IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 October 2024, accepted 2 November 2024, date of publication 8 November 2024, date of current version 25 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3493946

RESEARCH ARTICLE

A Privacy-Preserving Collaborative Federated Learning Framework for Detecting Retinal Diseases

SEEMA GULATI [1], KALPNA GULERIA [1], NITIN GOYAL [2], (Senior Member, IEEE), AHMAD ALI ALZUBI [3], AND ÁNGEL KUC CASTILLA [4,5,6]

[1]Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab 140401, India
[2]Department of Computer Science and Engineering, School of Engineering and Technology, Central University of Haryana, Mahendergarh, Haryana 123031, India
[3]Computer Science Department, Community College, King Saud University, Riyadh, Saudi Arabia
[4]Engineering Research & Innovation Group, Universidad Europea del Atlántico, 39011 Santander, Spain
[5]Department of Project Management, Universidad Internacional Iberoamericana, Campeche 24560, México
[6]Department of Project Management, Universidad Internacional Iberoamericana, Arecibo, PR 00613, USA

Corresponding author: Kalpna Guleria (guleria.kalpna@gmail.com)

This work was supported by the Researchers Supporting Project (RSP2024R395), King Saud University, Riyadh, Saudi Arabia.

ABSTRACT The rapid advancement in technology has simplified human life and provides convenience. However, this convenience has led to many lifestyle diseases like diabetes and obesity. The 2021 reports of the International Diabetes Federation (IDF) show 537 million diabetics, three-fourths of whom are from developing countries. About 28.5% of these diabetics over 40 suffer from Diabetic Retinopathy (DR), 4.4% face vision-threatening DR, and 3.8% suffer from Diabetic Macular Edema (DME). These conditions can lead to complete vision loss affecting health and quality of life. Early detection of DR and DME is crucial to prevent harmful effects. The proposed work employs a collaborative, privacy-preserving Federated Deep Learning (FDL) framework with lightweight MobileNetV2 architecture for early detection of DR and DME. The proposed FDL framework uses both Independent and Identically Distributed (IID) and non-IID data for 2 and 3-client architectures. In a 2-client scenario, the FDL implementation with FedAvg aggregation achieved 98.69% accuracy on IID data and 87.09% on non-IID data, while FedProx aggregation scored 98.03% on IID data and 98.25% on non-IID data. In a 3-client scenario, FedAvg aggregation achieved 97.62% accuracy on IID data and 96.28% on non-IID data, whereas FedProx aggregation achieved 98.69% on IID data and 97.77% on non-IID data. The results demonstrate that the FedProx aggregation is more stable and converges earlier than FedAvg aggregation in the non-IID settings of an FDL framework. The proposed FDL framework with its collaborative training feature, preserves privacy and maintains high prediction accuracy.

INDEX TERMS Automatic disease detection, early convergence, federated deep learning framework, healthcare, machine learning, MobileNetV2, model stability, retinal diseases.

I. INTRODUCTION

Artificial Intelligence (AI) revolutionizes disease detection by leveraging advanced algorithms to analyze medical data with high speed, accuracy, and scale. Disease detection is done by the analysis of diverse datasets including symptoms, medical images, genetic markers, and patient records. AI systems can identify patterns and anomalies indicative of various diseases at earlier stages than the traditional methods. Early detection increases the success rate of treatment and allows for proactive measures to stop the disease progression [1], [2]. AI-driven predictive analytics empowers healthcare providers to predict and mitigate disease outbreaks on a large scale [3]. This improves public health surveillance and quick

The associate editor coordinating the review of this manuscript and approving it for publication was Abedalrhman Alkhateeb.

responses to control pandemics. As AI technologies continue to evolve, they offer more accurate diagnoses, personalized treatments, and better health outcomes for individuals and communities [4], [5], [6].

Early detection of Diabetic Eye Diseases (DEDs) is essential in preventing vision loss among individuals with diabetes. Eye diseases caused by diabetes, such as diabetic macular edema and diabetic retinopathy, often develop without any noticeable symptoms in early stages which makes regular eye screenings crucial for timely diagnosis and intervention [7]. When left untreated, diabetic eye diseases can lead to irreversible damage to the retina, resulting in vision impairment or complete loss of sight [8], [9]. Early detection enables healthcare providers to provide proper treatments, such as laser therapy or intravitreal injections, to either completely stop the progression of the disease or at least slow down and preserve the patient's vision. Moreover, detecting diabetic eye diseases not only safeguards visual health but also serves as an indicator of overall diabetes management [10]. Thus, enabling adjustments to treatment plans to control sugar levels in the blood and consequently, reduce the growth of complications [11]. Therefore, prioritizing the detection of diabetic eye diseases through routine eye exams is essential for preserving vision and enhancing the quality of life of individuals with diabetes [12], [13].

AI plays a vital role in automatic disease detection, transforming traditional diagnostic methods with advanced data-driven approaches using Machine Learning (ML) algorithms and even more sophisticated neural network-based Deep Learning (DL) methods. These technologies handle vast amounts of medical data, including imaging, genetic, and clinical data, to identify patterns and anomalies that may indicate the presence of diseases. ML algorithms excel in tasks such as classification and regression, helping to distinguish between healthy and diseased states, predict disease progression, and identify high-risk patients [14], [15]. DL is a subset of ML that uses complex neural networks to automatically extract features from raw data, which makes it particularly effective in analysing complex datasets like medical images and genomic sequences. Together, ML and DL enhance the accuracy, speed, and efficiency of disease detection, enabling earlier diagnosis, and improving patient outcomes [16], [17], [18].

Federated Learning (FL) is an innovative approach to machine learning (ML) that was introduced by Google Research in 2016 for applying machine learning to multiple edge devices in a decentralised manner without sharing the local device information. FL has gained significant attention due to its ability to enhance privacy and efficiency in training models across multiple decentralised devices and servers [19], [20]. The fundamental framework of federated machine learning is based on the principle that the model is trained across many devices while keeping the data localized rather than centralizing the data and pooling it in a single location. This approach to learning addresses the privacy concerns arising while sharing vital information with the central devices, also it reduces the need for data transfer. Data transfer is practically a bottleneck in traditional machine learning workflows [21], [22].

The FL framework has certain key characteristics which make it suitable for practical scenarios. These attributes are decentralisation of data, collaborative model training, protection of sensitive user data, and efficiency and scalability. The data of edge devices like smartphones or other IoT-based smart devices or data of an organization on its local servers is never transferred out. This contrasts with the conventional machine learning scenarios, which require pooling all the data and forming data banks at a central location for training. Collaborative training is one of the primary characteristic features that makes a federated framework more credible. Collaborative training includes training from many devices to form an effective central or global model while maintaining privacy [23]. The training process is carried out with a central server that coordinates the whole process. The training takes place on devices that hold the data and model parameters. The weights of the global model are transported from the server and the weights of the local models are transported to the server. The improvements are made to the global model at the server using the clients' weights (participating devices) and by a series of communication rounds, the model converges. Therefore, the model updates are shared instead of the original data which means the private user data is not put at any risk. This is of utmost importance in today's world, especially in applications like healthcare, finance and personal services. Thus, one of the most significant features offered by a federated collaborative framework is privacy preservation [24].

In a centralized machine learning scenario, all data is consolidated and stored in a central location, often a server or a data centre. The pooled dataset is thus used entirely for training machine learning models and usually, the training is done in batches. Although centralised machine learning (ML) builds better models, it poses a significant privacy threat since all the user-sensitive data is moved from various locations creating a data pool at the central location causing privacy concerns like data breach and data misuse. This is in particular a big cause of concern in the healthcare industry where the data is personal and sensitive which the organizations usually restrict in sharing due to privacy policies. All the data used in the machine learning system is transmitted to a data center which is located at the central server. This data is transferred over networks, resulting in network congestion and network latency since the data is of enormous volume. As the data keeps increasing the system becomes increasingly complex and costly. Thus, scaling the system becomes a serious challenge in the centralised learning paradigms [25], [26].

In contrast to a centralised learning scenario, there are many advantageous features in a collaborative FL framework, making it more popular among machine learning enthusiasts. Foremost, the processing occurs in a decentralized manner since FL enables models to be trained directly on distributed devices without needing to pool any data. Therefore, data is not pooled and the training occurs at the local site containing the data. The data never leaves the host device and parameters plus model weights are sent to a central location. Thus, maintaining the privacy of data to be used and preventing potential threats. Next, the communication overhead is reduced significantly compared to centralised approaches, since the communication includes only model updates which are pretty small in size compared to the raw dataset. In addition, FL leverages edge computing resources by training models directly on devices, allowing for real-time, personalised model updates tailored to individual users' data. FL is inherently scalable as it distributes the computational load across multiple devices or servers. Also, it can adapt to dynamic and heterogeneous data sources, making it suitable for large-scale, diverse datasets [21], [27].

From the above, it can be said that the centralised learning models gather and process data in a single location, posing privacy and scalability challenges in contrast with FL computations and training is distributed across multiple devices or servers. Thus, preserving privacy, reducing communication overhead, and enabling scalable and adaptable ML models [28], [29].

The work proposed in the article implements a privacy-preserving FDL framework for detecting diabetic eye diseases. The significant contributions and findings from the implementation of the collaborative privacy-protecting framework are given below:

- The article presents a privacy-preserving and collaborative Federated Deep Learning (FDL) model, using the lightweight MobileNetV2 architecture as its backbone for detecting diabetic retinopathy and diabetic macular edema.
- The proposed federated deep learning model performs a comprehensive evaluation and comparison of two popular aggregation techniques, FedAvg and FedProx, across both Independent and Identically Distributed (IID) and non-IID data scenarios.
- The results demonstrate that the FDL implementations with the FedProx aggregation strategy have a higher prediction accuracy than the FDL implementations with FedAvg aggregation. The consistent superior performance is more prominent in the FDL implementations with non-IID data at the clients due to the proximal term of the FedProx aggregation scheme, which stabilizes the training process and enhances model convergence.
- The proposed work demonstrates the key advantage of federated learning in maintaining data privacy while achieving high prediction accuracy. This approach significantly reduces the possibility of data breaches, aligning with the stringent privacy norms of healthcare organizations.
- The proposed work also exhibits the capability of the FedProx aggregation strategy in stabilizing the model and attaining early convergence compared to the FedAvg aggregation strategy, particularly in non-IID data scenarios.

This article is organised into several sections to enhance the readability and understandability of the readers. The remaining sections are as follows: Section II is the related work that gives current research works in the domain. Section III is the background section which elaborates on the technical details and the challenges faced in implementing an FL framework, including handling IID and non-IID data. Section IV is the materials and methodology section that describes the datasets, models, and training processes used in the study. Section V is the most important since it shows the results and analytical discussion segment. This section showcases the outcomes and efficacy of various model configurations and methods backed by graphical data. Section VI, the conclusion reiterates the benefits of FL for medical diagnostics and suggests directions for future research.

II. RELATED WORK

This section details the research relevant to automatic disease detection in deep learning and FL, with a focus on privacy-preserving collaborative FL for disease detection.

The authors in reference [30] presented a FL system utilizing the FedAvg algorithm to update the global model. This system selects clients for each round using a mini-batch of 32. The dataset, comprising 142,952 mammography images from seven medical organizations, reflects real-world, non-IID, and heterogeneous data distributions. Class distributions for breast density varied significantly. Model evaluation was conducted using Cohen's Linear Weighted Kappa score. The authors in their research article presented in reference [31] proposed a diffusion technique for FL involving two decision-makers: Client Participation and Client Selection. A dataset of both CT and X-ray images was taken with 3706 total images, 3326 were selected, with 2800 used for training and 526 for testing. The study employed GhostNet, ResNet50, and ResNet101 models and compared their accuracy across various dataset partitions. The proposed approach improved accuracy scores but required more training time than the default federated model. The article presented in reference number [32] introduces a novel approach for segmenting 3D chest images and classifying them as COVID-affected or not, using FL to address data heterogeneity and preserve privacy. The study involved 1704 chest CT scans from three hospitals in China, Italy, and Japan, with 945 scans manually evaluated by radiologists. An ablation study showed that frequent aggregation rounds improved the performance of unsupervised clients, measured using Dice's score. The dataset included 28833 X-ray images from five sources, split into COVID-19, pneumonia, lung opacity, and normal categories. The results showed that the standard central model achieved 93.44% accuracy, the federated model 92.26%, and the communication-efficient model 92.24%.

The authors in reference [33] explored pneumonia detection using FL models on chest X-ray images, leveraging deep learning while protecting personal data by avoiding centralized data sharing. The dataset comprises chest X-ray images of two categories pneumonia sufferers and non-sufferers, resulting in 5856 images split into 4684 for training and 1172 for testing. The study focused on identifying foggy regions in the lungs as regions of interest. Four deep learning models were used: a customized CNN model, ResNet50, VGG16, and AlexNet. Results indicated that VGG16 was the best-performing model. The work presented in the article given in reference [34] presents an FL approach for diabetic retinopathy detection across a multi-center fundus screening network, utilizing the OPHDIAT dataset comprising 697,275 images from 96,023 patients, with 641,917 images used for training and 21,054 for testing. Employing the EfficientNet-B5 model, both weighted and unweighted aggregation methods were explored for client-server setup.

The research work given in the article referenced in [35] presents federated transfer learning for diabetic retinopathy (DR) detection using CNN architectures. It addresses the challenge of DR detection, a vision-threatening disease, by employing FL. The study utilizes public domain data from five DR datasets, totaling around 3000 images. The key algorithm used is AlexNet, with two aggregation strategies, FedAvg and FedProx, compared against a centralized deep learning system. Results show comparable accuracy between federated and standard models, with future directions focusing on enhancing privacy with differential privacy. The article in reference [36] introduces DRFL, a FL approach for grading diabetic retinopathy (DR) using fundus images. A novel Convolutional Neural Network (CNN) captures local, global, and intermediate image features. The FedAvg method with median cross-entropy loss integration is proposed, enhancing performance, especially for clients with underfitting or overfitting issues. The central server extracts multi-scale features for lesion identification. The dataset includes 5000 images from various sources. The results show high accuracy (98.6%), specificity (99.3%), precision (97.25%), and F1 score (97.5%). The authors plan to involve hardware-based collaboration to handle distinct datasets in the future.A proposal for a framework targeting retinal microvasculature segmentation and diabetic retinopathy classification using optical tomographic images was presented in the article given in reference [37]. The dataset utilized consists of 153 OCTA images for segmentation and 700 eyes for diabetic retinopathy classification. Key performance metrics included accuracy, dice similarity coefficient (DSC), and area under the receiver operating characteristic curve (AUROC). The study demonstrated that FL achieves performance comparable to internal models, showcasing its significance in medical image analysis while prioritizing data privacy, a crucial aspect of FL.

In conclusion, the reviewed research exemplifies the pivotal role of privacy-preserving collaborative FL in disease detection, particularly in the context of diabetic eye diseases. Through innovative approaches and robust methodologies, these studies have demonstrated the efficacy of FL models in achieving comparable performance to centralized models while preserving data privacy and addressing challenges such as data heterogeneity and class imbalance. The findings highlight the significance of adopting FL frameworks for scalable and privacy-conscious disease detection across diverse healthcare settings, paving the way for enhanced patient care and medical image analysis in real-world applications. A table providing a structured summary of the objectives, datasets, methodologies, results, and key metrics from the papers is given in Table 1.

III. BACKGROUND

This section will discuss in detail the general FL framework and also provide its classification. The aggregation strategies and commonly used optimizers are also described here. The section also includes the selection of data, preprocessing methods, the architecture of the neural network model, and training procedures.

A. GENERAL FRAMEWORK OF THE PRIVACY-PRESERVING FEDERATED LEARNING DISTRIBUTED SYSTEM

FL is a distributed and collaborative learning framework that enables machine learning models to be trained across multiple devices in a decentralised manner. These devices are servers or edge devices that hold the raw data samples and do not exchange any data samples. The general framework of a FL scenario is described step by step in the paragraph given below.

A central server is designated which is responsible for initializing the FL process. The initialisation phase includes defining a global or central model at the server and setting its parameters. In the next step, the model's parameters and weights are sent to the clients who are involved in the learning process. Thereafter, all the participating client devices start training on their datasets held locally by them.

The clients do not share any of the intermediate states or data while training. It is only after the completion of training that the clients compute all the gradients, forming their local models also called client models. Then the gradients are sent to the central server which collects all the gradients. The next step is to aggregate and incorporate the gradients into the global model by adopting an aggregation strategy like averaging (unweighted) or weighted averaging, etc. Usually weighted averaging is employed to ensure that updates from devices with more relevant data, contribute better to building a robust global model. The newly built model is sent again to the client nodes for local training. The clients start training again on their locally held subsets of data using the updated weights sent by the server. They create new client models once more and send them to the server, which then integrates the updated weights and gradients. The process is iterated in

TABLE 1. A structured summary of the literature presented in the related work.

