Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.045020

ARTICLE

Real-Time Prediction Algorithm for Intelligent Edge Networks with Federated Learning-Based Modeling

Seungwoo Kang[1], Seyha Ros[1], Inseok Song[1], Prohim Tam[1], Sa Math[2] and Seokhoon Kim[1,3,*]

[1]Department of Software Convergence, Soonchunhyang University, Asan, 31538, Korea

[2]Department of Telecommunication and Electronic Engineering, Royal University of Phnom Penh, Phnom Penh, 12156, Cambodia

[3]Department of Computer Software Engineering, Soonchunhyang University, Asan, 31538, Korea

*Corresponding Author: Seokhoon Kim. Email: seokhoon@sch.ac.kr

Received: 14 August 2023 Accepted: 27 September 2023 Published: 29 November 2023

ABSTRACT

Intelligent healthcare networks represent a significant component in digital applications, where the requirements hold within quality-of-service (QoS) reliability and safeguarding privacy. This paper addresses these requirements through the integration of enabler paradigms, including federated learning (FL), cloud/edge computing, software-defined/virtualized networking infrastructure, and converged prediction algorithms. The study focuses on achieving reliability and efficiency in real-time prediction models, which depend on the interaction flows and network topology. In response to these challenges, we introduce a modified version of federated logistic regression (FLR) that takes into account convergence latencies and the accuracy of the final FL model within healthcare networks. To establish the FLR framework for mission-critical healthcare applications, we provide a comprehensive workflow in this paper, introducing framework setup, iterative round communications, and model evaluation/deployment. Our optimization process delves into the formulation of loss functions and gradients within the domain of federated optimization, which concludes with the generation of service experience batches for model deployment. To assess the practicality of our approach, we conducted experiments using a hypertension prediction model with data sourced from the 2019 annual dataset (Version 2.0.1) of the Korea Medical Panel Survey. Performance metrics, including end-to-end execution delays, model drop/delivery ratios, and final model accuracies, are captured and compared between the proposed FLR framework and other baseline schemes. Our study offers an FLR framework setup for the enhancement of real-time prediction modeling within intelligent healthcare networks, addressing the critical demands of QoS reliability and privacy preservation.

KEYWORDS

Edge computing; federated logistic regression; intelligent healthcare networks; prediction modeling; privacy-aware and real-time learning

1 Introduction

By enabling the interaction of exchanging model parameters using local on-device computation and global server aggregation, federated learning (FL) presents a collaborative and privacy-preserving framework, which is cooperative for future digital healthcare systems that need assistance from machine learning and deep learning [1,2]. FL has been widely adopted in privacy-sensitive fields like smart healthcare services, where local participant devices such as the Internet of Healthcare Things (IoHT) utilize the sharing-restricted data to compute the models. By keeping patient data locally, healthcare institutions can uphold privacy regulations and maintain the trust of the patients while still benefiting from collective knowledge through scalable parties/organizations gained with collaborative model training. The main functionality phases of FL include 3 primary entities, including the parameter server, edge aggregator node, and IoHT devices. The framework starts by distributing the global model structure and parameters to the local participant, and then after the local computation, the model aggregation between local and edge is executed to assist the resource-constrained IoHT devices before updating to the global parameter server. The active status of local participants from the previous learning iteration requires requesting establishment for next-iteration model learning. The framework iteratively trains the model through multiple round communications until reaching the final convergence points. FL ensures five primary beneficial factors as follows for digital healthcare systems:

- **Healthcare Privacy Protection:** FL allows data to remain on local devices, ensuring internal confidentiality. Since the data is not shared in a centralized manner, the model can be trained on personal data locally without exposure to other parties.
- **Sensitive Data Security:** By keeping data decentralized, FL reduces the risk of data breaches or unauthorized access. The risk of data exposure during transmission or storage is minimized. FL safeguards patient information and protects against potential security vulnerabilities, which is particularly important when dealing with sensitive data such as medical records or financial information.
- **Collaboration and Integration:** FL enables the pooling of knowledge and data resources from different parties and healthcare organizations. Each party can contribute their local knowledge to the model while maintaining data control. The collaborative approach enhances the accuracy and robustness of predictive models, allowing for improved decision-making.
- **Massive Scalability:** As the number of participating institutions increases, the collective dataset grows larger, allowing for more diverse and representative training data. The scalability of FL helps in developing more accurate and generalized models that can be applied to a wider range of healthcare scenarios and populations.
- **Regulatory Compliance:** FL aligns with regulations such as the general data protection regulation (GDPR) and other data privacy and security standards [3,4]. The compliance factor ensures that healthcare institutions can leverage the benefits of FL without compromising legal and ethical obligations related to patient data handling.

In future digital healthcare systems, real-time disease prediction is one of the cutting-edge approaches that need support from the FL architecture [5,6]. However, achieving an accurate final learning model requires attention to several key aspects for sufficient real-time performance metrics, such as communication and computation resource placement, client selection, model aggregation scheduling, and offloading strategies. One of the key challenges in healthcare applications is the need for real-time predictions with high accuracy and reliability. Quality-of-service (QoS) reliability is a vital factor in healthcare because it directly affects the quality and timeliness of healthcare services and decision-making.

Therefore, this paper aims to design a well-formulated objective function with system models of complete FL interactions, termed federated logistic regression (FLR), that can deploy a real-time prediction model in intelligent healthcare networks and ensure maximum reliability with weighted high-quality data contribution of IoHT participants. The integration of FLR in edge networks enhances the accuracy and robustness of predictive models, thereby significantly impacting QoS reliability in healthcare. The key contribution can be summarized as follows:

- We define the final predictive modeling problem by specifying three key components for each healthcare service, including target variables, input variables, and optimization parameters. After specifying the key components, edge aggregation is pairing for optimizing each service following their QoS expectations and upper-bound threshold.
- Our system architecture provides an overview of the communication flow within the FLR system in healthcare networks, spanning three crucial phases, including framework setup, iterative round communications, and model evaluation and deployment. Furthermore, we employ a dataset from the 2019 annual data (Version 2.0.1) of the Korea Medical Panel Survey, a collaboration between the Korea Institute for Health and Social Affairs (KIHASA) and the National Health Insurance Service (NHIS).
- The experiment of our prediction model was conducted on a Mininet testbed to represent the model flows in communication networks, and from another perspective, we focused on the final model on hypertension prediction, considering factors such as age, gender, diabetes status, physical activity, smoking habits, occupation, and education level to develop a comprehensive policy.

2 Related Works

Healthcare institutions deal with highly confidential information related to patient's medical conditions, treatments, and personal details. Therefore, it is crucial to safeguard this data and ensure its privacy protection throughout the learning process. Practical FL needs to be flexible in a resource-constrained and scalable environment, which provides self-organizing capabilities in terms of resource awareness, personalization, incentive awareness, etc. [7–9]. Distributed edge FL plays a vital role in offering the mentioned capabilities by leveraging edge aggregation processes within micro data centers or other access points (equipped computing servers) [10–12]. These edge aggregators act as intermediaries that facilitate the aggregation of locally computed updates without exposing the raw data.

