Joceline Ziegler | Bjarne Pfitzner | Heinrich Schulz | Axel Saalbach | Bert Arnrich

Defending against Reconstruction Attacks through Differentially Private Federated Learning for Classification of Heterogeneous Chest X-ray Data

Suggested citation referring to the original publication:
Sensors 22 (2022), Art. 5195 pp. 2 - 25
DOI https://doi.org/10.3390/s22145195
ISSN 1424-8220

Journal article | Version of record

sensors

MDPI

Article

Defending against Reconstruction Attacks through Differentially Private Federated Learning for Classification of Heterogeneous Chest X-ray Data

Joceline Ziegler [1,2,*], Bjarne Pfitzner [1,2], Heinrich Schulz [3], Axel Saalbach [3] and Bert Arnrich [1,2]

[1] Digital Engineering Faculty, University of Potsdam, 14482 Potsdam, Germany; bjarne.pfitzner@hpi.de (B.P.); bert.arnrich@hpi.de (B.A.)

[2] Hasso Plattner Institute for Digital Engineering gGmbH, 14482 Potsdam, Germany

[3] Philips Research, 22335 Hamburg, Germany; heinrich.schulz@philips.com (H.S.); axel.saalbach@philips.com (A.S.)

* Correspondence: post@jocelineziegler.de

Abstract: Privacy regulations and the physical distribution of heterogeneous data are often primary concerns for the development of deep learning models in a medical context. This paper evaluates the feasibility of differentially private federated learning for chest X-ray classification as a defense against data privacy attacks. To the best of our knowledge, we are the first to directly compare the impact of differentially private training on two different neural network architectures, DenseNet121 and ResNet50. Extending the federated learning environments previously analyzed in terms of privacy, we simulated a heterogeneous and imbalanced federated setting by distributing images from the public CheXpert and Mendeley chest X-ray datasets unevenly among 36 clients. Both non-private baseline models achieved an area under the receiver operating characteristic curve (AUC) of 0.94 on the binary classification task of detecting the presence of a medical finding. We demonstrate that both model architectures are vulnerable to privacy violation by applying image reconstruction attacks to local model updates from individual clients. The attack was particularly successful during later training stages. To mitigate the risk of a privacy breach, we integrated Rényi differential privacy with a Gaussian noise mechanism into local model training. We evaluate model performance and attack vulnerability for privacy budgets $\varepsilon \in \{1, 3, 6, 10\}$. The DenseNet121 achieved the best utility-privacy trade-off with an AUC of 0.94 for $\varepsilon = 6$. Model performance deteriorated slightly for individual clients compared to the non-private baseline. The ResNet50 only reached an AUC of 0.76 in the same privacy setting. Its performance was inferior to that of the DenseNet121 for all considered privacy constraints, suggesting that the DenseNet121 architecture is more robust to differentially private training.

Keywords: federated learning; privacy and security; privacy attack; X-ray

check for updates

Citation: Ziegler, J.; Pfitzner, B.; Schulz, H.; Saalbach, A.; Arnrich, B. Defending against Reconstruction Attacks through Differentially Private Federated Learning for Classification of Heterogeneous Chest X-ray Data. *Sensors* **2022**, *22*, 5195. https://doi.org/10.3390/s22145195

Academic Editors: Fiammetta Marulli and Laura Verde

Received: 6 May 2022
Accepted: 5 July 2022
Published: 11 July 2022

1. Introduction

The development of machine learning models for medical use cases often requires collecting large amounts of sensitive patient data. Medical datasets are usually scattered across multiple sites and underlie rigorous privacy constraints of both ethical and regulatory natures [1]. The effectiveness of anonymization to enable data sharing is dependent on the type of data and cannot always prevent re-identification [2]. Federated learning gains increasing attention as a method for training machine learning models on distributed data in a privacy-preserving manner. In a federated learning setting, holders of sensitive data can make their data available for machine learning without sharing it with other parties. In several iterations, a central server distributes an initial model to several clients holding the data, e.g., medical institutions, which then individually train their models and provide them to the server for aggregation.

Federated learning provides a basic level of privacy by the principle of data minimization, i.e., data collection and processing are restricted to a necessary minimum. However, it cannot by itself formally guarantee privacy [3]. It has been shown that input data can successfully be reconstructed from model gradients [4,5]. In addition to the threat of data reconstruction, attacks disclosing the presence of a specific data sample or property in the training data imply a serious privacy risk for individual contributors [6].

Measures to prevent privacy breaches of machine learning models are subject to ongoing research. Differential privacy is a concept actively explored in this field. Intuitively, the goal of differential privacy is to limit the impact of a single data sample or a subset of the data on the outcome of a function computed on the data, thereby providing a guarantee that no or little information can be inferred about individual samples [6]. However, the application of differential privacy is known to decrease the utility of the machine learning model, characterized by a trade-off between utility and privacy specific to each use case [3,7–9]. Despite the potential of differentially private federated learning in healthcare, there has been an increased research interest only recently in selected use cases.

Vast amounts of medical image data are currently produced in daily medical practice. Chest X-rays play an essential role in diagnosing a variety of diseases, such as pneumonia [10], as well as recently in studying COVID-19 [11]. Automatic diagnosis assistance may substantially support the work of radiologists, which is particularly of interest in the face of ongoing medical specialist shortages [12]. Digital support systems may also mitigate the impact of error sources in human assessment that occur systematically, e.g., due to increased workload and varying professional experience [13]. Increased costs for the healthcare system and potentially fatal misdiagnoses can thereby be avoided.

We evaluate the potential of privacy-preserving federated learning for the use case of disease classification on chest X-ray images. As a key contribution, we directly compare two popular image classification model architectures, DenseNet121 and ResNet50, in terms of the effects of differentially private training on model performance and privacy preservation. Extending previous work, we introduce a federated environment that is subject to data heterogeneity and imbalance. We demonstrate that the basic federated learning setting is vulnerable to privacy violation through the successful application of reconstruction attacks. We specifically compare the vulnerability to privacy breach and the effect of differential privacy on a previously unconsidered complex model, DenseNet121, with the previously studied ResNet architecture. Our results endorse the conjecture that reconstruction attacks pose a realistic threat within the federated learning paradigm, even for large and complex model architectures. We integrate Rényi differential privacy into the federated learning process and investigate how it affects the utility-privacy trade-off for our use case. Two measures of privacy are addressed: The privacy budget ε as part of the formal differential privacy guarantee and the susceptibility of the local models to reconstruction attacks. Our results suggest that the DenseNet121 is a promising architecture for feasible privacy-preserving model training on X-ray images. This novel insight may direct future research and applications in that area.

This paper is structured as follows: In Section 2, we briefly present previous work related to privacy-preserving federated learning for the task of X-ray classification. We introduce the used datasets in Section 3.1 and explain our federated learning setup in Section 3.2. We provide background information on the *Deep Leakage from Gradients (DLG)* attack (Section 3.3) and on the integration of differential privacy into the training of neural networks (Section 3.4). We present the results on model performance in a basic federated learning setting (Section 4.1), demonstrate the susceptibility of our federated learning models to reconstruction attacks (Section 4.2), and finally evaluate the impact of differential privacy on model performance and attack vulnerability (Section 4.3). We discuss and summarize our findings in Sections 5 and 6.

2. Related Work

The healthcare sector especially profits from privacy-preserving machine learning due to the natural sensitivity of the underlying patient data [1,14,15]. A wide range of applications demonstrate that federated learning is a potential fit for leveraging diverse types of medical data, including electronic health records [16], genomic data [17], and time-series data from wearables [18]. Examples related to medical image classification include brain tumor segmentation [9,19], classification and survival prediction on whole slide images in pathology [20], classification of functional magnetic resonance images (fMRI) [21], and breast density classification from mammographic images [22]. One large research area is concerned with the classification of chest X-ray images. Çallı et al. [23] provided an overview of recent deep learning advances in this field, but do not consider federated learning. The feasibility of federated learning on chest X-rays has previously been benchmarked for both the CheXpert [24] and the Mendeley [25] dataset. Chakravarty et al. [26] enhance a ResNet18 architecture with a graph neural network for federated learning on CheXpert data with site-specific data distributions. Nath et al. [27] deploy a DenseNet121 model for a real-world, physically distributed implementation of federated learning on CheXpert. Banerjee et al. [28] determine the ResNet18 architecture to be superior for federated learning on the Mendeley data in comparison with ResNet50, DenseNet121, and MobileNet. Table 1 summarizes related work on chest X-ray classification with DenseNet or ResNet architectures.

Table 1. Overview of related works evaluating deep neural networks on the CheXpert or Mendeley datasets using DenseNet or ResNet architectures. The mentioned models are not necessarily exhaustive; some papers evaluate additional ResNet and DenseNet architectures. We also include our paper at the bottom for comparison with related work. Note: *(non-)IID* corresponds to a (not) independent and identical data distribution. *DP* corresponds to the use of differential privacy ($\varepsilon = 6$).

Data	Reference	Model	Federated Learning	AUC
CheXpert	Irvin et al. [24]	DenseNet121	no	0.889
	Bressem et al. [29]	DenseNet121 ResNet50	no	0.869 0.881
	Ke et al. [30]	DenseNet121 ResNet50	no	0.859 0.859
	Chakravarty et al. [26]	ResNet18	5 sites, non-IID	0.782
	Nath et al. [27]	DenseNet121	5 sites, IID	0.803
Mendeley	Banerjee et al. [28]	ResNet50	3 sites, non-IID (the data distribution between the three hospitals is 30:32:38, with slightly varying class distribution)	0.976 (no AUC given, the value corresponds to the binary accuracy)
	Kaissis et al. [8]	ResNet18	no 3 sites (data distribution unknown) 3 sites (data distribution unknown), DP	0.93 0.92 0.89
Both	This paper	DenseNet121	36 sites, non-IID 36 sites, non-IID, DP	0.935 0.937
		ResNet50	36 sites, non-IID 36 sites, non-IID, DP	0.938 0.764

Surveys on current developments in the field of privacy-preserving machine learning and federated learning describe potential threat models and privacy attacks [3,31,32]. Zhu et al. [5] originally proposed the Deep Leakage from Gradients (DLG) attack, which allows a malicious server instance to reconstruct complete data samples from received model gradients. Subsequent improvements to the idea include analytical label reconstruction [33], improved loss functions for gradient matching [4,34] and an extension towards larger batch sizes [35]. DLG and other privacy attacks have been identified as a severe threat to federated learning. Wei et al. [36] evaluate the impact of attack initialization, optimization method, and training parameters including batch size, image resolution, and activation function on DLG attack's success with a small network.