Paper Reference	OBJECTIVE	Dataset	Methodology	Results	Key Metrics
Roth et al. [30]	FL for breast density classification	142,952 mammography images from seven medical organizations	DenseNete121 backbone model, FedAvg algorithm, TensorFlow, NVIDIA Clara Train SDK, normalization, augmentation	Performance is measured by Cohen's Linear Weighted Kappa score	Kappa score
Zhang et al. [31]	FL with dynamic fusion-based technique	CT and X-ray images (3326 selected from 3706)	Client Participation and Selection, three backbone models- GhostNet, ResNet50, ResNet101	Better accuracy, and more training time required	Accuracy
Yang et al. [32]	Segment and classify 3D chest images for COVID detection	1704 chest CT scans, 28833 X-ray images	Semi-supervised learning, ResNet architecture at the backbone, FedAvg, weight pruning, weight quantization	Accuracy: 93.44% (central), 92.26% (federated), 92.24% (communication-efficient)	Accuracy, Dice's score
Khan and Alam [33]	Pneumonia detection using FL	Chest X-ray images (5856)	Canny filter, FL performance comparison with four backbone models - CNN, ResNet50, VGG16, AlexNet	VGG16 best-performing model	Accuracy, sensitivity, precision, specificity, AUC
Matta et al. [34]	FL for diabetic retinopathy detection	OPHDIAT (697,275 images)	EfficientNet-B5 model as the base model, weighted and unweighted aggregation	Centralized: AUC = 0.9482, Cross-grader= 0.9522, Cross-center = 0.9317,	AUC
Nasajpour et al. [35]	Federated transfer learning for DR detection	3000 images from five DR datasets	AlexNet as backbone DL model, FedAvg and FedProx aggregation, data privacy	Centralized Model: 92.19, FedAvg: 90.07% and FedProx: 85.81%	Accuracy, sensitivity, precision, specificity
Mohan et al. [36]	FL for grading diabetic retinopathy	5000 fundus images	Customized CNN as the central model, FedAvg, multi-scale feature extraction	Accuracy: 98.6%, Specificity: 99.3%, Precision: 97.25%, F1 score: 97.5%	Accuracy, Specificity, Precision, F1 score
Lo et al. [37]	Retinal microvasculature segmentation and DR classification	153 OCTA images, 700 eyes	VGG19 was used for feature extraction and 2 fully connected layers for classification, FL comparison with centralized methods	FL-Mean DSC: 0.793, Centralized- 0.807 and RDR Classification FL- AUROC of 0.954 and 0.960, Centralized- 0.956 and 0.973	Accuracy, Dice similarity coefficient (DSC), AUROC

a series of communication rounds until a predefined criterion is reached or the model converges and does not improve any further. The criteria could be attainment of a certain level of accuracy [38], [39]. The contrast between a centralized deep learning scenario and a distributed FL scenario is shown in a diagrammatic form in Figure 1, and the steps of FL are algorithmically given in Algorithm 1.

The FL framework was proposed to cater to privacy protection in a distributed environment. Since the data is present at different locations, in practicality, the amount and quality of data would vary significantly. Therefore, the scenario of IID data and non-IID data plays a crucial role in understanding the challenges and behaviours of machine learning models trained across decentralised datasets in the context of FL.

In FL, IID refers to the scenario where the data across different contributing clients is independent and identically distributed. In simpler words, it means each client's data is a fair representation of the overall data distribution. Thus, no matter where or which client the data comes from, it statistically shares the same properties (mean, variance, etc.) and is independent of each other. Due to this, training models in an IID setting are generally more straightforward because the data across all clients resembles the entire dataset. This similarity in data distribution helps in converging to a well-performing model quickly and efficiently, as each client's update is relevant and beneficial to the global model [40], [41].

In contrast, non-IID data in FL refers to the scenario where the data across different clients is not independent and identically distributed. This is a more common scenario in real-world applications, where each client's data might represent different user behaviours, preferences, or environments. Consequently, the data distribution varies significantly across clients or data-holding sites. Therefore, training on non-IID data presents several challenges like, models might perform well on data similar to what they were trained on but poorly on unseen or significantly different data distributions. Additionally, achieving convergence becomes harder as the global model tries to accommodate diverse data characteristics from all clients, which might sometimes be contradictory. Apart from this, there is a risk of model bias towards the majority data representation, leading to fairness issues for underrepresented groups [42], [43].

Algorithm 1 Generalised Federated Learning Framework

```
1. Initialization Phase: Initialize a global or central model at the server with parameters $\theta$.
2. Model Distribution: Send the parameters $\theta$ to participating client devices.
3. Client Training:
    i. Each of the clients trains a local model on their datasets.
    ii. After training, all clients compute gradients $\nabla\theta_i$ based on their local data, forming local models $\theta_i$.
4. Gradient Collection: Clients send their gradients $\nabla\theta i$ to the central server.
5. Aggregation: The central server aggregates gradients from all clients using an aggregation strategy (e.g., weighted averaging):
        $\nabla\theta = \frac{\Sigma_{i=1}^{N} w_i \cdot \nabla\theta_i}{\Sigma_{i=1}^{N} w_i}$
    where $N$ is the number of clients and $w_t$ is the weight assigned to the client $i$.
6. Model Distribution (Gradients): Send the updated gradients $\nabla\theta$ to all client nodes.
7. Client Model Update:
    i. Clients update their local models using the received gradients.
    ii. Clients train again on their locally held subsets of data, forming new local models $\theta i'$.
8. Model Aggregation: Clients send their updated local models $\theta i'$ to the central server.
9. Iteration: Repeat steps 4 to 8 for a predefined number of communication rounds or until a convergence criterion is met.
End
```

In order to mitigate the challenges posed by non-IID data, researchers and practitioners employ various strategies such as personalization, aggregation strategies, data augmentation, etc [44], [45], [46]. Personalization considers that instead of a one-size-fits-all model, personalized models must be trained to cater to the specific data distribution of individual clients. The advanced aggregation techniques, which are beyond simple averaging, more sophisticated aggregation methods like Federated Averaging (FedAvg) with modifications, weighted averaging, or clustering-based approaches are used to better handle the diversity in data distributions. Data preprocessing techniques like data augmentation or synthetic data generation can help in improving the quality of the data diversity at each client, making the local data more representative of the global distribution.

B. CLASSIFICATION OF FEDERATED LEARNING

FL can be classified based on the distribution and nature of the data samples across different clients. These classifications help in understanding the type of data distribution and collaboration setup in FL scenarios, guiding the design and implementation of appropriate FL algorithms [47], [48].

Horizontal Federated Learning (HFL): In HFL, also at times referred to as sample-based federated learning, the data sets are from different clients, they share the same feature space but differ in sample space. This means that each client has data of different users or entities but has the same set of features. For example, multiple banks may share a model to predict loan defaults, each with data about their customers but using the same financial attributes. In HFL the same features across clients are required and the data samples are different for different clients. An example of HFL can be a scenario where multiple hospitals collaborate to improve a diagnostic model, each hospital has their patient data with the same medical features but all the patient samples are different.

Vertical Federated Learning (VFL): VFL is referred to as feature-based FL, which occurs when the data sets are from different clients and also have different feature spaces but share the same sample space. This means that each client has different types of information about the same set of users or entities. For example, a bank and an e-commerce company might collaborate, where the bank has financial data and the e-commerce company has purchasing data about the same customers. The main characteristic feature of VFL is that different features across clients are required but the same data samples for each of the clients. Instantiating, a financial institution and a retail company collaborate to enhance a credit scoring model, with the financial institution providing banking details and the retail company providing purchase history.

Federated Transfer Learning (FTL): is used when both the feature spaces and sample spaces are different across clients. It is especially useful when there is limited overlap in the data distributions. This type of FL leverages transfer learning techniques to adapt models from one domain to another, addressing situations where the data heterogeneity is significant. In this type of FL the features, as well as the data samples are different for different clients. An example of FTL is where a healthcare provider and a wearable device manufacturer collaborate to improve health monitoring models, with the healthcare provider contributing medical records and the device manufacturer contributing activity data. A graphical presentation of the classification based on the sample space and feature space is shown in Figure 2.

C. AGGREGATION TECHNIQUES USED IN A COLLABORATIVE FEDERATED LEARNING FRAMEWORK FOR BUILDING AN EFFICIENT GLOBAL MODEL

The aggregation strategies are chosen according to the type of data, its distribution and the application at hand [49], [50]. Various aggregation strategies include FedAvg, FedProx, FedNova, MOON, Zeno, etc. A summary of each of these aggregation techniques is given subsequently.

1) FEDERATING AVERAGING

FedAvg is the earliest and most commonly used aggregation technique used in FL environments. In this type of aggregation, the participating clients are randomly selected in each round of learning. The parameters of the selected set of clients are weighted and then averaged to form a new set of parameters which form the updated global model. This global model is then distributed to all the client devices that use the updated parameters and train their local models. This

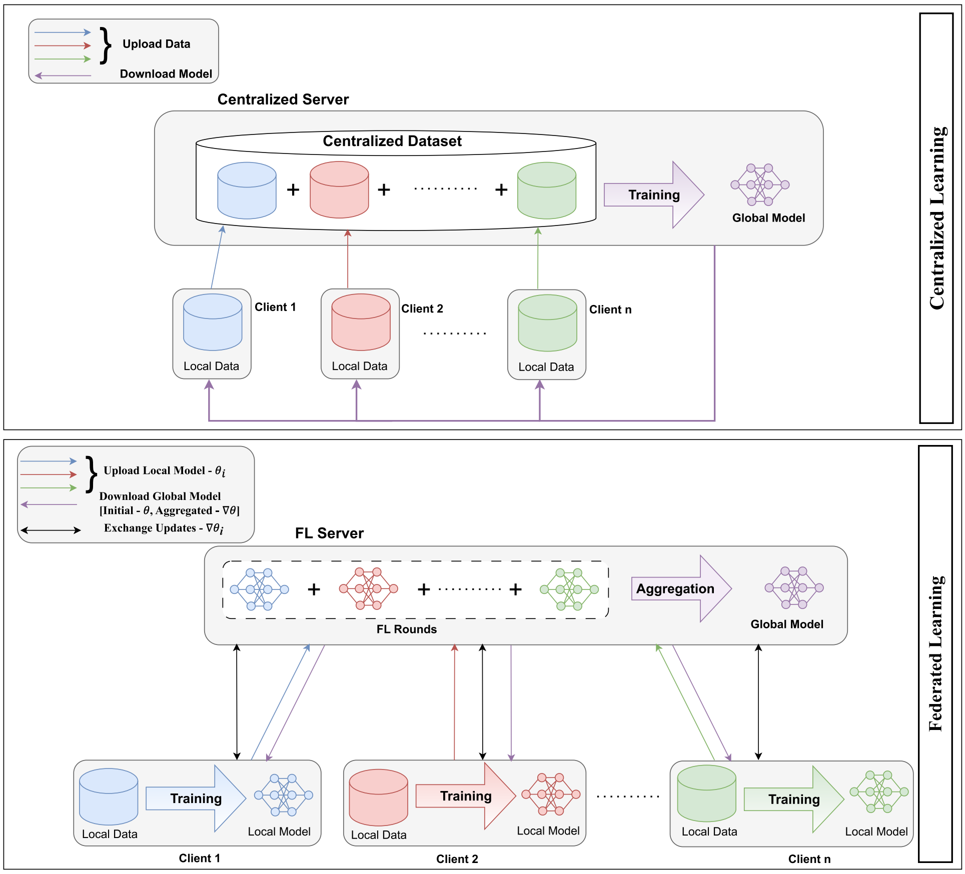


FIGURE 1. Graphical representation of the general framework of a federated learning scenario.

averaging strategy is suited for scenarios where the local data of the clients is large and IID (Independent and Identically Distributed). It is frequently used in applications with homogenously distributed data. In such applications usually, communication bandwidth is limited, and to decrease the communication messages for update transmission. Federated averaging is relatively simple to implement as well as computationally efficient, and effective in preserving model accuracy. Federated averaging is at times sensitive to devices with imbalanced or noisy data, which might lead to biased aggregation.

2) WEIGHTED FEDERATED AVERAGING

It is similar to the FedAvg the only difference is that weights are assigned based on factors other than dataset size, such as device reliability, data relevance, or model performance on validation data. Weighted federated averaging enhances the FedAvg process by introducing weights to the averaging step, where the influence of each client's update on the global model is adjusted based on certain criteria, most commonly the size of the local dataset. The most straightforward and commonly used weighting criterion is the size of the local dataset. In this approach, updates from clients that have more data samples have a greater share in the global model. This is motivated by the idea that clients with more data are likely to provide a more reliable and comprehensive update. Mathematically, w_i is the weight for the i-th client, which could be proportionate to the number of data points n_i it has, the global model update Wglobal is calculated as:

$$W_{global} = \frac{\sum_{i=1}^{K} w_i W_i}{\sum_{i=1}^{K} w_i}$$

where W_i is the update from the i-th client, and K is the total number of clients participating in that round. The Weighted Federated Averaging can mitigate some effects of non-IID data distributions amongst the different sets of clients by adjusting weights based on dataset sizes or potentially other criteria such as data diversity, data quality, or historical contributions to model performance. This is pivotal for real-world applications where data is inherently non-uniform across devices or users. Apart from this, the weighting mechanism can be designed to incorporate other factors, offering flexibility to tailor the FL process to specific application requirements, such as fairness, bias reduction, or incentivization for more active participation. Weighted Federated Averaging is a key development in making FL systems better and fairer, particularly where datasets vary greatly and are spread unevenly. This method boosts the

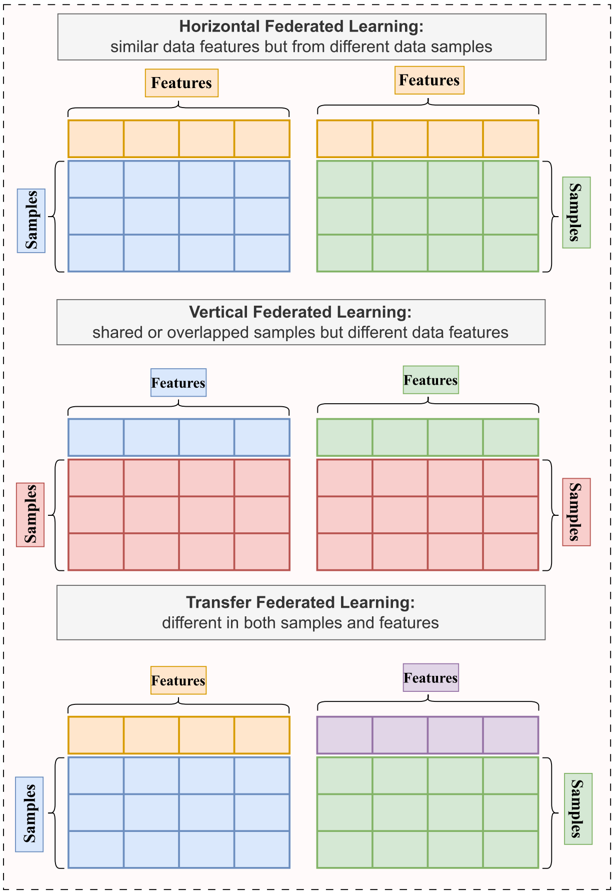

Horizontal Federated Learning:
similar data features but from different data samples
Features
Features
Samples
Samples
Vertical Federated Learning:
shared or overlapped samples but different data features
Features
Features
Samples
Samples
Transfer Federated Learning:
different in both samples and features
Features
Features
Samples
Samples

FIGURE 2. Classification of the privacy-preserving federated learning framework based on the dataset used.

model's accuracy and fairness by giving different weights to the inputs from various clients.

3) FEDPROX

It extends the FedAvg algorithm, the FedProx better handles issues related to system heterogeneity such as differences in client data distribution, computational capabilities, and availability. It introduces a proximal term to the local optimization problems solved by each client. This term penalizes the deviation of the local models from the global model, which helps to stabilize training across heterogeneously capable devices. Similar to FedAvg in FedProx each client trains a local model using its dataset. However, the objective function for training includes an additional proximal term. Mathematically, the local objective for client i can be represented as:

$$\min_{w} \left\{ f_i\left(w\right) + \frac{\mu}{2}\left\|w - w_t\right\|^2 \right\}$$

Here, $f_i\left(w\right)$ is the loss function based on the local data, W represents the local model parameters, WT is the global model parameters received from the server, and μ is a hyperparameter that controls the strength of the proximal term. The proximal term, $\frac{\mu}{2}\left\|w - w_t\right\|^2$ penalizes the deviation of the local model parameters from the current global model parameters. This term acts as a regularization that encourages the local models not to stray too far from the global model, mitigating the effect of extreme or noisy local updates that could result from non-IID data or varied local training epochs.

After training, each client returns their updated model parameters to the central server. The server aggregates these updates to create a new global model. This aggregation is usually a weighted average, with weights often proportional to the number of data points each client contributes, similar to the FedAvg method. FedProx is particularly useful in scenarios with significant system heterogeneity. It ensures that clients with less reliable internet connections or lower computational power can still contribute effectively without destabilizing the learning process.

4) FEDNOVA

Another advanced aggregation algorithm designed to enhance FL in heterogeneous environments, specifically addressing issues that arise when clients perform a different number of local updates. This is a common scenario in federated settings where clients have varying computational resources or data sizes. FedNova was developed to provide a more robust and efficient aggregation mechanism that improves upon traditional methods like FedAvg and FedProx, particularly in handling client heterogeneity in terms of the number of local updates (epochs). FedNova introduces a novel normalization technique in the aggregation process to deal with discrepancies in the number of local updates performed by different clients. The core idea is to normalize the local updates based on the number of local epochs each client performs before averaging them at the server. Each client trains a local model on their data. Unlike FedAvg, where each client typically performs the same number of epochs, in FedNova, clients can perform a different number of local updates or epochs based on their computational capabilities or availability. After training, instead of directly sending the model updates to the server, each client first normalizes their updates by the number of epochs they performed. This normalization is crucial to ensure that the contribution of each client's update to the global model is scaled appropriately, avoiding bias towards clients that performed more epochs. Mathematically, if client i performs ni epochs, and their update is Δwi, the normalized update is given by: $\Delta \tilde{w}_i = \frac{\Delta w_i}{n_i}$

This normalization helps to mitigate the skew in updates caused by varying numbers of epochs across clients. At the server, these normalized updates are aggregated to update the global model. The aggregation could be a simple average or a weighted average based on other factors like the amount of data each client has. FedNovais advantageous in many aspects such as having fair contribution by normalizing updates based on the number of local epochs, FedNova ensures that each client's contribution to the global model is proportional to their level of participation in terms of computation, rather than being overly influenced by clients that can afford more computation. Next, its stability and convergence, by normalizing updates the training process stabilizes and improves convergence properties, especially in highly heterogeneous environments where the number of local updates can vary significantly. Also, it has flexibility in client participation, which means FedNova allows for greater flexibility in client participation, accommodating clients with limited computational resources by enabling them to perform fewer epochs without diminishing their contribution to the mode's learning. Additionally, Fednova has efficiency in non-IID Settings since the normalization mechanism can also help in non-IID data distributions, as it reduces the potential dominance of any single client's data characteristics, which might be overrepresented due to more extensive local training.

