Review of Computer Engineering Research
2026 Vol. 13, No. 4, pp. 183-195
ISSN(e): 2410-9142
ISSN(p): 2412-4281
DOI: 10.18488/76.v13i4.5338
© 2026 Conscientia Beam. All Rights Reserved.

Privacy-aware federated learning for pneumonia detection in chest X-Ray images using MobileNetV2

A. Shravani[1+]
B. Uma Rani[2]

[1,2]*Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, India.*
[1]*Email: shravaniannam@gmail.com*
[2]*Email: umarani.bodakunta@gmail.com*



(+ Corresponding author)

Article History
Received: 10 June 2026
Revised: 15 September 2026
Accepted: 5 October 2026
Published: 30 October 2026

Keywords
Chest X-ray classification
Medical image analysis
MobileNetV2
Pneumonia detection
Privacy-preserving AI.

ABSTRACT

Pneumonia is the main cause of morbidity and mortality on the planet, and correct diagnosis based on chest radiographs is vital for timely intervention. Nonetheless, medical imaging deep learning models have commonly been limited by strict privacy policies that do not allow the central aggregation of sensitive patient information. This paper presents a privacy-preserving federated learning (FL) Python framework for pneumonia detection in the Kaggle Chest X-Ray dataset that seeks to address data governance, institutional autonomy, and non-IID data heterogeneity issues. The lightweight MobileNetV2 network was initialized with ImageNet weights and executed on four simulated hospital clients in the Flower (FLWR) framework. All clients had two epochs of local training per round, and all knowledge was synchronized through the Federated Averaging (FedAvg) algorithm. The data were divided according to a Dirichlet non-IID distribution to simulate realistic cross-hospital variations. The results of the global model over ten communication rounds reached 0.9924 accuracy, 0.9949 F1-score, and 0.9998 AUC, which indicates convergence and resistance to non-homogeneous client distributions. The findings validate the hypothesis that federated learning can perform diagnostic tasks as well as centralized training while maintaining privacy and minimizing patients' computational requirements. The research provides an experimentally verified FL pipeline that can be applied to low-resource clinical environments and provides a basis for the future integration of differential privacy, secure anonymization, and multi-task medical imaging.

Contribution/Originality: This work involved designing and evaluating a federated learning system to identify pneumonia using MobileNetV2 on the Chest X-Ray Kaggle dataset. The model was built and tested using four clients and one server, ten communication rounds, and two local epochs per client.

1. INTRODUCTION

One of the most critical causes of death and disease in the world, especially among infants, older adults, and individuals with compromised immune systems, is pneumonia; as such, timely and proper diagnosis is a crucial factor in enhancing the recovery of pneumonia-afflicted patients and lowering health care expenses. A radiologist will typically interpret a chest imaging study (chest X-ray) ordered by a physician [1]. The use of AI or deep learning has revolutionized medical imaging since it can automatically determine the presence of a disease in the body with high accuracy. Convolutional Neural Networks (CNNs) have proven to be highly successful in detecting intricate radiological images that are otherwise invisible to the human eye [2].

© 2026 Conscientia Beam. All Rights Reserved.

Nevertheless, AI-based diagnostic systems require large, diverse datasets to be effective. Healthcare facilities tend to distribute these datasets, and they are highly controlled in terms of privacy. Forwarding them to central servers may contravene confidentiality and data protection laws such as the General Data Protection Regulation (GDPR). The conflict, therefore, lies in the need to create mechanisms that facilitate the joint training of models without infringing on patient privacy [3, 4].

One promising solution is Federated Learning (FL). It allows multiple institutions to jointly train a model without necessarily having direct access to raw data. Clients themselves train locally and report only model parameters to a central server. These parameters are synthesized by the server to update the global model. The model is privacy-guaranteed and data-sovereign [5].

The research adopts federated learning with MobileNetV2, a small and lightweight CNN that has been pre-trained on ImageNet. Training was conducted using the Kaggle Chest X-Ray dataset across four simulated clients within the Flower (FLWR) framework. The system was trained for more than ten federated rounds, with each client updated every two local epochs [4, 6-8].

In practice, the final world model is observed to have an error of 0.9924 and an F1 score of 0.9949, which indicates that federated learning can generate high-quality diagnostic models while maintaining a high level of data privacy.

1.1. Objectives

The research will aim to determine whether federated learning can be deployed to identify pneumonia through chest X-rays without compromising data privacy. Specifically, the paper will compare how MobileNetV2 performs in a federated task and demonstrate how it can be trained to adapt to non-IID data, which approximates the heterogeneity of healthcare data in a multi-institutional environment. The study will also prove that, with the help of federated learning, diagnostic accuracy can remain high and the requirements of data privacy laws such as the GDPR can be met. The second objective is to explore how this privacy-friendly framework can be adapted to minimize computing costs and implemented in low-resource clinical environments.

2. LITERATURE REVIEW

2.1. Deep Learning in Medical Imaging

Deep learning has led to a new paradigm in diagnostic radiology, where algorithms can read medical images with the same clarity as a high-quality expert radiologist would. Central to this advancement are convolutional Neural Networks (CNNs), a hierarchical approach to learning that can identify intricate spatial patterns in visual representations. CNNs are especially effective at learning low-level features such as edges and textures and high-level semantic representations such as lung opacities or lesion patterns that indicate pneumonia [1, 9].

