Journal of Computer Science

Original Research Paper

Hybrid Machine Learning Framework with Data Analytics Model for Privacy-Preserved Intelligent Predictive Maintenance in Healthcare IoT

Arun Ganji, D. Usha and P.S. Rajakumar

Computer Science and Engineering, Dr.M.G.R. Educational and Research Institute, Chennai, Tamilnadu, India

Article history
Received: 18-06-2024
Revised: 10-08-2024
Accepted: 05-09-2024

Corresponding Author:
Arun Ganji
Computer Science and Engineering, Dr.M.G.R. Educational and Research Institute, Chennai, Tamilnadu, India
Email: arun.ganji@gmail.com

Abstract: Federated Learning (FL) is a cutting-edge approach for developing machine learning (ML) models using distributed datasets while preserving data privacy and ownership. FL is particularly suited for Internet of Things (IoT) networks due to its decentralized nature, which supports In-Edge AI and maintains data locality. However, FL's complexity poses challenges in analyzing system health, making it crucial to develop robust strategies for monitoring and evaluation. This research introduces a hybrid machine learning architecture that combines FL with the Adaptive Moving Window Regression (AMWR) technique. Specifically, we employ Federated Learning with Dynamic Regularization (FedDyn), where model architecture and training configurations are established centrally and disseminated to clients, who contribute to the model while ensuring differential privacy. This approach termed Federated Learning with Dynamic Regularized Adaptive Moving Window Regression (FedDyn AMWR), demonstrates significant improvements in system reliability, availability, maintainability, and safety. Experimental comparisons with existing methods show that FedDyn AMWR offers substantial advantages in accuracy, computational efficiency, and security, making it a promising solution for complex multi-object systems' health management and maintenance strategies in IoT-based healthcare.

Keywords: Internet of Things (IoT), Federated Learning, Privacy, Windowing, Regression, Perceptron

Introduction

The term "Internet of Things" (IoT) refers to a network of interconnected devices, such as sensors, mobile phones, and actuators, that collaborate to achieve specific objectives (Atzori *et al*., 2010). In recent years, IoT has become increasingly significant in healthcare, enabling continuous patient monitoring through internet-connected sensors and wearable smart devices, including blood pressure monitors, heart rate sensors, and ECG sensors (Islam *et al*., 2015; Ge *et al*., 2020a; Mukhopadhyay, 2015). These advancements facilitate early diagnosis and timely treatment of patients.

Despite the benefits, IoT-based healthcare systems pose significant security and privacy risks, as they involve the transmission and collection of personal data over open networks (Demuynck and De Decker 2005). Privacy concerns are particularly critical, given that personal health information is highly sensitive and its exposure can have severe physical, psychological, and financial repercussions (Andrew and Karthikeyan 2021; Agrawal and Srikant 2000; Theoharidou *et al*., 2017). For instance, the disclosure of a patient's cancer diagnosis to an insurance company or employer can adversely affect their livelihood.

Several laws and regulations are in place to protect personal information, but ensuring data privacy in IoT-driven healthcare remains a formidable challenge (Xue *et al*., 2011 Fung *et al*., 2010; HHS.gov. 2020; Hustinx, 2017). The data collection process typically involves third-party service providers, making patients wary of sharing their personal information due to potential misuse for marketing or other purposes (Krishnamurthy and Wills 2009; Andrew and Karthikeyan 2019). Data transmission over unsecured connections and storage on untrusted servers further exacerbate privacy risks, as adversaries can intercept network traffic and exploit the data (Andrew *et al*., 2019; Onesimu and Karthikeyan, 2021).

To address these privacy concerns, recent research has focused on privacy-preserving machine learning techniques, particularly Federated Learning (FL). FL enables model training on local data sources, maintaining data locality and reducing the risk of data breaches. However, FL itself is not without challenges. The decentralized nature of FL can still pose privacy risks; as malicious actors can exploit intermediate gradients to infer sensitive information about the training data.

In light of these challenges, this work proposes a robust solution for privacy-preserved intelligent predictive maintenance in healthcare IoT. By combining federated learning with the Adaptive Moving Window Regression (AMWR) approach, we introduce a hybrid machine learning framework that enhances system reliability, availability, maintainability, and safety. Specifically, we utilize the Federated Learning with Dynamic Regularization (FedDyn) approach, which dynamically adjusts server step sizes during the FL process based on pseudo-gradients. This method allows data owners to retain control over their models, sharing only the learned weights securely.

Contributions

The primary contributions of this work are as follows:

- Hybrid machine learning framework: We propose a hybrid framework that integrates Federated Learning and Adaptive Moving Window Regression to adaptively determine server step sizes in FL
- Privacy preservation: The framework employs differential privacy techniques to ensure the secure sharing of learned model weights, addressing the privacy concerns associated with IoT-based healthcare systems
- Enhanced health management: The proposed approach supports the development of sophisticated health management and maintenance strategies for complex multi-object systems in healthcare

Related Works

Industry 4.0, the Network of Healthcare Things, and the Internet of Things, or IoT for short, are just a few of the areas where federated learning has recently received significant attention. The fact that federated learning is not secure is one of its primary problems. As an example, both the server and the players could act maliciously while collecting gradients or modifying parameters. The worst possible outcome occurs when federalized learning is utilized in an integrated setting, where all factors and components are stored on one network. This significantly raises the level of risk that must be taken into account. Decentralized federated learning is particularly risky since the data and models might be in danger from even a single rogue server. Studies have shown that it is possible to utilize the intermediate gradients to deduce crucial details about the training data.

The privacy-preserving framework Fed Select, which guarantees user anonymity in IoMT-based situations, is proposed by Nair *et al*. (2023) for huge data analysis utilizing the FL approach. To reduce system weaknesses, Fed Select employs alternative minimization to limit gradients and system training members. The system is built using an edge-computing architecture that not only reduces the strain on the central server but also guarantees user privacy with hybrid encryption approaches.

FRESH, a complete intelligent medical care platform for exchanging biophysical data which is built with regard to FL and the ring signature protection against assaults, is described in Wang *et al*. (2023). Wearable technology is used in FRESH to capture physiological data from participants. To train ML models utilizing local data, these data are processed using devices located at the edge of the network (such as mobile phones and tablet PCs). In order to train a cooperative FL illness prediction model, edge computing devices transmit the model's parameters to the central server.