5) MOON (MODEL-ORIENTED OPTIMIZATION NUDGING)

It is a relatively new aggregation strategy designed for FL to specifically handle scenarios where data is non-IID (Non-Independently and Identically Distributed) across clients. This situation is quite common in real-world applications of FL, where different clients may have data distributions that vary significantly. These discrepancies can lead to issues with model convergence and performance when using traditional aggregation methods like FedAvg. MOON focuses on enhancing model convergence and performance in non-IID settings by incorporating an auxiliary loss term that nudges each client's local model towards a consensus representation, which is represented by a model-oriented optimization approach. The key elements of MOON include the local model training with auxiliary loss and aggregation. In MOON, each client trains their model not only based on the local data but also includes an additional regularization term that encourages the model to be similar to a consensus representation. This consensus is generally derived from the global model parameters, ensuring that while the local models learn from their specific data, they also incorporate learnings that are common across the federation. The loss function for a client in MOON can be formulated as:

$$L_{total} = L_{local}\left(w, D^{i}\right) + \lambda.L_{aux}\left(w\ ,\ w_{g}\right)$$

L_{local} is the local loss function computed on the client's dataset, w represents the local model parameters, and w_g denotes the global model parameters. L_{aux} is the auxiliary loss term, often chosen as a distance metric (like Euclidean distance) between the local model parameters w and the global model parameters w_g. λ is a hyper-parameter that controls the strength of the regularization. After training, clients send their updated models to the server. The server then performs an aggregation of these models, typically using a weighted average approach. The weights can be proportional to the number of data points each client contributes or could be uniform across all clients.

D. THE OPTIMIZERS USED IN THE LEARNING PHASE OF A FEDERATED LEARNING FRAMEWORK

In FL, choosing the right optimizer is crucial for training models efficiently and effectively. Optimizers are methods used to change the attributes of the neural network such as weights and learning rate to reduce the losses. Some of the most commonly used optimizers in machine learning, each with unique characteristics and applications, are given below.

1) STOCHASTIC GRADIENT DESCENT (SGD)

It is a simple yet very effective approach to fitting linear classifiers and neural networks. It updates the parameters using a fixed learning rate multiplied by the gradient of the loss function concerning the parameters. SGD can also include a momentum term to smoothen the updates. It is used in basic deep learning tasks and when simplicity and speed are preferred over sophistication.

2) SGD WITH MOMENTUM

Momentum is an approach that helps accelerate SGD in the relevant direction and dampens oscillations. It does this by adding a fraction of the update vector of the previous step to the current ste's update vector. It is quite useful in scenarios where we need to escape local minima and accelerate convergence, particularly in deep or complex networks.

3) NESTEROV ACCELERATED GRADIENT (NAG)

It is a variant of momentum that attempts to correct the course of the momentum by calculating the gradient not at the current weights but at the approximate future position of the weights. It is generally employed for training models faster than standard momentum and when a more refined update path is needed.

4) ADAGRAD (ADAPTIVE GRADIENT ALGORITHM)

This optimizer adapts the learning rate to the parameters by performing larger updates for infrequent parameters and smaller updates for frequent parameters. Additionally, Adagrad uses a different learning rate for every parameter i at every time step t, based on the past gradients that have been computed for each parameter. It is great for dealing with sparse data and is commonly used in natural language processing (NLP) and other applications where the dataset has frequent and infrequent features.

5) ADADELTA

Adadelta is an extension of Adagrad that aims to reduce its aggressive, monotonically decreasing learning rate. It uses a sliding window of gradient updates rather than accumulating all past gradients, thus addressing Adagra's radically diminishing learning rates. It is fruitful in settings where Adagrad's performance suffers due to its continuously decreasing learning rates, particularly in large-scale and complex datasets.

6) RMSprop

This optimizer addresses the problem of rapidly decreasing learning rates in Adagrad by dividing the learning rate for a weight by a running average of the magnitudes of recent gradients for that weight. It is particularly effective in online and non-stationary settings (e.g., noisy data).

7) ADAM (ADAPTIVE MOMENT ESTIMATION)

Adam combines the advantages of two other extensions of stochastic gradient descent. Namely, it computes adaptive learning rates for each parameter from estimates of the first and second moments of the gradients, hence combining the benefits of AdaGrad and RMSprop. Adam is generally regarded as being fairly robust to the choice of hyperparameters and works well on large datasets and high- dimensional parameter spaces, making it popular in many deep-learning applications.

8) ADAMW

It modifies the way weight decay is incorporated for Adam, decoupling the weight decay from the optimization steps taken based on the Adam update rules. This change improves training performance and convergence. It is being used in cases where the L2 penalty can interfere with the Adam update step, leading to suboptimal results.

9) NADAM (NESTEROV-ACCELERATED ADAPTIVE MOMENT ESTIMATION)

It is a combination of NAG and Adam optimizers. It brings together Nestero's accelerated gradient and Ada's methodologies to provide a more responsive approach to training. It is suitable for tasks where both fast convergence and fine-tuned, adaptive gradient methods are required. These optimizers are integral tools in the machine learning practitioner's toolkit, each providing different mechanics and advantages for training deep learning models under various scenarios.

IV. MATERIALS AND METHODOLOGY

The dataset used in the proposed work, the general methodology of a privacy-preserving federated framework, and the proposed methodology followed for the proposed work are presented in this section.

A. DATASET DESCRIPTION

The dataset used in the current study has been taken from two different repositories available publicly on GitHub and Kaggle. The first part is taken from the repository owned by "Translational Visual Health Laboratory" and it has been uploaded on GitHub under the name "Oct and Eye Fundus Dataset" [51]. The second subset of the data is from the Indian Diabetic Retinopathy Image Dataset available on Kaggle as well as their website called "Indian Diabetic Retinopathy Image Dataset (IDRiD) website" [52].

The dataset from "Translational Visual Health Laboratory" eye fundus and OCT images for the study of diabetic macular edema and diabetic retinopathy. The dataset is composed of 1548 Eye Fundus images and 1113 Macular Optical Coherence Tomography (OCT) images taken between 2015 and 2022. Out of which only eye fundus images were used for the simulation. These images are in the '.jpg' format and have a size of 1000 × 1000 pixels. The IDRiD dataset comprises 1053 images with DME and 495 images without DME. Some of the images are quite dark and the microvasculature is not visible at all. Therefore, the images are manually enhanced and some are deleted altogether.

The images from IDRID are original coloured fundus images 516 in number and have a'.jpg' format. The size of the images is 4288 × 2848, with good resolution and clear. Thus, manual preprocessing was not required in the case of the IDRID dataset. The good-quality images from both datasets were shortlisted to form a dataset of 2,284 images. The final split of the coloured fundus images was 675 images of DME, 825 of DR, and 784 from the Normal category. Some of the sample images from the coloured fundus image dataset are shown in Figure 3. The figure contains a sample from each of the three categories: Diabetic Macular Edema, Diabetic Retinopathy, and Normal. Table 2 gives the numerical presentation of the classes used in the implementation and the data in Table 3 gives the number of samples used for each of these classes.

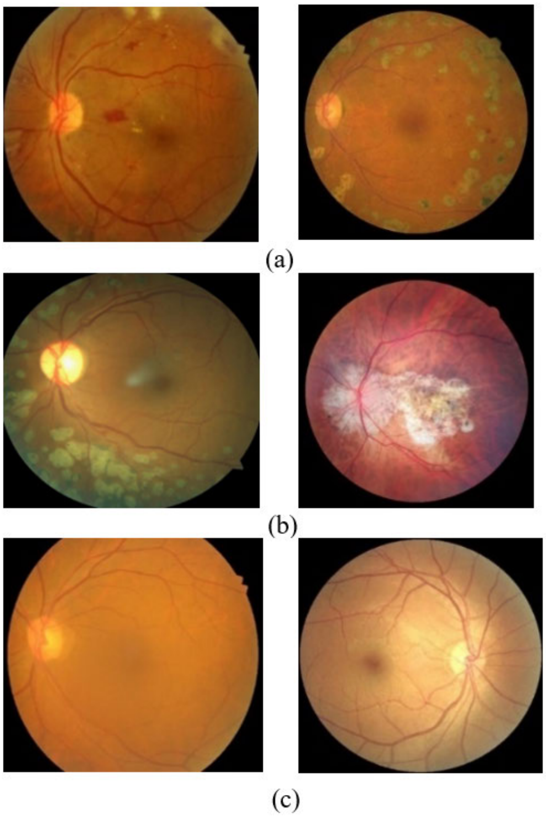


FIGURE 3. Images from the dataset of a sample eye suffering from (a) Diabetic Macular Edema, (b) Diabetic Retinopathy, and (c) Normal eye.

TABLE 2. Labels of the classes used in the implementation of the federated learning framework.

Class Number	Class Name
Class-0	DME
Class-1	DR
Class-2	Normal

TABLE 3. Bifurcation of the dataset into two subsets: training and testing.

	DME	DR	Normal	Total
Training	540	660	627	1827
Testing	135	165	157	457
Total	675	825	784	2284

B. METHODOLOGY

In the current work, a horizontal FL scenario is implemented using two aggregation techniques FedAvg and FedProx in two different scenarios where the data is IID (independently and identically distributed) and non-IID. Horizontal FL, also known as sample-based FL, is a machine learning technique where multiple organizations or devices collaboratively train a model without sharing their data. This approach is useful when the data held by different parties share the same feature space but have different sample spaces. In other words, each participant has data with the same features but different records. A horizontal FL scenario is explained in the diagram shown in Figure 4. A global model is initialized as per the architecture chosen and sent to all participants. Each participant then trains the model on their local data. After training, participants send their local model updates to a central server. The central server aggregates these updates to create an improved global model. The updated global model is sent back to the participants, and the process is repeated until the model converges.

The complete process of an HFL with aggregation scheme FedAvg being used at the server is given in the flowchart in Figure 5 and the algorithmic form with equations is given in Algorithm 2. FL with the FedAvg algorithm offers a transformative approach to machine learning that prioritises user privacy and regulatory compliance. By decentralising the training process and allowing data to remain local, FL opens up new possibilities for deploying machine learning in sensitive and regulated environments. However, it also requires careful management of heterogeneous data and resource constraints to ensure its successful implementation [53], [54]. FL with FedProx is an advanced version of the basic Federated Averaging (FedAvg) method, designed to handle challenges related to system heterogeneity—such as differences in data distribution, computational capabilities, and network connectivity among the participating clients.

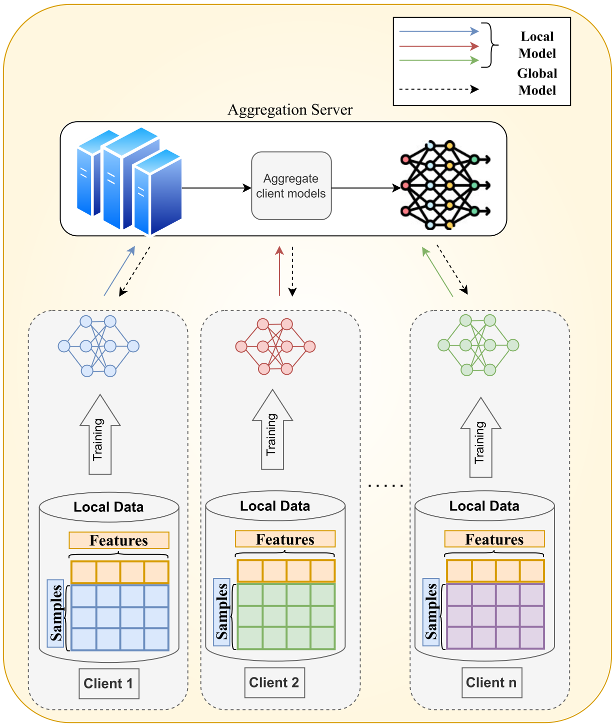


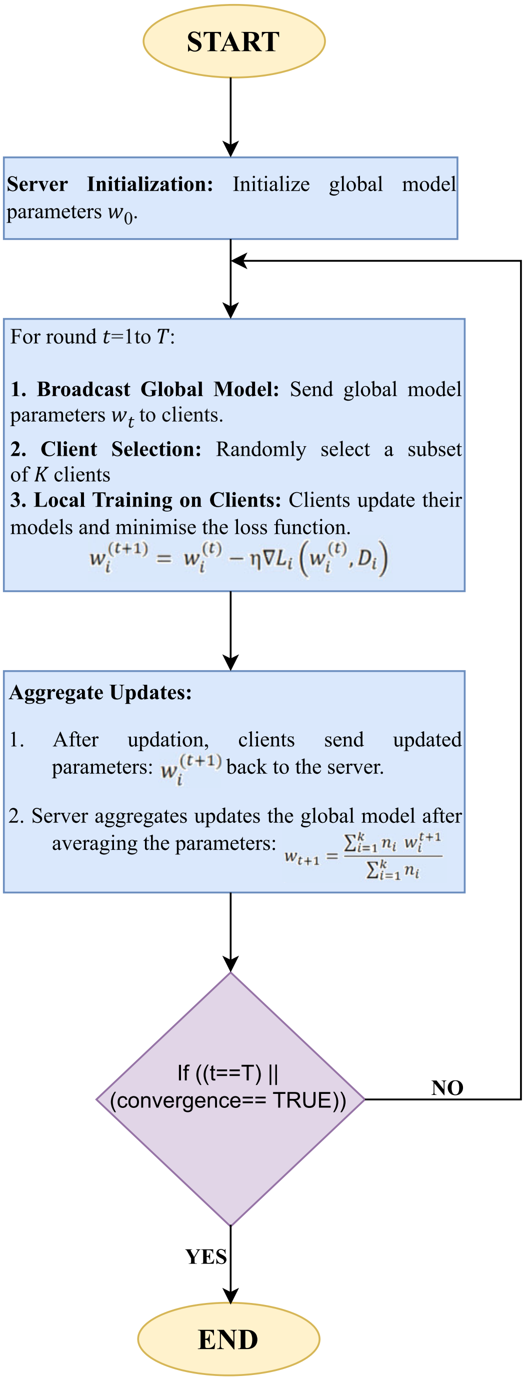
FIGURE 4. Horizontal federated learning architecture for maintaining the privacy of sensitive data.

FedProx modifies the FedAvg approach by introducing a proximal term to the local optimization problems solved by each client. This adjustment helps stabilize the training across devices with varying capabilities and ensures more consistent and robust global model performance. The FedProx algorithm extends the Federated Averaging (FedAvg) algorithm by incorporating a proximal term to handle system heterogeneity better [55], [56]. The algorithm is outlined step-by-step with relevant equations in Algorithm 3 given below and the flowchart for an HFL scenario with aggregation scheme FedProx aggregation mechanism at the server is given in the flow diagram in Figure 6.

FedProx effectively addresses challenges such as client data distribution, non-uniformity, and variability in client availability and computational power by incorporating a proximal regularization term. This term helps in maintaining the stability and consistency of the learning process across clients with diverse capabilities, thus enhancing the overall robustness of the federated learning system.

C. PROPOSED METHODOLOGY

This sub-section details the methodology followed in the current work to implement a privacy-preserving FL framework. The proposed method is based on the environmental setup where FL performance is compared in 2 scenarios, i.e., with IID and non-IID data. The aggregation mechanism also plays an important role in shaping the global model therefore two different aggregation strategies have been employed, FedAvg and FedProx. The complete set-up is 2 client-based and 3 client-based. The process is divided into eight scenarios based on the combination of these variables.



Server Initialization: Initialize global model parameters w_0.

For round t=1to T:

1. Broadcast Global Model: Send global model parameters w_t to clients.

2. Client Selection: Randomly select a subset of K clients

3. Local Training on Clients: Clients update their models and minimise the loss function.

$$w_i^{(t+1)} = w_i^{(t)} - \eta \nabla L_i \left(w_i^{(t)}, D_i\right)$$

Aggregate Updates:

1. After updation, clients send updated parameters: $w_i^{(t+1)}$ back to the server.

2. Server aggregates updates the global model after averaging the parameters: $w_{t+1} = \frac{\sum_{i=1}^{k} n_i \; w_i^{t+1}}{\sum_{i=1}^{k} n_i}$

If ((t==T) || (convergence== TRUE))

FIGURE 5. Flow diagram for implementing a horizontal federated learning with FedAvg aggregation at the central server.

Algorithm 2 Federated Learning Using FedAvg for Aggregation at the Cenral Server

```
1. Server Initialization:
    1.1 Initialize global model parameters $w0$.
    1.2 Choose hyperparameters like learning rate $\eta$,
        number of total communication rounds $T$, and
        number of clients to participate in each round $K$.
2. For each round $t = 1$to $T$:
    2.1 Client Selection: Randomly select a subset of $K$
        clients from the available pool.
    2.2 Broadcast Global Model: Send global model
        parameters $wt$ to each selected client.
    2.3 Local Training on Clients: Each client $i$ receives
        $wt$ and performs local training using their data.
        Specifically, each client updates their model by
        minimizing the local loss function $Li$ using an
        optimization algorithm (typically SGD):
            $w_i^{(t+1)} = w_i^{(t)} - \eta \nabla L_i \left(w_i^{(t)}, D_i\right)$
        where $Di$ is the dataset of client $i$ and $\nabla Li$ is the
        gradient of the loss function to the model
        parameters.
3. Aggregate Updates:
    3.1 After training, each client sends their updated
        parameters $w_i^{(t+1)}$ back to the server.
    3.2 The server aggregates these updates to update the
        global model using a weighted average, where
        weights are often proportional to the number of data
        samples each client holds:
            $w_{t+1} = \frac{\sum_{i=1}^{k} n_i w_i^{t+1}}{\sum_{i=1}^{k} n_i}$
        Here, $n_i$ is the number of samples the client $i$ has,
        and $K$ is the number of participating clients in round
        $t$.
4. Convergence Check:
    4.1 Evaluate the global model $wt+1$ on a validation set
        or check other convergence criteria (like improvement
        in loss less than a threshold).
    4.2 If the model has converged or the maximum number
        of rounds $T$ is reached, terminate the training.
        Otherwise, proceed to the next round.
        End of Training
```

In the initial phase, the dataset is carefully organized into three distinct folders: Diabetic Macular Edema, Diabetic Retinopathy, and Normal. Each of these folders contains images intended for both training and testing the models. Following the organization of the dataset, the subsequent step involves defining transformations for resizing, cropping, normalizing, and augmenting the images. These transformations are crucial for maintaining consistency in the input data format. Subsequently, the dataset is split into training and testing sets.