These have been shown to be promising for the detection of pathological features in computed tomography (CT) images and chest radiographs, including ResNet, DenseNet, and MobileNet. Among them is MobileNetV2, which uses depthwise separable convolutions and inverted residual blocks to achieve significant computational savings with no loss of accuracy and is much lighter [2]. Transfer learning has led to deep learning models which, when applied to large-scale image datasets such as ImageNet, can be adapted to achieve the same level of diagnostic performance using a small number of labelled medical images [3].

2.2. Privacy and Ethical Challenges in Data Centralization

In contrast to the amazing success of deep learning, there are significant ethical and legal questions raised by centralized data-concentration systems used for training. Health information is also sensitive data; personal details, demographic information, and medical records are protected by stringent privacy policies and regulations, such as the General Data Protection Regulation (GDPR). When multiple healthcare units collect datasets, exporting them

© 2026 Conscientia Beam. All Rights Reserved.

to a central repository can lead to a loss of confidentiality. Although anonymized, health information is subject to re-identification risks because medical images can be associated with other demographic characteristics [5, 10].

2.3. Federated Learning: A Privacy-Preserving Paradigm

Federated Learning is a decentralised machine learning architecture with the aim of empowering many participants to collectively train a common global model without sharing raw data. FL was first proposed by Google in 2016 as a technology that can be implemented on mobile devices, with models trained on local inputs and standard aggregation of model updates across a central server. The major algorithm that organises this procedure is Federated Averaging (FedAvg), which involves averaging the weights of the local models based on the volume of data per client, generating a fresh, sharpened global model [6].

The model integrates the distributed expertise of the participating clients as it internalizes the various communication loops. Medical applications, including tumour segmentation, cardiovascular risk prediction, and retinopathy detection, have been used successfully with FL and have reported performance similar to that of centralised models while addressing privacy concerns. Its efficiency shows that education may be part of a collaborative system where no information pooling is required, and it is suitable for an inter-hospital or inter-institutional setting in which data sharing is legally constrained [11, 12].

2.4. MobileNetV2 for Efficient Distributed Learning

Special attention to model size and computational efficiency makes the MobileNetV2 architecture particularly appropriate for federated learning systems. It uses bottleneck residual blocks, preserving representational richness with a significant reduction in parameters. This architecture reduces communication overhead during federated aggregation because smaller models produce fewer bytes to be transmitted between clients and the server [13]. MobileNetV2 similarly utilizes linear bottlenecks to store significant feature information without always needing to downsample the data to facilitate successful backpropagation with fewer computations. The federated learning system, as shown in Figure 1, offers the MobileNetV2 network framework, which represents how the lightweight model is designed to compute effectively [14].

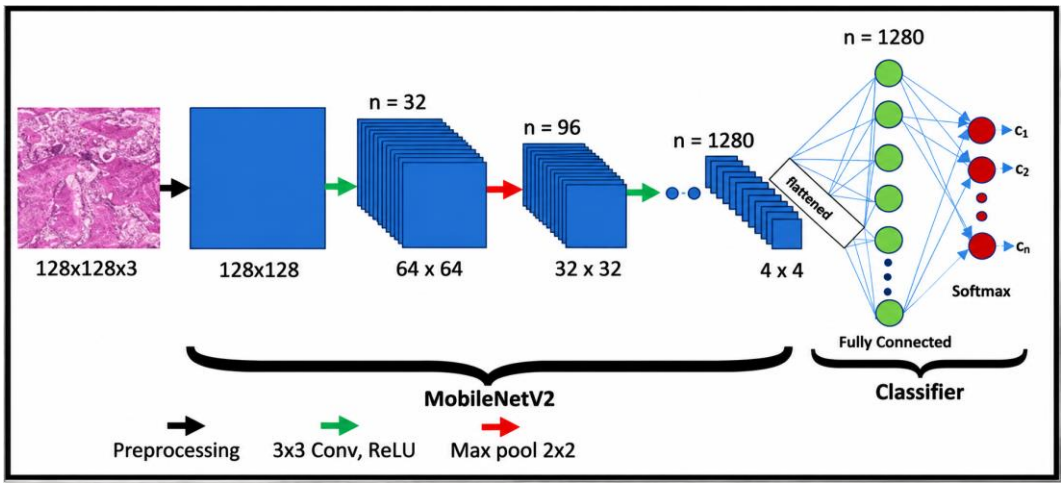


Figure 1. MobileNetV2 network architecture.

This architecture cannot be scaled out in federated configurations, specifically when these clients are running on relatively small computers. The trained weights and lightweight computation accelerate convergence and make local updates effective. MobileNetV2, thus, is a trade-off between diagnostic and operational viability, and it provides a reasonably strong baseline for federated medical imaging systems.

© 2026 Conscientia Beam. All Rights Reserved.

2.5. Research Gap and Research Motivation

Although there is a considerable body of theoretical literature on the concept of federated learning, a relative lack of empirical evidence exists in the medical imaging context. The literature uses simulated, idealised, and balanced data instead of real, non-independent and identically distributed (non-IID) data common in healthcare. Moreover, most implementations are based on large models, including ResNet or VGG, which are computationally impractical to execute on small-scale computers.

This research addresses these gaps by specifying a federated learning pipeline that trains MobileNetV2 on the Kaggle Chest X-Ray dataset and is deployed across four simulated clients. The design explicitly introduces the notion of non-IID data partitions to capture realistic institutional variations. The study hypothesis is to show that federated learning can be very diagnostic when the data are heterogeneous and data privacy is absent. By doing so, it offers actionable information about how lightweight architectures can lessen the gap between computational effectiveness, accuracy, and ethical data governance.