Ensuring privacy and protecting against reconstruction in a practicable manner is not yet fully explored and remains an open problem for the federated learning paradigm [1,3,37,38]. A key method that finds wide use among federated learning research is *differential privacy*, first proposed by Dwork [39] in the context of database systems. Generally, it describes the addition of carefully crafted noise into a system to prevent learning too much about single data instances and measuring the remaining risk. Mironov [40] introduced the variant of *Rényi differential privacy*, defining a tighter bound on the privacy loss. Differential privacy in federated learning is often achieved using *differentially-private stochastic gradient descent* (DP-SGD) [7,41,42], an algorithm that determines the appropriate noise scale and how to clip the model parameter. The combination of federated learning and differential privacy has been explored in multiple medical use cases, including prediction of mortality and adverse drug reactions from electronic health records [43], brain tumor segmentation [9], classification of pathology whole slide images [20], detection of diabetic retinopathy in images of the retina [44], and identification of lung cancer in histopathologic images [45].

Most similarly to this work, Kaissis et al. [8] demonstrate a framework for the implementation and evaluation of privacy-preserving machine learning in a federated learning setting on the Mendeley dataset evenly distributed among three clients. They combine a ResNet18 model with a secure multi-party computation protocol and differential privacy and compare the success of reconstruction attacks on centralized and federated learning models. We extend their setting by considering larger networks, simulating a scenario with heterogeneous data unevenly distributed among a larger number of clients, and evaluating the impact of different parameters on model performance and the model's vulnerability to reconstruction attacks.

3. Materials and Methods

In this section, we first provide information about the used datasets. Then, we go over the three central pieces of our article: the federated learning baseline and our heterogeneous data distribution, the reconstruction attack, and finally the introduction of differential privacy as a defense against the attack.

3.1. Data

CheXpert [24] comprises 224,316 images of 65,240 adult patients in total, where 234 images are labeled by professional radiologists for use as a validation set. We only considered frontal view images as this accounts for the higher prevalence of frontal view images in the clinical setting and ensures compatibility with the Mendeley dataset. Each image is labeled with one or more of thirteen classes referring to a medically relevant finding, or with "No Finding". Following previous work [46,47], uncertain labels were considered as negative (U-Zeroes method).

The Mendeley chest X-ray dataset version 3 [25] contains 5856 images of pediatric patients and is split into original training and test sets with 5232 and 624 images. Each image is labeled as either "Normal", with "Viral Pneumonia", or "Bacterial Pneumonia". For convenience, we assume that "Normal" in the Mendeley dataset corresponds to "No Finding" in the CheXpert dataset. To ensure compatibility between the dataset labels, our primary setting is a binary classification task based on the "No Finding" or "Normal" labels, indicating the presence or absence of a medically relevant condition.

3.2. Federated Learning

Successful training of a deep learning model usually relies on the availability of a single large, high-quality dataset, requiring prior data collection and curation, which are potentially associated with great expense in time and resources. Despite such efforts, data transfer or direct access to the data can still often not be granted due to patient privacy concerns. Federated learning enables model training on scattered data that remains at the participants' sites at all times [48].

A typical federated learning system consists of a central server that orchestrates the training procedure, and several clients that communicate with the server (Figure 1). The server initializes a model and distributes the model parameters to its clients. In parallel, a subset of clients trains the model individually on their data for a defined number of epochs, which is equivalent to local stochastic gradient descent (SGD) optimization. The clients send their local models back to the server, where they are aggregated through *federated averaging* [48]. The new global model is again distributed among the clients, and the process is repeated until convergence or until a defined number of communication rounds has been reached.

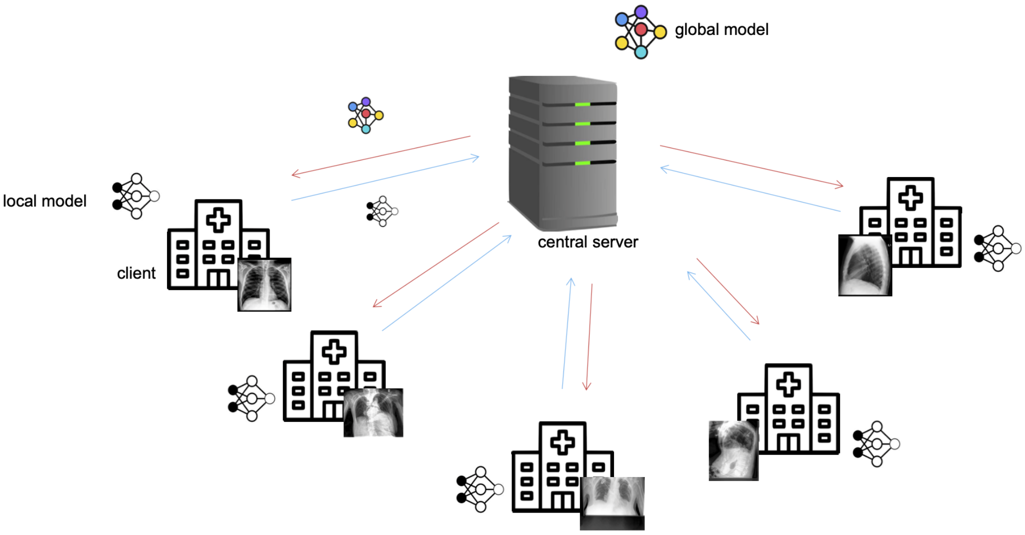


Figure 1. In the federated learning setup, the server first initializes a model and distributes the model parameters to its clients. Over several iterations, each client trains the model individually on its data for a defined number of local epochs, sends the parameters of its locally trained model back to the server for aggregation, and receives a global model, aggregated from all trained local models.

3.2.1. Experimental Setup

In real-world use cases, the expectation is that datasets between clients in a federated learning setting show some variety. We reflect this in our simulation of a federated environment by combining two public X-ray datasets representing heterogeneous target populations; adult and pediatric patients. Our federated learning setup comprises 36 clients that each hold a subset of X-ray images from either the CheXpert or the Mendeley dataset. The clients represent hospitals or other medical institutions that provide their collected X-rays for the development of a classification model. The sizes of the clients' datasets are chosen such that they create a highly imbalanced setting including clients with very few data points, which represents small institutions that make their limited amount of data available as soon as they are collected.

We simulated five clients with large subsets of the original CheXpert training set, and thirty-one clients with small subsets of either the original CheXpert validation set or the original Mendeley training set. We randomly split the patients whose images are part of the original CheXpert training set into five equal parts and assigned each part randomly to one of five clients. Table 2 shows the distribution of the CheXpert validation data and Mendeley data among the remaining 31 clients, split in training, validation and test set sizes. These clients are used as targets for the reconstruction attacks in Sections 4.2 and 4.3.3. Each client's dataset was further split into a dedicated training, validation, and test set, consisting of 70%, 15%, and 15% of the client's data, respectively. Clients' datasets that are smaller than 50 images were split equally among the subsets. Datasets comprising less than ten images were used solely for training, omitting local validation or testing. All splits were performed randomly. No specific label distribution was enforced. We ensured that there was no patient overlap between clients and between training, validation, and test splits within each client's dataset.

Table 2. Number of images from the Mendeley training dataset (**a**) and Chexpert validation dataset (**b**), distributed among 14 and 17 clients, respectively. We specify how many clients are included that hold the respective amount of data. The last row shows the total of previous rows, taking into account the number of clients.

(a) Mendeley Clients.

No. Clients	Train	Val.	Test	Total
2	350	75	75	500
2	140	30	30	200
2	70	15	15	100
2	10	10	10	30
2	4	3	3	10
2	2	0	0	2
2	1	0	0	1
14	1686	1154	266	266

(b) CheXpert Clients.

No. Clients	Train	Val.	Test	Total
2	10	10	10	30
5	4	3	3	10
5	2	0	0	2
5	1	0	0	1
17	125	55	35	35

3.2.2. Model Training

We compared a densely connected network (DenseNet) [49] and a residual network (ResNet) [50] because both architectures have proven especially successful for the task of X-ray image classification [30,51]. We monitored the local models' performance during training on their client's validation set. The global, aggregated model was evaluated using the average performance over the clients' validation sets. The client's test sets were held back for unbiased, internal evaluation of the final global model after training had finished. As the performance metric, we used the area under the receiver operating characteristics curve (AUC).

Both the DenseNet121 and ResNet50 models were initialized with parameters pre-trained on ImageNet data. A fully connected layer with sigmoid activation and the adjusted number of output neurons replaced the original final classification layer. We modified the model architectures to accept one channel grayscale instead of three-channel RGB inputs to reduce unnecessary model complexity. To still leverage pre-trained model parameters, we summed the three-channel parameters of the first model layer to obtain new weights for the one-channel input. Images were resized to 224 × 224 pixels and normalized with ImageNet parameters adapted to grayscale color encoding by averaging over the input channels, yielding the normalization parameters $\mu = 0.449$ and $\sigma = 0.226$. We did not apply any data augmentation methods.

Because multiple training rounds on the same dataset increase the risk of privacy leakage, we did not perform hyperparameter tuning and settled on standard hyperparameters. The training ran for at most 20 communication rounds. Each client participated in every round. We set the local batch size to ten and adapted it accordingly for clients with fewer than ten data points. To avoid overfitting, clients performed a single local epoch [2,19]. Early stopping was applied if the AUC value of the global model did not improve for five consecutive rounds. We minimized the binary cross-entropy loss using SGD with an initial learning rate of 0.01. The learning rate was reduced by a factor of 0.1 when reaching a performance plateau, i.e., after the AUC of the global model has not improved for three consecutive rounds. The global model with the highest mean AUC across all clients was selected as the best final model.

In private training, the privacy loss is difficult to track for some layer types. This includes active batch normalization layers, which are part of both the DenseNet and ResNet architectures, as they create arbitrary dependencies between samples within a single batch [52]. We experimented with different model layer freezing techniques to avoid training batch normalization layers resulting in intractable privacy loss. We refer to rendering model layers untrainable as *layer freezing*. We considered full model training (no layer freezing), freezing batch normalization layers, and freezing all layers but the final classification layer.

3.3. Reconstruction Attack

Federated learning enables model training on distributed data without the need for direct data sharing. However, while federated learning satisfies the principle of data minimization by eliminating the need for data transfer, it is not by itself sufficiently privacy-preserving. Sensitive information about the training data can be inferred from shared models, which has been demonstrated in a variety of privacy attacks including inference of class representatives [53], property inference [54], membership inference [55], and sample reconstruction [5].

We assume the server to be an honest-but-curious adversary with full knowledge of the federated as well as local training procedures [3]. It correctly orchestrates and executes the required computations. However, it has white-box access to shared model parameters and can passively investigate them without interfering with the training process.

Reconstruction attacks aim to recover data samples from trained model parameters. A disclosure implies a serious privacy risk as X-ray images may reveal information about the patient's identity [56] and sensitive properties, such as patient age [57]. Since reconstruction attacks can be conducted with little auxiliary information and in a passive manner, it is a relevant vulnerability within our threat model.