In scenarios involving non-IID settings, the data split ensures the reflection of realistic discrepancies in data distribution across different clients. The image ratio for training and testing is 80:20, and number of images used for training is 1827, and for testing is 457. The setup for FL is configured to accommodate scenarios with 2 and 3 clients. Separate data loaders are prepared for each client,

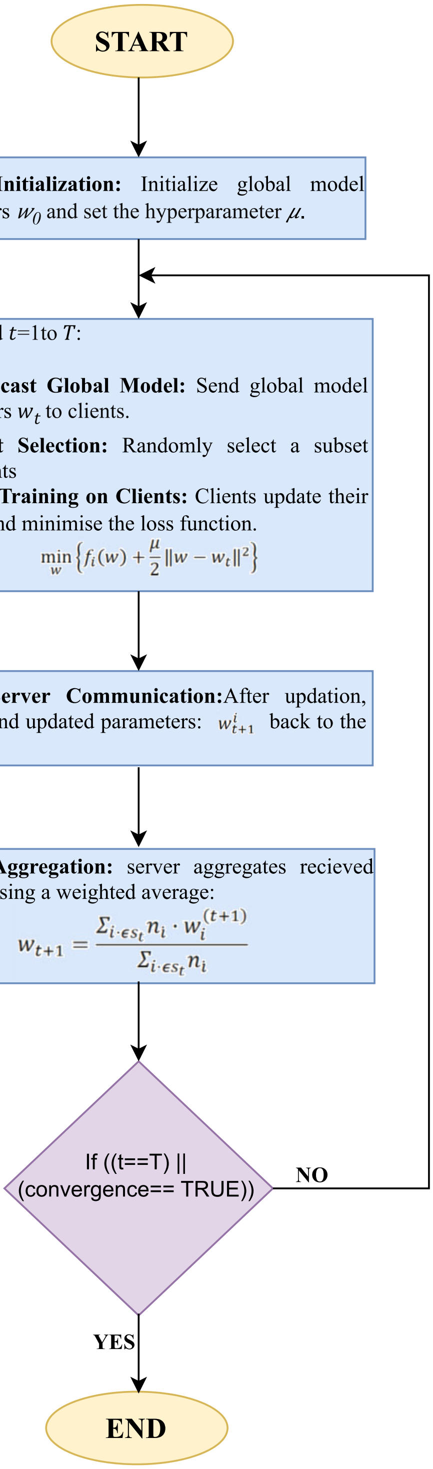

START
Server Initialization: Initialize global model parameters w_0 and set the hyperparameter μ.
For round t=1to T:
1. Broadcast Global Model: Send global model parameters w_t to clients.
2. Client Selection: Randomly select a subset of K clients
3. Local Training on Clients: Clients update their models and minimise the loss function.
$\min_w \left\{ f_i(w) + \frac{\mu}{2} \|w - w_t\|^2 \right\}$
Client Server Communication:After updation, clients send updated parameters: w_{t+1}^i back to the server.
Global Aggregation: server aggregates recieved updates using a weighted average:
$w_{t+1} = \frac{\Sigma_{i \cdot \epsilon s_t} n_i \cdot w_i^{(t+1)}}{\Sigma_{i \cdot \epsilon s_t} n_i}$
If ((t==T) \|\| (convergence== TRUE))
NO
YES
END

FIGURE 6. Flowchart for implementing a horizontal federated learning with FedProx aggregation at the central server.

considering both IID and non-IID distributions. For non-IID distributions is 40:60 samples for the 2-client scenario and 30:30:35 samples for a 3-client scenario. In IID settings the data distribution is equal in accordance with the number of clients.

Algorithm 3 Federated Learning Using FedProx for Aggregation at the Central Server

```
1.   Initialization:
  1.1  The server initializes global model
       parameters, $w_0$.
  1.2  Hyperparameter $\mu$ is set, controlling the strength
       of the proximal term.
2.   Server Operation:
  2.1  For each round $t = 1, 2, \ldots, T$:
         o   Server selects a subset $S_t$ of clients.
         o   Server sends the global model parameters $w^t$ to
             each selected client in $S_t$.
3.   Client Training:
  3.1  Each client $i$ in $St$ receives wt and performs local
       training by solving:
           $\min_w \left\{ f_i(w) + \frac{\mu}{2} \|w - w_t\|^2 \right\}$
         o   $f_i(w)$ is the local loss function computed
             using client $i$'s local data.
         o   $\|w - w_t\|^2$ represents the Euclidean
             distance between the local model
             parameters $w$ and the global model
             parameters $w_t$ penalizing deviations.
  3.2  Each client trains the model using its dataset and
       optimizes the above objective function. The
       training could involve multiple iterations of gradient
       descent or another optimization technique suitable
       for the client's computational capabilities.
4.   Client to Server Communication:
  4.1  After training, each client $i$ sends the updated
       model parameters $w_{t+1}^i$ back to the server.
5.   Global Aggregation:
  5.1  The server aggregates the received updates using
       a weighted average:
           $w_{t+1} = \frac{\Sigma_{i \cdot \epsilon s_t} n_i \cdot w_i^{(t+1)}}{\Sigma_{i \cdot \epsilon s_t} n_i}$
       where $n_i$ is the number of data points or
       the degree of participation weight of client
       $i$, and $w_i^{(t+1)}$ are the parameters returned
       by the client $i$.
6.   Convergence Check:
  6.1  The server checks for convergence. If the
       convergence criteria are met, the algorithm
       terminates. Otherwise, it proceeds to
       the next round.
7.   End of Training
  7.1  After the final round, the server holds the
       globally trained model parameters $w_T$, which
       are expected to be robust to client
       heterogeneity.
```

Furthermore, the FedAvg methods are configured for use in different scenarios and later on the FedProx method is configured, necessitating modifications in the local training function to incorporate the proximal term in FedProx. The model architecture used at the backbone of the FL setup is the MobileNet. This model is carefully selected and adapted to classify the three categories: Diabetic Macular Edema, Diabetic Retinopathy, and Normal. The final layers of these models are adjusted to output three classes. The process of local training on individual clients involves implementing a dedicated training function. This function encompasses data feeding, forward and backward propagation, and optimization. The specifics of this function are adapted based on the selected aggregation method, be it FedAvg or FedProx. Model updates from individual clients are aggregated using either FedAvg or FedProx, depending on the scenario. In cases where FedProx is utilized, a proximal term is introduced to effectively manage system heterogeneity and mitigate client drift.

After each training round, the global model undergoes evaluation on a centralized test set to gauge metrics such as accuracy, sensitivity, and specificity. Scenario-specific validation procedures are conducted to discern the impact of client numbers, data distribution, and aggregation methods on model efficacy.

Iterative rounds of training, updating, and evaluation are carried out to refine the models. In each round, the number of epochs of training is set to 5 for each client and the total number of such rounds is 20. Adjustments to learning rates and other hyperparameters are made based on feedback obtained from model performance. The initial learning rate is set to 0.001 and the optimizer used is the Adam optimizer. This optimizer is chosen since adaptive learning rates, momentum optimization, bias correction, efficient memory usage, versatility, and automatic parameter tuning make Adam a powerful optimizer for training deep neural networks, leading to faster convergence and improved performance on a variety of tasks. A tabular representation summarizing all the parameters used in the implementation of the FL scenario is given in Table 4.

TABLE 4. Parameters used in the implementation of the federated learning model.

Parameter	Value
Model Architecture	MobileNet V2 using Horizontal Federated Learning
Clients	2, 3
Dataset	Diabetic Macular Edema, Diabetic Retinopathy, and Normal.
Dataset Size	2,284 images
Learning Rate	0.001
Optimizer	Adam
Batch Size	32
Epochs (per client for each communication round)	5 (per round)
Communication Rounds	20
Aggregation Strategy	FedAvg and FedProx
Augmentation	Yes
Augmentation Techniques	Rotation, Flipping
Train Test Ratio	80:20
Performance Metrics	Accuracy, Precision, Recall, and F1-score

Comprehensive documentation of performance metrics from each scenario is conducted. Visualization tools are leveraged to graphically represent performance across different configurations. Insights derived from the analysis shed light on the efficacy of diverse FL setups, thereby offering valuable recommendations for real-world applications. Upon identification of optimal parameters and models, the final step entails preparing the model for deployment in clinical settings or for further research studies.

After the parameters, a flowchart of the experimental setup for horizontal FL using FedAvg and FedProx algorithms is shown in Figure 7. It illustrates the communication process between clients and the central server, highlighting the steps of model distribution, local training, and aggregation of model updates. The figure emphasizes the iterative nature of the FL process, showing how the global model is continuously refined through multiple rounds of local training and centralized aggregation. This visual representation helps in understanding the flow of data and model updates, ensuring clarity in the implementation and analysis of the FL framework used for detecting diabetic eye diseases. Furthermore, Figure 8 presents a pictorial representation of the IID (Independent and Identically Distributed) and non-IID (Non-Independent and Identically Distributed) data distributions in a horizontal FL scenario involving 2 and 3 client simulations. The figure visually differentiates between the two data distribution scenarios and highlights how data is split across different clients.

V. RESULTS AND ANALYTICAL DISCUSSION

This section shows the results obtained in a graphical form for an analytical discussion.

The proposed federated deep learning framework results have been obtained by implementing the proposed methodology on Google Colab with the NVIDIA T4 GPU virtual machine using TensorFlow framework version 2.17.0 integrated with PyTorch Library on a windows system with 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz processor and 8 GB RAM installed.

A. DISCUSSION AND ANALYSIS OF RESULTS OBTAINED FROM SIMULATIONS OF 2 CLIENTS

The figures 9 (a) to 9 (p) focus on the local results obtained on the client end and performance analysis of the federated deep learning framework with MobileNetV2 architecture using FedAvg and FedProx aggregation methods under different data distribution scenarios (IID and non-IID) with two clients.

Figures 9 (a) to 9 (d) show the accuracy scores achieved by FedAvg and FedProx in IID and non-IID scenarios for 2 clients. FedProx performs slightly better in both scenarios, especially in non-IID, where it demonstrates greater stability and higher accuracy than FedAvg. Figures 9 (e) to 9 (h) compare the precision scores for FedAvg and FedProx in 2-client IID and non-IID data. FedProx again maintains a better precision score, especially in the non-IID scenario, highlighting its robustness in heterogeneous environments. Figures 9(i) to 9(l) display the recall scores for FedAvg and FedProx in IID and non-IID scenarios. FedProx shows

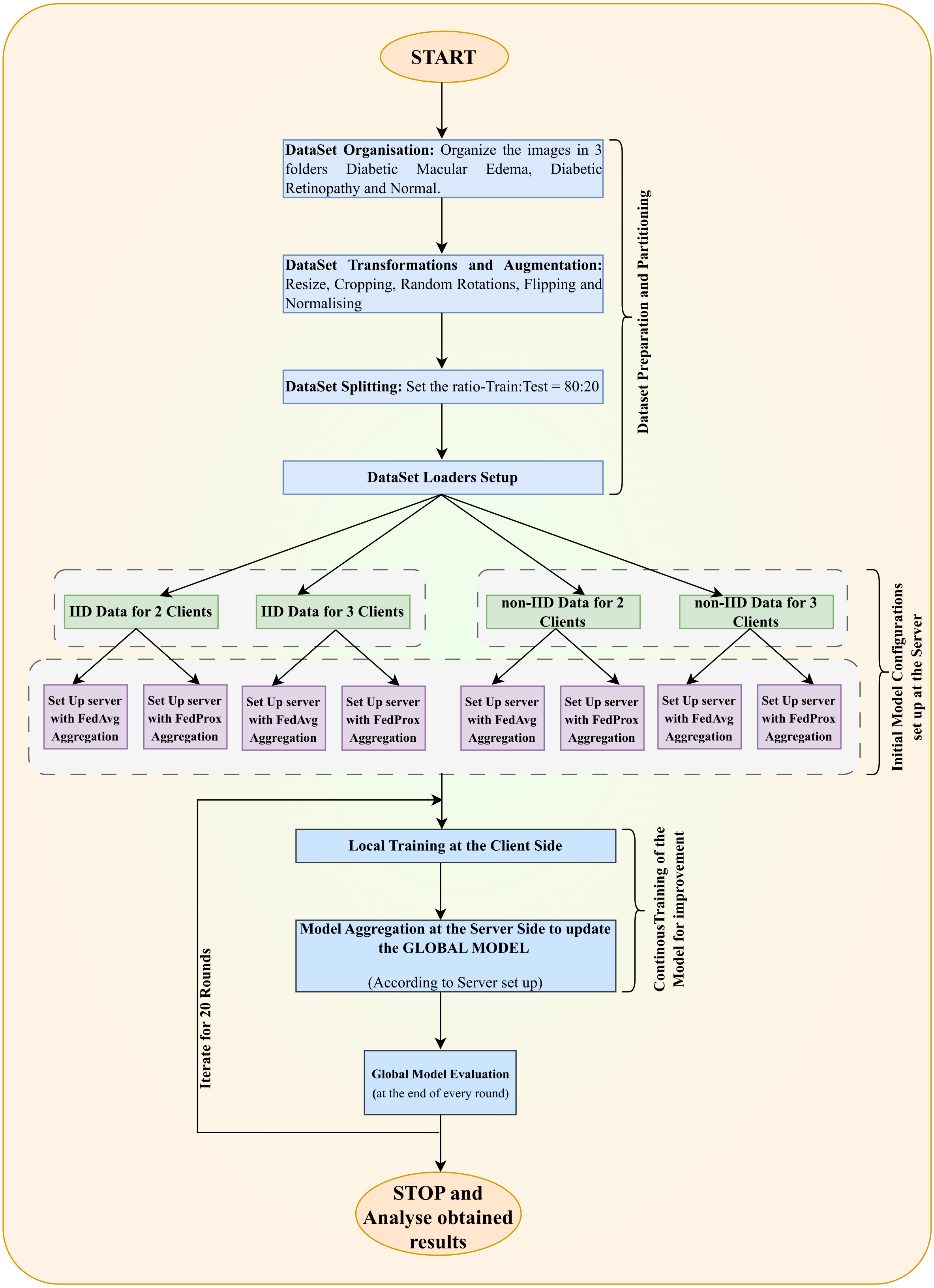

START
DataSet Organisation: Organize the images in 3 folders Diabetic Macular Edema, Diabetic Retinopathy and Normal.
DataSet Transformations and Augmentation: Resize, Cropping, Random Rotations, Flipping and Normalising
DataSet Splitting: Set the ratio-Train:Test = 80:20
DataSet Loaders Setup
Dataset Preparation and Partitioning
IID Data for 2 Clients
IID Data for 3 Clients
non-IID Data for 2 Clients
non-IID Data for 3 Clients
Set Up server with FedAvg Aggregation
Set Up server with FedProx Aggregation
Set Up server with FedAvg Aggregation
Set Up server with FedProx Aggregation
Set Up server with FedAvg Aggregation
Set Up server with FedProx Aggregation
Set Up server with FedAvg Aggregation
Set Up server with FedProx Aggregation
Initial Model Configurations set up at the Server
Local Training at the Client Side
Model Aggregation at the Server Side to update the GLOBAL MODEL
(According to Server set up)
ContinousTraining of the Model for improvement
Iterate for 20 Rounds
Global Model Evaluation
(at the end of every round)
STOP and Analyse obtained results

FIGURE 7. Flowchart of the horizontal federated learning using FedAvg and FedProx algorithms.

superior recall rates in non-IID data, indicating its better ability to identify true positive cases across diverse data distributions. Figures 9 (m) to 9 (p) show the F1-score, a balanced metric combining precision and recall. FedProx