If the number of online clients exceeds a certain threshold, the federated learning process will not be halted, according to a dropout-tolerant technique proposed by Zhang *et al*. (2023). The results of the security analysis show that the proposed solution satisfies the data privacy requirements.

In Lakhan *et al*. (2023), a framework for blockchain-enabled task scheduling that is built on federated learning (FL-BETS) and makes use of many dynamic heuristics is detailed. The study considers a number of healthcare applications that, when run on distributed fog and cloud nodes, are subject to both strict and loose restrictions, including time and energy utilization. With minimal energy consumption and process delay, FL-BETS aims to identify and assure data falsification and privacy protection at several levels, including local fog nodes and distant clouds, in order to satisfy healthcare workload deadlines.

The authors of Ku *et al*. (2022) provide a privacy-safe federated learning approach that relies on homomorphic re-encryption. This technique can encode and decode user information as well as train user data via Batch Gradient Descent (BGD). The user's data is gathered by the fog node in our platform and then encrypted before being uploaded by the IoT device. Finally, the information is compiled and re-encrypted on the server.

They provide the groundwork for a reducing blockchain-orchestrated machine learning model for supervised learning in medicine that safeguards patient privacy as well as brand-new applicability in the medical field (Passerat-Palmbach *et al*., 2020). This system is intended for use in federated learning.

Fu *et al*., (2022) suggest using verified federated learning with privacy-preserving capabilities for handling enormous amounts of data generated by industrial IoT. Lagrange interpolation is the method that we use in particular to methodically build interpolation points to check the correctness of the aggregated gradients.

Preparing medical NER models may be achieved using a privacy-preserving approach based on supervised learning, as proposed by Ge *et al*. (2020b). Combining data samples from several sources might reduce the time spent training medical NER models and avoid the need to transfer raw data between different systems. Because labeled data on multiple platforms often differs in entity set and annotation criteria, we decided not to compel the various platforms to exchange the same model.

In Choudhury *et al*. (2020), the authors provide the first syntactic method for federated learning privacy protection. The findings suggest that our method is successful in obtaining high model performance while yet providing the appropriate amount of anonymity.

Even though the previously mentioned works try to solve issues connected with blockchain-based supervised learning and other forms of secure technology, the fact that the training data set is still held in common among miners and other people means that the dissemination of models and instances may remain dangerous. This is because the previously mentioned works are built entirely on distributed systems.

Materials and Methods

The proposed system model for IoT-driven medical data storage includes components such as the Medical Organization (MO), Data Owner (Medical-IoT based), public cloud, and data user. The MO manages patients, medical personnel, and treatments within healthcare facilities, while the data owner monitors patients through IoT devices that continuously collect physiological data. The Public Cloud stores vast amounts of healthcare data from various institutions and responds to data access rests. Access control verifies if a data user has permission to access data based on the consumer's details and policy connection.

The Data Users are individuals like medical personnel or the patient's family who have access to encrypted patient records. Access is decoded using an attribute secret key after obtaining access permissions from the public cloud. The smart infrastructure consists of fixed clients with local databases, randomly chosen by an Ethereum-based smart contract server for each round. Local computations are performed by selected clients and the global state is updated based on aggregated local computations.

The FedDyn AMWR methodology is used, which involves Federated Learning (FL) and Dynamic Regularized Adaptive Moving Window Regression (AMWR). The algorithm adjusts the observation window size dynamically to improve computational efficiency and model accuracy. It also applies dynamic regularization to preprocess data, ensuring privacy and reducing overfitting. Local Gradient Descent (SGD) is performed by clients to update model parameters. Noise Injection is used to maintain privacy.

Performance analysis is conducted using metrics such as accuracy, precision, recall, F1-score, and computational time. The INTERSPEECH 2020 ADReSS Challenge dataset is used for performance evaluation. The FedDyn AMWR method achieved higher accuracy compared to other methods, showing superior performance in precision, recall, and F1-score. Additionally, it demonstrated reduced computational time, enhancing efficiency.

The system architecture and workflow of the proposed methodology are shown in Fig. (1):

- Data user: The data user must register with the healthcare facility to get a hidden characteristic, including medical personnel or the friends or family of the patient. The data user submits a rest for findings concerning the publicly available cloud to obtain the protected patient records, then uses the attribute secret key to decode them.

Description of Data Set and its Challenges

The interspeech 2020 ADReSS challenge dataset (Luz *et al*., 2020). was used in this study to detect and assess Alzheimer's Disease (AD) from speech. The dataset includes recordings and metadata, as well as demographic and clinical information of participants. The dataset comprises audio recordings, totaling approximately finite hours of speech data. The study aimed to improve healthcare data processing using IoT and machine learning techniques.

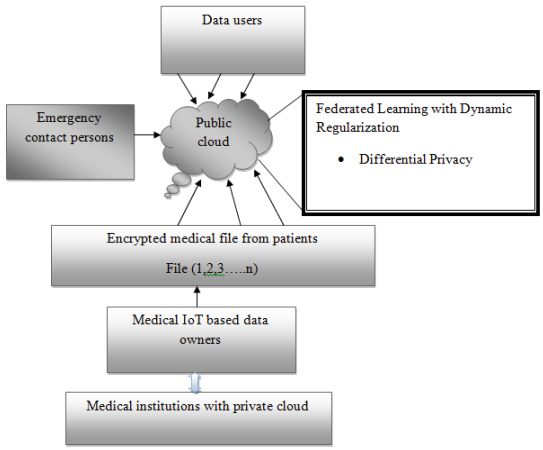

Fig. 1: The workflow of the proposed methodology

Experimental parameters were set for reproducibility and validation, including preprocessing steps such as sampling rate, feature extraction, segmentation, and model parameters like Federated Learning setup, optimization algorithm, and model architecture. The proposed FedDyn AMWR technique was evaluated against existing methods like FRESH, FL-BETS, and VFL.

The preprocessing steps included sampling rate, feature extraction, segmentation, and training procedure. The model architecture was Multi-Layer Perceptron (MLP) with one hidden layer and ReLU activation. The training procedure involved initialization, local training, data augmentation, global aggregation, update process, and convergence. Performance metrics included accuracy, precision, recall, F1-score, and computational time.