The DLG attack enables pixel-wise reconstruction of training images from the model gradients obtained during SGD [5]. The attack comprises the following steps:

1. Randomly initialize some dummy input data x' and dummy label y'.
2. Fit the given initial model with the dummy data and obtain dummy gradients $\nabla\theta'$.
3. Quantify the difference between the original and the dummy gradient by using the Euclidean (ℓ_2) distance as the cost function:

$$d_{grad} = \|\nabla\theta' - \nabla\theta\|^2 \tag{1}$$

4. Iteratively minimize the distance between the dummy and original gradients by adjusting the dummy input and label using the following objective:

$$x'^*, y'^* = \arg\min_{x',y'} \|\nabla\theta' - \nabla\theta\|^2 \tag{2}$$

5. End the optimization process when the loss is sufficiently small, indicating complete reconstruction of the input data, or when reaching a maximum number of iterations.

Following subsequent work, we used an improved version of the attack. We assume that labels can be reconstructed analytically [33] and restrict the optimization to the image data. We used a loss function based on the cosine similarity between original and dummy gradients and the Adam optimizer as proposed by Geiping et al. [4]. The cosine similarity loss is defined as follows:

$$d_{grad} = 1 - \frac{\langle \nabla\theta', \nabla\theta \rangle}{\|\nabla\theta' - \nabla\theta\|} + \alpha TV(\mathbf{x}'). \tag{3}$$

where $TV(\mathbf{x}')$ is the total variation of the dummy image $\mathbf{x}'$, with factor α as a small prior. The loss is minimized based on the sign of its gradient.

To evaluate the vulnerability of the local models to the DLG attack in our federated learning setting, we simulated an adversarial server that applies the reconstruction attack to model updates received from individual clients. We chose an arbitrary client holding one image from the Mendeley dataset to evaluate the impact of model layer freezing and attack time on image reconstruction quality. We then attacked other clients holding up to ten training images to demonstrate that they are also susceptible to a privacy breach. We conducted three trials per attack, initializing dummy images from a random normal distribution. We determined the best result as the trial with the lowest cosine similarity loss. The initial learning rate of the Adam optimizer for the attack was 0.1. We adopted the strategy from Geiping et al. [4] and reduced the learning rate by a factor of 0.1 after 3/8th, 5/8th, and 7/8th of the maximum number of iterations. Each trial ran for 20,000 optimization steps. The total variation factor α for the cosine similarity loss was 0.01. We inferred the model gradients by computing the absolute difference between the original model parameters and the local model parameters after local training.

For quantitative evaluation of attack success, we used the peak signal-to-noise ratio (PSNR), measured in the unit of decibels (dB):

$$PSNR = 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right), \tag{4}$$

where MAX_I is the difference between the minimum and the maximum possible pixel value, and MSE is the mean squared error (MSE) between two images.

In addition to quantifying attack success with the PSNR measure, we demonstrate to what degree sensitive patient information can be derived from reconstructed X-ray images. Even if an individual cannot always be identified directly from a particular image, statistical knowledge about demographic information and other sensitive properties in a given dataset may lead to unwanted conclusions about individuals. We compare the performance of auxiliary models that predict demographic patient information from original and reconstructed X-rays. Because there is no demographic patient information available for the Mendeley data, we focus our evaluation on clients holding parts of the CheXpert dataset. We centrally trained two auxiliary ResNet50 models with the original CheXpert training data to predict patient sex and age. Sex was encoded as a binary category. The corresponding loss function for model training was binary cross-entropy. The loss function for age prediction was the MSE in years between the true and the predicted age. The sigmoid activation function in the age prediction model was replaced by a rectified linear unit (ReLU). Validation was carried out on a dedicated part of the CheXpert training data. The models achieved a validation AUC of 0.97 on sex classification and a mean absolute error (MAE) of 6.0 on age prediction. We applied the auxiliary classification on reconstructed images from clients that hold subsets of the original CheXpert validation data, thus ensuring that the model was only used for inference on images with which it has not been trained with.

3.4. Differential Privacy

Dwork [39] originally proposed the notion of differential privacy in the context of database systems. Differential privacy guarantees that the amount of information revealed about any individual record during a query remains unchanged regardless of whether the record is included in the database at the time of the query or not. Put differently, the probability of receiving a specific output from a query on a database should be almost the same when an individual record is part of the database or not. *Almost* means that the probabilities do not differ by more than a specific factor, which is captured by the *privacy* *budget* or *privacy loss* ε. In the context of machine learning, we regard model training as a function of a dataset equivalent to a query that runs on a database. Intuitively, differential privacy applied to machine learning means that training a model on a dataset should likely result in the same model that would be obtained when removing a single sample from the dataset.

The formal definition of (ε, δ)-differential privacy is as follows:

$$Pr[\mathcal{M}(x) \in \mathcal{S}] \leq \exp(\varepsilon) \cdot Pr[\mathcal{M}(y) \in \mathcal{S}] + \delta, \tag{5}$$

where $\mathcal{M}(x)$ is the (randomized) query or function, x and y are parallel databases that differ in at most one entry, $\mathcal{S} \subseteq Range(\mathcal{M})$, and δ is a small term relaxing the guarantee, usually interpreted as the probability that it fails. A randomized mechanism $\mathcal{M}(x)$ can be obtained by adding noise to the original function drawn from a statistical random distribution, e.g., the Laplacian or the Gaussian distribution. The amount of noise necessary to achieve (ε, δ)-differential privacy is scaled to the ℓ_2-*sensitivity* of the function, which is the maximum distance between the outputs of a function run on two parallel databases. Differential privacy has two important qualities important to its application to machine learning. The output of a differentially private random mechanism remains differentially private during the application of another data-independent function (closure under post-processing). The privacy loss can be analyzed cumulatively over several applications of a mechanism on the same database (composability). We use the variant of Rényi differential privacy, based on the Rényi divergence, in combination with a Gaussian noise mechanism that allows for a tighter estimate of the privacy loss over composite mechanisms than (ε, δ)-differential privacy [40].

Differentially-private stochastic gradient descent (DP-SGD) is commonly deployed for integrating differential privacy into model training [7]. DP-SGD adds two main steps to the SGD algorithm:

1. Bounding the function's sensitivity by clipping per-sample gradient ℓ_2-norms to a clipping value C.
2. Adding Gaussian noise to the gradient, scaled to the sensitivity enforced by Step 1.

We applied DP-SGD locally during training at the clients' sites. Private training was limited to at most ten communication rounds. A privacy accountant tracked the ε-guarantees for a specified list of orders α of the Rényi divergence over communication rounds. This yields the optimal (α, ε)-pair at the end, where ε is the lowest bound on the privacy loss in combination with the respective α. Because the differentially private mechanism is closed under post-processing, aggregation of private model parameters yields a private global model that does not incur a larger privacy loss on individual clients' data than that upper bounded by local DP-SGD.

To investigate the relationship between privacy and model performance for our use case, we limited the privacy budget to $\varepsilon \in \{1, 3, 6, 10\}$. We tracked the α values in $[1.1, 10.9]$ in steps of 0.1, and the values in $[12, 63]$ in steps of 1.

If δ is equal to or greater than the inverse of the size of the dataset, it would allow for leakage of a whole record or data sample without violation of the privacy constraint [58]. As this is unacceptable, δ should be smaller than the inverse of the dataset size [8]. Because the size of each individual client's dataset varies, we determined δ as follows:

$$\delta_k = \min(\frac{1}{\|x_k\|_1} \cdot 0.9, 10^{-2}), \tag{6}$$

where $\|x_k\|_1$ is the number of data samples in the training dataset of client k. Because some clients' datasets are very small, leading to high probabilities for the privacy guarantee to be violated, we defined a minimum value of $\delta = 10^{-2}$.

We bounded the sensitivity of the training function by clipping per-sample gradients. An effective bound is a compromise between excessive clipping, which leads to biased aggregated gradient estimates that do not adequately represent the underlying true gradient values, and a loose clipping bound that forces addition of an exaggerated amount of noise to the gradients. We employed global norm clipping, i.e., gradients were clipped uniformly over the course of training. Abadi et al. propose to use the median of unclipped gradient ℓ_2 norms [7]. We could not obtain unclipped gradient norms directly because the clients' datasets were not considered available for non-private training. As a solution, we ran a few epochs of non-private, centralized training on an auxiliary chest X-ray dataset with the same training parameters as in our federated learning scenario [8]. We used the original Mendeley test set, which was not part of any of the clients' datasets. We randomly picked 5% of the dataset as validation and test sets to validate the training procedure. Because centralized training on the Mendeley test set converged quickly, we tracked the medians over the first three epochs. We obtained median gradient norms of 0.42 (DenseNet121) and 0.62 (ResNet50) for models with frozen batch normalization layers, and 1.24 (DenseNet121) and 0.72 (ResNet50) for models with all layers frozen but the final layer.

3.5. Implementation

The implementation of all experiments is based on PyTorch (https://pytorch.org/, last accessed 3 December 2021) version 1.9. It is available at https://github.com/linev8k/cxr-fl-privacy (last accessed 8 July 2022). For differentially private model training, we used Opacus (https://opacus.ai/, last accessed 28 November 2021) version 0.14.0. Our privacy attacks follow the implementation published by Geiping et al. (https://github.com/JonasGeiping/invertinggradients, last accessed 3 December 2021) Our modified version is available at https://github.com/linev8k/invert-gradients-cxr (last accessed 8 July 2022).

4. Results

This section follows a similar structure as the previous one. We first show the results of the federated learning baseline. Then, we go over the evaluation of the reconstruction attack on the system, where we analyze different factors that impact attack success. Finally, we assess the impact of differential privacy, first on the federated learning effectiveness, and then on the reconstruction attack.

4.1. Federated Learning Baseline

We trained both models on the binary classification of the "No Finding" label. The best global models achieved an AUC value of 0.935 (DenseNet121) and 0.938 (ResNet50). The average AUC was larger on clients holding Mendeley data (0.96 for DenseNet121 and 0.95 for ResNet50) compared to clients with CheXpert data (0.85 for DenseNet121 and 0.87 for ResNet50). These results confirm the ability of deep learning models to reach a high classification performance on the Mendeley dataset [8,28], even when trained in a heterogeneous setting.

Given the implications of different layer freezing techniques for privacy, we compared the outcomes of full model training (no layer freezing), freezing batch normalization layers, and freezing all layers but the final classification layer (Table 3).

For both models, the performance after full model training and training with frozen batch normalization layers was similar with a maximum difference in AUC of 0.022 on the test sets. We conclude that freezing batch normalization layers did not impede model training in our setting. In contrast, rendering all layers untrainable except for the final layer significantly decreased performance. This confirms outcomes from previous work where this transfer learning technique was found to be inferior to including more layers in training updates [51].