The healthcare domain has witnessed extensive exploration of FL, particularly in real-time medical data processing and applications such as brain tumor segmentation [13,14]. The ability to learn from distributed data sources while preserving privacy has opened up new avenues for improving medical diagnostics and treatment outcomes. FL harnesses the concept of learning over networks, allowing healthcare institutions to collaborate and share knowledge intelligently. Furthermore, bandwidth efficiency is a key aspect, especially when integrating FL with the message queuing telemetry transport (MQTT) protocol. MQTT, known for its lightweight and efficient messaging capabilities, facilitates seamless communication between the central server and local clients. This protocol minimizes the overhead associated with data transmission, enabling faster and more efficient model updates [13].

The integration of real-time medical data processing within the FL architecture has proven to be highly signification. By continuously incorporating subsets of medical data with varying timestamps and conditions, the models can adapt to dynamic healthcare scenarios and improve their diagnostic capabilities [14]. The workflow procedures involve multiple data rounds, where streaming data is collected in each iteration. The model stages leverage the FL paradigm to aggregate and average between the old-timeslot and new-timeslot models, allowing for incremental learning and continuous improvement. Exemplar stages help establish a new set of exemplars based on the previous diagnosis model, contributing to the ongoing refinement and accuracy of the model [14].

The proposed solution in [6] leverages homomorphic encryption to implement logistic regression for vertical FL and model prediction. By designing a privacy-preserving logistic regression training scheme based on homomorphic encryption in vertical FL, the paper contributes to improving security while maintaining an acceptable level of efficiency. The scheme overcomes limitations of existing approaches, such as protecting gradient information during training, avoiding the leakage of party labels, and significantly safeguarding data features of the host. The authors also introduce a multi-party vertical FL framework that eliminates the need for a third-party to address the multi-party logistic regression problem. The proposed framework enables effective model training among multiple participants while ensuring privacy and data protection. By removing the reliance on a trusted third-party coordinator, the proposed framework simplifies the complexity and enhances security.

The aforementioned studies collectively emphasize the critical significance of FL in the context of real-time healthcare services. The ability to leverage distributed data while maintaining privacy and confidentiality enables healthcare networks to unlock new possibilities for improving patient care, medical research, and decision-making processes. However, to harness the full potential of FL in prediction modeling within healthcare systems, it is essential to converge the FL system architecture with logistic regression. Logistic regression is a well-established and interpretable machine learning technique, particularly suited for binary classification tasks common in healthcare, such as disease prediction. Combining FL with logistic regression allows us to leverage the strengths of each approach: (1) FL facilitates collaborative learning from distributed data while preserving privacy, and (2) logistic regression provides transparent and interpretable models crucial for gaining medical professional trust. Therefore, the concept of convergence enables real-time and incremental learning, which is crucial in healthcare where data continually evolves. Logistic regression within FL ensures that models adapt incrementally to changing healthcare scenarios while maintaining accuracy.

3 System Architectures and Models for Federated Logistic Regression

Resource-constrained and mission-critical healthcare environments pose unique challenges to the FL framework, requiring efficient management of communication resources and modification of real-time prediction performance. In this section, the network setup, system architectures, and models for the proposed FLR are presented. Table 1 presents the key notations and its description used in this paper.

Table 1: Definition of symbols

Notations	Descriptions
$N = \{1, 2, \ldots, n\}$	Set of local participants
$T = \{1, 2, \ldots, t\}$	Set of round communications in FLR participant-server
$W_G^T = \left\{w_G^0, w_G^1, \ldots, w_G^t\right\}$	Set of global models in a centralized parameter server
$w_{l(n)}^T = \left\{w_{l(n)}^0, w_{l(n)}^1, \ldots, w_{l(n)}^t\right\}$	Set of local models from participant-n
α	Learning rate
$\omega\left(n, s\right)$	Weighted metric of participant-n contributing to building the model in healthcare service-s

(Continued)

Table 1 (continued)

Notations	Descriptions
$Y_s = \left\{y_s^1,\ y_s^2, \ldots, y_s^i\right\}$	Set of target variables consists of *i*-labels in healthcare modeling of service-*s*
$X_{n(s)} = \left\{x_{n(s)}^1,\ x_{n(s)}^2, \ldots, x_{n(s)}^j\right\}$	Set of input variables consists of *j*-labels from participant-*n* in healthcare modeling of service-*s*
$\tau_s^{\forall n} = \left\{\tau_s^1,\ \tau_s^2, \ldots, \tau_s^n\right\}$	Set of optimization parameters in health modeling of service-*s* from all participants (∀*n*) that jointly trained in that particular round communication
$D_M^n = \left\{d_1^n,\ d_2^n, \ldots, d_m^n\right\}$	Set of feeding data batches in each local participant
$\theta_M^n = \left\{\theta_1^n,\ \theta_2^n, \ldots, \theta_m^n\right\}$	Set of model parameters that outputted by different *m* data batches in each local participant-*n*
$L(.)$	Loss function
$T_{n(t)}^{total}\left(T_{n(t)}^{loc},\ T_{n(t)}^{off},\ T_{n(t)}^{comp}\right)$	Total time completion of local model requirements, including local execution times to obtain θ_m^n, offloading times, and computing times
$\varphi\left(w_{l(n)}^t\right), \beta\left(w_{l(n)}^t\right), f\left(R; w_{l(n)}^t\right)$	Model parameter sizes, required resources, and allocated resources in aggregator node in processing of local models

3.1 Algorithm Objections

The proposed system defined the final predictive modeling problem by determining the 3-tuple information for each healthcare modeling in service-*s*: (1) target variables consist of *i*-labels, (2) input variables consist of *j*-labels from participant-*n*, and (3) optimization parameters for prediction on that particular service labeling from participant-*n*, which are denoted as a set of (1) $Y_s = \left\{y_s^1,\ y_s^2, \ldots, y_s^i\right\}$, (2) $X_{n(s)} = \left\{x_{n(s)}^1,\ x_{n(s)}^2, \ldots, x_{n(s)}^j\right\}$, and (3) $\tau_s^{\forall n} = \left\{\tau_s^1,\ \tau_s^2, \ldots, \tau_s^n\right\}$, respectively. Eq. (1) describes in terms of prediction of target variables $\widehat{y_s^i}$ based on the input variables from all selected participants in that round iteration index, denoted as $\forall n\,(t)$. By using sigmoid function as expressed in Eq. (2), the relation with the parameter vector in FLR can be described in Eq. (3) for prediction from all the training samples in *n*-participants using all the gathered features at point k. With given input features from all participants at each round iteration, the predicted probability of the target variable is formulated.

$$p\left(\widehat{y_s^i} = i\right) = p\left(X_{n(s)} | \forall n\,(t)\right) \tag{1}$$

$$p\left(X_{n(s)} | \forall n\,(t)\right) = \frac{1}{1 + e^{-\tau_s^{\forall n} X_{n(s)}}} \tag{2}$$

$$\sum\nolimits_{n \in \forall n(t)} \sum\nolimits_{k \in K} log\left[\frac{p\left(\widehat{y_s^i} = i \mid x_{n(s)}^k\right)}{1 - \ p\left(\widehat{y_s^i} = i \mid x_{n(s)}^k\right)}\right] = \sum\nolimits_{n \in \forall n(t)} \left[\tau_s^n * X_{n(s)}\right] \tag{3}$$