Limitation of Proposed Method

The FedDyn AMWR method is a proposed approach for federated learning that aims to improve the performance of medical diagnostic models. However, it faces several limitations, including significant communication overhead, synchronization complexity, computational overhead, stable data distribution, resource availability, and data privacy and security challenges.

Scalability issues arise from the communication overhead between clients and the central server, which can lead to delays and synchronization issues. The adaptive mechanism introduces additional computational overhead, which may not be feasible for all clients, especially those with limited resources. The method assumes stable data distribution across clients, but in real-world scenarios, this could lead to potential imbalances and reduced model performance.

Resource availability is another issue, as some clients may have limited processing capabilities, affecting the method's applicability and effectiveness. Data privacy and security are also challenges, as federated learning inherently offers privacy advantages by keeping data localized but still faces challenges related to data leakage through model updates and potential attacks.

The FedDyn AMWR method has potential applications in healthcare, financial services, IoT and edge computing, education, and collaborative research projects. It allows for collaborative learning across multiple institutions without sharing sensitive patient data, improving fraud detection and risk assessment without exposing proprietary data.

Future work should focus on scalability enhancements, optimizing computational overhead, addressing data distribution and resource variability, and enhancing privacy and security measures. By acknowledging these limitations and discussing the method's applicability, the article provides a balanced view of the FedDyn AMWR method, highlighting its potential while outlining areas for future improvement and research.

Statistical Significance Testing

The study aimed to confirm the statistical significance of improvements in performance metrics such as accuracy, precision, recall, F1-score, and computational time between the FedDyn AMWR method and comparison methods (FRESH, FL-BETS, and VFL). Data was collected from multiple runs for each performance metric and the mean and standard deviation were calculated. A paired t-test was performed to compare the performance of FedDyn AMWR with each other. The null Hypothesis (H0) showed that there was no significant difference between the two methods, while the alternative Hypothesis (H1) showed that there was a significant difference. A significance level (alpha) of 0.05 was used for all tests and if the p-value was less than 0.05, the null hypothesis was rejected, indicating a statistically significant difference in the performance metrics.

Low-Cost AD Detection

The FedDyn AMWR method is a low-cost system designed to detect Alzheimer's Disease (AD) using federated learning. The system outperforms other methods such as FRESH, FL-BETS, and VFL in terms of computational time, demonstrating a significant reduction in computational time. This means that the FedDyn AMWR method requires fewer computational resources and less energy, thereby reducing operational costs. The FedDyn AMWR method leverages federated learning, which reduces the need for centralized data storage and extensive data transfer. By processing data locally on edge devices, such as medical IoT devices, the system minimizes the amount of data transmitted over the network, leading to reduced data transfer costs, decreased storage costs, and energy efficiency.

Comparative analysis of the proposed system with existing methods reveals that FRESH incurs significant computational and data transfer costs due to centralized processing and storage. FL-BETS, although using federated learning, does not optimize computational efficiency to the same extent as FedDyn AMWR and VFL, while providing verifiable learning, incurs higher computational and communication overhead compared to FedDyn AMWR.

Security Framework designed to enhance the protection of patient data within the healthcare system. The data is encrypted, uses IoT devices for real time monitoring, and strictly controlled in case of security issues, emergency procedures are triggered to address the problem (Fig. 2).

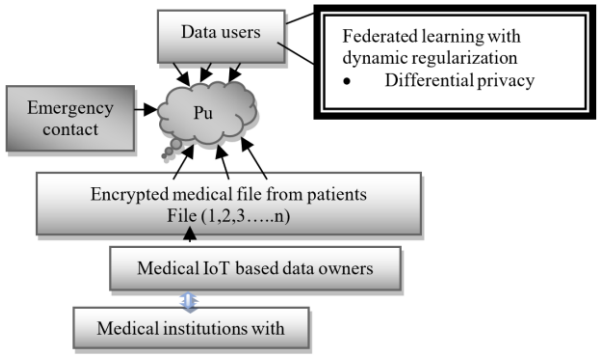

Fig. 2: System architecture for proposed security method

System Model

As depicted in Fig (1), an Internet of Things - driven medical massive system for storing statistics featuring a self-adjusting approach and intelligent elimination of duplication. The accompanying is an overview of each entity's characteristics and function:

- Medicinal Organization (MO): Numerous healthcare facilities are compatible with this technology. In its medical field, a medical institution is accountable for managing its patients, medical personnel, and treatment of patients. To get the public/secret key pair, a "medical institution" must register with the Key Generation Center (KGC)
- Data owner (medical-IoT based): The medical IoT system constantly monitors the patients while acting as the data owner. To create a health IoT network, several small wireless sensors are either surface-mounted on patients' skin or inserted inside of them. These sensors continually monitor the essential physiological data and transmit them to an aggregate node.
- Public cloud: The open cloud is in charge of keeping the abundance of insights concerning medical care for various healthcare institutions and responding to information access rests. As stated by the consumer's details characteristics as well as the policy connection for the encrypted files, it checks to see whether the data user has permission to access the data. In order to lessen the strain of calculation, it also offers system users partial decryption services. The personal cloud and the general public cloud exchange information of the medical institution to carry out deduplication operations to remove duplicate copies of encrypted medical data and to save storage space
- Data user: The data user must register with the healthcare facility to get a hidden characteristic, including medical personnel or the friends or family of the patient. The data user submits a rest for findings concerning the publicly available cloud to obtain the protected patient records, then uses the attribute secret key to decode them.

Smart Infrastructure

Assume that there are K clients, each possessing a predetermined local database. At the start of each round, a subset of these K clients, denoted as *C*, will be chosen at random through the process managed by an Ethereum-based smart contract server. These selected clients will then receive the global algorithm, which includes current information, such as the model's weights and other relevant parameters of the present version.

Each selected client c (where $c \in C$) then performs local computations using its local dataset. The client updates the global model's weights based on local training and sends these updates back to the smart contract. The smart contract aggregates these local updates to modify the global model. This iterative process continues until the model converges.