Table 3. Mean AUCs of the best global DenseNet121 and ResNet50 models, evaluated on the clients' test sets. *Batch norm.* refers to freezing of batch normalization layers, *All but last* to freezing all parameters except for the final classification layer. Training with frozen batch normalization layers delivered similar results to full model training.

Model	**AUC**		
	No Freezing	**Batch Norm.**	**All but Last**
DenseNet121	0.947	0.935	0.714
ResNet50	0.916	0.938	0.813

4.2. Reconstruction Attack

We attacked local models of arbitrary clients with varying layer freezing techniques, attack time points and batch sizes.

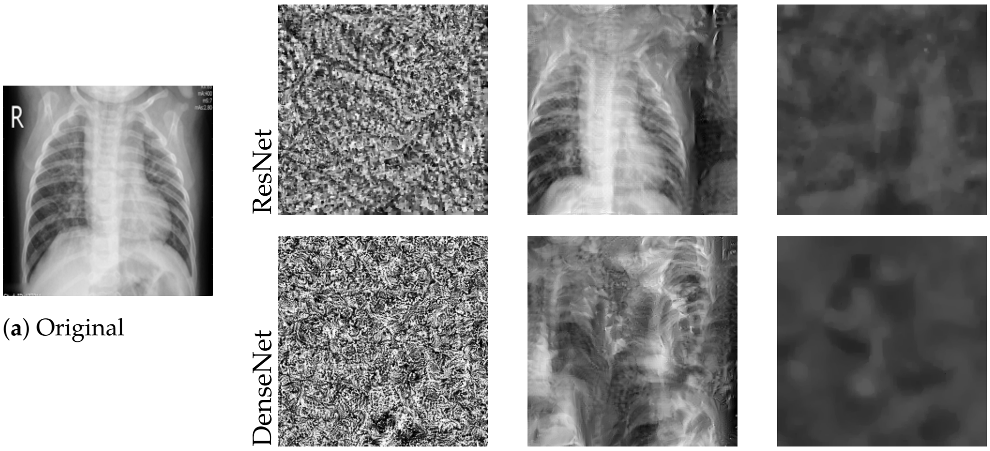
4.2.1. Impact of Layer Freezing

We applied the reconstruction attack to a single client's local model with a batch size of one during the first communication round. Table 4 reports the mean PSNR and sample standard deviation over three trials per experiment. Figure 2 shows the reconstructed images of the best attack trials. The attack was only successful in the case of batch normalization layer freezing, indicated by larger mean PSNR values of 12.29 (ResNet50) and 10.98 (DenseNet121). Training the full model as well as fine-tuning only the output layer prevented the recovery of any useful image features in this setting. We further observed that the DenseNet121 seems to be more robust to leakage from gradients in this example, although the ResNet50 is the larger architecture in terms of parameter count, containing more than three times as many trainable parameters as the DenseNet121.

Table 4. Impact of layer freezing on the attack success during early training. We report the mean PSNR and sample standard deviation (STD) over all images obtained from three attack trials per setting. The batch size is kept constant at one. The attack was only successful on models with frozen batch normalization layers.

Model	**PSNR ± STD**		
	None	**Batch Norm.**	**All but Last**
ResNet50	9.73 ± 0.09	12.29 ± 0.71	8.50 ± 0.12
DenseNet121	8.16 ± 0.03	10.98 ± 0.18	8.07 ± 0.03

The results highlight that shared model updates with partial layer freezing are practically relevant targets for privacy violation. Cases of attack failure, however, do not provide a formal privacy guarantee. Other factors such as the privacy-breaking properties of active batch normalization layers in the case of full model training need to be considered for a comprehensive assessment of model privacy.



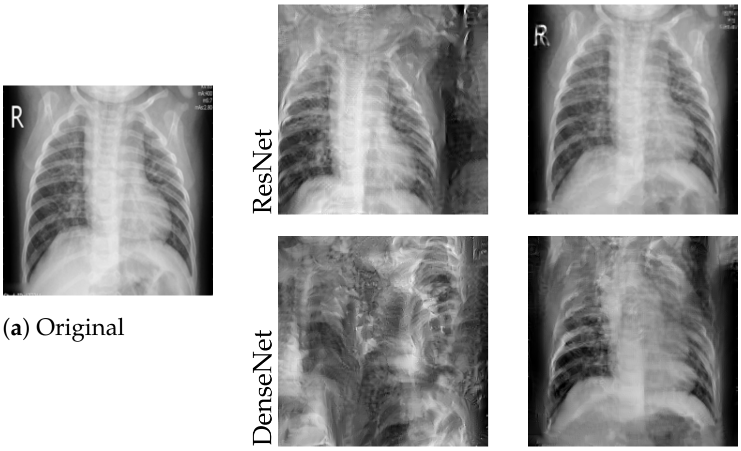
(**a**) Original

(**b**) None (**c**) Batch norm. (**d**) All but last

Figure 2. Best reconstructed images with varying layer freezing techniques. We attacked the locally trained model from a client holding a single Mendeley image (**a**). *None* (**b**) refers to full model training, *Batch norm.* (**c**) to freezing batch normalization layers, and *All but last* (**d**) to only training the output layer.

4.2.2. Impact of Training Stage

We applied the attack during the initial communication round and after four rounds of training. We refer to the settings as an attack in *early* and *late* training stages, respectively. Figure 3 compares the images obtained from early and late attacks. The late attack was significantly more successful on both ResNet50 (mean PSNR 18.42 ± 5.25) and DenseNet121 (mean PSNR 11.7 ± 2.23). At the same time, the variation between trials was greater for the late attack.



(**a**) Original

(**b**) Early training (**c**) Late training

Figure 3. Best reconstructions of the image shown in (**a**) after the first (**b**) and after the fourth (**c**) communication round. Reconstruction quality increased significantly after several rounds of training.

The observation that the attack was more successful as training progressed does not confirm previous evidence, which suggests that reconstruction is less successful from pre-trained models [4] and during later training stages [8]. Attack success has been associated with the magnitude of the gradients' ℓ_2-norms, which are usually largest at the beginning when the model starts training on previously unseen data [8]. In Figure 4, we investigate how the ℓ_2-norms of our models' gradients changed as training progressed. We show the exemplary case of the pre-trained DenseNet121. Results were similar for the ResNet50, for which we refer to Appendix A. For each layer of every client's local model, we tracked the median ℓ_2-norm during training. We display the per-layer mean values of all tracked medians over the local models. For both model architectures, the norms were greater during the first round of training than in the following iterations. Subsequent changes are more subtle and lack continuity. We validated that the attacked client's model did not pose an exception to this behavior.

Since our attacks were more successful during late training and we observed overall smaller gradient norms as training progressed, we could not associate larger gradient norms with increased attack success.

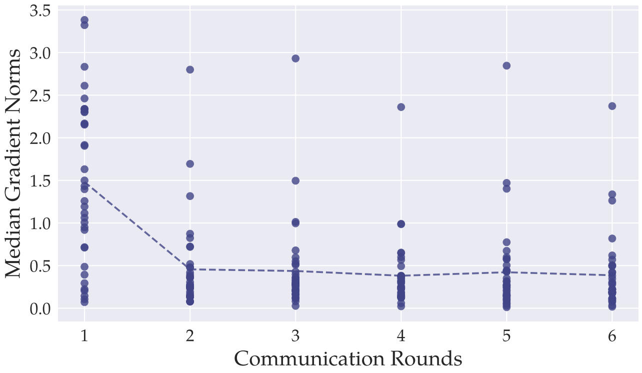


Figure 4. For each model layer of the DenseNet121, we tracked the median ℓ_2-norms during training of every local model. Each dot represents the mean of one layer-median over all local models. The dashed line depicts the overall mean of all per-layer ℓ_2-norm medians. Most layers' ℓ_2-norms were greater during the first round of training than in later stages. In our experiments, image reconstruction was better on models from later rounds, suggesting that the magnitude of gradient ℓ_2-norms is not a primary indicator for attack success.

4.2.3. Impact of Batch Size

We investigated the impact of the training batch size in the setting where the attack was most successful, i.e., on models trained with frozen batch normalization layers attacked during late training. We attacked clients for which the considered batch size was equal to the available number of training images. The setting is equivalent to clients with larger datasets sharing model updates after every processed batch. A batch size of ten reduced attack success as the mean PSNR values over the batch decreased to 9.01 (ResNet50) and 8.15 (DenseNet121) (Table 5). While the quality of the reconstructions varied for individual images within a batch, at least one image out of each batch became recognizable. We note that the order of images in a batch may not be preserved in the reconstruction of larger batches, preventing a direct comparison between original and reconstructed data points. To assign a reconstructed image to its original for evaluation, we first obtained the PSNR of each original image with each reconstructed image. We then determined the first original-reconstruction pair as the one with the largest PSNR value. The next best pair was determined considering the PSNR values between the remaining original and reconstructed images. We iterated the procedure until all images have been assigned.

Figure 5 shows the best-reconstructed images out of each batch, demonstrating that all considered batch sizes permit severe privacy breaches on individual data samples.

Table 5. Impact of batch size on attack success during late training. We report the mean PSNR and sample standard deviation (STD) over all images obtained from three attack trials per setting. Models were trained with frozen batch normalization layers (cf. Figure 2). Attack success deteriorated with a batch size of ten, but not significantly with smaller batch sizes.

Model	**PSNR ± STD**			
	1	**2**	**4**	**10**
ResNet50	18.42 ± 5.25	11.79 ± 1.2	14.60 ± 2.86	9.01 ± 3.13
DenseNet121	11.7 ± 2.23	12.47 ± 3.24	12.67 ± 2.24	8.15 ± 2.82

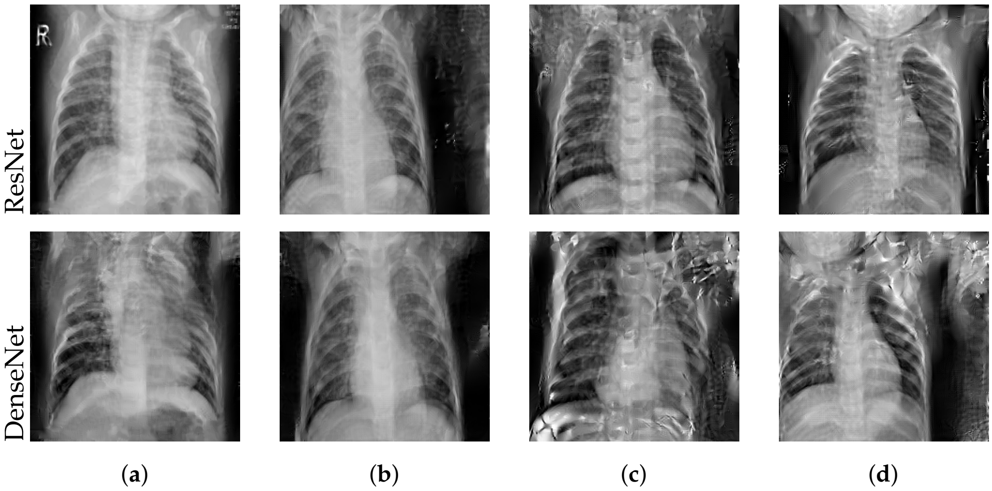


Figure 5. Best reconstructed images out of each batch of size (**a**) 1, (**b**) 2, (**c**) 4, and (**d**) 10. While other samples from those batches were not affected by the attack, the privacy of these examples' original X-rays has been severely breached, regardless of the batch size.