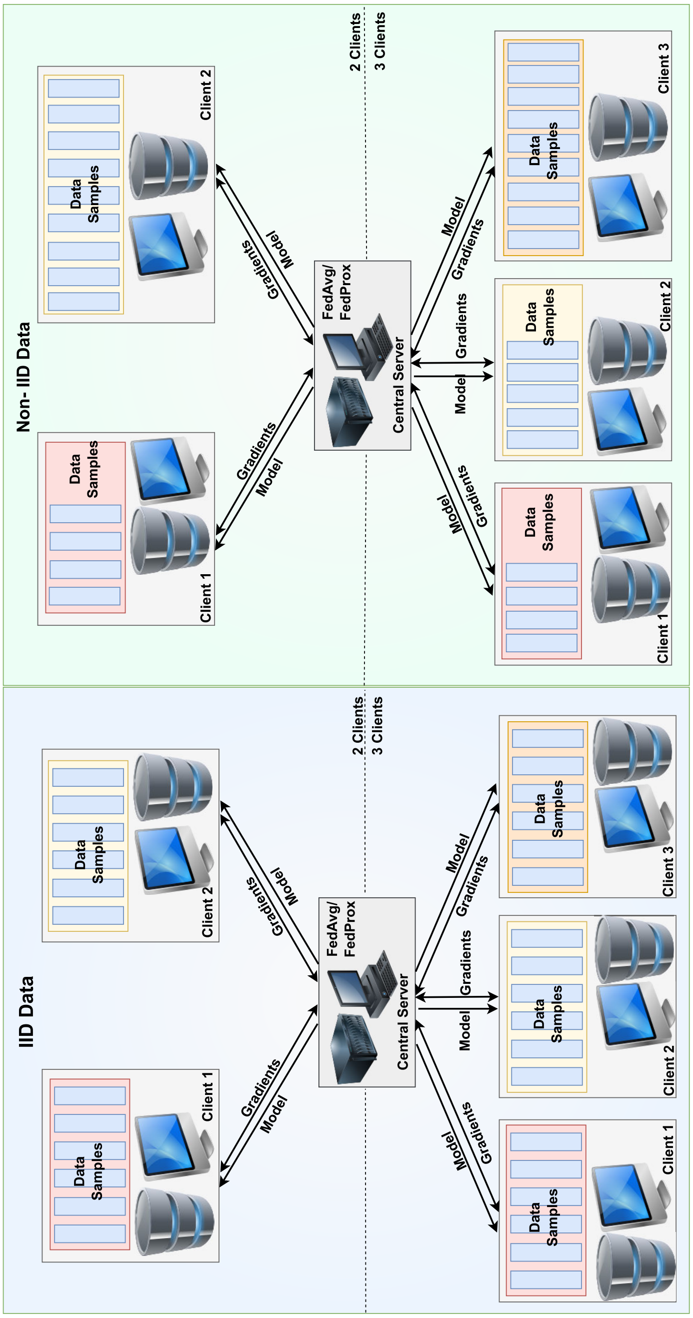
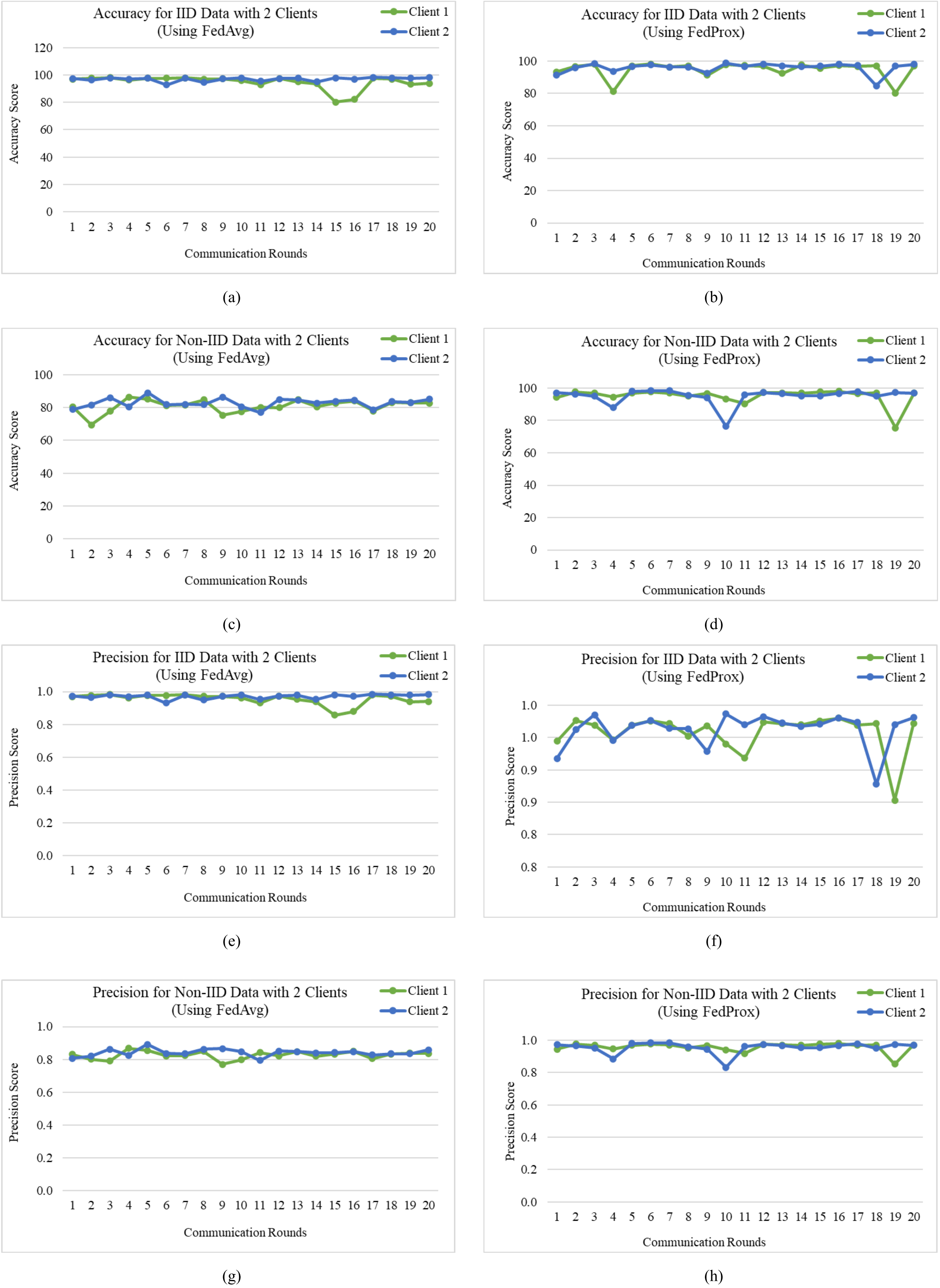


FIGURE 8. Pictorial presentation of the IID and non-IID data distributions in a horizontal federated learning scenario with 2 and 3 client simulations.

consistently outperforms FedAvg, especially in non-IID scenarios, due to its ability to handle data heterogeneity and maintain model stability.

The figures 10 to 17 focus on the performance evaluation of the global model. Hence the figures are for the global metrics of the MobileNetV2 based federated deep learning model for detecting retinal diseases using FedAvg and FedProx on IID and non-IID data with 2 clients.

The graphs in Figures 10 and 11 present the accuracy results of two aggregation methods, FedAvg and FedProx, for handling IID and non-IID data configurations in a 2-client FL scenario. Referring to Figure 8, FedAvg shows high accuracy levels, which are generally consistent and above 90% for most of the communication rounds. This indicates that FedAvg performs well when data across clients is homogeneously distributed (IID), which simplifies the learning process as the data is similar and evenly representative of the overall dataset. FedProx also performs well, often slightly better than FedAvg in the same IID scenario. The enhancements in FedProx, particularly the proximal term, help in maintaining model


Accuracy for IID Data with 2 Clients (Using FedAvg)

(a)

Accuracy for IID Data with 2 Clients (Using FedProx)

(b)

Accuracy for Non-IID Data with 2 Clients (Using FedAvg)

(c)

Accuracy for Non-IID Data with 2 Clients (Using FedProx)

(d)

Precision for IID Data with 2 Clients (Using FedAvg)

(e)

Precision for IID Data with 2 Clients (Using FedProx)

(f)

Precision for Non-IID Data with 2 Clients (Using FedAvg)

(g)

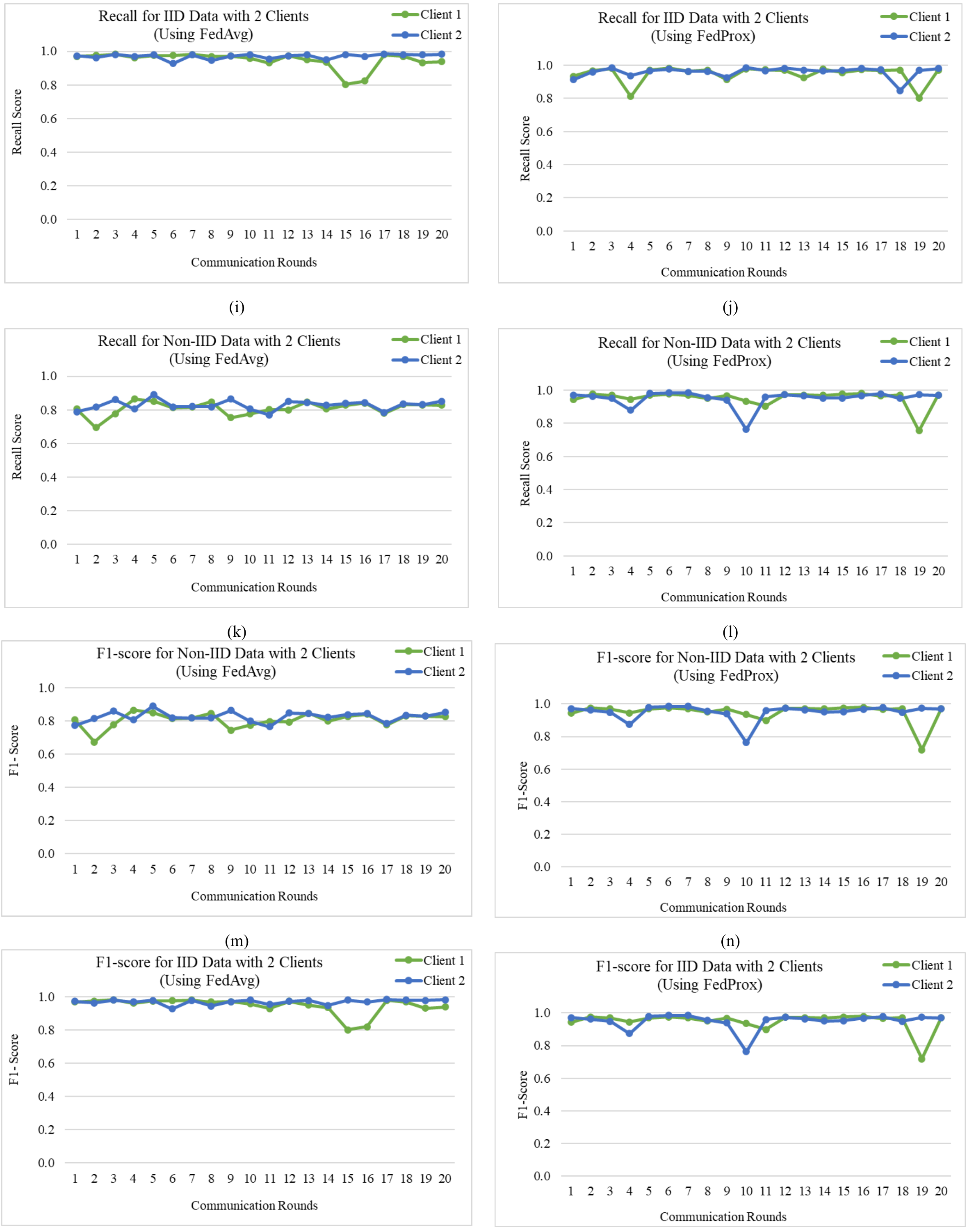
Precision for Non-IID Data with 2 Clients (Using FedProx)

(h)

FIGURE 9. Pictorial presentation of the Local Model for IID and non-IID data distributions in a horizontal federated learning scenario with 2 clients using FedAvg and FedProx algorithms.

stability and robustness even when the data is already IID, thus occasionally providing marginal gains over FedAvg.

Referring to Figure 10, FedAvg experiences a noticeable drop in performance when handling non-IID data. The



(i) (j)

(k) (l)

(m) (n)

(o) (p)

FIGURE 9. ***(Continued.)*** **Pictorial presentation of the Local Model for IID and non-IID data distributions in a horizontal federated learning scenario with 2 clients using FedAvg and FedProx algorithms.**

accuracy significantly fluctuates and is generally lower compared to the IID scenario. This variation suggests that FedAvg struggles with the diversity and complexity introduced by non-IID data, where each client's data might represent different subsets with varying characteristics. FedProx demonstrates a clear advantage in the non-IID

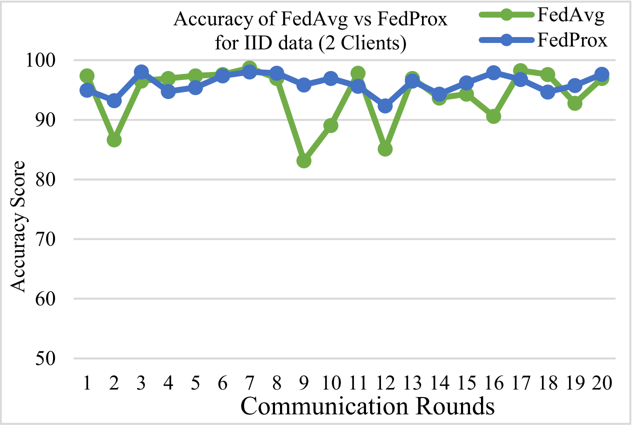


FIGURE 10. Representation of the accuracy achieved by FedAvg and FedProx for IID data in 2 clients.

setting, consistently maintaining higher accuracy levels across rounds. The proximal term in FedProx likely helps in mitigating the divergence of local updates due to the heterogeneous data, ensuring that the global model remains robust and less susceptible to discrepancies in local data distributions.

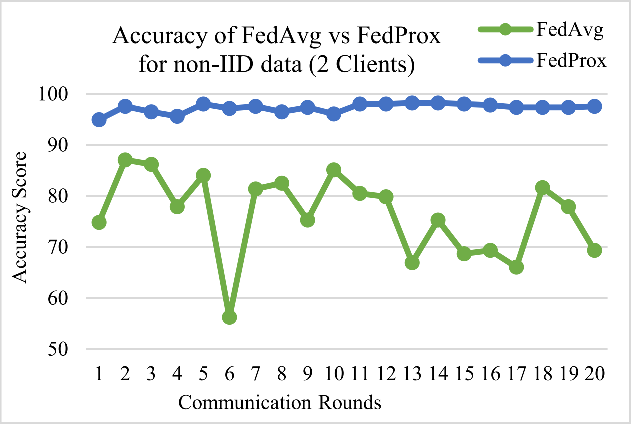


FIGURE 11. Representation of the accuracy achieved by FedAvg and FedProx for non-IID data in 2 clients.

The precision score comparison of FedAvg and FedProx for IID data in 2 Clients is graphically presented in Figure 12, which is given below. The precision scores for IID data show similar trends to the accuracy scores. Both aggregation methods perform well, but FedProx generally achieves slightly higher precision. This indicates that FedProx not only predicts accurately but also returns more relevant results per query, which is crucial in medical image analysis like diabetic retinopathy detection.

The precision score comparison of FedAvg and FedProx for Non-IID data in 2 Clients is shown in a graphical form in Figure 13. For non-IID data, the precision of FedAvg decreases, reflecting its struggle with diverse data sources. In contrast, FedProx maintains high precision levels, reinforcing its robustness in heterogeneous environments. This makes FedProx a preferred choice in practical applications where client data might not be uniformly distributed. In the analysis of precision scores from FL methods FedAvg and FedProx across IID and non-IID data scenarios, it is evident that both methods perform adequately in IID settings, with FedProx slightly outperforming FedAvg.

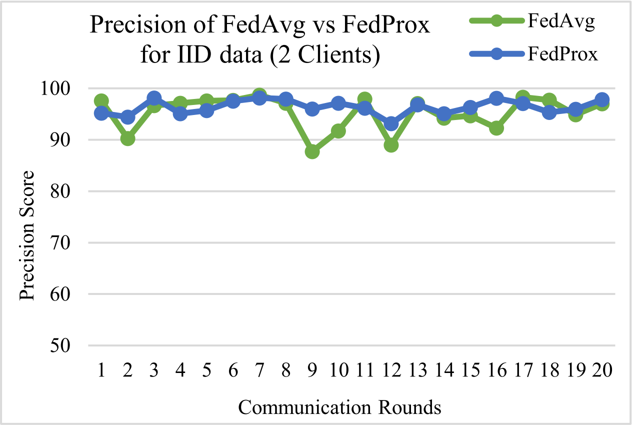


FIGURE 12. Representation of the precision score obtained by FedAvg and FedProx for IID data in 2 clients.

However, the distinction becomes notable in non-IID scenarios where FedAvg shows a marked decrease in precision, indicating difficulty working with diverse and heterogeneous data. In contrast, FedProx maintains robust precision scores, demonstrating its superior ability to handle non-uniform data distributions effectively.

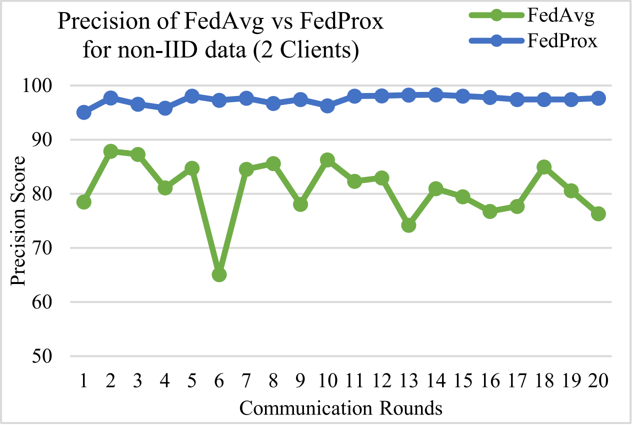


FIGURE 13. Representation of the precision score obtained by FedAvg and FedProx for non-IID data in 2 clients.

The recall score of FedAvg and FedProx for IID Data in 2 Clients is graphically presented in Figure 14. Both FedAvg and FedProx show high recall scores for IID data, with FedProx occasionally outperforming FedAvg. High recall indicates that the models are capable of identifying most positive cases correctly, which is essential for medical diagnostics. The recall score of FedAvg and FedProx for Non-IID Data in 2 Clients is depicted in Figure 15.

The recall graph for non-IID data highlights the challenges FedAvg faces in environments with data variance across clients. FedProx exhibits superior recall rates, suggesting it is better at managing the diversity in training data and effectively recognizing positive cases across different client datasets.