For input variables from multi-participants, all the features are mostly not complete/matching and consist of null values, which requires a collaborative normalization process from experience feature batch from global server. Healthcare feature normalization is formulated as a problem with the solutions by standard scaling or imputation technique. Edge-assisted data feature filtering can be used to expedite the preprocessing. At the output layer of FLR, sigmoid function is employed, which requires the utilization of optimal $\tau_s^{\forall n}$ parameters with full awareness from all selected participant models. Eqs. (4) to (7) present the flow of formulating parameter valuation as the objective of prediction algorithm learning, which is later used to approximate the output likelihood that resulting the label class of healthcare services. $y_s^i(k)$ represents the actual label value of that FLR-enabled service at point k.

$$f^l\left(\tau_s^{\forall n}\right) = \sum\nolimits_{k \in K} y_s^i(k) \log\left(p\left(\widehat{y_s^i} = i \mid x_{n(s)}^k\right)\right) + \left(1 - y_s^i(k)\right) \log\left(1 - p\left(\widehat{y_s^i} = i \mid x_{n(s)}^k\right)\right) \tag{4}$$

$$= \sum\nolimits_{k \in K} y_s^i(k) \log\left(\frac{1}{1 + e^{-\tau_s^{\forall n} x_{n(s)}^k}}\right) + \left(1 - y_s^i(k)\right) \log\left(\frac{e^{-\tau_s^{\forall n} x_{n(s)}^k}}{1 + e^{-\tau_s^{\forall n} x_{n(s)}^k}}\right) \tag{5}$$

$$= \sum\nolimits_{k \in K} y_s^i(k) \left[\log\left(\frac{1}{1 + e^{-\tau_s^{\forall n} x_{n(s)}^k}}\right) - \log\left(\frac{e^{-\tau_s^{\forall n} x_{n(s)}^k}}{1 + e^{-\tau_s^{\forall n} x_{n(s)}^k}}\right)\right] + \log\left(\frac{e^{-\tau_s^{\forall n} x_{n(s)}^k}}{1 + e^{-\tau_s^{\forall n} x_{n(s)}^k}}\right) \tag{6}$$

$$= \sum\nolimits_{k \in K} y_s^i(k)\, \tau_s^{\forall n} x_{n(s)}^k + \log\left(\frac{1}{1 + e^{\tau_s^{\forall n} x_{n(s)}^k}}\right) \tag{7}$$

The objectives of this proposed prediction algorithm using FLR are expressed in Eqs. (8) and (9) by optimizing the reliability of prediction output measuring by cost function, normalization of collaborative input features per services, and optimization of argument $\tau_s^{\forall n}$ to maximize the log-likelihood of parameter vector $\widehat{\tau_s^{\forall n}}$. Model parameters of participant-n, denoted as $\theta_M^n = \left\{\theta_1^n,\ \theta_2^n, \ldots, \theta_m^n\right\}$, are differed within the same round training by different feeding data batches, represented as $D_M^n = \left\{d_1^n,\ d_2^n, \ldots, d_m^n\right\}$.

$$L\left(\left(\theta_m^n\right)^*\right) = \underset{\theta_m^n}{\arg\min} \left[y_s^i(k) - p\left(\widehat{y_s^i} = i \mid x_{n(s)}^k\right)\right]^2 \tag{8}$$

$$\left(\tau_s^{\forall n}\right)^* = \underset{\tau_s^n}{\arg\max} \sum\nolimits_{n \in \forall n(t)} f^l\left(\tau_s^n\right) \tag{9}$$

3.2 Working Flow

This section introduces the communications flow for FLR systems in healthcare networks, encompassing three key phases: framework setup, iterative round communications, and model evaluation and deployment. Framework setup introduces the involved entities and controlling policies for optimal FLR initialization. Iterative round communications include the global model distribution, local model computation (missing input values handling), local model transmission, secure aggregation and updates, and global model re-distribution in the next round of communication. Finally, model evaluation and deployment cover the efficiency weighting of trained FLR models and decide based on evaluation metrics before whether to re-train or compress for final implementation. This section primarily presents the objective in communication networking perspectives in the execution of FLR for reliable healthcare QoS requirements, which essentially aims for minimizing the latency in constructing the final converged FLR model. Fig. 1 illustrates the interactions between the key entities and execution functions in the FLR system, which describes the overall functionalities formulated in the following sub-sections.

3.2.1 Framework Setup

The collaborative FLR framework involves two essential entities, namely (1) the locally selected participants that compute the local model $w_{l(n)}^t$ by feeding data batches d_m^n, including IoHT, and (2) the global parameter server G that initialize the model $W_G^T = \left\{w_G^0, w_G^1, \ldots, w_G^t\right\}$ for every collaborative local-global round communication. The interactions between these two entities are bound by communication and resource orchestration policies. The participant selection and scheduling policies of joint model training within each round of communications are required to be identified by the central controller including the model distribution and partition strategies among entities.

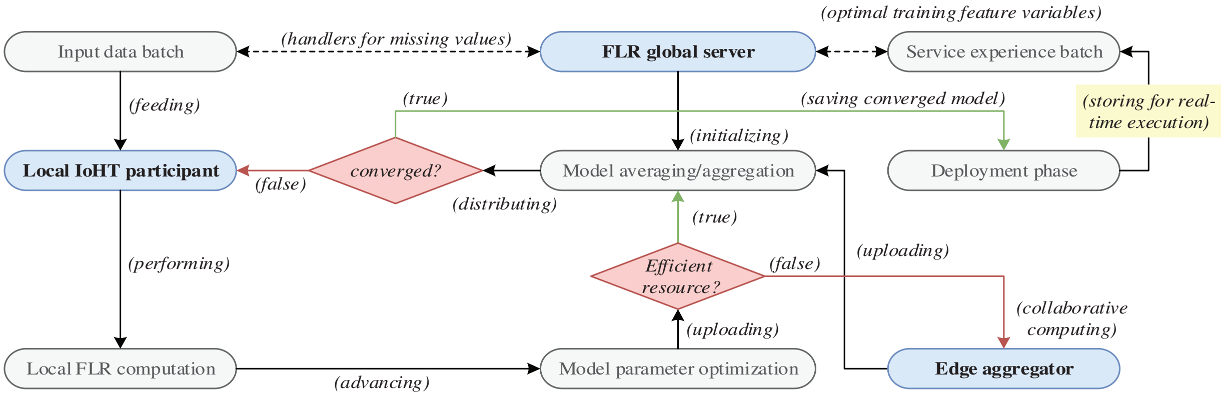

Figure 1: Working flow of the proposed FLR framework

The decision on whether to perform central model aggregation and averaging in the cloud, at the edge, or through a hybrid approach requires adaptability from the controller. In order to accommodate the heterogeneity of the IoHT environment, where participants and services may have varying features and labels, multi-service containers are employed. To be selected as an IoHT participant, certain criteria must be met, including computation capabilities, secure communications, and the ability to provide high-quality data. If these requirements are not met, edge-assisted model computation is employed to aid in the training and computing processes, ensuring effective FL. The setup of the FLR framework includes the following major procedures:

- **Network Topology:** In the context of FLR, network topology refers to the structure of connections among the participating IoHT devices and other nodes, which interact by two topology settings. 1) Centralized topology, where the central server acts as the coordinator of the FLR execution. The participating IoHT devices n communicate with the central server G to exchange model updates $\left(w^{t}_{l(n)},\ w^{t+1}_{G}\right)$ by following the orchestration policies. The coordination mechanism requires a reliable and scalable healthcare system installation rule with a stable connection to the server G. 2) Decentralized topology, where IoHT node n communicates to horizontal FLR with other nodes and alleviates the need for a central server G and distributes the coordination tasks among the devices. Decentralized networks are resilient but may struggle with global feature normalization. The setup relies on factors like system scale and coordination needs, enabling edge-assisted model processing in FLR frameworks.
- **Participant Selection:** The framework controller determines which IoHT devices will participate in communication rounds based on various selection metrics, such as device heterogeneity, healthcare data privacy, connectivity and availability, and representativeness. These metrics ensure that the chosen devices meet specific criteria to ensure the effectiveness of the FLR process for healthcare prediction services.
- **Data Batch Partitioning:** FLR policy settings ensure the completion time of local training to avoid heavy delays on model updates and later affect the global convergence speed. The partition process divides the training data batches into smaller subsets and assigns each subset d^{n}_{m} for feeding to local model training $w^{t}_{l(n)}$. If the training dataset features have class imbalances for

fitting the requirement of healthcare service prediction algorithm, the framework controlling platform essentially ensures a fair distribution of data subsets across selected IoHT devices. The hyperparameter in model structures requires a deep understanding of experienced completion times, which can be listed in terms of batch sizes, number of training epochs, and number of layers.

- **Model Initialization:** w_G^0 serves as the starting point from the global server to distribute in the first index round communication. The pre-trained FLR model is used with initialized model layer structures and hyperparameter settings. The global model is initialized with random parameter values, depending on the availability of pre-trained models, the complexity of the learning task, and the expected convergence latencies for each healthcare service.

3.2.2 Global Model Distribution

The central server fine-tunes the model architectures, hyperparameters, and target use cases as a global learning model during the initial iteration, aiming to distribute it to chosen IoHT participants. When it comes to multi-service IoHT prediction, the diverse models endure the deployment process by considering participant clustering, service prioritization, and aggregation strategies. The models are distributed to selected IoHT participants with matching services and enable the structure for the next phase of local model computation.

3.2.3 Local Model Distribution

Once the chosen IoHT nodes receive the distributed model w_G^0, the nodes utilize the data available, namely d_m^n, at that specific timeslot to train the local prediction models. Eq. (10) presents the local model updating $w_{l(n)}^{t+1}$ relevant to the optimization by loss alleviation. Each local IoHT node possesses distinct data characteristics and continuously collects/generates data at different timeslots. The local data batches d_m^n are fed into the received model w_G^t, initiating the training process for the local prediction model $w_{l(n)}^{t+1}$ with iterative learning. α represents the learning rate. By using gradient optimization algorithm, model parameters are iteratively updated in direction to alleviate the error and biases. Eq. (12) presents the objective of all selected IoHT participant-n is to find the optimal parameter $\theta_m^n(t)^*$ at the next-index round that minimize the cost function. Later, the updated and optimal local parameters are transmitted to the aggregation node.

$$w_{l(n)}^{t+1} \leftarrow w_G^t - \alpha \Delta L \left[w_{l(n)}^t;\ d_m^n(t) \right] \tag{10}$$

$$\theta_m^n(t)^* = \arg\min_{\theta_m^n(t)} L\left(w_{l(n)}^t \right) \tag{11}$$

3.2.4 Local Model Transmission

Once the optimal parameter $\theta_m^n(t)^*$ and the model with the minimal loss is determined, the chosen IoHT participants, as per defined iteration, can proceed to update the model for aggregation in the FLR server. This update process adheres to the specified update scheduling and aggregation policies. In the context of edge-assisted procedures, the diversity of system capabilities and the number of direct round communications are mitigated by leveraging the edge server's capacities for aggregating service labels near the local nodes. The transmission process encompasses various steps, including the initial handshake for client selection, distribution of the global model, training and optimizing local prediction models, scheduling updates for models, and implementing aggregation policies. The proposed FLR is to minimize the completion time $T_{n(t)}^{total}$, expressed in Eqs. (12) to (14), which can be described in wireless networks by the offloading time and edge computation time, denoted as $T_{n(t)}^{off}$, and the local computation time to obtain the optimal parameter in t-round, denoted as $T_{n(t)}^{loc}$. Starting from expressing the uplink data rate $U_{\theta}^{n(t)}$ of uploading the local model parameters, the proposed system formulates the execution time by model parameter sizes, required resources, and allocated resources in aggregator node, denoted as $\varphi\left(w_{l(n)}^{t}\right)$, $\beta\left(w_{l(n)}^{t}\right)$, and $f\left(R; w_{l(n)}^{t}\right)$, respectively. The completion time and key parameters to consider in the phase of local model transmission are mentioned in the equations below:

$$T_{n(t)}^{off} = \varphi\left(w_{l(n)}^{t}\right) / U_{\theta}^{n(t)} \tag{12}$$

$$T_{n(t)}^{comp} = \beta\left(w_{l(n)}^{t}\right) / f\left(R; w_{l(n)}^{t}\right) \tag{13}$$

$$T_{n(t)}^{total} = T_{n(t)}^{loc} + T_{n(t)}^{off} + T_{n(t)}^{comp} \tag{14}$$

3.2.5 *Global Model Aggregation/Updates and Re-Distribution*

In the aggregation process, a weighted metric $\omega(n, s)$ is employed as an influential factor for IoHT participant-n that contribute a larger quantity or higher-quality data features to the FLR systems in service-s. Each participant's contribution is assigned different weight classes to enhance aggregation strategies and prioritize factors critically to achieve an optimized final accuracy. To ensure FLR reliability in specific scenarios such as communication-critical, computation-intensive, or energy-constrained situations, a balancing strategy is applied. Weighted sum models, which consider multiple objectives, provide near-optimal solutions for balancing performance and optimizing the process through iterative iterations. Eq. (15) presents the generic loss optimization in a global server as a part of model parameter aggregation objectives, using each local model w_n^t based on the valuation of parameter estimation $f^l\left(\tau_s^n\right)$, updated optimal model parameter $\theta_m^n(t)^*$, and the weight metric $\omega(n, s)$. The next-round global model w_G^{t+1} is obtained after averaging aggregation, as described in (16).

$$L\left(w_G^t\right) = \frac{1}{\forall n(t)} \sum\nolimits_{n \in \forall n(t)} L\left(w_n^t | .f^l\left(\tau_s^n\right),\ \theta_m^n(t)^*, \omega(n, s)\right) \tag{15}$$

$$w_G^{t+1} = \frac{1}{\sum_{n \in \forall n(t+1)} d_m^n} \sum\nolimits_{n \in \forall n(t+1)} d_m^n w_n^{t+1} \tag{16}$$

This phase of the proposed FLR system aims to achieve several functionalities, including (1) promotes collaboration and knowledge sharing among the IoHT participating devices leading to a collectively improved global model, (2) allows for privacy-preserving learning and adaptive exchanges of model updates, and (3) enables continuous learning and adaptation, as the global model is refined over multiple iterations, capturing weight factors from various IoHT participants and data sources.