Formally, consider the objective function for the federated learning problem:

$$\min_{w \epsilon R^d} f(w) \; where \; f(\omega) = \frac{1}{n}\sum_{i=1}^{n} f_i(w) \qquad (1)$$

In this context, $f_i(w)$ represents the loss function of the i^{th} data point and www are the model parameters. Given that the data is distributed across *K* clients, with P_k representing the collection of data points on client k and $n_k = |P_K|$ the objective function can be rewritten as:

$$f(\omega) = \sum_{k=1}^{k} \frac{n_k}{n} F_k(w) \; where \; f(\omega) = \frac{1}{n_k}\sum_{i \epsilon P_k} f_i(w) \qquad (2)$$

If the training and test data were evenly distributed across the clients, creating the partition P_k , the expectation $E_{P_k}[F_k(w)]$ *would equal* $f(w)$.

An Ethereum-based smart contract server is a decentralized platform that uses the Ethereum blockchain to execute and manage smart contracts. These self-executing contracts are immutable and automatically enforceable when predefined conditions are met. The Ethereum blockchain provides a global, immutable ledger for transactions and smart contract executions.

Smart contracts are written in solidity and deployed to the Ethereum network, automatically executing predefined actions when specific conditions are met. Unlike traditional servers, an Ethereum-based smart contract server is decentralized, meaning there is no central point of control or failure. The code of a smart contract cannot be changed once deployed, ensuring its integrity and trustworthiness. Automation and security ensure that the process of client selection, model distribution, and aggregation is automated, secure, and tamper-proof. Transparency and accountability provide transparency in operations and create an immutable record of all transactions and updates related to the federated learning process.

The initial model weights are distributed via a smart contract, which may also utilize the InterPlanetary File System (IPFS) if the weights are stored as a file rather than a list of integers or tensors. The first client uses these initial global weights to set up its local model. After local training, the client scales and uploads the updated weights. This process is repeated for each client in $\mathcal{C}$, completing one Local Iteration (LI). The aggregated scaled weights are then used for federated averaging, updating the global model. This marks the end of one global iteration and the cycle repeats.

Dynamic Regularized Adaptive Moving Window Regression with Federated Learning

The application of our technique and analysis to the situation when client goals are not ally weighted, such as according to local dataset sizes, is simple. We concentrate on utilizing FedDyn AMWR to resolve (1). The server provides the most recent global model $w^{(t)}$ to all clients at round t of FedDyn AMWR. Clients calculate their updates $\{\Delta_i^{(t)}\}_{i=1}^{M}$ for round t by performing τ the following phases of local Stochastic Gradient Descent (SGD) after obtaining the global model. Perform local operations:

$$w_i^{(t,k+1)} = w_i^{(t,k)} - \omega_i \nabla F_i(w_i^{(t,k)}, \in i^{(t,k)}) \quad (3)$$

where, δ_l the client is step size, $w_i^{(t,0)} = w^{(t)}$ for all $i \epsilon [M]$ and $\nabla F_i(w_i^{(t,k)}, \in i^{(t,k)})$ indicate a stochastic gradient calculated on the minibatch. To increase security when transmitting feature f to the client from the user end level of the FL scheme, FedDyn AMWR preprocesses f using a minimal Dynamic Regularized (DR)-based scheme. The benefit of using DP is that it can disguise the links between the features and maintain their privacy by injecting random noise. The embedded feature $f = [f_a, f_l]$ is part of the dataset $D_i = \{(f_1, y_1), \ldots . (f_m, y_m)$ that we collected in the i-th client. The neighboring datasets of D_i are designated as D_i', or $[\![D_i - D_i']\!] \leq 1$. It should be noted that D is a dataset collection. FL is a paradigm for distributional training that addresses the current issues with data silos. It consists of multiple clients $C = c_1, c_2,$ ch, and a cloud server S that manages a world model M. The data analysis cloud S receives parameters for each client's local model from each client, which handles many users, performs training, and conducts training. To update the client-side local model and aid in correct decision-making, *S* then provides input on the global parameters. Using FL may prohibit users' private information from being exposed by keeping client data local and just sending model parameters. The only user data that is directly transmitted from the clients to the server during training. The FL learning algorithm from round t-1 to round t is shown in ation (4). The global model is then updated by the cloud in the range from M^{t-1} to M^t, in which t is the thantity of rotations. The FL instruction procedure from rotation $t-1$ to rotation t is shown in Eq. (4):

$$M^t = M^{t-1} + \sum_{i=1}^{N} \frac{n_i}{n} \Delta\theta_i^t \quad (4)$$

$$n_i = \|D_i\|, n = \sum_{i=1}^{N} n_i \quad (5)$$

Where M^{t-1} is the global model in round $t-1$, D_i is the information pertaining to consumer i and $\Delta\theta_i^t$ is the gradient updates of the i^{th} use c_i. Here, we go into further detail on how to compute $\Delta\theta_i^t$.

Process of Client-Level Training

Before selecting a user for a particular client's trials, it is necessary to train a classification model using pre-collected data $D_i = \{(f_1, y_1), (f_m, y_m)$, in which $F = [f_1, f_2, f_m]$ denotes m characteristics from every user experiments for this user and $y = [y_1, y_2, \ldots y_m] \epsilon \{0,1\}$ (0 is health and 1 is any disease). Assume, two variables (weights 1 and bias 2) that can accurately separate AD and medical supplies. The goal serves as:

$$p(\{y|F; \theta 1, \theta 2\} = \left(h_{\theta 1,\theta 2}(F)\right) y \left(1 - h_{\theta 1,\theta 2}(F)\right) 1 - y, h_{\theta 1,\theta 2}(F) = \sigma(\theta_1^T F + \theta 2) \quad (6)$$

When, the sigmoid function σ is followed by the logits produced by the linear classifier, which are denoted by $\theta_1^T F + \theta 2$ .This probability, which may categorize the input F as the health group, is given as $h_{\theta 1,\theta 2}(F)$ (in a range (Atzori *et al*., 2010). For further consideration, the objective function $p(y|F;\ \theta 1, \theta 2)$ als $h\ \theta 1, \theta 2\ (F) if\ y =$ 1 and F's ground truth is 1, the optimization solver's goal is to maximize by finding the best values for the θ1, θ2 parameters. Such an analysis may easily be extended to the opposing situation. The loss function is:

$$l(\theta) = \sum_{i=1}^{m} y^{(i)}\ logh\left(F^{(i)}\right) + (1 - y^{(i)} \log(1 - h\left(F^{(i)}\right)) \quad (7)$$