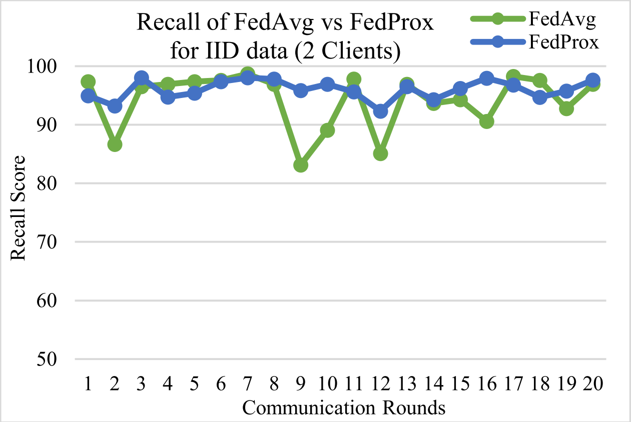


FIGURE 14. **Representation of the recall score obtained by FedAvg and FedProx for IID data in 2 clients.**

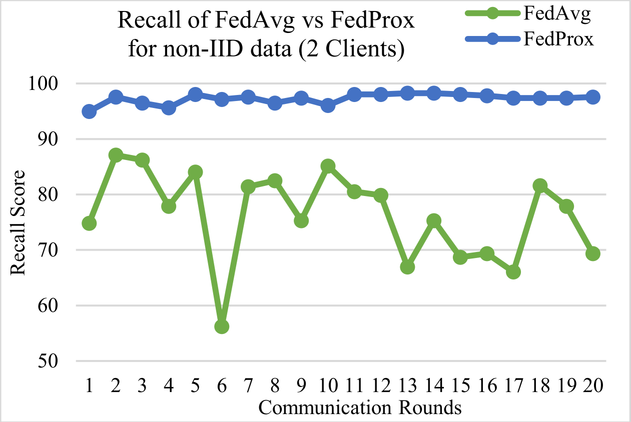


FIGURE 15. **Representation of the recall score obtained by FedAvg and FedProx for non-IID data in 2 clients.**

Figures 16 and 17 depict the FL methods FedAvg and FedProx performance in terms of F1-score across different data distributions (IID and non-IID) in a 2-client scenario. In Figure 16, FedAvg generally achieves a high F1-score, consistently remaining above 90% across most rounds. This indicates an effective balance between precision and recall, which is crucial in scenarios where both false positives and false negatives are costly. FedProx also shows high F1 scores, comparable to or slightly better than those of FedAvg in some rounds. The proximity term included in FedProx may contribute to this performance by helping to stabilize the model even in the IID setting, where data homogeneity typically simplifies training. Both methods display relatively stable performance across the rounds, with minor fluctuations. This stability is a positive indicator of their robustness in IID environments where the data distribution across clients does not vary significantly.

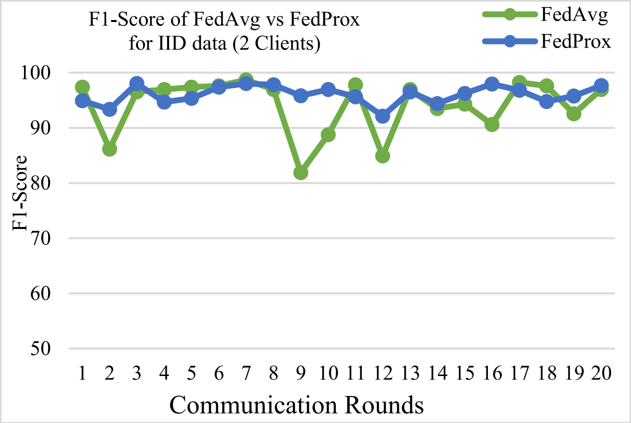


FIGURE 16. **Representation of the F1-score obtained by FedAvg and FedProx for IID data in 2 clients.**

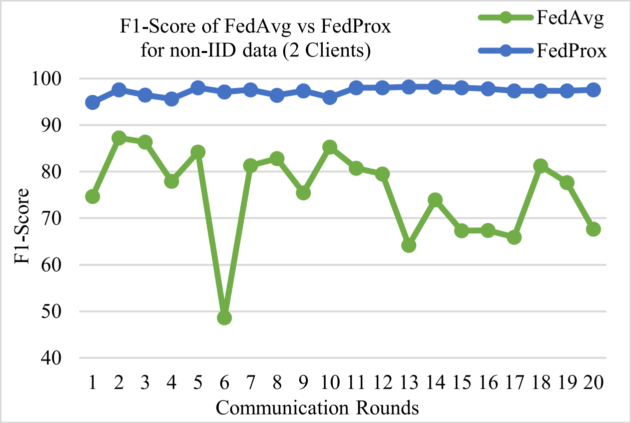


FIGURE 17. **Representation of the F1-score obtained by FedAvg and FedProx for non-IID data in 2 clients.**

In Figure 17, FedAvg shows more variability and a generally lower F1 score than the IID scenario. The scores are noticeably reduced, indicating challenges in handling non-IID data effectively. This reduction reflects difficulties in maintaining a balance between precision and recall when data distributions vary significantly across clients.

FedProx maintains a higher F1-score than FedAvg, indicating superior performance in non-IID settings. This suggests that FedProx is more adept at managing discrepancies in data distribution among clients. Its methodology, which incorporates a regularization term, seems to mitigate the impact of data heterogeneity more effectively.

The gap between the performance of FedAvg and FedProx is more pronounced in the non-IID setting than in the IID setting. FedProx's resilience in maintaining higher F1-scores highlights its potential for applications where client data is inherently diverse and non-uniform. Overall, Figures 16 and 17 reveal that both FedAvg and FedProx perform well in IID data scenarios, but FedProx demonstrates superior resilience and consistently higher F1-scores in non-

IID settings, effectively managing data heterogeneity across clients.

B. DISCUSSION AND ANALYSIS OF RESULTS OBTAINED FROM SIMULATIONS OF 2 CLIENTS

The figures 18 (a) to 18 (p) focus on the local results obtained on the client end after training and performance analysis of the federated deep learning framework with MobileNetV2 architecture using FedAvg and FedProx aggregation methods under different data distribution scenarios (IID and non-IID) with three clients.

The Figures 18(a) to 18(d) depict the accuracy achieved by FedAvg and FedProx across both IID and non-IID settings. FedProx shows slightly higher and more stable accuracy in non-IID settings compared to FedAvg, demonstrating its robustness in heterogeneous environments. The Figures 18(e) to 18(h) compare the precision scores of FedAvg and FedProx. Similar to accuracy, FedProx outperforms FedAvg in precision, especially in non-IID data scenarios. The Figures 18(i) to 18(l) illustrate the recall scores, where FedProx again exhibits superior performance, particularly in non-IID settings, indicating its ability to identify true positives effectively. The Figures 18(m) to 18(p) display the F1-scores, a balanced metric combining precision and recall. FedProx consistently outperforms FedAvg, with the advantage being more pronounced in non-IID settings due to better handling of data heterogeneity.

The figures 19 to 26 focus on the global model performance of the federated deep learning framework with FedAvg and FedProx on IID and non-IID data with three clients.

Figures 19 and 20 provide insights into the performance of FedAvg and FedProx in handling IID and non-IID data distributions respectively, in a scenario involving 3 clients. FedAvg displays relatively high accuracy, mostly above 90% throughout the rounds. Thus FedAvg can effectively coordinate and aggregate updates from three clients when the data is uniformly distributed (IID), leading to a robust model. FedProx performs slightly better than FedAvg, achieving nearly perfect accuracy in several rounds. The proximal term in FedProx likely contributes to this slight edge by stabilizing the model updates even in a triple-client setup, enhancing the convergence towards an optimal model. Both methods show consistency in their performance, with FedProx exhibiting slightly less variability and maintaining peak accuracy levels more frequently.

In Figure 19, FedAvg shows a significant drop in performance compared to the IID scenario, with more fluctuation in accuracy across rounds. This highlights FedAvg's challenges in managing the more complex and varied data structures inherent in non-IID settings across multiple clients. FedProx maintains higher accuracy than FedAvg, although it too experiences some reduction compared to the IID data scenario. The superior performance of FedProx suggests that its methodology, particularly the inclusion of a proximal term, effectively mitigates the impact of data diversity, helping to maintain model integrity and convergence.

The gap between the performances of FedAvg and FedProx is more pronounced in the non-IID setting. This gap reinforces FedProx's robustness and adaptability in environments where client data distributions are not identical, a common situation in real-world FL scenarios. In both figures, FedProx demonstrates superior accuracy to FedAvg, which is especially notable in non-IID data scenarios with three clients.

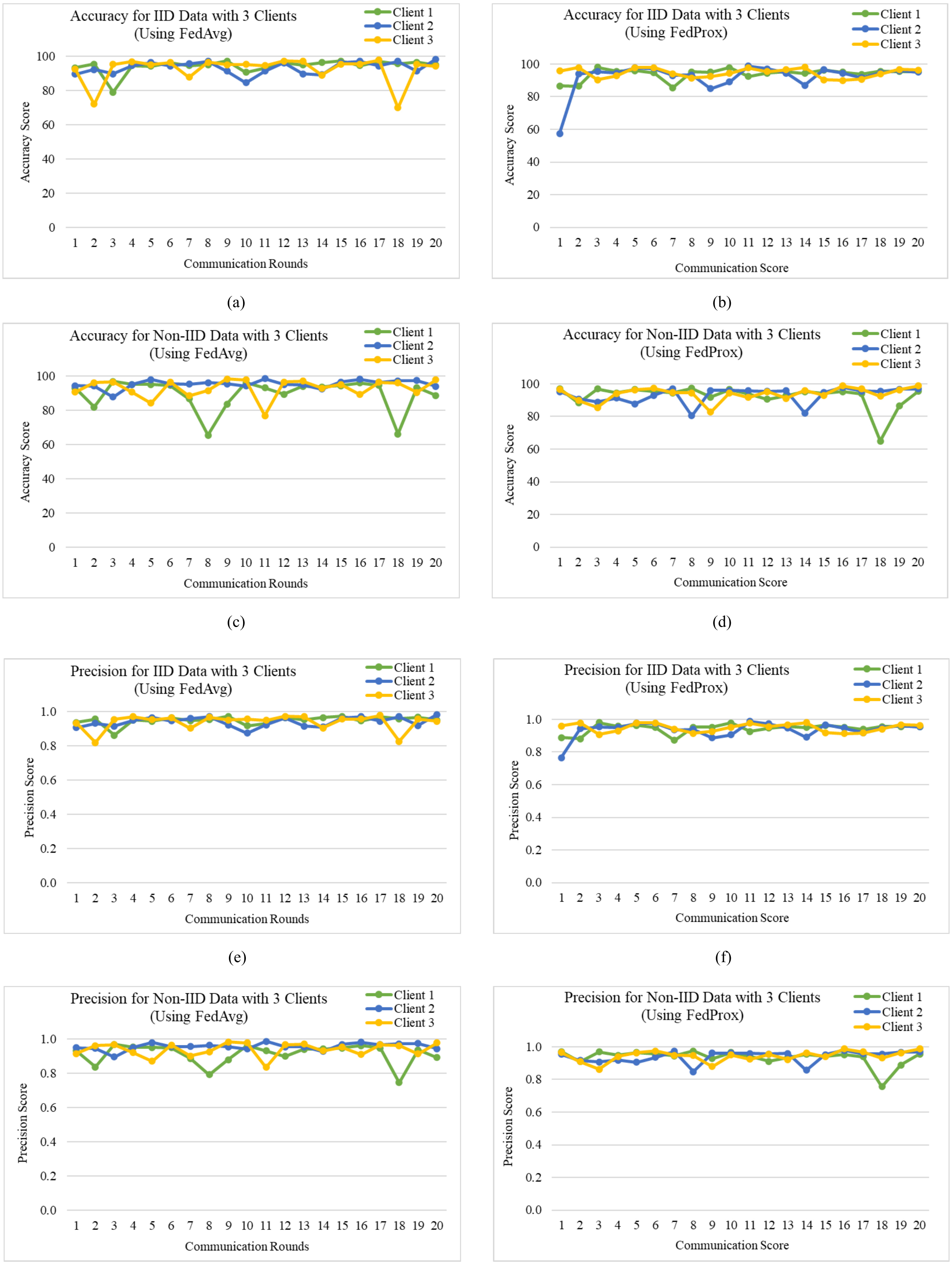
Figures 21 and 22 shown below, illustrate the performance of FedAvg and FedProx in terms of precision across IID and non-IID data distributions in a scenario involving 3 clients. In the IID setting shown in Figure 21 FedAvg performs consistently well in the IID data scenario, with precision scores mostly hovering around the high 90s. This high level of precision indicates that FedAvg can effectively minimize false positives when the data distribution among the clients is uniform. FedProx shows similar or slightly superior performance to FedAvg in terms of precision. Its scores are occasionally perfect or near-perfect, suggesting that the proximal term in FedProx not only helps in managing data heterogeneity but also in refining the model's accuracy and consistency even when data is IID. Both methods display robust precision across the rounds, with FedProx often exhibiting a bit more stability.

In the non-IID setting, referring to Figure 22, FedAvg experiences a decline in precision, as reflected by more variable and generally lower precision scores compared to the IID scenario. This decline indicates challenges in maintaining model accuracy when data characteristics vary significantly among the clients. FedProx maintains higher precision than FedAvg, though it also shows some reduction compared to its performance in the IID scenario. The generally higher precision scores of FedProx underline its effectiveness at dealing with data heterogeneity, likely due to the regularization effect of its proximal term which helps to control the divergence in client updates.

The gap in performance between FedAvg and FedProx becomes more evident in the non-IID setting, where FedProx's advanced features significantly contribute to its ability to produce more accurate and reliable results despite the complexity of the data distribution. Thus, FedProx outperforms FedAvg in both IID and non-IID scenarios, with a more pronounced advantage in environments where data is not uniformly distributed across clients.

The recall score comparison of FedAvg and FedProx for IID Data in 3 Clients is presented graphically in Figure 23. Here, FedAvg shows strong performance in the IID scenario, with recall scores mostly above 90%. This indicates that FedAvg is effective in identifying true positive cases when the data distribution is uniform across the clients, minimizing the number of false negatives.

FedProx exhibits slightly higher or comparable recall scores to FedAvg, occasionally reaching near-perfect recall. This suggests that the proximal term included in FedProx not only helps in enhancing precision but also contributes



(a) (b)

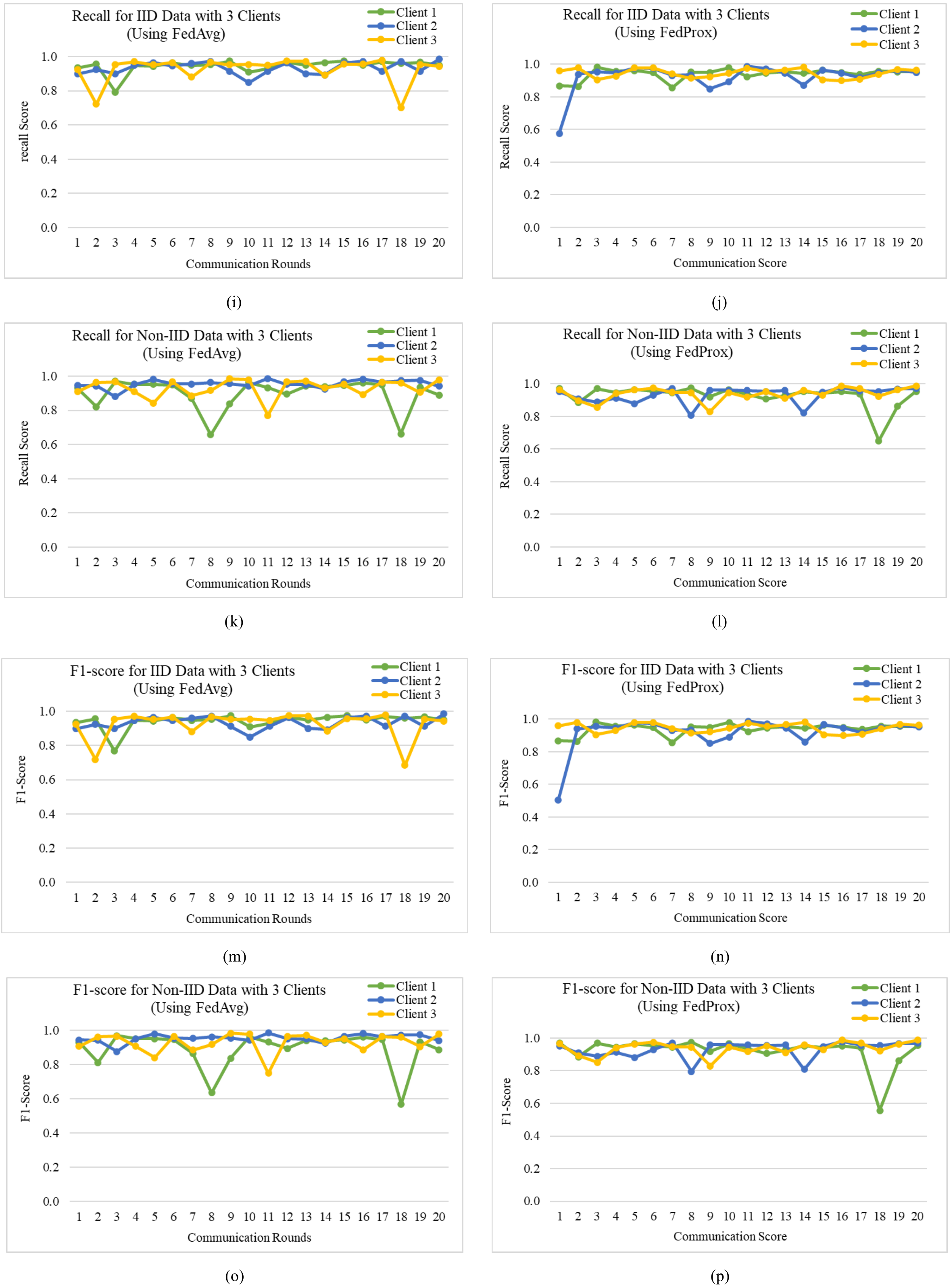
(c) (d)

(e) (f)

(g) (h)

FIGURE 18. Pictorial presentation of the Local Model for IID and non-IID data distributions in a horizontal federated learning scenario with 3 clients using FedAvg and FedProx algorithms.

positively to recall, ensuring that nearly all positive cases are captured. Both methods demonstrate consistency in their recall scores, with FedProx showing a slight edge in terms of stability and peak performance. This consistency is essential



(i) (j)

(k) (l)

(m) (n)

(o) (p)

FIGURE 18. ***(Continued.)*** **Pictorial presentation of the Local Model for IID and non-IID data distributions in a horizontal federated learning scenario with 3 clients using FedAvg and FedProx algorithms.**

for applications where missing a positive case can have significant consequences.