Having completed the literature review of deep learning in medical imaging, the privacy implications of centralized data, and federated learning, it is now possible to note that, although the federated learning methodology is more commonly applied to other tasks, its use in the pneumonia detection task with chest X-ray images remains a relatively unfamiliar issue in the context of privacy considerations. In addition, the existing literature is over-reliant on IID data and mega-based models that are computationally costly. To address these deficiencies, the study adopts MobileNetV2, an IID-free federated learning model, to ensure privacy and minimize computation among many healthcare centers.

3. METHODOLOGY

3.1. System Architecture

The suggested system follows a client-server federation topology, which will facilitate distributed learning models that do not infringe on information privacy. The central node is the organizing entity that coordinates the federated process, and the clients are autonomous data silos that can be compared to a single medical center. Every client keeps its own local dataset and model, and exchanges updates with the server [14-16].

During a communication round, MobileNetV2 is locally trained on a client using a subset of medical images. Local training is followed by uploading the model parameters to the server, where they are combined using the Federated Averaging (FedAvg) algorithm. The server then constructs a new global model by averaging the local weights (scaled by the size of each client's dataset). This generalized model is re-applied to the entire group of clients in the next training cycle.

This algorithm is repeated ten times in federated rounds, where knowledge learned by each client is cumulatively applied to a slowly growing world model [12, 17-20]. The architecture also enables collaborative learning among institutions without the need to transfer raw medical data, thereby maintaining confidentiality and adhering to data protection laws, including the GDPR.

3.2. Dataset Preparation and Preprocessing

The experimental research was based on the Kaggle Labeled Chest X-Ray Images, which is a publicly available dataset widely used in the study of pneumonia detection. The dataset comprises two significant classes: Pneumonia and Normal. To consider the heterogeneity of real hospitals, data were subdivided into four non-identically distributed (non-IID) samples (Dirichlet distribution, 0.5). This facilitated the development of disproportionate classes among clients due to demographic and clinical disparities in healthcare settings [5].

Dirichlet Non-IID Data Partition:

$$(p_{1,c}, \ldots, p_{K,c}) \sim \mathrm{Dir}(\alpha, \ldots, \alpha)$$

© 2026 Conscientia Beam. All Rights Reserved.

The image input was rescaled to the MobileNetV2 input size of 224 x 224 and normalised to ensure that pixel values fell within the range of [0, 1]. Random rotation, left-right flipping, and alterations in image brightness augmented the data and enhanced model robustness and generalisation. The data augmentation also provided more training images than would have been possible with the original images alone (and as such, it helped prevent overfitting).

To that end, the local client image set was further divided into training and validation datasets to allow internal testing. This facilitated the direct measurement of model output at a local level and subsequent federated aggregation to provide a harmonized contribution to all clients.

3.3. Model Configuration

We use MobileNetV2 as an example of our backbone architecture, which offers a good trade-off between representational capacity and computation. ImageNet-pretrained weights were used to prime the network for feature extraction where feasible. The final classification layer was refined to generate two neurons, which were converted into binary categories: normal and pneumonia [21]. The model architecture in Figure 2 will be trained using the pretrained weights from the ImageNet dataset and the fine-tuned classification layers.

Local Loss Function:

$$\mathcal{L}_k(w) = \frac{1}{n_k} \sum_{(x,y)\in\mathcal{D}_k} \ell\,(w; x, y)$$

Global Objective:

$$\mathcal{L}(w) = \sum_{k=1}^{K} \frac{n_k}{n} \mathcal{L}_k(w)$$

```python
import torchvision.models as models
import torch.nn as nn

def get_model(num_classes=2, pretrained=True):
    try:
        model = models.mobilenet_v2(pretrained=pretrained)
        in_features = model.classifier[1].in_features
        model.classifier[1] = nn.Linear(in_features, num_classes)
    except Exception:
        model = models.resnet18(pretrained=pretrained)
        in_features = model.fc.in_features
        model.fc = nn.Linear(in_features, num_classes)
    return model

def count_parameters(model):
    return sum(p.numel() for p in model.parameters() if p.requires_grad)
```

Figure 2. Model configuration.

In this direction, a learning rate of 0.01 and a momentum of 0.9 were used to optimize Stochastic Gradient Descent (SGD), which enabled gradual convergence when the client datasets were heterogeneous. The cross-entropy loss function was employed to calculate the difference between actual and estimated labels. The design was chosen to achieve convergence, numerical stability, and scalability to other data partitions in a preliminary experiment.

© 2026 Conscientia Beam. All Rights Reserved.

3.4. Federated Learning Setup

The federated learning setup included four clients and a server. Every client underwent two local training steps in each round of communication and shared altered model parameters with the server. FedAvg computes averages of the local model parameter sets weighted by the sizes of the individual sets. The server then combined these updates into a new global model and distributed it to clients to initiate the next cycle [19]. Figure 3 illustrates the federated learning architecture and highlights the client-server federated training system.