3.2.6 *Model Evaluation and Deployment*

Once the FLR process is complete, the system evaluates the performance of the trained global model using appropriate metrics and saves the converged model for the deployment phase, which can be handled using a held-out validation set. If the model is not converged following the expectation metrics of particular healthcare service requirements, false conditions lead to the distribution of the global model to local IoHT participants for next-round communication training.

The optimal trained global model $\left(w_G^t\right)^*$ is stored for making predictions on new unseen data features of assigned healthcare service in real-time, and the proposed FLR system ensures to maintain privacy and handle the missing values of low-quality IoHT data batches by synchronizing the feature normalization module. The procedure of FLR outlines the implementation details, which vary depending on the prediction labels, and the quality-of-service requirements. Fig. 2 presents the flow of evaluation to gather network metrics and FLR objective values before exploiting the service optimal replays. Deployment is primarily presented in software-defined healthcare networking.

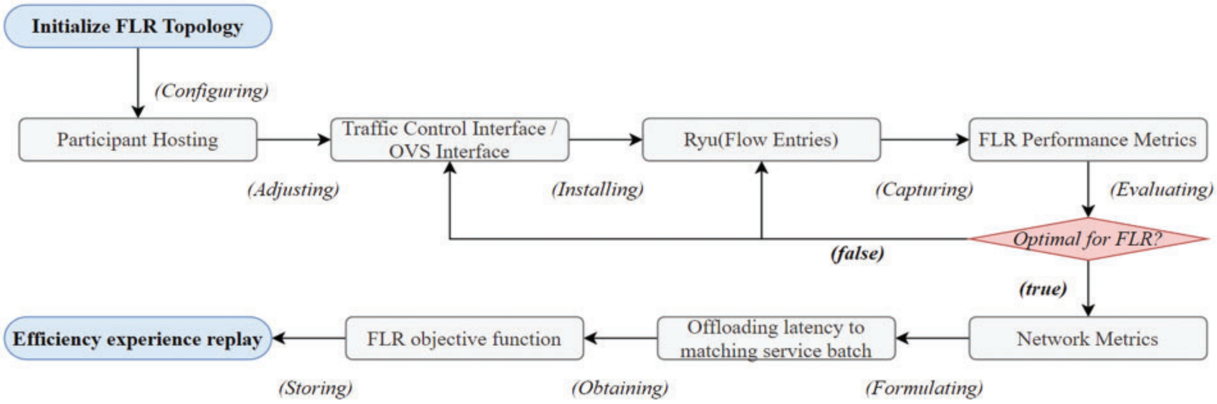

Figure 2: Flow for evaluation and primary deployment execution

4 Performance and Evaluation

4.1 Experiment Setting

The dataset utilized in this research is derived from the 2019 annual data (Version 2.0.1) of the Korea Medical Panel Survey, a collaborative effort between KIHASA and NHIS. The focus of the prediction model developed in this study is on hypertension, incorporating factors such as age, gender, diabetes status, physical activity, smoking status, occupation, and education level. By using this dataset and selected features, we aim to measure and identify the coefficient weights in each category and the high/low capability of current risk analysis, providing a foundation and recommendation for future health improvements. The selection of features is partially influenced by previous studies addressing risk factors related to hypertension prediction [15–17], which highlights the importance of critically examining the performance of existing hypertension risk models.

The proposed FLR method is compared to two baseline approaches that utilize computation-intensive mechanisms to guide the policies of FL networks, termed CI-FL and Conv-FL. Fig. 3 is given as a FLR network topology representation. The employed topology is primarily based on Mininet software-defined networking (SDN) emulator and RYU controller (using Python-based custom scripts) [18–22], utilizing a testbed that supports programmable networking as follows:

- ***CPULimitedHost*** is used to configure the 5 participants (P1 to P5) and 1 global parameter server (G1) with IoHT capacities for simulating resource-constrained hosts with limited CPU capacity. The setting parameter determines the maximum CPU utilization allowed for the host and specifies the 70% of CPU resources that the host is allowed to utilize in the FLR execution.
- ***Open vSwitch*** (OVS) and ***RYU*** adjusts the functionalities and uses *TCIntf* with *OVSIntf* in topology settings and flow entry management. *OVSIntf* offers the creation and management of virtual switches, ports, and flows within the proposed FLR topology. The bridge name, ports, VLANs, flows, and QoS settings are assigned. *TCIntf* configures the control parameters on bandwidth, delay, loss, jitter, txo, rxo, and max_queue_size. The port configurations and other OVS-specific settings are set within *OVSIntf.* This functionality serves as a crucial component in our framework, simulating the behavior and functionality of edge devices in a controlled

environment. While it may not replicate all aspects of real-world devices, it allows us to assess the impact of edge-structured functionality on the FLR framework and evaluate the contribution to real-time aspects.

- ***FL Setting*** partially follows the integration process with TensorFlow-Federated to capture the results and execute the FL aspect. However, this paper extends the contribution by introducing FLR models and IoHT nodes to evaluate the resource-constrained and mission-critical healthcare metrics with real-time prediction models and new dataset distribution as non-IID. The hyperparameters of forecasting and deep learning [23–25], applied for the proposed FLR method are set to 0.01 for the learning rate α, and t-numbers of round communications are set to 2500.
- ***Hosting Server*** in the experiments is equipped with an Intel(R) Xeon(R) Silver 4280 CPU @ 2.10 GHz, 128 GB memory, and an NVIDIA Quadro RTX 4000 GPU.