4.2.4. Inference of Demographic Properties

Finally, to investigate the leakage of sensitive patient information from reconstructed images, we applied the attack to 15 clients holding CheXpert validation data subsets. We included five clients each, holding one, two, and four training images, yielding 35 images in total. The setting was the same as for the attacks on Mendeley clients. We attacked models trained with frozen batch normalization layers during late training. Then, we predicted the patients' age and sex from the original X-rays and from the reconstructed images using auxiliary models to demonstrate that the images leak sensitive information.

Table 6 summarizes the auxiliary model predictions. The low baseline performance of the auxiliary models on original images compared to the classifier validation estimate is probably due to the small sample size of 35 images. Superior results on images reconstructed from the ResNet50 in the case of sex prediction suggest an increased susceptibility to privacy violation of this architecture compared to the DenseNet121.

Table 6. Performance of the auxiliary models for predicting patient sex and age from X-ray images. We compare the classification/regression of original images, and images reconstructed from local ResNet50 and DenseNet121 models. All attacked clients provided 35 images in total. Metrics reported are AUC for sex prediction and the mean absolute error (MAE) in years for age regression.

Attacked Model	Sex (AUC)	Age (MAE)
-	0.71	11.51
ResNet50	0.69	15.42
DenseNet121	0.56	15.22

4.3. Differentially Private Federated Learning

As a countermeasure to the reconstruction attack, we evaluate the introduction of local differential privacy into our training process. The following sections detail the implications that come with that added protection.

4.3.1. Model Performance

Table 7 reports the models' performance with privacy budgets $\varepsilon \in \{1, 3, 6, 10\}$. We include the non-private baseline performance for comparison. Batch normalization layer parameters were not updated during model training. We report the exact privacy budget spent by each local model as optimal (α, ε)-pairs in Appendix C.

Table 7. Mean AUC of global DenseNet121 and ResNet50 models, evaluated on the clients' test sets for non-private training and private training with varying ε values. Stronger privacy guarantees decreased model performance.

Model	**AUC**				
	-	$\boldsymbol{\varepsilon = 10}$	$\boldsymbol{\varepsilon = 6}$	$\boldsymbol{\varepsilon = 3}$	$\boldsymbol{\varepsilon = 1}$
DenseNet121	0.935	0.925	0.937	0.854	0.800
ResNet50	0.938	0.861	0.764	0.711	0.622

We compare the utility-privacy trade-off between the two model architectures in Figure 6. The DenseNet121 performed better than ResNet50 for all considered privacy budgets. As expected, a stronger privacy guarantee claimed a higher cost in accuracy for both models. The degradation was more pronounced in the ResNet50 with an AUC difference of 0.24 between $\varepsilon = 10$ and $\varepsilon = 1$. The private DenseNet121 performed equally well compared to its non-private counterpart for both $\varepsilon = 10$ and $\varepsilon = 6$, suggesting that a increasing the privacy budget beyond $\varepsilon = 6$ does not benefit model performance. For $\varepsilon = 6$, the DenseNet121 achieved an AUC of 0.937, the ResNet50 only 0.764.

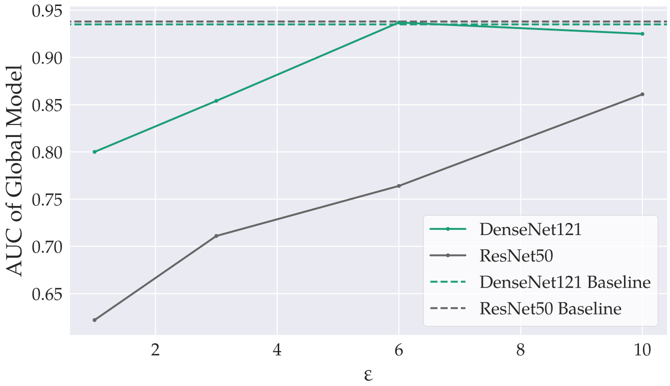


Figure 6. Model performance, evaluated on the clients' test sets in dependence on the privacy budget ε. Baselines mark the peak performance of global non-private models. Stronger privacy guarantees degraded model accuracy.

We expect that model evaluation in our setting with imbalanced data distribution tends to be unreliable on clients with less data. Incidental good results on those clients may bias the global model's performance estimate. To provide a more meaningful assessment of the model performance under privacy conditions, we investigated the performance of the best global private DenseNet121 model on individual clients compared to the best non-private model. We visualize the comparison for $\varepsilon = 6$ in Figure 7. We provide the figures for other considered ε-values in Appendix B. Private training demanded a systematic cost in performance for clients holding large amounts of CheXpert data. AUC values on those clients' datasets decreased by 0.03 ($\varepsilon = 10$ and $\varepsilon = 6$) and 0.06 ($\varepsilon = 3$) on average from non-private to private training.

We conclude that the impact of private training on model accuracy, also at moderate privacy budgets, needs to be carefully assessed on the client level. Further potential weak points of the resulting model, such as performance on underrepresented patient subgroups, require additional consideration.

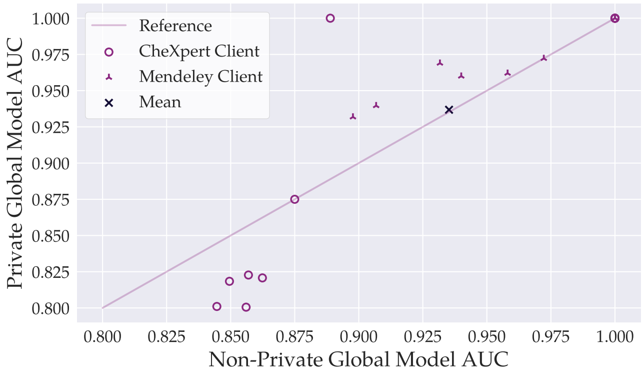


Figure 7. Comparison of per-client AUC values achieved by the best global DenseNet121 model between private ($\varepsilon = 6$) and non-private training. The bottom left circle markers belong to CheXpert clients with large datasets. Privacy demanded a higher cost in model accuracy on CheXpert clients. The figure does not consider AUC values below 0.79. Markers may overlap.

4.3.2. Additional Training Techniques

We evaluate the effect of additional training techniques on model performance: Training only the final layer (*All but last* layer freezing), client subsampling, and employment of layer-wise gradient clipping. All experiments were carried out with a privacy budget of $\varepsilon = 10$. We found that none of the techniques introduced an advantage for private model training. When restricting training to the final layer, the performance of the DenseNet121 decreased significantly compared to *Batch norm.* freezing (AUC 0.707 vs. 0.925) and that of the ResNet50 remained similar (AUC 0.871 vs. 0.861).

In a separate experiment, we introduced a client subsampling procedure where the maximum number of global communication rounds was set to ten and the maximum number of rounds that each client can be selected to five. The fraction of clients chosen each round was 0.3, resulting in eleven clients selected per round. This way, less of the available privacy budget was effectively spent during the clients' local training because each client participated in fewer training rounds in total. However, there was no improvement in model performance. The DenseNet121 and ResNet50 achieved AUC values of 0.836 and 0.819, respectively. A potential explanation is that clients with small datasets got selected frequently during subsampling, but could not contribute as effectively to the global model as clients with larger datasets. Model accuracy degraded more heavily after a few rounds during the subsampling experiment, indicating stronger local overfitting which was amplified by the lower number of contributing clients.

Finally, instead of uniformly clipping the norm of each gradient value, we specified an individual clipping bound for each model layer. We utilized the per-layer median gradient norms from the auxiliary training experiment on the Mendeley test set (Section 3.4). The models' AUC values converged to 0.5, indicating that model training failed for our use case when employing layer-wise gradient clipping. The variation between individual clipping values may be too large, preventing the model parameters to retain any information that is usable in combination with other layers' parameters.

4.3.3. Vulnerability to Reconstruction Attack

We attempted to reconstruct the training image from the local model shared by a Mendeley client during private training. We performed the attack on models with frozen batch normalization layers during late training. Table 8 compares the mean PSNR over three trials between non-private and private training. The PSNR on all images from private models was significantly smaller than in the non-private setting. Figure 8 confirms that the reconstructed images from both model architectures did not leak any visual parts of the training images. Differentially private training under all considered privacy budgets therefore successfully prevented the attack.

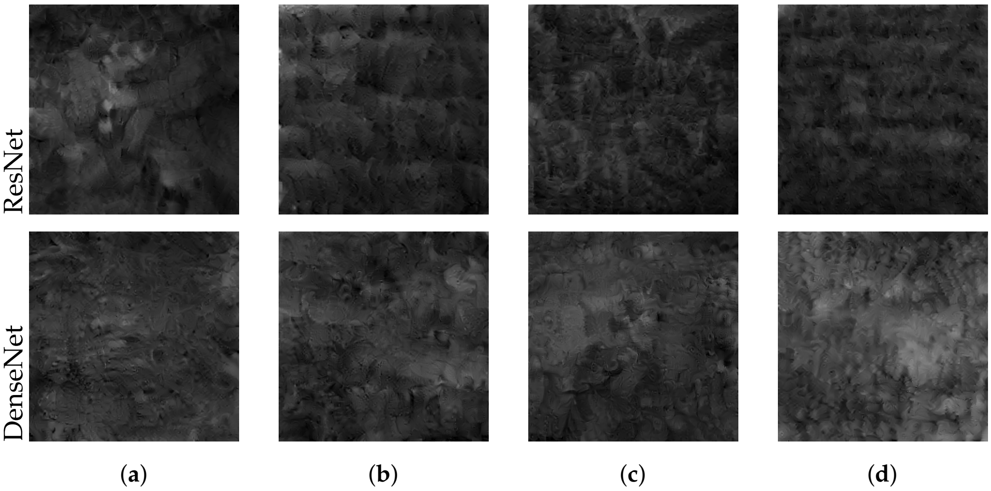


Figure 8. Best reconstructed images under different privacy constraints with (**a**) $\varepsilon = 10$, (**b**) $\varepsilon = 6$, (**c**) $\varepsilon = 3$, (**d**) $\varepsilon = 1$. Private training successfully prevents the leakage of any visible features.