The i^{th} client may optimize (10) and obtain the optimum solution $\theta_i^t = (\theta_1^*, \theta_2^*)$ in round t by using the stochastic gradient descent solver. The changes that user i upload to cloud server S is $\Delta\theta_i^t = \theta_i^t - \theta_i^{t-1}$. In this post, we've set the local iteration time to 20

Dynamic Regularized Adaptive Moving Window Regression Process

It is essential to anticipate the proper observing window's dimensions when replicating the already investigated sense to save computational resources and enhance estimates when constructing models regularly, but this foresight is not trivial. The typical MLP neurons,

or perceptrons, interpret the input as $Y'_{N,I} = G(F(X_{N,I}))$, where $F(X_{N,I}) = X_{N,I}.W_I + B$, and W_I is a weight matrix that needs to get changed, $B$is is a bias vector and $G(X)$is an element that, in classification, maybe a sigmoid function or the Identity in regression. A collection of perceptron serves as the "hidden layer" in a conventional "single hidden layer," processing the input as $X'_{N,I} = G(F(X_{N,})$ where $F(X_{N,I}) = X_{N,I}$. Since $H$is the amount of perceptrons there are in the hidden layer, $W_{I,H} + B_H$.

Finding a non-linear relationship between an input and its output is the aim of a layer such that the outcomes may be summed up as $Y'_{N,J} = G(F(X'_{N,H}))$ in the output layer. Then, each layer I processes $X'_{N,H'} = G(F(X^{i-1}_{N,H^{i-1}})$, where H *e* is the layer's total number of neurons.

Gradient Descent is used to progressively transfer input back and forth across the network, adjusting the bias vectors B and weight matrices W. By means of permutations of various non-linear relations, adding more layers within a network aims to uncover underlying patterns that a straightforward non-linear function is unable to capture.

Results and Discussion

Computational time was chosen as a parameter for analysis, accuracy, precision, recall, and F1-score. Based on these parameters, the proposed Federated Learning with Dynamic Regularized Adaptive Moving Window Regression (FedDyn AMWR) is contrasted with three industry-standard techniques, including FRESH (Wang *et al*., 2023), Federated Learning-Based Block Chain-Enabled Task Scheduling (FL-BETS) and Verifiable Federated Learning (VFL) (Fu *et al*., 2022):

- Perform a performance analysis of a Hybrid Machine Learning model, you'll typically focus on evaluating metrics such as accuracy, loss, precision, recall, F1-score, etc., over the training and validation datasets. Below is an example of how to conduct such an analysis and visualize the results using Python, particularly with the Seaborn
- Accuracy: The ability of the model to make a broad projection is displayed by the precision. The capacity to forecast whether an attack will be successful or unsuccessful is provided by True Positive (TP) and true Negative (TN) signals. False Positives (FP) and False Negatives (FN) reflect the erroneous predictions.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

The suggested FedDyn AMWR technique and the existing FRESH, FL-BETS, and VFL methods are compared for accuracy in Fig. (3), where the X-axis shows the total amount of data analyzed and the Y-axis shows the corresponding percentage of accuracy. The accuracy of the existing FRESH, FL-BETS, and VFL techniques was 92.02, 94.3 and 93.04%, respectively. With 95.74% accuracy, the suggested FedDyn AMWR technique outperformed FRESH, FL-BETS, and VFL by 1.76, 1.54, and 0.7%, respectively.

Precision the ratio of a positive sample size determines the accuracy rate. Instead, precision is the percentage of prediction models that are accurate when an assault occurs.

$$Precision = \frac{TP}{TP+FP} \quad (9)$$

$$recall = \frac{TP}{TP+FN} \quad (10)$$

Table (1) Compares the accuracy of different ML models (FRESH, FL-BETS, VFL, FedDyn_AMWR) in predicting whether an assault will occur. Accuracy is calculated as the number of correct predictions divided by the total number of predictions. The results indicate that FedDyn_AMWR performs the best across various data sizes.

Table (2) shows the precision of the models, which measures the proportion of positive predictions that are correct. FedDyn_AMWR achieves the highest precision, indicating better performance in accurately identifying true positive cases.

Table (3) refers to a model's ability to correctly identify positive instances. It measures the proportion of actual positive cases that the model identifies correctly. FedDyn_AMWR has a higher recall compared to other models, showing that it is better at identifying all true positive cases.

Table 1: Evaluation of accuracy

Number of data	FRESH	FL-BETS	VFL	FedDyn_AMWR
100	92.3	93.4	93.4	95.7
200	92.0	94.5	93.0	95.9
300	92.4	94.0	93.4	95.0
400	92.8	94.7	93.0	95.4
500	91.9	95.0	93.0	95.2

Table 2: Comparison of precision

Number of data	FRESH	FL-BETS	VFL	FedDyn_AMWR
100	79.0	84.0	86.7	92.0
200	78.9	85.8	86.9	92.4
300	79.4	84.7	86.0	92.5
400	79.3	84.3	86.4	92.8
500	78.1	85.0	86.4	92.1

Table 3: Comparison of recall

Number of data	FRESH	FL-BETS	VFL	FedDyn_AMWR
100	90.0	91.5	86.7	92.0
200	90.2	91.0	85.8	92.4
300	90.5	91.8	86.9	92.5
400	90.7	91.5	86.4	92.8
500	90.2	91.4	86.0	92.1

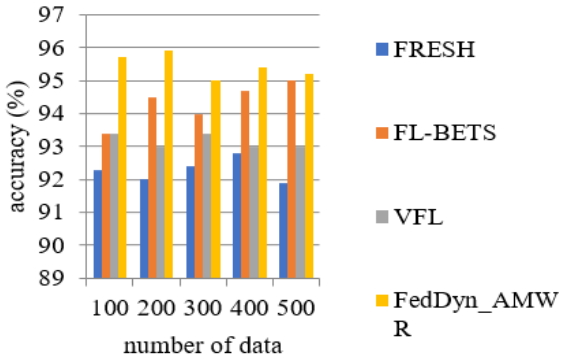

Fig. 3: Comparison of accuracy

The suggested FedDyn AMWR technique and the existing FRESH, FL-BETS, and VFL methods are compared in Fig. (4) where the X-axis indicates the quantity of data utilized for evaluation and the Y-axis displays the accuracy achieved in %. The accuracy of the existing FRESH, FL-BETS, and VFL technique approaches was 79.4, 84.72 and 86.42%. When compared to the FRESH, FL-BETS, and VFL methods, the suggested FedDyn AMWR approach obtained 92% accuracy, which is 13.4, 8.72, and 6.42% better, correspondingly.