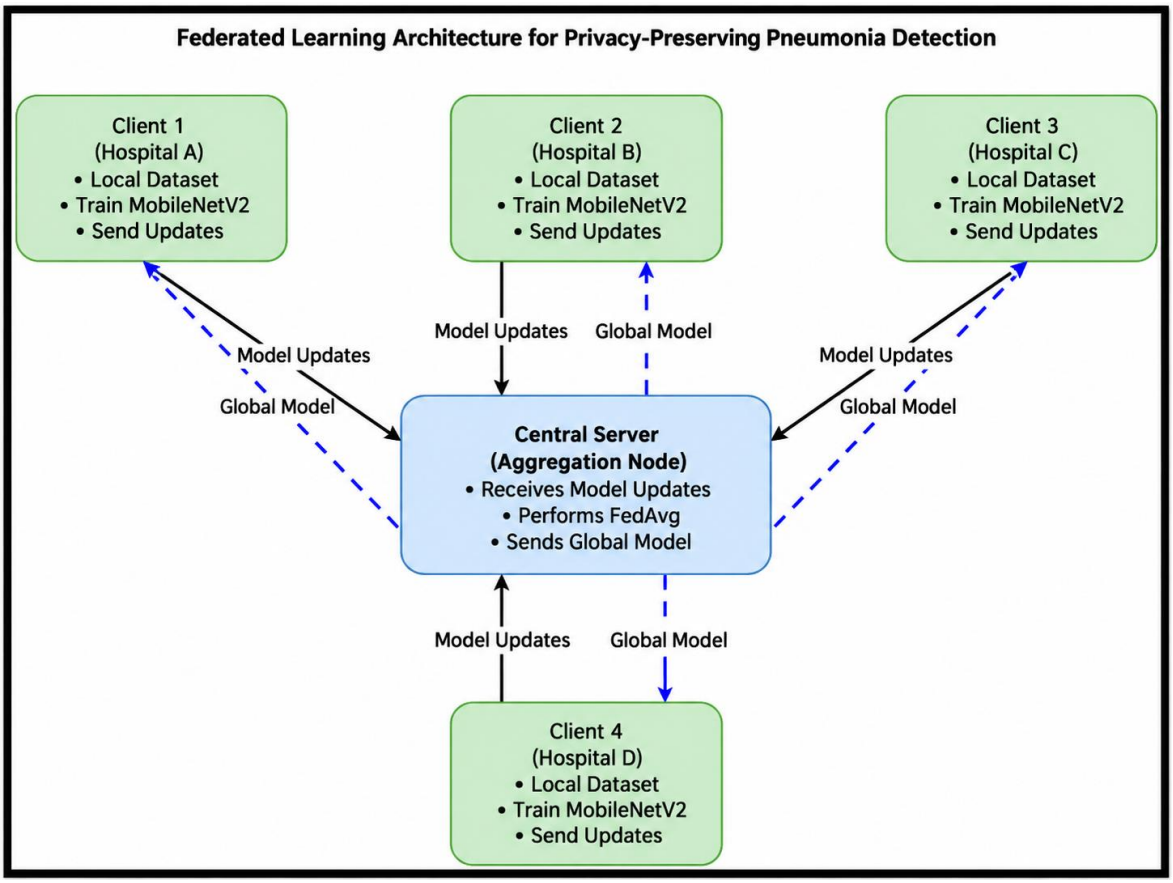


Figure 3. Federated learning architecture.

Federated Averaging (FedAvg):

$$w_{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} w_t^{(k)}$$

Local SGD Update:

$$w \leftarrow w - \eta \cdot \frac{1}{|B|} \sum_{i \in B} \nabla \ell(w; x_i, y_i)$$

Communication Cost:

$$C_{\text{comm}} = R \cdot K \cdot \left(|w|_{\text{upload}} + |w|_{\text{download}} \right)$$

This provided sufficient iterations to monitor convergence since the system had experienced ten rounds of communication.

The Flower (FLWR) codebase managed orchestration, client-server communication, and metric logging. This architecture enabled messages to flow through gRPC, asynchronous jobs to be scheduled, and distributed training to be coordinated effectively.

© 2026 Conscientia Beam. All Rights Reserved.

Algorithm 1 — Server: Federated Averaging (FedAvg)
Inputs:
R: Total number of communication rounds
K: Total number of clients
C_f: Client participation fraction ($0 < C_f \leq 1$)
w0: Initial global model parameters
use_deltas: Boolean; if true, clients send Δw; otherwise, full w
val_set: Optional validation dataset for server-side evaluation

Output:
w_R: Final global model parameters

Procedure:
1. w ← w0
2. for t = 0 to R-1 do
3. S_t ← select ⌈C_f · K⌉ clients (random sampling or scheduled selection)
4. for each client k ∈ S_t in parallel do
5. send current model w to client k
6. end for
7. receive from each k ∈ S_t: (n_k, payload_k)
8. where payload_k = (Δw_k) if use_deltas else (w_k)
9. if use_deltas then
10. for each k ∈ S_t do
11. w_k ← w + Δw_k
12. end for
13. end if
14. compute normalization denominator N_t ← ∑_{j∈S_t} n_j
15. w_new ← 0 (same shape as w)
16. for each k ∈ S_t do
17. weight ← n_k / N_t
18. if use_deltas then
19. w_new ← w_new + weight · (w + Δw_k)
20. else
21. w_new ← w_new + weight · w_k
22. end if
23. end for
24. w ← w_new
25. if val_set is provided then
26. evaluate w on val_set and log metrics
27. end if
28. end for
29. return w

The process of server-side orchestration in the federated learning framework is described in Algorithm 1. The server has to actively select a group of existing clients and send them the up-to-date global model at the beginning of each round of communication. The server receives locally updated models provided by the participating clients and performs Federated Averaging (FedAvg), in which the contribution of each client is weighted based on its dataset size. This creates a new global model, which is in turn transmitted to clients in the second round. The process is repeated a certain number of times until it converges.