The recall score comparison of FedAvg and FedProx for non-IID Data in 3 Clients is presented graphically in

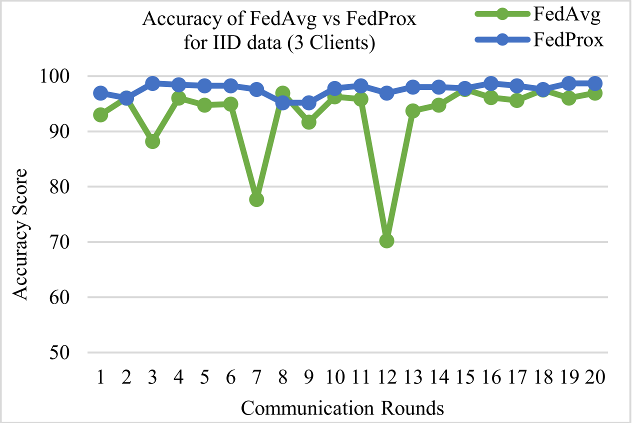


FIGURE 19. Representation of the accuracy score obtained by FedAvg and FedProx for IID data in 3 clients.

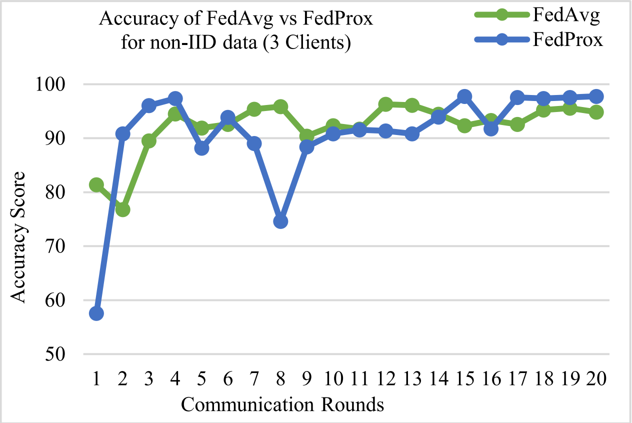


FIGURE 20. Representation of the accuracy score obtained by FedAvg and FedProx for non-IID data in 3 clients.

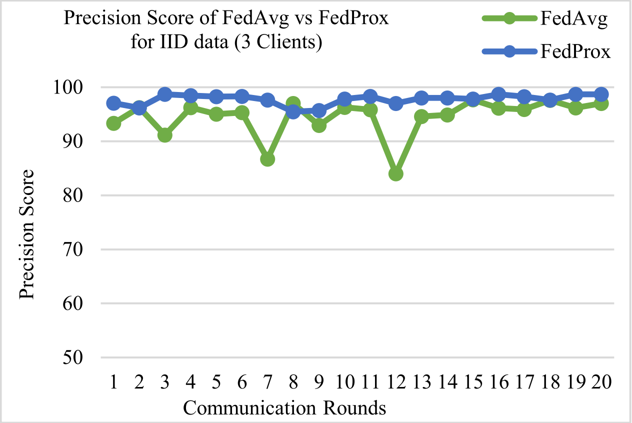


FIGURE 21. Representation of the precision score obtained by FedAvg and FedProx for IID data in 3 clients.

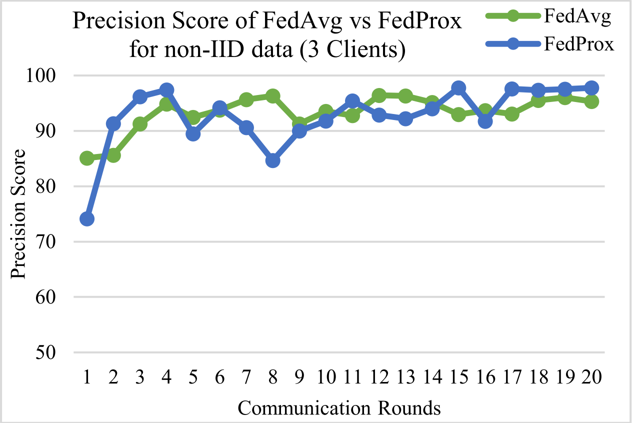


FIGURE 22. Representation of the precision score obtained by FedAvg and FedProx for non-IID data in 3 clients.

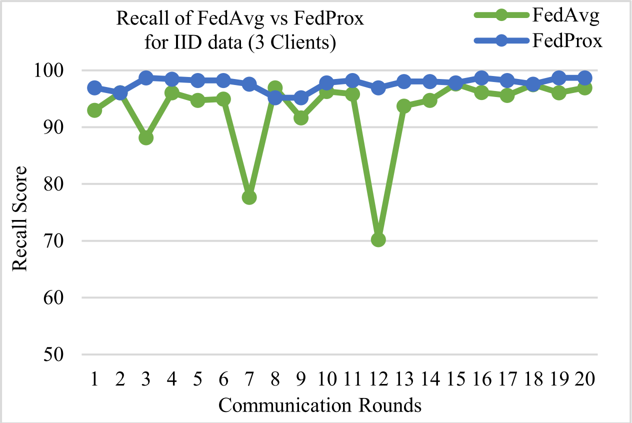


FIGURE 23. Representation of the recall score obtained by FedAvg and FedProx for IID data in 3 clients.

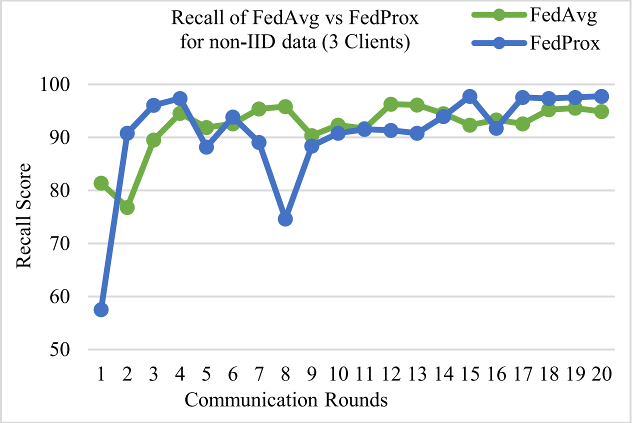


FIGURE 24. Representation of the recall score obtained by FedAvg and FedProx for non-IID data in 3 clients.

Figure 24, and the results show that FedAvg displays a noticeable drop in recall compared to the IID scenario, with increased variability in scores. This reduction reflects the challenges that FedAvg faces in environments where client data is diverse and not uniformly distributed, leading to potential misses in positive case identification. Whereas, FedProx maintains higher recall scores than FedAvg in the non-IID setting, although it too experiences some fluctuations compared to the IID data scenario. The generally better performance of FedProx indicates its robustness and ability to handle data heterogeneity, reducing the likelihood of false negatives even when data distributions are complex.

The gap between the performances of FedAvg and FedProx is considerably higher in the non-IID setting. In both figures, FedProx demonstrates superior recall to FedAvg, which is particularly notable in non-IID data scenarios with three clients. While both methods are effective in IID contexts, FedProx's ability to maintain higher recall in more complex, real-world scenarios where data uniformity cannot be assumed is evident.

Figure 25 depicts the F1 score for IID data. In this scenario, both FedAvg and FedProx show high performance, with F1-scores generally above 90%. FedProx slightly outperforms FedAvg, maintaining a higher and more stable F1 score across most rounds. This suggests that while both methods are effective in a homogeneous data distribution (IID), FedProx has a marginal advantage due to its proximal term, which aids in stabilizing the model updates and preventing significant deviations during the training process. Figure 26, on the other hand, illustrates the F1 score for non-IID data. The performance gap between FedAvg and FedProx becomes more pronounced in this setting. FedAvg experiences a notable drop in F1 score, exhibiting increased variability and generally lower scores compared to the IID scenario. This decline highlights the challenges FedAvg faces in dealing with data heterogeneity, where the data distribution across clients is diverse. Conversely, FedProx maintains a relatively high F1 score even in the non-IID scenario, showcasing its robustness and superior ability to handle non-uniform data distributions. The proximal term in FedProx plays a crucial role in mitigating the adverse effects of data heterogeneity, ensuring more consistent and reliable model performance. It is evident from the graphs that while both FedAvg and FedProx perform well with IID data, FedProx demonstrates a significant advantage in non-IID settings.

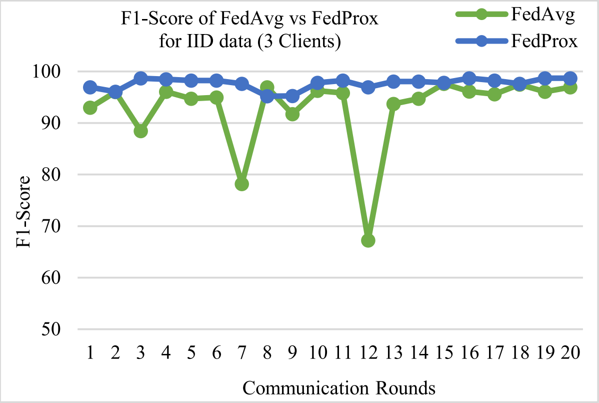


FIGURE 25. Representation of the F1-score obtained by FedAvg and FedProx for IID data in 3 clients.

Conclusively, across the various performance metrics of accuracy, precision, and recall in FL scenarios involving 2 and 3 clients with IID and non-IID data distributions, FedProx consistently outperforms FedAvg. While both methods show competence in IID settings, FedProx demonstrates superior stability and higher scores in non-IID conditions, reflecting its robustness against data heterogeneity. This advantage is evident in its ability to maintain higher accuracy, precision, and recall, suggesting that its proximal term significantly mitigates the challenges posed by diverse and complex data distributions. Thus, FedProx validates as a more reliable method for FL applications, particularly in real-world environments where uniform data distribution cannot be guaranteed. Thus, making it a preferable choice for ensuring effective, accurate, and dependable model outcomes.

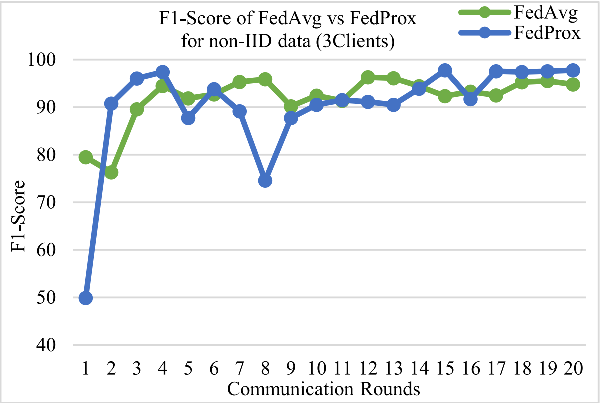


FIGURE 26. Representation of the F1-score obtained by FedAvg and FedProx for non-IID data in 3 clients.

VI. CONCLUSION

Federated Deep Learning models are transforming disease diagnosis technology, particularly in applications requiring robust privacy measures and high model accuracy, such as diabetic retinopathy detection. Deep learning provides powerful tools for analyzing complex medical images, enabling early and accurate disease identification. However, the traditional centralized training methods for deep learning models often require aggregation of sensitive data, posing privacy risks. FDL provides a privacy-preserving framework that allows for the training of models across multiple decentralized devices without sharing or transferring the raw data which is usually sensitive and confidential.

The article thoroughly evaluates the effectiveness of FDL in detecting diabetic macular edema and diabetic retinopathy. Also, it compares two popular FDL aggregation methods FedAvg and FedProx. The evaluation and analysis are based on the key performance metrics accuracy, precision, recall, and F1-score. These are observed under two different data scenarios—IID (where data across clients is similar) and non-IID (where data varies significantly among clients). The FDL framework using lightweight MobileNetV2 architecture and with FedAvg aggregation scheme, obtained an accuracy of 98.69% and 97.62% when IID data was used for 2 and 3 clients, respectively. The accuracy of FedAvg dropped to 87.09% and 96.28% when the data was non-IID in 2 and 3 client scenarios. The accuracy achieved by the FDL framework using the MobileNetV2 architecture and with FedProx aggregation was 98.03% and 98.69% with IID data for 2 and 3 client scenarios. Unlike the scenario where FedAvg aggregation was used in non-IID data, the accuracy did not drop as much for the scenario with FedProx aggregation. The prediction accuracy score of the FDL setup with FedProx aggregation and non-IID data was 98.25% and 97.77% for 2 clients and 3-client scenarios. These results consistently demonstrate that an FDL scenario with a FedProx aggregation strategy performs better and more stable results are obtained as compared to the FDL setup with FedAvg as the aggregation strategy, especially in non-IID settings. The superior performance of the FedProx aggregation in stabilizing the model and converging early is due to the proximal term, which effectively manages the challenge posed by diverse data distributions. This term stabilizes the training process and helps the model converge early. Thus, ensuring robust performance across different client datasets. In simpler terms, FedProx aggregation is more reliable in real-world applications of FDL where data consistency across different sources cannot be assumed. In IID scenarios, both aggregation methods show strong performance, while the FedProx aggregation method still improves performance and convergence. However, the FedProx aggregation method has a low improvement margin in terms of metrics like precision and recall for IID scenarios. This slight advantage suggests that FedProx is better suited for challenging non-IID environments but also performs well when the data is uniform.

The role of federated learning in healthcare is crucial, particularly in disease diagnosis, where data privacy and model accuracy are paramount. Data privacy in a federated learning scenario is maintained by enabling data to remain on local devices and only sharing model updates. Therefore, federated learning aligns with the strict privacy requirements of healthcare data, reducing the risk of sensitive data breaches while still allowing for developing highly accurate diagnostic tools. This capability makes federated learning an invaluable approach in the healthcare sector. Thus, promising to enhance the privacy of data while maintaining the effectiveness of disease diagnosis processes.

Model heterogeneity and scalability are the major open research challenges for implementing federated learning in real-world scenarios. The challenges also include selecting communication-efficient strategies and making the communication secure while reducing communication overhead. In the future, the federated learning framework demonstrated for retinal diseases can be expanded to adapt to other diseases where privacy is paramount. Also, other aggregation strategies, such as FedNova and MOON, can be explored for enhancement and better model generalizations. In addition, some data security methods, such as differential privacy and homomorphic encryption, can be incorporated to prevent any potential threats.

DECLARATIONS

COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

AUTHOR CONTRIBUTIONS

Conceptualization: Seema Gulati, Kalpna Guleria; Methodology: Seema Gulati, Kalpna Guleri; Formal Analysis: Seema Gulati, Kalpna Guleria, Nitin Goyal, Ahmad Ali AlZubi, Ángel Gabriel Kuc Castill; Validation: Seema Gulati, Kalpna Guleria, Nitin Goyal, Ahmad Ali AlZubi, Ángel Kuc Castilla; Investigation: Seema Gulati, Kalpna Guleria, Nitin Goyal, Ahmad Ali AlZubi, Ángel Gabriel Kuc Castill; Writing—original draft preparation: Seema Gulati, Kalpna Guleri; Writing—Review and editing: Seema Gulati, Kalpna Guleri; Supervision: Kalpna Guleria, Nitin Goyal; All authors have read and agreed to this version of the manuscript.

ACKNOWLEDGMENT

The authors would like to thank the editors and reviewers for their review and recommendations and also to extend their thanks to King Saud University for funding this work through the Researchers Supporting Project (RSP2024R395), King Saud University, Riyadh, Saudi Arabia.

AVAILABILITY OF DATA AND MATERIAL

The dataset used in this research is available at the following repositories.

https://github.com/Traslational-Visual-Health-Laboratory/OCT-AND-EYE-FUNDUS-DATASET

https://ieee-dataport.org/open-access/indian-diabetic-retinopathy-image-dataset-idrid

REFERENCES

[1] T. Soni, D. Gupta, M. Uppal, and S. Juneja, "Explicability of artificial intelligence in healthcare 5.0," in *Proc. Int. Conf. Artif. Intell. Smart Commun. (AISC)*, Jan. 2023, pp. 1256–1261, doi: 10.1109/AISC56616.2023.10085222.

[2] B. L. Jimma, "Artificial intelligence in healthcare: A bibliometric analysis," *Telematics Informat. Rep.*, vol. 9, Mar. 2023, Art. no. 100041, doi: 10.1016/j.teler.2023.100041.

[3] F. Ahmed, S. Abbas, A. Athar, T. Shahzad, W. A. Khan, M. Alharbi, M. A. Khan, and A. Ahmed, "Identification of kidney stones in KUB X-ray images using VGG16 empowered with explainable artificial intelligence," *Sci. Rep.*, vol. 14, no. 1, p. 6173, Mar. 2024, doi: 10.1038/s41598-024-56478-4.

[4] F. Jiang, Y. Jiang, H. Zhi, Y. Dong, H. Li, S. Ma, Y. Wang, Q. Dong, H. Shen, and Y. Wang, "Artificial intelligence in healthcare: Past, present and future," *Stroke Vascular Neurology*, vol. 2, no. 4, pp. 230–243, Dec. 2017.

[5] J. S. Roppelt, D. K. Kanbach, and S. Kraus, "Artificial intelligence in healthcare institutions: A systematic literature review on influencing factors," *Technol. Soc.*, vol. 76, Mar. 2024, Art. no. 102443, doi: 10.1016/j.techsoc.2023.102443.

[6] R. Gupta, S. Kumari, A. Senapati, R. K. Ambasta, and P. Kumar, "New era of artificial intelligence and machine learning-based detection, diagnosis, and therapeutics in Parkinson's disease," *Ageing Res. Rev.*, vol. 90, Sep. 2023, Art. no. 102013, doi: 10.1016/j.arr.2023.102013.

[7] S. Gulati, K. Guleria, and N. Goyal, "Classification and detection of diabetic eye diseases using deep learning: A review and comparative analysis," *AIP Conf. Proc.*, vol. 2916, no. 1, Dec. 2023, Art. no. 020005, doi: 10.1063/5.0177682.

[8] R. Pradeepa and V. Mohan, "Epidemiology of type 2 diabetes in India," *Indian J. Ophthalmol.*, vol. 69, no. 11, pp. 2932–2938, Nov. 2021, doi: 10.4103/ijo.ijo_1627_21.