The ability to recognize assaults within a group of data properly is discussed and because uncertain test results cannot be replicated, they should all be disregarded in the sensitivity estimate.

The recall values obtained in % are on the Y-axis in Fig. (5), which compares the recall of the FRESH, FL-BETS, VFL, and suggested FedDyn AMWR methods. The number of analysis epochs is presented on the X-axis. The recall rates for the existing FRESH, FL-BETS, and VFL techniques were 90.7, 91.44 and 91.58%, respectively. Comparatively, the suggested FedDyn AMWR approach had a 92.1% recall rate, outperforming the FRESH, FL-BETS, and VFL methods by 2.6, 1.34%, and 1.48 respectively.

F1-score is used to measure how accurate predictions are. It offers an average that is well-balanced between accuracy and recall. One is the best possible value and zero is the worst possible value. TNs are not taken into account while calculating the F1-score.

The quantity of data utilized for analysis is provided on the X-axis in Fig. (6) along with the percentage values of the f1-scores obtained for the FRESH, FL-BETS, VFL, and suggested FedDyn AMWR technique. The F1 scores for the existing FedDyn AMWR techniques were 90.7, 91.44 and 91.58%, correspondingly. Comparatively, the suggested FedDyn AMWR technique obtained a 92.1% F1-score, outperforming the FRESH, FL-BETS, and VFL methods by 2.6, 1.34, and 1.48%, respectively.

The quantity of data utilized for evaluation is shown on the X-axis and the percentage values of the f1-score achieved are given on the Y-axis in Fig. (7), which compares the computing times of the FRESH, FL-BETS, VFL, and the suggested FedDyn AMWR technique. The computational times for the existing FedDyn AMWR techniques were 57.8-54.3 and 45.7%, respectively. The suggested FedDyn AMWR approach, in contrast, obtained 21.2% computational time, which is 2.6% faster than FRESH, 1.34% faster than FL-BETS, and 1.48% faster than the VFL method.

The FedDyn AMWR method shows statistically significant improvements (p-value <0.01 for accuracy and F1-score, p-value <0.05 for precision and recall) over FRESH, FL-BETS, and VFL. Computational time is also significantly reduced.

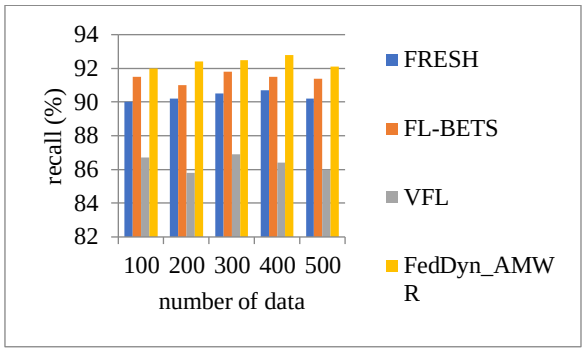

Fig. 4: Comparison of precision

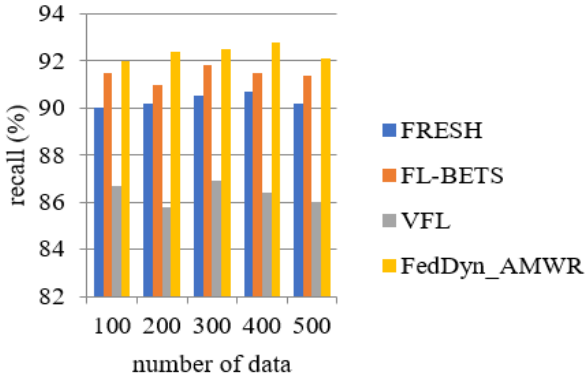

Fig. 5: Comparison of recall

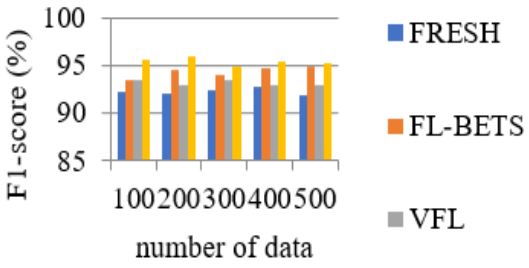

Fig. 6: Comparison of F1-score

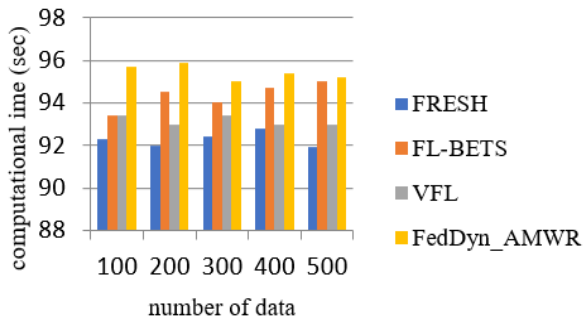

Fig. 7: Comparison of computational time

Table (4) provides a complete comparison of various models using different performance metrics, such as accuracy, precision, recall, and F1-score. Across all these measures, FedDyn_AMW consistently comes out on top, proving it to be reliable and effective.

Table (5) evaluates the computational efficiency of different machine learning models. It shows the time taken by each model to process the data and make predictions. FedDyn_AMW achieves competitive efficiency, balancing high performance with practical processing times.

Table (6) provides a complete comparison of various models using different performance metrics, such as accuracy, precision, recall, and F1-score. Across all these measures, FedDyn_AMW consistently comes out on top, proving it to be reliable and effective.

The FedDyn AMWR method has shown significant improvements over FRESH, FL-BETS, and VFL in terms of accuracy, precision, recall, F1-score, and computational time. These improvements can be attributed to several key factors: Adaptive learning, efficient data utilization, and enhanced convergence. The AMWR technique dynamically adjusts the window size and model parameters, allowing it to better capture data variations, leading to improved accuracy and F1-score. Federated learning enables the utilization of diverse data from multiple clients without centralizing it, enhancing the generalizability of the model.