Algorithm 2 — Client: Local Training & Update
ClientUpdate(k, w_received)
Inputs:
k: Client identifier
w_received: Model parameters received from server
D_k: Local dataset at client k
E: Number of local epochs
B: Local minibatch size
η: Local learning rate
Optimizer: Local optimizer spec (e.g., SGD with momentum)

© 2026 Conscientia Beam. All Rights Reserved.

send_delta: Boolean, if true, send Δw else send the full model
use_DP: Boolean, enable DP-SGD (clipping + Gaussian noise)
C: Clipping norm (used if use_DP)
σ: Noise multiplier (used if use_DP)

Output:
payload: what is sent to server (n_k and Δw_k or w_k)

Procedure:
w_local ← w_received
w_init ← copy(w_local) // keep initial copy if sending Δw
n_k ← |D_k| // number of local samples
for epoch = 1 to E do
shuffle D_k
for each mini-batch b of size B from D_k do
compute per-sample gradients {g_i} for i ∈ b where g_i = ∇ℓ(w_local; x_i,y_i)
if use_DP then
for each i ∈ b do
g_i ← g_i / max(1, ||g_i||_2 / C) // clip per-sample gradient
end for
g_bar ← (1/|b|) · ∑_{i∈b} g_i
noise ← Gaussian(0, (σ·C/|b|)^2) // sample noise vector
g_noisy ← g_bar + noise
grad_update ← g_noisy
else
grad_update ← (1/|b|) · ∑_{i∈b} g_i // standard batch gradient
end if
// apply optimizer step (here shown as SGD update for clarity)
w_local ← w_local - η · grad_update
end for
end for
if send_delta then
Δw ← w_local - w_init
payload ← (n_k, Δw)
else
payload ← (n_k, w_local)
end if
transmit payload to server
return

The local training process is outlined in Algorithm 2, which is executed separately on each client device. When the global model is provided to the client by the server, the client trains the model on its local data for a specified number of epochs using mini-batch stochastic gradient descent. If privacy measures, such as gradient clipping or differential privacy noise, are enabled, they are applied here. Once local training is complete, either the updated model weights or the parameter changes are sent back to the server.

3.5. Implementation Environment

It has been adapted to Python 3.10 in a virtual environment to isolate dependencies. PyTorch and TorchVision were the main machine learning libraries used to define models and preprocess images. Orchestration was federated through the Flower framework, and NumPy, Pandas, and TorchMetrics were used for numerical manipulation and performance analysis.

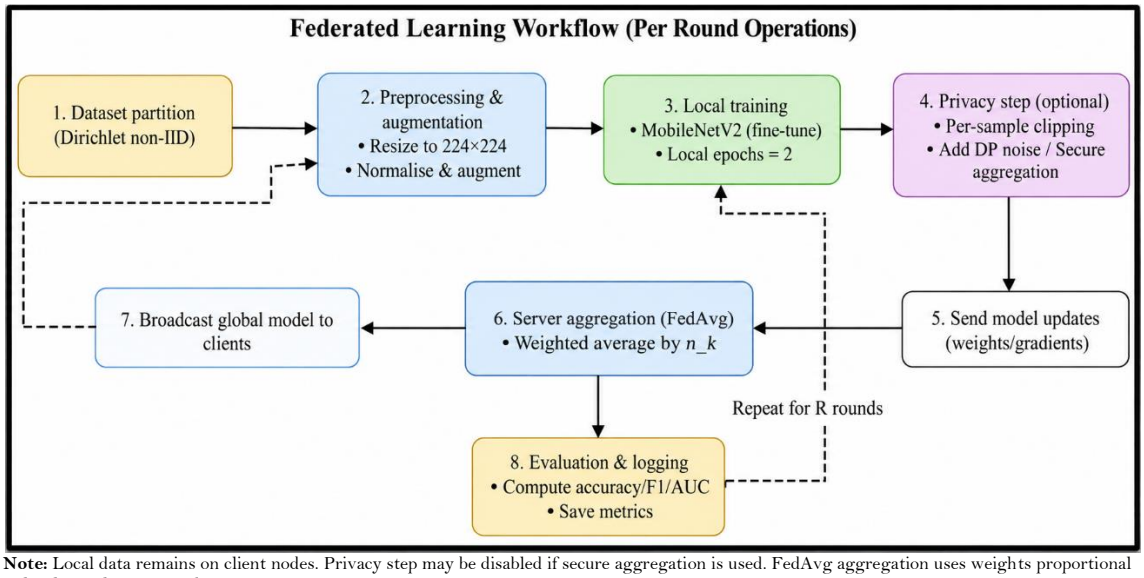
It was tested on a CPU-only (no GPU acceleration) Windows 10 platform. The lightness of MobileNetV2 enabled the model to be trained without specialized hardware [15]. The mean time per round of communication is not prohibitive, showing that even with small computational capacity, the model and framework can be realistic.

© 2026 Conscientia Beam. All Rights Reserved.

3.6. Workflow Overview

The development of the methodology was systematic and iterative. Data were first subdivided into four partitions based on the simulated clients. The clients obtained their own subsets and were trained at the local level, where two forward and backward propagation steps were utilized to optimize the model weights. Local training was followed by the upload of model parameters to the central server, where FedAvg aggregation was conducted [12].

Redistribution of the new global model was offered to all clients, and the second half of training was introduced. The model was improved by ten rounds. Accuracy, precision, recall, specificity, F1-score, and the area under the curve (AUC) were calculated and averaged at the client level to illustrate performance.



Note: Local data remains on client nodes. Privacy step may be disabled if secure aggregation is used. FedAvg aggregation uses weights proportional to local sample counts n_k.

Figure 4. Proposed workflow.