[9] P. N. Thotad, G. R. Bharamagoudar, and B. S. Anami, "Diabetes disease detection and classification on Indian demographic and health survey data using machine learning methods," *Diabetes Metabolic Syndrome: Clin. Res. Rev.*, vol. 17, no. 1, Jan. 2023, Art. no. 102690, doi: 10.1016/j.dsx.2022.102690.

[10] F. Ahmed, A. Fatima, M. Mamoon, and S. Khan, ''Identification of the diabetic retinopathy using ResNet-18,'' in *Proc. 2nd Int. Conf. Cyber Resilience (ICCR)*, Feb. 2024, pp. 1–6.

[11] S. Gulati, K. Guleria, and N. Goyal, ''Classification of diabetic retinopathy using pre-trained deep learning Model- DenseNet 121,'' in *Proc. 14th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, Jul. 2023, pp. 1–6, doi: 10.1109/icccnt56998.2023.10308181.

[12] R. Raman, ''Prevalence of diabetic retinopathy in India stratified by known and undiagnosed diabetes, urban–rural locations, and socioeconomic indices: Results from the SMART India population-based cross-sectional screening study,'' *Lancet Global Health*, vol. 10, no. 12, pp. e1764–e1773, Dec. 2022, doi: 10.1016/s2214-109x(22)00411-9.

[13] J. Ding and T. Y. Wong, ''Current epidemiology of diabetic retinopathy and diabetic macular edema,'' *Current Diabetes Rep.*, vol. 12, no. 4, pp. 346–354, Aug. 2012.

[14] S. Gulati, K. Guleria, and N. Goyal, ''Classification and detection of coronary heart disease using machine learning,'' in *Proc. 2nd Int. Conf. Advance Comput. Innov. Technol. Eng. (ICACITE)*, Apr. 2022, pp. 1728–1732, doi: 10.1109/ICACITE53722.2022.9823547.

[15] S. Gulati, K. Guleria, and N. Goyal, ''Classification of migraine disease using supervised machine learning,'' in *Proc. 10th Int. Conf. Rel., Infocom Technol. Optim. (Trends Future Directions) (ICRITO)*, Oct. 2022, pp. 1–7.

[16] A. Kaur, K. Guleria, and N. K. Trivedi, ''Rice leaf disease detection: A review,'' in *Proc. 6th Int. Conf. Signal Process., Comput. Control (ISPCC)*, Oct. 2021, pp. 418–422.

[17] S. Sharma and K. Guleria, ''A systematic literature review on deep learning approaches for pneumonia detection using chest X-ray images,'' *Multimedia Tools Appl.*, vol. 83, no. 8, pp. 24101–24151, Aug. 2023, doi: 10.1007/s11042-023-16419-1.

[18] G. O. Kuttan and M. S. Elayidom, ''Review on computer aided breast cancer detection and diagnosis using machine learning methods on mammogram image,'' *Current Med. Imag. Formerly Current Med. Imag. Rev.*, vol. 19, no. 12, pp. 1361–1371, Oct. 2023.

[19] S. Sharma and K. Guleria, ''A comprehensive review on federated learning based models for healthcare applications,'' *Artif. Intell. Med.*, vol. 146, Dec. 2023, Art. no. 102691, doi: 10.1016/j.artmed.2023.102691.

[20] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, ''Federated learning: Challenges, methods, and future directions,'' *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020, doi: 10.1109/MSP.2020.2975749.

[21] H. Kaur, V. Rani, M. Kumar, M. Sachdeva, A. Mittal, and K. Kumar, ''Federated learning: A comprehensive review of recent advances and applications,'' *Multimedia Tools Appl.*, vol. 83, no. 18, pp. 54165–54188, Nov. 2023, doi: 10.1007/s11042-023-17737-0.

[22] T. Sood, R. Bhatia, and P. Khandnor, ''Cancer detection based on medical image analysis with the help of machine learning and deep learning techniques: A systematic literature review,'' *Current Med. Imag. Formerly Current Med. Imag. Rev.*, vol. 19, no. 13, pp. 1487–1522, Nov. 2023.

[23] A. Suryavanshi, V. Kukreja, S. Mehta, S. Chattopadhyay, and A. Verma, ''Disruptive farming intelligence: Federated learning CNN for broccoli leaf disease classification,'' in *Proc. Int. Conf. Cognit. Robot. Intell. Syst. (ICC-ROBINS)*, Apr. 2024, pp. 697–702, doi: 10.1109/icc-robins60238.2024.10534024.

[24] K. M. J. Rahman, F. Ahmed, N. Akhter, M. Hasan, R. Amin, K. E. Aziz, A. K. M. M. Islam, M. S. H. Mukta, and A. K. M. N. Islam, ''Challenges, applications and design aspects of federated learning: A survey,'' *IEEE Access*, vol. 9, pp. 124682–124700, 2021, doi: 10.1109/ACCESS.2021.3111118.

[25] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao, ''Heterogeneous federated learning: State-of-the-art and research challenges,'' *ACM Comput. Surv.*, vol. 56, no. 3, pp. 1–44, Mar. 2024.

[26] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, ''Federated learning: A survey on enabling technologies, protocols, and applications,'' *IEEE Access*, vol. 8, pp. 140699–140725, 2020.

[27] Z. Zhao, Y. Mao, Y. Liu, L. Song, Y. Ouyang, X. Chen, and W. Ding, ''Towards efficient communications in federated learning: A contemporary survey,'' *J. Franklin Inst.*, vol. 360, no. 12, pp. 8669–8703, Aug. 2023, doi: 10.1016/j.jfranklin.2022.12.053.

[28] D. Ng, X. Lan, M. M.-S. Yao, W. P. Chan, and M. Feng, ''Federated learning: A collaborative effort to achieve better medical imaging models for individual sites that have small labelled datasets,'' *Quant. Imag. Med. Surg.*, vol. 11, no. 2, pp. 852–857, Feb. 2021, doi: 10.21037/qims-20-595.

[29] H. Zhang, J. Bosch, and H. H. Olsson, ''Federated learning systems: Architecture alternatives,'' in *Proc. 27th Asia–Pacific Softw. Eng. Conf. (APSEC)*, Dec. 2020, pp. 385–394.

[30] H. R. Roth et al., ''Federated learning for breast density classification: A real-world implementation,'' in *Proc. Domain Adapt. Represent. Transf., Distrib. Collaborative Learn. 2nd MICCAI Workshop, DART, 1st MICCAI Workshop, DCL, Held Conjunction MICCAI*, 2020, pp. 181–191, doi: 10.1007/978-3-030-60548-3_18/COVER.

[31] W. Zhang, T. Zhou, Q. Lu, X. Wang, C. Zhu, H. Sun, Z. Wang, S. K. Lo, and F.-Y. Wang, ''Dynamic-fusion-based federated learning for COVID-19 detection,'' *IEEE Internet Things J.*, vol. 8, no. 21, pp. 15884–15891, Nov. 2021, doi: 10.1109/JIOT.2021.3056185.

[32] D. Yang, Z. Xu, W. Li, A. Myronenko, H. R. Roth, S. Harmon, S. Xu, B. Turkbey, E. Turkbey, X. Wang, W. Zhu, G. Carrafiello, F. Patella, M. Cariati, H. Obinata, H. Mori, K. Tamura, P. An, B. J. Wood, and D. Xu, ''Federated semi-supervised learning for COVID region segmentation in chest CT using multi-national data from China, Italy, Japan,'' *Med. Image Anal.*, vol. 70, May 2021, Art. no. 101992, doi: 10.1016/j.media.2021.101992.

[33] S. H. Khan and M. G. R. Alam, ''A federated learning approach to pneumonia detection,'' in *Proc. Int. Conf. Eng. Emerg. Technol. (ICEET)*, Oct. 2021, pp. 1–6, doi: 10.1109/ICEET53442.2021.9659591.

[34] S. Matta, M. B. Hassine, C. Lecat, L. Borderie, A. L. Guilcher, P. Massin, B. Cochener, M. Lamard, and G. Quellec, ''Federated learning for diabetic retinopathy detection in a multi-center fundus screening network,'' in *Proc. 45th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2023, pp. 1–4, doi: 10.1109/embc40787.2023.10340772.

[35] M. Nasajpour, M. Karakaya, S. Pouriyeh, and R. M. Parizi, ''Federated transfer learning for diabetic retinopathy detection using CNN architectures,'' in *Proc. SoutheastCon*, Mar. 2022, pp. 655–660, doi: 10.1109/SoutheastCon48659.2022.9764031.

[36] N. J. Mohan, R. Murugan, T. Goel, and P. Roy, ''DRFL: Federated learning in diabetic retinopathy grading using fundus images,'' *IEEE Trans. Parallel Distrib. Syst.*, vol. 34, no. 6, pp. 1789–1801, Jun. 2023.

[37] J. Lo, T. T. Yu, D. Ma, P. Zang, J. P. Owen, Q. Zhang, R. K. Wang, M. F. Beg, A. Y. Lee, Y. Jia, and M. V. Sarunic, ''Federated learning for microvasculature segmentation and diabetic retinopathy classification of OCT data,'' *Ophthalmol. Sci.*, vol. 1, no. 4, Dec. 2021, Art. no. 100069, doi: 10.1016/j.xops.2021.100069.

[38] X. Yin, Y. Zhu, and J. Hu, ''A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions,'' *ACM Comput. Surv.*, vol. 54, no. 6, pp. 1–36, Jul. 2022.

[39] F. Hartmann, *Federated Learning*. Berlin, Germany: Freie Universität, 2018.

[40] J. Liu, ''From distributed machine learning to federated learning: A survey,'' *Knowl. Inf. Syst.*, vol. 64, pp. 885–917, Mar. 2022, doi: 10.1007/s10115-022-01664-x.

[41] S. Abdulrahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, ''A survey on federated learning: The journey from centralized to distributed on-site learning and beyond,'' *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5476–5497, Apr. 2021, doi: 10.1109/JIOT.2020.3030072.

[42] R. S. Antunes, C. A. D. Costa, A. Küderle, I. A. Yari, and B. Eskofier, ''Federated learning for healthcare: Systematic review and architecture proposal,'' *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 4, pp. 1–23, Aug. 2022.

[43] H. Zhu, J. Xu, S. Liu, and Y. Jin, ''Federated learning on non-IID data: A survey,'' *Neurocomputing*, vol. 465, pp. 371–390, Nov. 2021.

[44] X. Ma, J. Zhu, Z. Lin, S. Chen, and Y. Qin, ''A state-of-the-art survey on solving non-IID data in federated learning,'' *Future Gener. Comput. Syst.*, vol. 135, pp. 244–258, Oct. 2022.

[45] A. Dogra, B. Goyal, and S. Agrawal, ''Osseous and digital subtraction angiography image fusion via various enhancement schemes and Laplacian pyramid transformations,'' *Future Gener. Comput. Syst.*, vol. 82, pp. 149–157, May 2018.

[46] A. Dogra, B. Goyal, and S. Agrawal, ''Bone vessel image fusion via generalized Reisz wavelet transform using averaging fusion rule,'' *J. Comput. Sci.*, vol. 21, pp. 371–378, Jul. 2017.

[47] A. Alferaidi, K. Yadav, Y. Alharbi, W. Viriyasitavat, S. Kautish, and G. Dhiman, ''Federated learning algorithms to optimize the client and cost selections,'' *Math. Problems Eng.*, vol. 2022, no. 1, 2022, Art. no. 8514562, doi: 10.1155/2022/8514562.

[48] T. Zhang and S. Mao, ''An introduction to the federated learning standard,'' *Mobile Comput. Commun.*, vol. 25, no. 3, pp. 18–22, 2022.

[49] P. Qi, D. Chiaro, A. Guzzo, M. Ianni, G. Fortino, and F. Piccialli, "Model aggregation techniques in federated learning: A comprehensive survey," *Future Generat. Comput. Syst.*, vol. 150, pp. 272–293, Jan. 2023.

[50] K. Pillutla, S. M. Kakade, and Z. Harchaoui, "Robust aggregation for federated learning," *IEEE Trans. Signal Process.*, vol. 70, pp. 1142–1154, 2022, doi: 10.1109/TSP.2022.3153135.

[51] H. Cano, J. A. O. Pinto, and U. Thébault, "Dataset of eye fundus and OCT images for the study of diabetic macular edema and diabetic retinopathy," Translational Visual Health Laboratory, Instituto de Neurobiología, Universidad Nacional Autónoma de México (UNAM), Querétaro, Mexico, Tech. Report CF-2019-1759 and IN 205420, 2022. [Online]. Available: https://github.com/Traslational-Visual-Health-Laboratory/OCT-AND-EYE-FUNDUS-DATASET

[52] P. Porwal, "Indian diabetic retinopathy image dataset (IDRiD)," *IEEE Dataport*, Jul. 2018, doi: 10.21227/H25W98.

[53] J. Mills, J. Hu, and G. Min, "Communication-efficient federated learning for wireless edge intelligence in IoT," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5986–5994, Jul. 2020.

[54] A. Nilsson, S. Smith, G. Ulm, E. Gustavsson, and M. Jirstrand, "A performance evaluation of federated learning algorithms," in *Proc. 2nd Workshop Distrib. Infrastructures Deep Learn.*, Dec. 2018, pp. 1–8.

[55] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst. (MLSys)*, V. Sze, Eds., 2020, pp. 429–450. [Online]. Available: https://proceedings.mlsys.org/paper_files/paper/2020/file/1f5fe83998a09396ebe6477d9475ba0c-Paper.pdf

[56] X. Yuan and P. Li, "On convergence of FedProx: Local dissimilarity invariant bounds, non-smoothness and beyond," in *Proc. Adv. Neural Inf. Process. Syst.*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh, Eds., Red Hook, NY, USA: Curran Associates, 2022, pp. 10752–10765. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/45ecdd6cf1f507d378a3442ed89e580b-Paper-Conference.pdf

SEEMA GULATI received the Bachelor of Technology degree in computer engineering from Kurukshetra University, India, in 2011, and the Master of Technology degree from Punjab Technical University, India, in 2014. She is currently pursuing the Ph.D. degree in computer science and engineering with Chitkara University, Punjab, India. She is currently a full-time Scholar with Chitkara University. Her research interests include artificial intelligence and federated deep learning-based automatic disease detection systems. She is also a Seasoned Educator with more than five years of experience in academic teaching.

KALPNA GULERIA received the B.Tech. and M.Tech. degrees (Hons.) in computer science and engineering and the Ph.D. degree in computer science engineering from Thapar University, Patiala, India. She is currently a Professor (Research) with Chitkara University Research and Innovation Network (CURIN), Chitkara University, Punjab, India. She has been honored with a meritorious service award and has received various awards of merit in academics as well. She is actively involved in research and has filed more than 35 patents. She is currently working on two government funded projects. She is also the Principal Investigator for a project titled "Awareness and Training Program for Teachers on Teaching Mathematics through Origami for Patiala and Mohali Districts of Punjab" under the banner of the National Council for Science & Technology (NCSTC), Department of Science and Technology (DST), Government of India, New Delhi. She is also a Co-Principal Investigator in the Project "Smart Ergonomic Portable Commode Chair" under the Technology Interventions for Disabled and Elderly (TIDE) Scheme by the Ministry of Science and Technology, Government of India, New Delhi. She has research publications in SCI—journals and International conferences of high repute. Her research interests include wireless sensor networks, VANETS, the IoV, machine learning, deep learning, and federated learning. She is also a reviewer of various international journals.

NITIN GOYAL (Senior Member, IEEE) received the B.Tech. and M.Tech. degrees in computer science and engineering from Kurukshetra University, India, in 2007 and 2009, respectively, and the Ph.D. degree in computer engineering from NIT Kurukshetra, India, in 2018. He is currently an Assistant Professor with the Central University of Haryana, Mahenderagarh. He has around 15 years of teaching experience and academic work experience. He has published approximately 150 research papers in various international/national journals, books, and conferences, out of which 70 are SCI, five are Scopus, 13 are book chapters, and 30 are conference papers. He has filed more than 25 patents, out of which seven are published and 15 are granted. Also, he has attended 13 workshops and three FDPs. He is an editor of two books from different publishers, such as IGI Global and CRC Press. He has guided ten M.Tech. candidates, two Ph.D. candidates, and four Ph.D. candidates are in process. He has also delivered seven expert lectures on NS2, Paper Writing, Patents Filing, Innovation, and Entrepreneurship. He is also a GATE, HTET, and UGC-NET Qualifier. His research interests include MANET, FANET, UWSN, and WSN. He is an editorial board member in the different journals and conferences. He is a Guest Editor of the *International Journal of Distributed Sensor Networks* (SAGE) (SCI Journal), *Energies* (MDPI) (SCI Journal), *Frontiers in Blockchain* (ESCI) (Frontier Scientific Publishing), and *Journal of Autonomous Intelligence* (Scopus) (Frontier Scientific Publishing).

AHMAD ALI ALZUBI received the Ph.D. degree in computer networks engineering from the National Technical University of Ukraine, Kyiv, Ukraine, in 1999. He is currently a Professor with King Saud University, Riyadh, Saudi Arabia. His current research interests include computer networks, grid computing, cloud computing, big data, and data extraction. He was a Consultant and a member for three years with Saudi National Team for Measuring e-Government in Saudi Arabia.

ÁNGEL KUC CASTILLA received the Ph.D. degree in engineering with a specialization in marine and coastal engineering from the National Autonomous University of Mexico. He is currently with the Department of Engineering and Projects, Universidad Internacional Iberoamericana, Mexico. Additionally, he holds affiliations with several esteemed institutions, including the International University Foundation of Colombia, Bogotá, Colombia, and the University of La Romana, La Romana, Dominican Republic. He is currently a Professor and a Researcher in the field of coastal engineering. His work focuses on developing innovative solutions for the sustainable management of marine and coastal ecosystems. He has published numerous articles in top-tier journals and prestigious organizations, such as the National Science Foundation and Mexican Council for Science and Technology have funded his research.