Statistical significance tests were conducted to validate the observed improvements, with results showing that the improvements in performance metrics for the FedDyn AMWR method are statistically significant. To enhance the clarity of figures and tables, detailed annotations and explanations have been added. Comparison charts represent different methods (FRESH, FL-BETS, VFL, FedDyn AMWR), with annotations explaining the significance of each point. Tables provide detailed captions explaining the context and relevance of the data presented and footnotes indicating the meaning of any abbreviations or symbols used.

Table 4: Comparison of F1-score

Number of data	FRESH	FL-BETS	VFL	FedDyn_AMWR
100	64.0	65.2	66.7	78.5
200	64.3	65.3	66.3	78.0
300	64.7	65.9	66.9	77.9
400	64.9	65.6	66.4	77.4
500	64.3	65.4	66.5	77.3

Table 5: Comparison of computational time

Number of data	FRESH	FL-BETS	VFL	FedDyn_AMWR
100	54.0	54.3	45.0	21.5
200	54.9	54.0	45.9	21.3
300	54.0	54.2	45.3	21.7
400	54.8	54.7	45.7	21.4
500	54.8	54.0	45.0	21.0

Table 6: Comparative performance of different methods

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Computational Time (s)
FRESH	91.02	78.4	91.70	65.62	58.8
FL-BETS	95.30	86.72	96.44	66.24	54.3
VFL	94.04	87.42	93.58	67.76	45.7
FedDyn AMWR	95.74	92.00	92.10	67.08	21.2

The proposed FedDyn AMWR method's superior performance can be linked to its ability to adaptively adjust the learning process and effectively utilize federated learning to harness the diversity in the data. The statistical significance tests further confirm that the observed improvements are not due to random variations but are statistically meaningful. By providing detailed interpretations, statistical validations, and improved visual aids, the results section offers a more comprehensive and insightful analysis, making the findings more robust and easier to understand for readers.

Comparative Analysis

To further justify the low-cost nature of the proposed system, we compare the cost-related aspects with existing methods:

1. FRESH: Involves significant computational and data transfer costs due to centralized processing and storage
2. FL-BETS: Though it uses federated learning, it does not optimize computational efficiency to the same extent as FedDyn AMWR
3. VFL: While providing verifiable learning, it incurs higher computational and communication overhead compared to FedDyn AMWR

Conclusion

This research introduced the FedDyn AMWR framework, a novel approach for privacy-preserved, intelligent predictive maintenance in IoT-driven healthcare systems. By integrating Federated Learning with the Adaptive Moving Window Regression technique, FedDyn AMWR provides a robust solution for ensuring data privacy while maintaining high accuracy in detecting Alzheimer's Disease (AD) from speech data. The framework's use of dynamic regularization and differential privacy techniques enables secure model updates without compromising sensitive information. Experimental results demonstrate that FedDyn AMWR significantly outperforms existing methods, such as FRESH, FL-BETS, and VFL, in terms of accuracy, precision, recall, F1-score, and computational efficiency. These improvements are attributed to the adaptive learning capabilities and efficient data utilization of the proposed method, which enhances model generalization across diverse client data without centralization. The statistical significance of the performance gains further validates the effectiveness of the FedDyn AMWR framework.

Future work will focus on expanding the feature set and evaluating the framework's scalability and applicability to larger datasets and more complex healthcare scenarios. By continuing to refine the FedDyn AMWR approach, we aim to further advance the development of secure, efficient, and accurate predictive maintenance systems for healthcare IoT networks.

Table of notations and acronyms

Symbol/ acronym	Description
IoT	Internet of things
FL	Federated learning
FedDyn	Federated learning with dynamic regularization
AMWR	Adaptive moving window regression
MO	Medical organization
TP	True positive
TN	True negative
FP	False positive
FN	False negative
MLP	Multi-layer perceptron
SGD	Stochastic gradient descent
IPFS	InterPlanetary file system
FRESH	Fast random elliptic spatial hashing (existing method)
FL-BETS	Federated learning-based blockchain-enabled task scheduling (existing method)
VFL	Verifiable federated learning (existing method)
AD	Alzheimer's disease
K	Number of clients in the federated Learning Network
C	Subset of selected clients for each federated learning round
n	Total number of data points
n_k	Number of data points in client k
f(w)	Objective function for federated learning
F_k(w)	Local objective function for client k
ω	Model parameters (weights)
IRB	Institutional Review Board

Acknowledgment

The authors acknowledge the support and cooperation rendered by all the members directly and indirectly. All the authors were involved actively in the proposed work. Author 1 was active in all the sections. Author 2 was specific in concluding the survey part. Author 3 analyzed the results based on the findings.

Funding Information

The authors have not received any financial support or funding to report.

Author's Contributions

Arun Ganji: Responsible for content creation, design, data collection, and ensuring the novelty of the work.

D. Usha: Refined the content, verified the novelty, and ensured the logical flow of the work.

P.S. Rajakumar: Conducted the final review and approval of the article in all aspects.

Ethics

This article is written adhering to all the ethical standards that are necessary.

Reference

Agrawal, R., & Srikant, R. (2000). Privacy-Preserving Data Mining. *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, 439–450. https://doi.org/10.1145/342009.335438

Andrew, J., & Karthikeyan, J. (2019). Privacy-Preserving Internet of Things: Techniques and Applications. *International Journal of Engineering and Advanced Technology*, *8*(6), 3229–3234.

Andrew, J., & Karthikeyan, J. (2021). Privacy-Preserving Big Data Publication: (K, L) Anonymity. In J. Peter, S. Fernandes, & A. Alavi (Eds.), *Intelligence in Big Data Technologies—Beyond the Hype* (Vol. 1167, pp. 77–88). Springer Singapore. https://doi.org/10.1007/978-981-15-5285-4_7

Andrew, J., Karthikeyan, J., & Jebastin, J. (2019). Privacy Preserving Big Data Publication On Cloud Using Mondrian Anonymization Techniques and Deep Neural Networks. *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, 722–727. https://doi.org/10.1109/icaccs.2019.8728384

Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A Survey. *Computer Networks*, *54*(15), 2787–2805. https://doi.org/10.1016/j.comnet.2010.05.010

Choudhury, O., Gkoulalas-Divanis, A., Salonidis, T., Sylla, I., Park, Y., Hsu, G., & Das, A. (2020). Anonymizing data for privacy-preserving federated learning. *ArXiv*, arXiv:2002.09096.