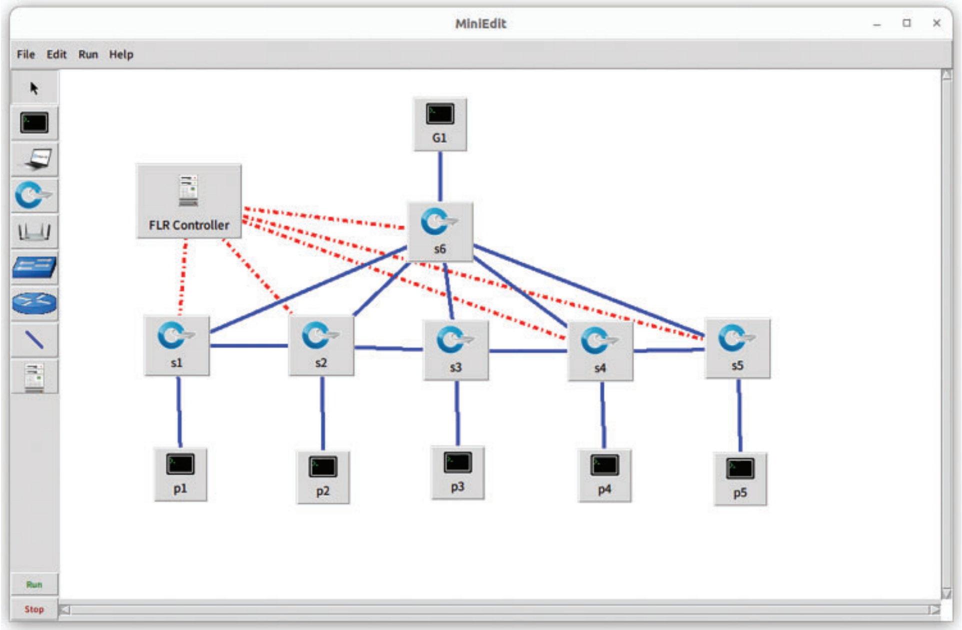

Figure 3: Representation of FLR network topology

4.2 Result and Discussion

Data preprocessing involves the use of an imputation technique, resulting in a reduction of null target classes from 14,741 to 13,834. Then, FLR is employed to determine the essential coefficients. The selected features, descriptions, and coefficients after the final FLR prediction model are given in Table 2. The results of coefficients represent the relationship between the independent variables (features of patient information) and the log-odds of the dependent variable (target variable of hypertension status) being in a particular category. The coefficient value indicates the direction and strength of the association between a specific independent variable from a given patient condition and the log-odds of the target variable. A positive coefficient suggests the value increment of the independent variable is related to the increment of target variable log-odds being in the hypertension status class. Otherwise, a negative coefficient suggests that the value increment of the independent variable is related to the decrement of the target instead. The magnitude of the coefficient indicates the strength of the relationship. Larger coefficient values indicate a stronger association between the independent variable and the log-odds. In other words, a larger coefficient implies a larger impact of the independent variable on the predicted hypertension outcome (0 to 1). Additionally, the coefficient values can be used to interpret the odds ratio associated with each independent variable. The odds ratio is calculated by exponentiating the coefficient value. For example, if the coefficient of physical activity is -0.2326, the odds ratio would be $\exp(-0.2326)$, which represents the multiplicative change in the odds of the target variable for a one-unit increase in the independent variable.

Table 2: Coefficient results of the proposed FLR-based hypertension prediction model

In-dataset features	Descriptions	Coefficient
BIRTH_Y	Birth year	0.0798
CD1_DM	Diabetes conditions (0 or 1)	1.1276
D4	Drinking assessment per year	0.0150
D5	Age of start drinking	−0.0064
ECO2	Employment conditions	−0.0408
EDU	Educational conditions	0.1061
P3	Cost of physical activity	−0.2416
P3_1	Whether to register for an exercise program	−0.0582
P3_2	Whether to use sports facilities	−0.0599
P3_4	Other physical activities	−0.2326
S1	Condition of smoking experience	−0.1748
S2	Age of start smoking	0.0038
S3	Current smoking level (every day, often, quit)	−0.0707
S4	(Every day) number of cigarettes smoked per day	−0.0089
S5	(Often) number of days smoking	−0.0496
S6	(Often) time of smoking	0.00329
SEX	Gender	−0.0028

Overall, the coefficient values from FLR provide insights into the direction, strength, and magnitude of the relationship between the independent variables in the general dataset features and the log-odds of the predicted target variable of hypertension status, allowing for an understanding of the impact of each variable on predicted disease likelihood.

In FLR communication perspective, IoHT nodes in the Mininet network represent resource-constrained devices, and their configuration is designed to exceed the maximum resource threshold. The proposed offloading strategies and fast-convergence FLR aim to highlight the significance of reliable model flows by incorporating congestion and resource limitations. Selected performance metrics include the local model update delivery/drop ratios, end-to-end execution latencies, and final accuracies per round communication. Each metric is presented as follows:

- ***Local model update delivery/drop ratio:*** the delivery ratio refers to the ratio of model updates that are successfully transmitted or delivered from the local IoHT nodes (P1 to P5) to the central server G1 during the FL process. This indicator measures the effectiveness of communication and data transmission between the local devices and the central server. A higher drop ratio indicates a higher rate of unsuccessful transmission, which can be due to network congestion, limited resources, or other computation-intensive modeling issues. A higher delivery ratio suggests a reliable and efficient transmission of local model updates and leads to applicable FL in healthcare.

- ***End-to-end execution latencies:*** this metric measures the time the system takes for the entire round communication to complete, from the initialization to the ending round point, until the model reaches a satisfactory level of accuracy. The end-to-end latency $T^{e2e}_{n(t)}$ includes the time taken for local devices to perform local computations $T^{loc}_{n(t)}$, model updating/offloading $T^{off}_{n(t)} + T^{queue}_{n(t)}$, controlling policy delay $T^{c}_{n(t)}$, and time for the central server to aggregate and update the global model, as $T^{comp}_{n(t)}$. Lower execution latencies indicate faster and more efficient FL, allowing for quicker model convergence and more timely updates. Eq. (17) is given to illustrate this metric.

$$T^{e2e}_{n(t)} = T^{loc}_{n(t)} + T^{off}_{n(t)} + T^{queue}_{n(t)} + T^{c}_{n(t)} + T^{comp}_{n(t)} \tag{17}$$

- ***Final model accuracies:*** to evaluate the final FL model performance in predicting the target hypertension condition, the training and testing phases are captured for accuracy measurement. Higher model accuracies indicate a better predictive performance and a more reliable model. Assessing the final model accuracies helps evaluate the effectiveness of the FL approach in improving the predictive performance of healthcare.

In FLR architectures, the consideration of local-global model communications leads to the possible drop of local model updates based on simulated resource efficiency and offloading scheduling, which severely affects the reliability of the system, particularly in real-time healthcare applications. The development of optimal prediction modeling and network topology setup have to be balanced for practical systems in real-world scenarios that can be highly congested, communicating/computing resource constraints, multi-personalized systems, or energy limitations. Fig. 4a presents the results on delivery ratios within 270 s of simulation throughout 3 different congestion states, from 30 to 270. The results show an average output of 99.96% for the proposed FLR, which is 0.06% and 0.1% higher than CI-FL and Conv-FL, respectively. The primary reason of CI-FL for aiming high delivery ratio is the target setting on resource-intensive computations. CI-FL prioritized delivering updates from resource-efficient nodes or with less computational load. While CI-FL ensures efficient communication, it leads to a high delivery ratio by favoring nodes that can handle computation-intensive tasks effectively. However, delivery efficiency comes at the expense of scalability and adaptability to resource-constrained devices, which may not perform well in real-time healthcare scenarios. Conv-FL demonstrates a high delivery ratio, but it appears to perform the least efficiently among the three approaches. The reason behind the delivery ratio of Conv-FL is attributed to its conventional communication strategies. Conv-FL is not adapted as well to network congestion or resource limitations. Consequently, while it maintains a high delivery ratio, it might be less suitable for real-time healthcare applications due to longer communication latencies. In the context of FL model communications, a high delivery ratio refers to a high proportion of successfully transmitted or delivered model updates, which indicates that the majority of the model updates sent by the local devices have reached the intended destination for aggregation/averaging without being lost or dropped during transmission. A high delivery ratio is desirable and achieved in the proposed FLR because the controlling policies and weight $\omega(n, s)$ placement signifies the high-quality contribution of different nodes and sets the priority level in serving resources. The missing features can be adapted following the high-weight nodes, which enhances the final predictive learning model. This result of efficient FLR in delivery ratio also comes from robust network infrastructure and connectivity, efficient data transmission protocols, optimized resource allocation, and effective loss/error mechanisms for logistic regression integration. By achieving the maximization of the delivery ratio, the proposed FLR can leverage a larger volume of diverse and distributed data from the local IoHT nodes, leading to improved global model accuracy and generalization. Contrary to delivery ratios, the proposed FLR, CI-FL, and Conv-FL reached the average drop ratios of 0.04%, 0.1%, and 0.14%, respectively, illustrated in Fig. 4b.