Table 8. Mean PSNR over three attack trials on non-private and private local models from a Mendeley client holding one training image. The attack failed for all considered ε values.

Model	**PSNR ± STD**				
	-	$\boldsymbol{\varepsilon = 10}$	$\boldsymbol{\varepsilon = 6}$	$\boldsymbol{\varepsilon = 3}$	$\boldsymbol{\varepsilon = 1}$
DenseNet121	10.98 ± 0.18	6.58 ± 0.07	6.50 ± 0.08	6.46 ± 0.07	6.41 ± 0.03
ResNet50	12.29 ± 0.71	7.22 ± 0.04	7.13 ± 0.16	7.12 ± 0.17	8.49 ± 0.32

To validate that no sensitive information was leaked, we applied the auxiliary models (first introduced in Section 4.2.4) to predict patient age and sex from images reconstructed from private models. Table 9 compares their performance on original and recovered images in private and non-private settings. We attacked the model with the weakest privacy guarantee of $\varepsilon = 10$. The AUC values of 0.49 and 0.47 on sex prediction indicate that the classifier's performance was equivalent to random label assignment in the private setting. The age predictions deviated around 19 years on average from the true patients' age. Differentially private model training prevented both auxiliary models to predict usable information about the patients' demographic properties.

Table 9. Performance of the auxiliary models for predicting patient sex and age from X-ray images. We compare the predictions on original images, and images reconstructed from local ResNet50 and DenseNet121 models in the non-private and private setting with $\varepsilon = 10$. Images reconstructed from private models leaked no usable information about the selected properties.

Attacked Model	ε	**Sex (AUC)**	**Age (MAE)**
-	-	0.71	11.51
ResNet50	-	0.69	15.42
	10	0.49	19.23
DenseNet121	-	0.56	15.22
	10	0.47	18.82

We conclude, that in our federated learning setting, differential privacy is an effective countermeasure against sample reconstruction from gradients, and no sensitive information could be inferred from the reconstructed images.

5. Discussion

In our federated learning setup, the effectiveness of model aggregation is limited by data heterogeneity and imbalance. The federated averaging algorithm weights the local model updates with respect to the clients' dataset size in relation to the overall amount of available data [48]. This led to a strong emphasis on model updates from clients with large CheXpert subsets in our case, while updates from clients with fewer images contributed less to model aggregation. One option to mitigate data imbalance is to aggregate models after a specified number of batches instead of local epochs [8,59]. However, sharing intermediate models more frequently will increase the susceptibility to reconstruction attacks since the updates are obtained on small batches rather than the client's whole dataset. Improving the aggregation process under consideration of privacy costs is left for future work.

The applied attack has shown that the two considered deep machine learning models are susceptible to reconstruction of sensitive data from gradients. Most notably, and contrary to previous work, we found that the attack was more successful in later training stages and for pre-trained models. Our privacy evaluation framework is limited by the choice of the DLG attack as a qualitative measure for model vulnerability. Even though we found reconstruction not successful under certain conditions, including full model training, restricting training to the final layer, and attacking the DenseNet121 at an early training stage, it cannot be assumed that model training would be privacy-preserving in these cases. Minor modifications of the attack scheme may improve attack success even in supposedly safe settings. Moreover, reconstruction attacks are only one example among a range of deliberate privacy breaches that neural networks are vulnerable to. Extending our privacy evaluation framework to include other privacy threats, e.g., property inference without data reconstruction, will provide further insights into potential vulnerabilities of the federated learning paradigm. Since the main limitation of DLG is its restriction to small datasets, it will be particularly valuable to capture the consequences of privacy breaches for clients with large amounts of data. From a security perspective, demonstrating that these attacks are practically feasible, albeit under limited circumstances, is sufficient for considering the machine learning process vulnerable to privacy violation. Countermeasures must constantly be re-evaluated for their effectiveness as a better understanding of privacy threats evolves.

Our privacy evaluation was further constrained to a limited choice of privacy budgets. While choosing $\varepsilon = 6$ delivered the best utility-privacy trade-off for our use case, which is in line with previous work [8], it may not be the optimal lower bound. We specifically suggest empirically examining choices of ε in the range $[3, 6]$ to potentially improve upon our results in future work. We also note that while all considered privacy constraints prevented the success of reconstruction attacks, smaller ε values still formally provide stronger privacy guarantees that offer protection against threats beyond the limited case of the reconstruction attack.

A key implication of our results is that the DenseNet121 architecture proved more robust against private model training with regard to performance than the ResNet50. This observation is potentially related to the greater ability of the DenseNet121 to withstand reconstruction attacks. Although the model contains overall fewer parameters than the ResNet50, its dense structure may, to a certain degree, offer a natural defense against reconstruction from trained parameters as well as perturbation of parameter updates during private training. This outcome suggests substantial differences in the suitability of individual model types for privacy-preserving machine learning, which requires further validation.

In the medical context, fairness is crucial for the safe deployment of machine learning algorithms. Rare diseases or conditions must be reliably detected despite the restricted availability of representative data. Furthermore, a model should perform with equal accuracy on all patient subgroups. We uncovered performance differences between individual clients' data in our federated learning baseline, revealing that the classification produced better results on Mendeley than on CheXpert data. This potentially reflects the ability of deep learning models to recognize pneumonia as an abnormal finding particularly well since the pathologic X-rays in the Mendeley dataset only include cases of pneumonia. More thorough investigations are required to reveal other potential biases, e.g., with respect to patient subgroups. It is further known that underrepresented classes and population subgroups are potentially affected more strongly by model performance degradation when applying differential privacy [60]. Because model performance evaluation on Mendeley clients was less reliable due to smaller amounts of data, it remains an open question how exactly these clients were affected by the integration of differential privacy. For practical applications, it is mandatory to thoroughly investigate how privacy mechanisms affect the model's performance on different types of data to identify a potential underlying bias.

We did not consider the practical implementation of federated learning between different institutions with regard to communication time, required infrastructure, costs and validation of correct computational execution. The focus of our paper lies in analyzing the threat of data reconstruction and the effectiveness of differential privacy against it. While simulated use cases like ours are vital to prepare for leveraging differential privacy in real-world cases where sensitive data is involved, further case studies are required to investigate aspects of practicability for privacy-preserving federated learning on a large scale.

6. Conclusions and Future Work

We simulated a collaborative machine learning use case in which 36 institutions provide their diverse chest X-ray data collections for the development of a classification model. Two main concerns in this scenario are the physical separation of the data sources and the privacy of patients to whom the data belongs. We employed the paradigm of federated learning as a solution for machine learning on dispersed data. Throughout our experiments, we compared two large network architectures: DenseNet121 and ResNet50. Extending previous evidence, we demonstrated that individual X-rays can be reconstructed from shared model updates within the federated learning setting from those networks using the DLG attack. It is especially successful during later training stages.

As a step towards privacy-preserving distributed learning, we integrated Rényi differential privacy with a Gaussian noise mechanism into the federated learning process. The DenseNet121 achieved the best utility-privacy trade-off with a mean AUC of 0.937 for $\varepsilon = 6$, where we identified an expected cost in accuracy of 0.03 in terms of the AUC on CheXpert clients' data compared to the non-private baseline. The results suggest that $\varepsilon \in [3, 6]$ are suitable candidates for private model training depending on the specific demands on model privacy and performance for the respective application. Overall, we found the DenseNet121 model superior to ResNet50 with regard to private model training for all considered ε values.

The adverse impact of differential privacy on model performance must be carefully considered, particularly for medical use cases. Our results endorse that differentially private federated learning is feasible at a small cost in model accuracy for the classification of heterogeneous chest X-ray data. As real-world medical use cases become more complex in practice, future work may elaborate on the potential of differentially private federated learning for multi-label X-ray classification where heterogeneous data from a broader range of sources is effectively integrated under consideration of an improved bound on the privacy budget. We identified the DenseNet121 as a robust model architecture suitable for differentially private training. Further comparison with other neural network architectures may reveal key indicators for the suitability of different model types and provide guidance in the choice of models for privacy-preserving machine learning. We further suggest to extend our evaluation framework in future work to consider the vulnerability to other types of privacy breaches, enabling a comprehensive qualitative assessment of model privacy. Finally, other variants of differential privacy, e.g., Gaussian differential privacy [61], may offer suitable alternatives to the application of Rényi differential privacy providing yet tighter bounds on the privacy loss.

Author Contributions: Conceptualization, J.Z., B.P., H.S., A.S. and B.A.; methodology, J.Z.; software, J.Z.; validation, J.Z. and B.P.; formal analysis, J.Z.; investigation, J.Z.; resources, J.Z., B.P. and B.A.; data curation, J.Z.; writing—original draft preparation, J.Z.; writing—review and editing, J.Z., B.P., H.S., A.S. and B.A.; visualization, J.Z.; supervision, B.P., H.S., A.S. and B.A.; project administration, B.A., B.P. and J.Z.; funding acquisition, B.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The CheXpert dataset [24] is a public chest X-ray dataset available at https://stanfordmlgroup.github.io/competitions/chexpert/ (last accessed 8 July 2022). The Mendeley dataset version 3 can be found at https://data.mendeley.com/datasets/rscbjbr9sj/3 (last accessed 8 July 2022).

Acknowledgments: Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)—Projektnummer 491466077.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AUC	Area under the receiver operating characteristics curve
DLG	Deep Leakage from Gradients
DP-SGD	Differentially private stochastic gradient descent
MAE	Mean absolute error
MSE	Mean squared error
PSNR	Peak signal-to-noise ratio
ReLU	Rectified linear unit
ROC	Receiver operating characteristic
SGD	Stochastic gradient descent

Appendix A. Gradient ℓ_2-Norms

We investigated the models' gradients' ℓ_2-norms during training to assess how they correlate with attack success. Figure A1 shows how the norms change in the ResNet50. The norms were greater during the first round of training than in the following iterations. In our experiments, image reconstruction was better on models from later rounds, suggesting that the magnitude of gradient ℓ_2-norms is not a primary indicator for attack success.

