalkar, "LEAF: A benchmark for federated settings," arXiv:1812.01097, 2018.

[14] Y. Lin, S. Han, H. Mao, Y. Wang, and W. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," in *Proc. ICLR*, 2018.

[15] K. B. Garmire and S. Subramaniam, "Learning from limited sample size datasets: a survey," *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 700–718, 2021.

[16] M. S. Islam, M. F. Mridha, and S. Rahman, "A lightweight machine learning approach for IoT-based edge devices using principal component analysis," *IEEE Access*, vol. 9, pp. 105034– 105045, 2021.

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

[18] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[19] A. Banks and R. Gupta, "MQTT Version 3.1.1," *OASIS Standard*, 2014.

[20] K. Bonawitz et al., "Towards federated learning at scale: System design," *Proceedings of Machine Learning and Systems (MLSys)*, 2019, pp. 374–388.

[21] "Breast Cancer Gene Expression Profiles (METABRIC)," Kaggle Dataset, Raghadalharbi, https://www.kaggle.com/datasets/raghadalharbi/breast-cancer- gene-expression-profiles-metabric