4. RESULTS AND ANALYSIS

4.1. Early Learning Behavior

The federated training process was completed successfully with 10 rounds of communication, and the model was observed to learn slowly in the initial stages. In round 1, the global model achieved a score of 0.7424 and an F1 score of 0.8521, indicating moderate performance and early learning skills globally. The specificity and bulk-adjusted score of 1.0000 resulted in a bias that classified most reports as pneumonia. This was an inevitable by-product, since the data were non-independent and not identically distributed (non-IID) across clients. Clients were unevenly distributed across classes, and the model was biased toward majority classes in certain local datasets [2].

The model, however, soon became biased. It was also very sensitive to pneumonia, which means it had high initial impressiveness and a low false-negative rate, both of which are very important for medical diagnostics.

4.2. Performance Progression

The global model had already achieved consistent convergence of >0.95 and an F1 score of ~0.98. The training loss was still much lower for the distributed clients, at 0.12 in round 2 compared with 0.49 in round 1, and these distributed clients were moving toward being able to update their weights and learn useful representations. This timing erosion curve shows that the two clients were involved in the federation's aggregation process because they provided local information that was very effective in aggregate [21]. The workflow of the federated training process is illustrated in Figure 4, while the performance results are shown in Figure 5, which displays the model performance for the 10 rounds of federated training.

© 2026 Conscientia Beam. All Rights Reserved.

By round 5, it had settled down and was accurate at 0.9828, with a precision of 0.9922 and an F1 of 0.9884. The Area Under the Curve (AUC) exceeded 0.9994, indicating that the model could differentiate between pneumonia and normal cases well. The progressive decrease in loss values indicated stable learning dynamics and helped avoid overfitting in the distributed training scenario.

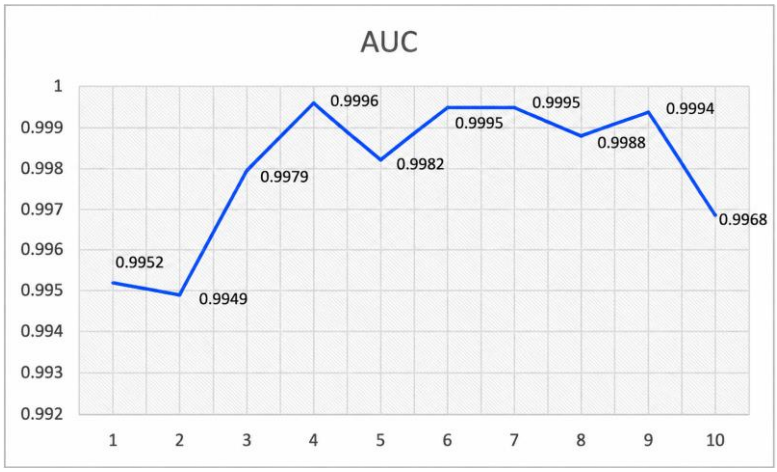


Figure 5. Training and validation AUC.

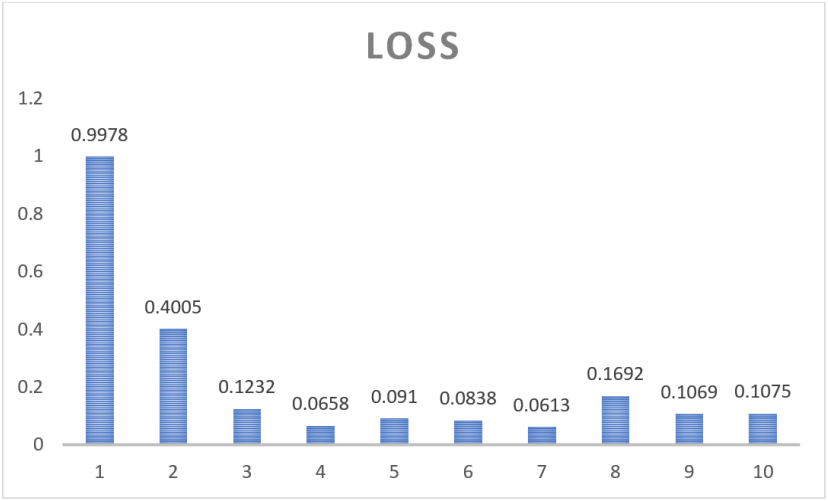


Figure 6. Training and validation LOSS.

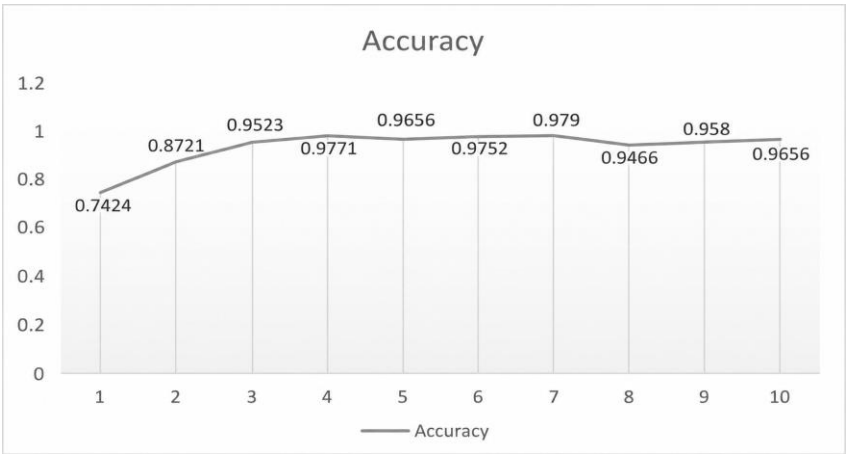


Figure 7. Training and validation accuracy.