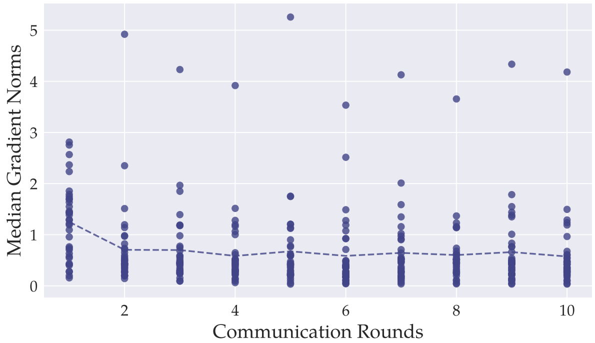
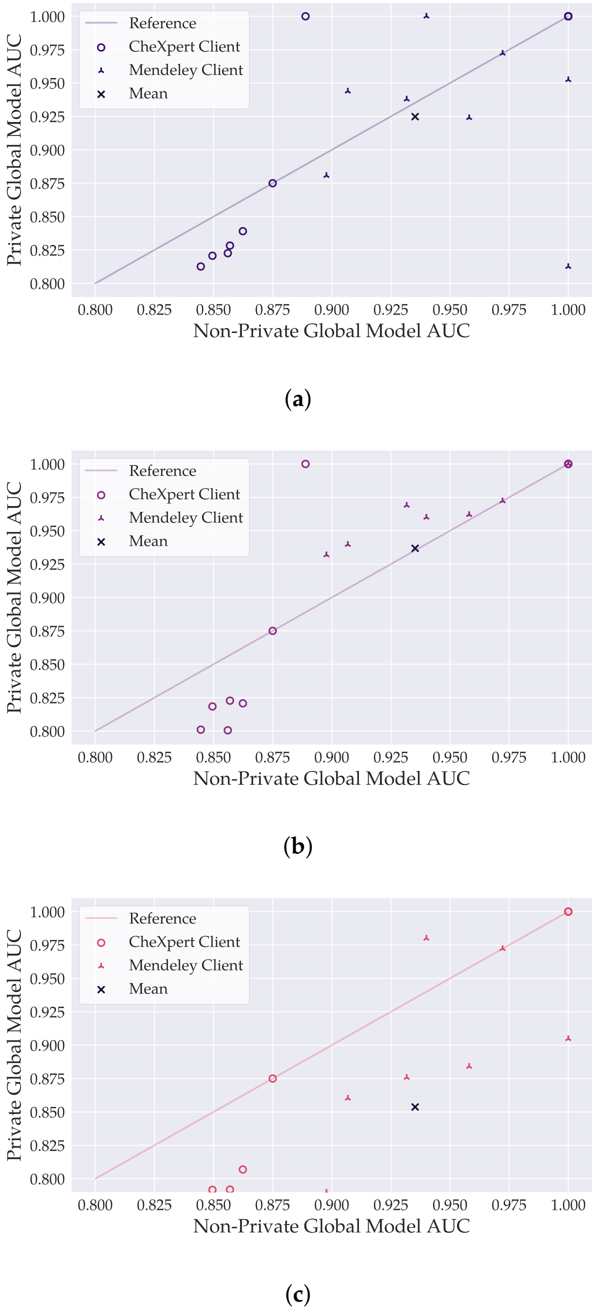


Figure A1. For each model layer in the ResNet50, we tracked the median ℓ_2-norms during training of every local model. Each dot represents the mean of one layer-median over all local models. The dashed line depicts the overall mean of all per-layer ℓ_2-norm medians. Most layers' ℓ_2-norms were greater during the first round of training than in later stages.

Appendix B. Client-Level AUC of Private DenseNet121 Models

We investigated the performance of the best global private DenseNet121 model on individual clients compared to the best non-private model. We visualize the comparison for $\varepsilon \in 3, 6, 10$ in Figure A2. Because training was unsuccessful for $\varepsilon = 1$, we do not evaluate model performance for this case in detail. Private training demanded a systematic cost in performance for clients holding large amounts of CheXpert data for all considered privacy budgets. Because clients with Mendeley data hold fewer images, the results on individual test sets of those clients was subject to greater variation.



(**a**)

(**b**)

(**c**)

Figure A2. Comparison of per-client AUC values achieved by the best global DenseNet121 models between non-private and private training with (**a**) $\varepsilon = 10$, (**b**) $\varepsilon = 6$, (**c**) $\varepsilon = 3$. The bottom left circle markers belong to CheXpert clients with large datasets. Privacy demanded a higher cost in model accuracy on CheXpert clients. The figures do not consider AUC values below 0.79. Markers may overlap.

Appendix C. List of the Private Models' (α, ε)-Pairs

Table A1 reports the exact privacy budget spent by each local model trained with differential privacy as optimal (α, ε)-pairs.

Table A1. Optimal (α, ε)-pairs under fixed privacy budgets for each client's local model of our private training baseline. True ε values may deviate slightly from the defined privacy budget as the noise multiplier is estimated before model training in order to meet the privacy constraint. Clients 0 to 13 hold Mendeley data subsets, clients 14 to 18 the CheXpert training data, and clients 19 to 35 parts of the CheXpert validation data.

Client	**No. Images**	**(α, ε)**			
		$\varepsilon = 10$	**$\varepsilon = 6$**	**$\varepsilon = 3$**	**$\varepsilon = 1$**
0	350	$(1.9, 10.49)$	$(2.4, 6.12)$	$(3.7, 2.87)$	$(7.9, 1.01)$
1	350	$(1.9, 10.49)$	$(2.4, 6.12)$	$(3.7, 2.87)$	$(7.9, 1.01)$
2	140	$(1.8, 10.26)$	$(2.3, 6.07)$	$(3.4, 2.84)$	$(6.6, 0.99)$
3	140	$(1.8, 10.26)$	$(2.3, 6.07)$	$(3.4, 2.84)$	$(6.6, 0.99)$
4	70	$(1.8, 9.83)$	$(2.5, 6.03)$	$(3.2, 2.83)$	$(6.0, 1.0)$
5	70	$(1.8, 9.83)$	$(2.2, 6.03)$	$(3.2, 2.83)$	$(6.0, 1.0)$
6	10	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
7	10	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
8	4	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
9	4	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
10	2	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
11	2	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
12	1	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
13	1	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
14	27,325	$(2.3, 9.15)$	$(2.8, 6.29)$	$(4.2, 2.91)$	$(9.0, 0.99)$
15	26,463	$(2.3, 9.22)$	$(2.7, 6.34)$	$(4.2, 2.93)$	$(9.0, 0.99)$
16	27,259	$(2.3, 9.15)$	$(2.8, 6.29)$	$(4.2, 2.91)$	$(9.0, 0.99)$
17	26,875	$(2.3, 9.18)$	$(2.7, 6.32)$	$(4.2, 2.92)$	$(9.0, 0.99)$
18	26,344	$(2.3, 9.23)$	$(2.7, 6.34)$	$(4.2, 2.93)$	$(9.0, 0.99)$
19	1	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
20	1	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
21	1	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
22	1	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
23	1	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
24	2	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
25	2	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
26	2	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
27	2	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
28	2	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
29	4	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
30	4	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
31	4	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
32	4	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
33	4	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
34	10	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$
35	10	$(2.1, 10.03)$	$(2.5, 6.0)$	$(3.6, 2.79)$	$(6.2, 1.0)$

References

1. Rieke, N.; Hancox, J.; Li, W.; Milletarì, F.; Roth, H.R.; Albarqouni, S.; Bakas, S.; Galtier, M.N.; Landman, B.A.; Maier-Hein, K.; et al. The Future of Digital Health with Federated Learning. *Npj Digit. Med.* **2020**, *3*, 119. [CrossRef]
2. Kaissis, G.A.; Makowski, M.R.; Rückert, D.; Braren, R.F. Secure, Privacy-Preserving and Federated Machine Learning in Medical Imaging. *Nat. Mach. Intell.* **2020**, *2*, 305–311. [CrossRef]
3. Kairouz, P.; McMahan, H.B. Advances and Open Problems in Federated Learning. *Found. Trends Mach. Learn.* **2021**, *14*. [CrossRef]
4. Geiping, J.; Bauermeister, H.; Dröge, H.; Moeller, M. Inverting Gradients—How Easy Is It to Break Privacy in Federated Learning? *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 16937–16947.