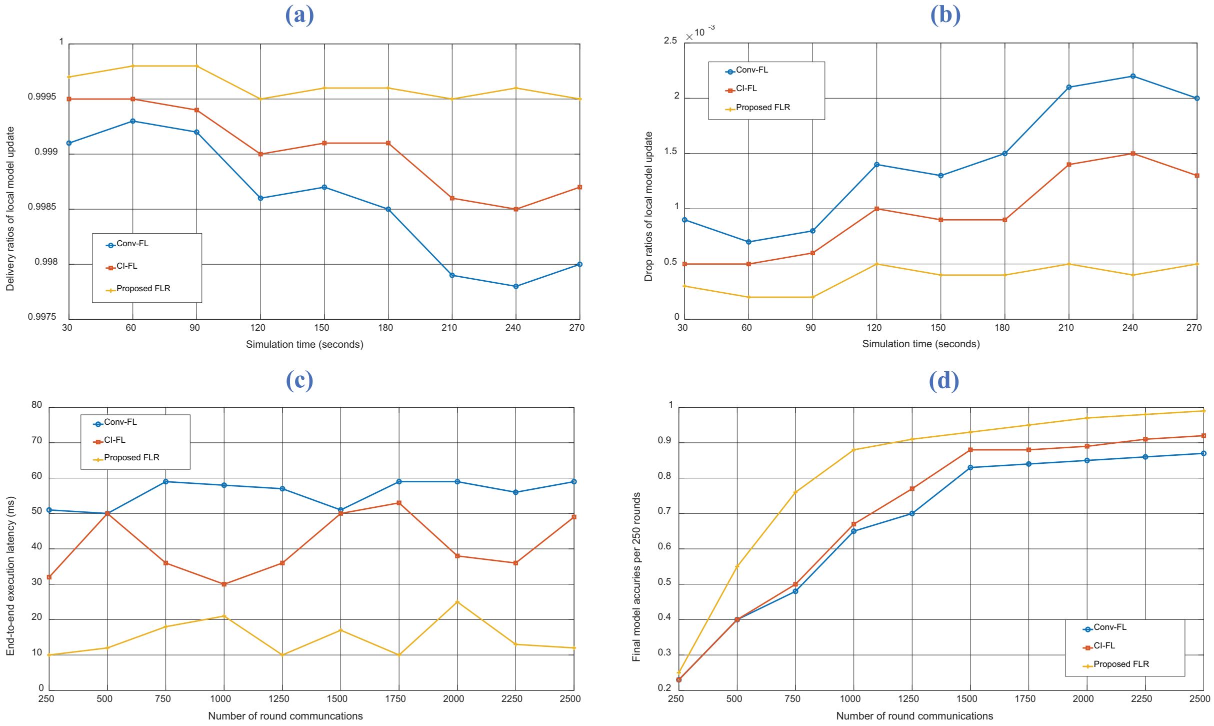

Figure 4: Results of proposed and baseline schemes on (a) delivery ratios of local model update, (b) drop ratios of local model update, (c) end-to-end execution latency, and (d) final model accuracies

The end-to-end execution latency is illustrated in Fig. 4c, which determines the total time for the FL operation to finish within each round of communication. The proposed scheme achieved an average latency of 14.8 ms, which is 26.2 and 41.1 ms faster than CI-FL and Conv-FL, respectively. This low latency output can be attributed to several factors. Firstly, FLR employs efficient controlling policies and weight placement (ω) strategies. The controlling measurement optimized communication between local devices and the central server. Secondly, the emphasis of FLR on high-quality data contribution from different nodes ensures that communication is streamlined, and data transmission is efficient. Thirdly, robust network infrastructure and efficient data transmission protocols play a role in reducing latency. Overall, FLR is designed to perform communication and computation tasks efficiently, allowing for rapid progress in each round of iteration. For CI-FL, it obtains higher end-to-end latency, which indicates that its resource-intensive computation methods contribute to longer communication times. While CI-FL excels in computational tasks, this approach might introduce latency when transmitting updates between nodes and the central server, which is due to the heavy computational load on resource-constrained devices, leading to slower communication. For Conv-FL, it also experiences relatively high end-to-end latency, which is attributed to its reliance on traditional communication methods that are not as efficient as the communication infrastructure employed by FLR. While Conv-FL maintains reliability in data transmission, it obtained results in longer latencies.

Low end-to-end execution latency in completing FL model communications in each round iteration refers to the short amount of time it takes for the entire process of transmitting and updating the model between the local devices and the central server to occur. This metric improvement indicates that the communication and computational tasks involved in healthcare FL systems are performed efficiently, enabling rapid progress in each round of iteration. The proposed FLR allows a better connection between healthcare entities for a seamless and fast exchange of information, facilitating the aggregation and integration of the local models into the global prediction model. The proposed FLR is particularly significant in real-time applications of healthcare networks, where the availability of up-to-date models is crucial for timely decision-making and accurate predictions.

Fig. 4d shows the final model accuracies per 250 rounds of the proposed and baseline schemes. The proposed FLR reached the final accuracy at the 2500-th round with 99.97%, which is attributed to the efficient communication and aggregation strategies. The baseline schemes, namely CI-FL and Conv-FL, can only reach 92.92% and 87.09% only for this hypertension prediction problem, which is not efficient enough. The resource-intensive computation of CI-FL contributed to lower accuracy compared to FLR, as it could prioritize computation over edge placement and aggregation methods. For Conv-FL, it obtained less efficient data exchange and model aggregation. Conv-FL struggled to adapt to the evolving data landscape and real-time healthcare demands, which resulted in reduced accuracy. These results highlight the effectiveness of the proposed FLR architecture in terms of accuracy, convergence time, and resource utilization in the restricted healthcare environment.

5 Conclusion and Future Works

This paper aims to minimize the latency in constructing the final converged FLR model while ensuring reliable healthcare QoS requirements. The interactions between key entities and execution functions in the FLR system were illustrated, highlighting the functionalities and the topology deployment phase. The results demonstrated the effectiveness of the proposed FLR architecture in terms of drop/delivery ratios of local model update, end-to-end execution latency, and final model accuracies. The proposed scheme achieved significantly lower latency compared to baseline schemes, with an average latency of 14.8 ms. We indicated that the communication and computational tasks involved in healthcare FL systems are performed efficiently by enabling rapid progress in each round of iteration. The proposed FLR architecture facilitated a seamless and fast exchange of information allowing for the aggregation and integration of local models into a reliable and real-time global prediction model. Furthermore, the proposed FLR architecture achieved better accuracy compared to baseline schemes. The final accuracy of the proposed FLR reached 99.97% after 2500 rounds, surpassing the baseline schemes, which achieved 92.92% and 87.09% for this particular hypertension prediction dataset. Overall, the proposed scheme offered beneficial factors in healthcare networks in terms of latency reduction, improved accuracy, and efficient utilization of resources. The integration of FLR into healthcare networks has the potential to enhance real-time prediction-based applications, ensuring timely decision-making and accurate modeling.