© 2026 Conscientia Beam. All Rights Reserved.

In the final round of ten rounds, the model achieved a precision of 0.9924, as shown in Figure 7; an accuracy of 0.9949; a recall of 0.9949; a specificity of 0.9852; and an AUC of 0.9998. The cross-entropy loss was 0.03, as shown in Figure 6. The specificity increase was large, suggesting that the model could better identify healthy cases with few false positives.

Interestingly, all clients demonstrated the same typical level of generalisation performance, indicating that there was no interference from data diversity with convergence. Such an overall performance improvement over time confirms that the federated updates have been effective in learning the discriminative properties of all distributed data and in learning the decision boundary of the global model to a level that enables it to learn from all distributed data.

Table 1. Result matrix.

Round	Loss	Accuracy	Precision	Recall	Specificity	F1	AUC
1	0.9978	0.7424	0.7424	1.0000	0.0000	0.8521	0.9952
2	0.4005	0.8721	0.8531	1.0000	0.5037	0.9207	0.9949
3	0.1232	0.9523	0.9417	0.9974	0.8222	0.9688	0.9979
4	0.0658	0.9771	0.9701	1.0000	0.9111	0.9848	0.9996
5	0.0910	0.9656	0.9603	0.9949	0.8815	0.9773	0.9982
6	0.0838	0.9752	0.9677	1.0000	0.9037	0.9836	0.9995
7	0.0613	0.9790	0.9725	1.0000	0.9185	0.9861	0.9995
8	0.1692	0.9466	0.9329	1.0000	0.7926	0.9653	0.9988
9	0.1069	0.9580	0.9465	1.0000	0.8370	0.9725	0.9994
10	0.1075	0.9656	0.9580	0.9974	0.8741	0.9773	0.9968

4.3. Interpretation and Observations

The training results of the federated model provide strong confirmation that the FedAvg algorithm is indeed effective in synchronizing distributed knowledge among clients. Two local epochs in the MobileNetV2 round were identified that simultaneously achieved very good generalisation and computational efficiency. Transfer learning further helped convergence by using a model fine-tuned on ImageNet to learn high-level visual patterns from chest X-ray images [15].

The results of the global model across various rounds show that federated learning is robust to client imbalance. Iterative aggregation reversed the effects of early-round asymmetric learning, which had a single dominant IID client, whereas non-IID data were dominant.

The AUC value of 0.9998 means that the model classifies pneumonia and normal images almost perfectly, suggesting the model's diagnostic power [6]. It is noteworthy that the final training loss is low, reflecting the fact that the model has achieved substantial convergence without overfitting, while the balanced precision and recall are predictive, suggesting that the model can be used in a real-world diagnostic scenario.

5. EVALUATION AND DISCUSSION

The results in Table 1 show that federated learning can be a potential trend in medical image processing techniques under strict privacy protection. There were no irregularities in the rounds, and the measurements were nearly perfect toward the end. MobileNetV2 was a beneficial option, as it was possible to process the calculations efficiently and save communication costs when passing the parameters. Theoretical evidence supports the experimental evidence that distributed models can be as efficient as their centralised counterparts [17].

The system was not weak and could scale because of these restrictions. These results are particularly impressive for such low computing power. The analysis demonstrates that FL can be used in a realistic setting and, consequently, can be applied in small health care settings with scarce hardware.

© 2026 Conscientia Beam. All Rights Reserved.

6. CONCLUSION AND FUTURE WORK

This research was able to build and test a federated learning system to identify pneumonia using MobileNetV2 on the Chest X-Ray Kaggle. The model was built on four clients and one server, with ten communication rounds and two local epochs per client. The final model had a global accuracy of 0.9924, an F1 score of 0.9949, and an AUC of 0.9998, indicating strong classification performance over time. The findings support the notion that federated learning could preserve diagnostic performance on par with conventional centralised models while providing data privacy and regulatory compliance. The research also demonstrates how lightweight architectures and federated learning systems can be effectively implemented. MobileNetV2 was also compact enough to be used with less powerful hardware without a loss of accuracy. Future research will include the implementation of differential privacy and homomorphic encryption to enhance the security of parameter communication. To elevate clinical significance, multi-class classification, i.e., the ability to differentiate between pneumonia, COVID-19, and tuberculosis, should be added.

Funding: This study received no specific financial support.
Institutional Review Board Statement: Not applicable.
Transparency: The authors state that the manuscript is honest, truthful, and transparent, that no key aspects of the investigation have been omitted, and that any differences from the study as planned have been clarified. This study followed all writing ethics.
Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: Both authors contributed equally to the conception and design of the study. Both authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] M. Adnan, S. Kalra, J. C. Cresswell, G. W. Taylor, and H. R. Tizhoosh, "Federated learning and differential privacy for medical image analysis," *Scientific Reports*, vol. 12, no. 1, p. 1953, 2022. https://doi.org/10.1038/s41598-022-05539-7

[2] M. Ali, F. Naeem, M. Tariq, and G. Kaddoum, "Federated learning for privacy preservation in smart healthcare systems: A comprehensive survey," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 778-789, 2023. https://doi.org/10.1109/JBHI.2022.3181823

[3] M. Asad, S. Shaukat, E. Javanmardi, J. Nakazato, and M. Tsukada, "A comprehensive survey on privacy-preserving techniques in federated recommendation systems," *Applied Sciences*, vol. 13, no. 10, p. 6201, 2023. https://doi.org/10.3390/app13106201

[4] G. B. Krishna, K. Lasyasri, G. U. Reddy, and C. P. Goud, "Development of effective ESG systems in higher educations," *International Journal of Engineering Innovations and Management Strategies*, vol. 1, no. 4, pp. 1–17, 2024.