5. Zhu, L.; Liu, Z.; Han, S. Deep Leakage from Gradients. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
6. Naseri, M.; Hayes, J.; De Cristofaro, E. Toward Robustness and Privacy in Federated Learning: Experimenting with Local and Central Differential Privacy. *arXiv* **2020**, arXiv:2009.03561.
7. Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H.B.; Mironov, I.; Talwar, K.; Zhang, L. Deep Learning with Differential Privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 308–318. [CrossRef]
8. Kaissis, G.; Ziller, A.; Passerat-Palmbach, J.; Ryffel, T.; Usynin, D.; Trask, A.; Lima, I.; Mancuso, J.; Jungmann, F.; Steinborn, M.M.; et al. End-to-End Privacy Preserving Deep Learning on Multi-Institutional Medical Imaging. *Nat. Mach. Intell.* **2021**, 3, 473–484. [CrossRef]
9. Li, W.; Milletarì, F.; Xu, D.; Rieke, N.; Hancox, J.; Zhu, W.; Baust, M.; Cheng, Y.; Ourselin, S.; Cardoso, M.J.; et al. *Privacy-Preserving Federated Brain Tumour Segmentation. Machine Learning in Medical Imaging*; Lecture Notes in Computer Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 133–141. [CrossRef]
10. Rajpurkar, P.; Irvin, J.; Zhu, K.; Yang, B.; Mehta, H.; Duan, T.; Ding, D.; Bagul, A.; Langlotz, C.; Shpanskaya, K.; et al. CheXNet: Radiologist-Level Pneumonia Detection on Chest X-rays with Deep Learning. *arXiv* **2017**, arXiv:1711.05225.
11. Feki, I.; Ammar, S.; Kessentini, Y.; Muhammad, K. Federated Learning for COVID-19 Screening from Chest X-ray Images. *Appl. Soft Comput.* **2021**, *106*, 107330. [CrossRef]
12. Rimmer, A. Radiologist Shortage Leaves Patient Care at Risk, Warns Royal College. *BMJ* **2017**, *359*, j4683. [CrossRef]
13. Itri, J.N.; Tappouni, R.R.; McEachern, R.O.; Pesch, A.J.; Patel, S.H. Fundamentals of Diagnostic Error in Imaging. *RadioGraphics* **2018**, *38*, 1845–1865. [CrossRef]
14. Qayyum, A.; Qadir, J.; Bilal, M.; Al-Fuqaha, A. Secure and Robust Machine Learning for Healthcare: A Survey. *IEEE Rev. Biomed. Eng.* **2021**, *14*, 156–180. [CrossRef]
15. Shah, U.; Dave, I.; Malde, J.; Mehta, J.; Kodeboyina, S. Maintaining Privacy in Medical Imaging with Federated Learning, Deep Learning, Differential Privacy, and Encrypted Computation. In Proceedings of the 2021 6th International Conference for Convergence in Technology (I2CT), Maharashtra, India, 2–4 April 2021. [CrossRef]
16. Brisimi, T.S.; Chen, R.; Mela, T.; Olshevsky, A.; Paschalidis, I.C.; Shi, W. Federated Learning of Predictive Models from Federated Electronic Health Records. *Int. J. Med. Inform.* **2018**, *112*, 59–67. [CrossRef]
17. Li, Y.; Jiang, X.; Wang, S.; Xiong, H.; Ohno-Machado, L. VERTIcal Grid lOgistic Regression (VERTIGO). *J. Am. Med. Inform. Assoc.* **2016**, *23*, 570–579. [CrossRef]
18. Chen, Y.; Qin, X.; Wang, J.; Yu, C.; Gao, W. FedHealth: A Federated Transfer Learning Framework for Wearable Healthcare. *IEEE Intell. Syst.* **2020**, *35*, 83–93. [CrossRef]
19. Sheller, M.J.; Reina, G.A.; Edwards, B.; Martin, J.; Bakas, S. Multi-Institutional Deep Learning Modeling Without Sharing Patient Data: A Feasibility Study on Brain Tumor Segmentation. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*; Lecture Notes in Computer Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 92–104. [CrossRef]
20. Lu, M.Y.; Kong, D.; Lipkova, J.; Chen, R.J.; Singh, R.; Williamson, D.F.K.; Chen, T.Y.; Mahmood, F. Federated Learning for Computational Pathology on Gigapixel Whole Slide Images. *Med. Image Anal.* **2022**, *76*, 102298. [CrossRef]
21. Li, X.; Gu, Y.; Dvornek, N.; Staib, L.H.; Ventola, P.; Duncan, J.S. Multi-Site fMRI Analysis Using Privacy-Preserving Federated Learning and Domain Adaptation: ABIDE Results. *Med. Image Anal.* **2020**, *65*, 101765. [CrossRef]
22. Roth, H.R.; Chang, K.; Singh, P.; Neumark, N.; Li, W.; Gupta, V.; Gupta, S.; Qu, L.; Ihsani, A.; Bizzo, B.C.; et al. Federated Learning for Breast Density Classification: A Real-World Implementation. In *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning*; Lecture Notes in Computer Science; Albarqouni, S., Bakas, S., Kamnitsas, K., Cardoso, M.J., Landman, B., Li, W., Milletari, F., Rieke, N., Roth, H., Xu, D., et al., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; Volume 12444, pp. 181–191. [CrossRef]
23. Çallı, E.; Sogancioglu, E.; van Ginneken, B.; van Leeuwen, K.G.; Murphy, K. Deep learning for chest X-ray analysis: A survey. *Med. Image Anal.* **2021**, *72*, 102125. [CrossRef]
24. Irvin, J.; Rajpurkar, P.; Ko, M.; Yu, Y.; Ciurea-Ilcus, S.; Chute, C.; Marklund, H.; Haghgoo, B.; Ball, R.; Shpanskaya, K.; et al. CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison. In Proceedings of the AAAI Conference on Artificial Intelligence, Atlanta, GA, USA, 8–12 October 2019.
25. Kermany, D.; Zhang, K.; Goldbaum, M. Labeled Optical Coherence Tomography (OCT) and Chest X-ray Images for Classification. *Mendeley Data* **2018**, 2.
26. Chakravarty, A.; Kar, A.; Sethuraman, R.; Sheet, D. Federated Learning for Site Aware Chest Radiograph Screening. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 1077–1081. [CrossRef]
27. Nath, V.; Abidin, A.; Genereaux, B.; Younis, K.; Singla, N.; Lakhani, P.; Gentili, A.; Swinburne, N.; Qu, L.; Landman, B.; et al. Empirical Evaluation of Federated Learning for Classification of Chest X-rays. In Proceedings of the Conference on Machine Intelligence in Medical Imaging, Montreal, QC, Canada, 6–8 July 2020.

28. Banerjee, S.; Misra, R.; Prasad, M.; Elmroth, E.; Bhuyan, M.H. Multi-Diseases Classification from Chest-X-ray: A Federated Deep Learning Approach. In *AI 2020: Advances in Artificial Intelligence*; Lecture Notes in Computer Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; Volume 12576, pp. 3–15. [CrossRef]
29. Bressem, K.K.; Adams, L.C.; Erxleben, C.; Hamm, B.; Niehues, S.M.; Vahldiek, J.L. Comparing Different Deep Learning Architectures for Classification of Chest Radiographs. *Sci. Rep.* **2020**, *10*, 13590. [CrossRef]
30. Ke, A.; Ellsworth, W.; Banerjee, O.; Ng, A.Y.; Rajpurkar, P. CheXtransfer: Performance and Parameter Efficiency of ImageNet Models for Chest X-ray Interpretation. In Proceedings of the Conference on Health, Inference, and Learning, Online, 8–10 April 2021; pp. 116–124. [CrossRef]
31. Enthoven, D.; Al-Ars, Z. An Overview of Federated Deep Learning Privacy Attacks and Defensive Strategies. In *Federated Learning Systems*; Studies in Computational Intelligence; Springer International Publishing: Berlin/Heidelberg, Germany, 2021; Volume 965, pp. 173–196.
32. Lyu, L.; Yu, H.; Zhao, J.; Yang, Q. Threats to Federated Learning. In *Federated Learning*; Lecture Notes in Computer Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; Volume 12500, , pp. 3–16. [CrossRef]
33. Zhao, B.; Mopuri, K.R.; Bilen, H. iDLG: Improved Deep Leakage from Gradients. *arXiv* **2020**, arXiv:2001.02610.
34. Wang, Y.; Deng, J.; Guo, D.; Wang, C.; Meng, X.; Liu, H.; Ding, C.; Rajasekaran, S. SAPAG: A Self-Adaptive Privacy Attack From Gradients. *arXiv* **2020**, arXiv:2009.06228.
35. Yin, H.; Mallya, A.; Vahdat, A.; Alvarez, J.M.; Kautz, J.; Molchanov, P. See Through Gradients: Image Batch Recovery via GradInversion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 16337–16346.
36. Wei, W.; Liu, L.; Loper, M.; Chow, K.H.; Gursoy, M.E.; Truex, S.; Wu, Y. A Framework for Evaluating Client Privacy Leakages in Federated Learning. In *Computer Security—ESORICS*; Lecture Notes in Computer Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 545–566. [CrossRef]
37. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated Learning: Challenges, Methods, and Future Directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [CrossRef]
38. Pfitzner, B.; Steckhan, N.; Arnrich, B. Federated Learning in a Medical Context: A Systematic Literature Review. *Acm Trans. Internet Technol.* **2021**, *21*, 1–31. [CrossRef]
39. Dwork, C. Differential Privacy. In *ICALP 2006: Automata, Languages and Programming*; Lecture Notes in Computer Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2006; Volume 4052, pp. 1–12.
40. Mironov, I. Rényi Differential Privacy. In Proceedings of the 2017 IEEE 30th Computer Security Foundations Symposium (CSF), Santa Barbara, CA, USA, 21–25 August 2017; pp. 263–275. [CrossRef]
41. Li, Y.; Chang, T.H.; Chi, C.Y. Secure Federated Averaging Algorithm with Differential Privacy. In Proceedings of the 2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP), Espoo, Finland, 21–24 September 2020; pp. 1–6. [CrossRef]
42. Truex, S.; Liu, L.; Chow, K.H.; Gursoy, M.E.; Wei, W. LDP-Fed: Federated Learning with Local Differential Privacy. In Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking, Heraklion, Greece, 27 April 2020; pp. 61–66. [CrossRef]
43. Choudhury, O.; Gkoulalas-Divanis, A.; Salonidis, T.; Sylla, I.; Park, Y.; Hsu, G.; Das, A. Differential Privacy-enabled Federated Learning for Sensitive Health Data. *arXiv* **2019**, arXiv:1910.02578.
44. Malekzadeh, M.; Hasircioglu, B.; Mital, N.; Katarya, K.; Ozfatura, M.E.; Gündüz, D. Dopamine: Differentially Private Federated Learning on Medical Data. *arXiv* **2021**, arXiv:2101.11693.
45. Adnan, M.; Kalra, S.; Cresswell, J.C.; Taylor, G.W.; Tizhoosh, H. Federated Learning and Differential Privacy for Medical Image Analysis. *Sci. Rep.* **2022**, *12*, 1953. [CrossRef]
46. Lenga, M.; Schulz, H.; Saalbach, A. Continual Learning for Domain Adaptation in Chest X-ray Classification. *Proc. Mach. Learn. Res.* **2020**, *121*, 413–423.
47. Mitra, A.; Chakravarty, A.; Ghosh, N.; Sarkar, T.; Sethuraman, R.; Sheet, D. A Systematic Search over Deep Convolutional Neural Network Architectures for Screening Chest Radiographs. In Proceedings of the 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Montreal, QC, Canada, 20–24 July 2020; Volume 2020, pp. 1225–1228. [CrossRef]
48. McMahan, H.B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), Fort Lauderdale, FL, USA, 20–22 April 2017.
49. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [CrossRef]
50. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
51. Baltruschat, I.M.; Nickisch, H.; Grass, M.; Knopp, T.; Saalbach, A. Comparison of Deep Learning Approaches for Multi-Label Chest X-ray Classification. *Sci. Rep.* **2019**, *9*, 6381. [CrossRef]

52. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; Volume 37, pp. 448–456.
53. Hitaj, B.; Ateniese, G.; Perez-Cruz, F. Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 603–618. [CrossRef]
54. Melis, L.; Song, C.; De Cristofaro, E.; Shmatikov, V. Exploiting Unintended Feature Leakage in Collaborative Learning. In Proceedings of the IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 691–706. [CrossRef]
55. Shokri, R.; Stronati, M.; Song, C.; Shmatikov, V. Membership Inference Attacks Against Machine Learning Models. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2017; pp. 3–18. [CrossRef]
56. Packhäuser, K.; Gündel, S.; Münster, N.; Syben, C.; Christlein, V.; Maier, A. Is Medical Chest X-ray Data Anonymous? *arXiv* **2021**, arXiv:2103.08562.
57. Sabottke, C.F.; Breaux, M.A.; Spieler, B.M. Estimation of Age in Unidentified Patients via Chest Radiography Using Convolutional Neural Network Regression. *Emerg. Radiol.* **2020**, *27*, 463–468. [CrossRef]
58. Dwork, C.; Roth, A. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* **2014**, *9*, 211–407. [CrossRef]
59. Wang, S.; Tuor, T.; Salonidis, T.; Leung, K.K.; Makaya, C.; He, T.; Chan, K. Adaptive Federated Learning in Resource Constrained Edge Computing Systems. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 1205–1221. [CrossRef]
60. Bagdasaryan, E.; Poursaeed, O.; Shmatikov, V. Differential Privacy Has Disparate Impact on Model Accuracy. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
61. Dong, J.; Roth, A.; Su, W.J. Gaussian Differential Privacy. *arXiv* **2019**, arXiv:1905.02383.