In future studies, the testbed platform for federated servers will be further developed to integrate our systems to serve more deep learning-based modeling in IoHT services. Furthermore, we will deploy fine-tuning edge aggregator components to better emulate real-world edge devices and patterns within healthcare networks. Performance metrics on (1) convergence speed, (2) resource consumption, and (3) energy consumption will be optimized in further study as a joint reward function in deep reinforcement learning agents. Autonomy and long-term sufficiency will be appended to the FLR framework.

Acknowledgement: Not applicable.

Funding Statement: This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2022-00167197, Development of Intelligent 5G/6G Infrastructure Technology for the Smart City), in part by the National Research Foundation of Korea (NRF), Ministry of Education, through Basic Science Research Program under Grant NRF-2020R1I1A3066543, in part by BK21 FOUR (Fostering Outstanding Universities for Research) under Grant 5199990914048, and in part by the Soonchunhyang University Research Fund.

Author Contributions: Study conception and design: S. Kang, S. Ros, S. Kim; data collection: I. Song, P. Tam; analysis and interpretation of results: S. Kang, I. Song, S. Kim; draft manuscript preparation: S. Kang. S. Ros, I. Song, P. Tam, S. Math, S. Kim. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. H. Brendan, E. Moore, D. Ramage, S. Hampson and A. B. Agüera, “Communication-efficient learning of deep networks from decentralized data,” arXiv:1602.05629, 2016.

[2] F. Cremonesi, V. Planat, V. Kalokyri, H. Kondylakis, T. Sanavia *et al.,* “The need for multimodal health data modeling: A practical approach for a federated-learning healthcare platform,” *Journal of Biomedical Informatics*, vol. 141, pp. 104338–104338, 2023.

[3] N. Truong, K. Sun, S. Wang, F. Guitton and Y. Guo, “Privacy preservation in federated learning: Insights from the gDPR perspective,” arXiv:2011.05411, 2020.

[4] A. Brauneck, L. Schmahorst, M. Majdabadi, M. Bakhtiari, U. Völker *et al.,* “Federated machine learning in data-protection-compliant research,” *Nature Machine Intelligence*, vol. 5, no. 1, pp. 2–4, 2023.

[5] N. Rieke, J. Hancox, W. Li, F. Milletari, H. Roth *et al.,* “The future of digital health with federated learning,” arXiv:2003.08119, 2020.

[6] D. He, R. Du, S. Zhu, M. Zhang, K. Laing *et al.,* “Secure logistic regression for vertical federated learning,” *IEEE Internet Computing*, vol. 26, no. 2, pp. 61–68, 2022.

[7] M. G. Crowson, D. Moukhheiber, A. Arévalo, B. Lam, S. Mantena *et al.,* “A systematic review of federated learning applications for biomedical data,” *PLoS Digital Health*, vol. 1, no. 5, pp. e0000033, 2022.

[8] A. Z. Tan, H. Yu, L. Cui and Q. Yang, “Towards personalized federated learning,” arXiv:2103.00710, 2021.

[9] S. Ros, P. Tam and S. Kim, “Modified deep reinforcement learning agent for dynamic resource placement in IoT network slicing,” *Journal of Internet Computing and Services*, vol. 23, no. 5, pp. 17–23, 2022.

[10] L. U. Khan, S. Pandey, N. Tran, W. Saad, Z. Han *et al.,* “Federated learning for edge networks: Resource optimization and incentive mechanism,” *IEEE Communications Magazine*, vol. 58, no. 10, pp. 88–93, 2020.

[11] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen *et al.,* “In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning,” *IEEE Network*, vol. 33, no. 5, pp. 156–165, 2019.

[12] Y. Ye, S. Li, F. Liu, Y. Tang and W. Hu, “EdgeFed: Optimized federated learning based on edge computing,” *IEEE Access*, vol. 8, pp. 209191–209198, 2020.

[13] B. Camajori Tedeschini, S. Savazzi, R. Stoklasa, L. Barbieri, L. Stathopoilos *et al.,* "Decentralized federated learning for healthcare networks: A case study on tumor segmentation," *IEEE Access*, vol. 10, pp. 8693–8708, 2022.

[14] K. Guo, T. Chen, S. Ren, N. Li, M. Hu *et al.,* "Federated learning empowered real-time medical data processing method for smart healthcare," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pp. 1–12, 2022.

[15] J. B. Echouffo-Tcheugui, G. D. Batty, M. Kivimäki and A. P. Kengne, "Risk models to predict hypertension: A systematic review," *PLoS One*, vol. 8, no. 7, pp. e67370, 2013.

[16] L. A. AlKaabi, L. S. Ahmed, M. F. Al Attiyah and M. E. Abdel-Rahman, "Predicting hypertension using machine learning: Findings from Qatar biobank study," *PLoS One*, vol. 15, no. 10, pp. e0240370, 2020.

[17] M. Alotaibi and M. A. Uddin, "Effectiveness of e-health systems in improving hypertension management and awareness: A systematic review," *KSII Transactions on Internet and Information Systems*, vol. 16, no. 1, pp. 173–187, 2022.

[18] B. Lantz, B. Heller and N. McKeown, "A network in a laptop," in *Proc. of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, Monterey, California, pp. 1–6, 2010.

[19] "Ryu controller," [Online]. Available: https://github.com/faucetsdn/ryu

[20] Y. Ren, A. Guo and C. Song, "Multi-slice joint task offloading and resource allocation scheme for massive MIMO enabled network," *KSII Transactions on Internet and Information Systems*, vol. 17, no. 3, pp. 794–815, 2023.

[21] Y. Zhu, C. Liu, Y. Zhang and W. You, "Research on 5G core network trust model based on NF interaction behavior," *KSII Transactions on Internet and Information Systems*, vol. 16, no. 10, pp. 3333–3354, 2022.

[22] R. L. S. de Oliveira, C. M. Schweitzer, A. A. Shinoda and L. R. Prete, "Using mininet for emulation and prototyping software-defined networks," in *2014 IEEE Colombian Conf. on Communications and Computing (COLCOM)*, Bogota, Colombia, pp. 1–6, 2014.

[23] A. Parizad and C. Hatziadoniu, "Deep learning algorithms and parallel distributed computing techniques for high-resolution load forecasting applying hyperparameter optimization," *IEEE Systems Journal*, vol. 16, no. 3, pp. 3758–3769, 2022.

[24] K. E. Hoque and H. Aljamaan, "Impact of hyperparameter tuning on machine learning models in stock price forecasting," *IEEE Access*, vol. 9, pp. 163815–163830, 2021.

[25] G. Peter and M. Matskevichus, "Hyperparameters tuning for machine learning models for time series forecasting," in *2019 Sixth Int. Conf. on Social Networks Analysis, Management and Security (SNAMS)*, Granada, Spain, pp. 328–332, 2019.