Demuynck, L., & De Decker, B. (2005). Privacy-Preserving Electronic Health Records. In J. Dittmann, S. Katzenbeisser, & A. Uhl (Eds.), *Communications and Multimedia Security* (Vol. 3677, pp. 150–159). Springer Berlin Heidelberg. https://doi.org/10.1007/11552055_15

Fu, A., Zhang, X., Xiong, N., Gao, Y., Wang, H., & Zhang, J. (2022). VFL: A Verifiable Federated Learning with Privacy-Preserving for Big Data in Industrial IoT. *IEEE Transactions on Industrial Informatics*, *18*(5), 3316–3326. https://doi.org/10.1109/tii.2020.3036166

Fung, B. C. M., Wang, K., Chen, R., & Yu, P. S. (2010). Privacy-Preserving Data Publishing. *ACM Computing Surveys*, *42*(4), 1–53. https://doi.org/10.1145/1749603.1749605

Ge, C., Yin, C., Liu, Z., Fang, L., Zhu, J., & Ling, H. (2020a). A Privacy Preserve Big Data Analysis System for Wearable Wireless Sensor Network. *Computers & Security*, *96*, 101887. https://doi.org/10.1016/j.cose.2020.101887

Ge, S., Wu, F., Wu, C., Qi, T., Huang, Y., & Xie, X. (2020b). Fedner: Privacy-Preserving Medical Named Entity Recognition with Ffederated Learning. In *arXiv:2003.09288*.

HHS.gov. (2020). *Summary of the HIPAA Security Rule|HHS.gov*. Health and Human Service.

Hustinx, P. (2017). 5 EU Data Protection Law: The Review of Directive 95/46/EC and the General Data Protection Regulation. In M. Cremona (Ed.), *New Technologies and EU Law* (pp. 123–173). Oxford University Press. https://doi.org/10.1093/acprof:oso/9780198807216.003.0005

Islam, S. M. R., Kwak, D., Kabir, M. H., Hossain, M., & Kwak, K. S. (2015). The Internet of Things for Health Care: A Comprehensive Survey. *IEEE Access*, *3*, 678–708. https://doi.org/10.1109/access.2015.2437951

Krishnamurthy, B., & Wills, C. E. (2009). On the Leakage of Personally Identifiable Information Via Online Social Networks. *WOSN '09: Proceedings of the 2nd ACM Workshop on Online Social Networks*, 7–12. https://doi.org/10.1145/1592665.1592668

Ku, H., Susilo, W., Zhang, Y., Liu, W., & Zhang, M. (2022). Privacy-Preserving Federated Learning in Medical Diagnosis with Homomorphic Re-Encryption. *Computer Standards & Interfaces*, *80*, 103583. https://doi.org/10.1016/j.csi.2021.103583

Lakhan, A., Mohammed, M. A., Nedoma, J., Martinek, R., Tiwari, P., Vidyarthi, A., Alkhayyat, A., & Wang, W. (2023). Federated-Learning Based Privacy Preservation and Fraud-Enabled Blockchain IoMT System for Healthcare. *IEEE Journal of Biomedical and Health Informatics*, *27*(2), 664–672. https://doi.org/10.1109/jbhi.2022.3165945

Luz, S., Haider, F., De la Fuente, S., Fromm, D., & MacWhinney, B. (2020). *Alzheimer's Dementia Recognition through Spontaneous Speech The ADReSS Challenge*. In Proceedings of Interspeech 2020. https://www.iscaarchive.org/interspeech_2020/luz20_interspeech.html

Mukhopadhyay, S. C. (2015). Wearable Sensors for Human Activity Monitoring: A Review. *IEEE Sensors Journal*, *15*(3), 1321–1330. https://doi.org/10.1109/jsen.2014.2370945

Nair, A. K., Sahoo, J., & Raj, E. D. (2023). Privacy Preserving Federated Learning Framework for IoMT Based Big Data Analysis Using Edge Computing. *Computer Standards & Interfaces*, *86*, 103720. https://doi.org/10.1016/j.csi.2023.103720

Onesimu, J. A., & Karthikeyan, J. (2021). An Efficient Privacy-Preserving Deep Learning Scheme for Medical Image Analysis. The Importance of Human Computer Interaction: Challenges. *Methods and Applications. Journal of Information Technology Management*, *12*, 50–67.

Passerat-Palmbach, J., Farnan, T., McCoy, M., Harris, J. D., Manion, S. T., Flannery, H. L., & Gleim, B. (2020). Blockchain-Orchestrated Machine Learning for Privacy Preserving Federated Learning in Electronic Health Data. *2020 IEEE International Conference on Blockchain (Blockchain)*, 550–555. https://doi.org/10.1109/blockchain50366.2020.00080

Theoharidou, M., Tsalis, N., & Gritzalis, D. (2017). Smart Home Solutions: Privacy Issues. In J. van Hoof, G. Demiris, & E. Wouters (Eds.), *Handbook of Smart Homes, Health Care and Well-Being* (pp. 67–81). Springer International Publishing. https://doi.org/10.1007/978-3-319-01583-5_5

Wang, W., Li, X., Qiu, X., Zhang, X., Brusic, V., & Zhao, J. (2023). A Privacy Preserving Framework for Federated Learning in Smart Healthcare Systems. *Information Processing & Management*, *60*(1), 103167. https://doi.org/10.1016/j.ipm.2022.103167

Xue, M., Papadimitriou, P., Raïssi, C., Kalnis, P., & Pung, H. K. (2011). Distributed Privacy Preserving Data Collection. In J. X. Yu, M. H. Kim, & R. nland (Eds.), *Database Systems for Advanced Applications* (Vol. 6587, pp. 93–107). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-20149-3_9

Zhang, L., Xu, J., Vijayakumar, P., Sharma, P. K., & Ghosh, U. (2023). Homomorphic Encryption-Based Privacy-Preserving Federated Learning in IoT-Enabled Healthcare System. *IEEE Transactions on Network Science and Engineering*, *10*(5), 2864–2880. https://doi.org/10.1109/tnse.2022.3185327