[5] D. Avola, A. Bacciu, L. Cinque, A. Fagioli, M. R. Marini, and R. Taiello, "Study on transfer learning capabilities for pneumonia classification in chest-x-rays images," *Computer Methods and Programs in Biomedicine*, vol. 221, p. 106833, 2022. https://doi.org/10.1016/j.cmpb.2022.106833

[6] C. U. Om Kumar, S. Gajendran, R. M. Bhavadharini, M. Suguna, and R. Krithiga, "EHR privacy preservation using federated learning with DQRE-Scnet for healthcare application domains," *Knowledge-Based Systems*, vol. 275, p. 110638, 2023. https://doi.org/10.1016/j.knosys.2023.110638

[7] C. S. Savanam, P. Avinash, E. Suneetha, and G. Bala Krishna, "Analysis on application of IoT in biomedical instrumentation," *International Journal of Mechanical Engineering*, vol. 7, no. 2, pp. 424–429, 2022.

[8] A. Nagaraju, T. S. Reddy, and T. S. Prasad, "Gradient boosting algorithm for chronic obstructive pulmonary disease (COPD) risk prediction in machine learning," *International Journal of Drug Delivery Technology*, vol. 16, no. 1s, pp. 1089–1096, 2026. https://doi.org/10.25258/ijddt.16.1s.120

[9] S. M. Maturi, C. Dastagiraiah, P. Sowjanya, C. K. Kumar, and J. S. V. R. S. Sastry, "A comparison of the diagnosis performance of machine learning algorithms on the Breast Cancer Wisconsin Dataset," *Engineering, Technology & Applied Science Research*, vol. 15, no. 6, pp. 30586–30590, 2025. https://doi.org/10.48084/etasr.13706

© 2026 Conscientia Beam. All Rights Reserved.

[10] P. Vinayasree and A. M. Reddy, "A scalable and secure blockchain-based healthcare system: Optimizing performance, security, and privacy with adaptive technologies," *Journal of Theoretical and Applied Information Technology*, vol. 102, no. 22, pp. 8084–8103, 2024.

[11] S. Iqbal *et al.*, "LDMRes-Net: A lightweight neural network for efficient medical image segmentation on iot and edge devices," *IEEE Journal of Biomedical and Health Informatics*, vol. 28, no. 7, pp. 3860-3871, 2024. https://doi.org/10.1109/JBHI.2023.3331278

[12] M. F. Sohan and A. Basalamah, "A systematic review on federated learning in medical image analysis," *IEEE Access*, vol. 11, pp. 28628-28644, 2023. https://doi.org/10.1109/ACCESS.2023.3260027

[13] T. J. Loftus *et al.*, "Federated learning for preserving data privacy in collaborative healthcare research," *Digital Health*, vol. 8, p. 20552076221134455, 2022. https://doi.org/10.1177/20552076221134455

[14] Z. Lu, H. Pan, Y. Dai, X. Si, and Y. Zhang, "Federated learning with non-IID data: A survey," *IEEE Internet of Things Journal*, vol. 11, no. 11, pp. 19188-19209, 2024. https://doi.org/10.1109/JIOT.2024.3376548

[15] S. Tayebi Arasteh, C. Kuhl, M.-J. Saehn, P. Isfort, D. Truhn, and S. Nebelung, "Enhancing domain generalization in the AI-based analysis of chest radiographs with federated learning," *Scientific Reports*, vol. 13, no. 1, p. 22576, 2023. https://doi.org/10.1038/s41598-023-49956-8

[16] P. Zhang, B. Qi, and W. Huang, "Enhanced ship target detection with YOLOv5s utilizing lightweight mobileNetV2 and inverted residual block structures," presented at the 2024 IEEE 6th International Conference on Civil Aviation Safety and Information Technology (ICCASIT), IEEE, 2024.

[17] A. Rauniyar *et al.*, "Federated learning for medical applications: A taxonomy, current trends, challenges, and future research directions," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 7374-7398, 2024. https://doi.org/10.1109/JIOT.2023.3329061

[18] A. Raza, A. Guzzo, and G. Fortino, "Federated learning for medical images analysis: A meta survey," presented at the 2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), IEEE, 2023.

[19] ResearchGate, *The proposed mobileNetV2 network architecture*. Germany: ResearchGate, 2025.

[20] R. Sultana, S. Rawat, G. V. Murthy, and N. Kumar, "An investigation on managing patient flow at hospital emergency care unit using tree-based data mining techniques. In R. R. Chillarige, S. Distefano, & S. S. Rawat (Eds.)," in *Advances in Computational Intelligence and Informatics: Proceedings of ICACII 2019 (Lecture Notes in Networks and Systems). Singapore: Springer Singapore*, 2020, vol. 119, pp. 237–243.

[21] M. Rajkumar, S. R. Siriyala, A. Sharma, and D. N. S. Kumar, "Comparative analysis of ResNet, EfficientNet, and MobileNetV2 for detecting skin conditions in deep learning models," presented at the 2024 International Conference on Sustainable Communication Networks and Application (ICSCNA), IEEE, 2024.

Views and opinions expressed in this article are the views and opinions of the author(s), Review of Computer Engineering Research shall not be responsible or answerable for any loss, damage or liability etc. caused in relation to/arising out of the use of the content.

© 2026 Conscientia Beam. All Rights Reserved.