Federated Learning and Differential Privacy for Medical Image Analysis

Mohammed Adnan

University of Waterloo

Shivam Kalra

University of Waterloo

Jesse C. Cresswell

Layer 6 AI

Graham W. Taylor

University of Guelph

Hamid Tizhoosh (✉ tizhoosh@uwaterloo.ca)

University of Waterloo

Research Article

Keywords: medical, learning, datasets, study, private, images

Posted Date: November 9th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1005694/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at Scientific Reports on February 4th, 2022.
See the published version at <https://doi.org/10.1038/s41598-022-05539-7>.

Federated Learning and Differential Privacy for Medical Image Analysis

Mohammed Adnan^{1,2,4}, Shivam Kalra^{1,2,3}, Jesse C. Cresswell³, Graham W. Taylor^{2,4}, and Hamid R. Tizhoosh^{*1,2}

¹Kimia Lab, University of Waterloo, Canada

²Vector Institute, MaRS Discovery District, Toronto, Canada

³Layer 6 AI, MaRS Discovery District, Toronto, Canada

⁴University of Guelph, Ontario, Canada

ABSTRACT

The artificial intelligence revolution has been spurred forward by the availability of large-scale datasets. In contrast, the paucity of large-scale medical datasets hinders the application of machine learning in healthcare. The lack of publicly available multi-centric and diverse datasets mainly stems from confidentiality and privacy concerns around sharing medical data. To demonstrate a feasible path forward in medical image imaging, we conduct a case study of applying a differentially private federated learning framework for analysis of histopathology images, the largest and perhaps most complex medical images. We study the effects of IID and non-IID distributions along with the number of healthcare providers, i.e., hospitals and clinics, and the individual dataset sizes, using The Cancer Genome Atlas (TCGA) dataset, a public repository, to simulate a distributed environment. We empirically compare the performance of private, distributed training to conventional training and demonstrate that distributed training can achieve similar performance with strong privacy guarantees. We also study the effect of different source domains for histopathology images by evaluating the performance using external validation. Our work indicates that differentially private federated learning is a viable and reliable framework for the collaborative development of machine learning models in medical image analysis.

Introduction

Deep neural networks have achieved and established state-of-the-art results in many domains. However, deep learning models are data-intensive, i.e., they often require millions of training examples to learn effectively. Medical images may contain confidential and sensitive information about patients that often cannot be shared outside the institutions of their origin, especially when complete de-identification cannot be guaranteed. The European General Data Protection Regulation (GDPR) and the United States Health Insurance Portability and Accountability Act (HIPAA) enforce guidelines and regulations for storing and exchanging personally identifiable data and health data. Ethical guidelines also encourage respecting privacy, that is, the ability to retain complete control and secrecy about one's personal information¹. As a result, large archives of medical data from various consortia remain widely untapped sources of information. For instance, histopathology images cannot be collected and shared in large quantities due to the aforementioned regulations, as well as due to data size constraints given their high resolution and gigapixel nature. Without sufficient and diverse datasets, deep models trained on histopathology images from one hospital may fail to generalize well on data from a different hospital (out-of-distribution)^{2,3}. The existence of bias or the lack of diversity in images from a single institution brings about the need for a collaborative approach which does not require data centralization. One way to overcome this problem is by collaborative data sharing (CDS) or federated learning among different hospitals⁴.

In this paper, we explore federated learning (FL) as a collaborative learning paradigm, in which models can be trained across several institutions without explicitly sharing patient data. We show that using federated learning with additional privacy preservation techniques can improve the performance of histopathology image analysis compared to training without collaboration. We discuss its benefits, drawbacks, potential weaknesses, as well as technical implementation considerations. Finally, we use lung cancer images from The Cancer Genome Atlas (TCGA) dataset⁵ to construct a simulated environment of several institutions to validate our approach.

Federated Learning (FL). Federated learning algorithms learn from decentralized data distributed across various client devices, in contrast to conventional learning algorithms. In most examples of FL, there is a *centralized server* which facilitates training a shared model and addresses critical issues such as data privacy, security, access rights, and heterogeneity⁶. In FL, every client locally trains a copy of the centralized model, represented by the model weights ω , and reports its updates back to

Figure 1. The proposed federated learning algorithm to train a MEM model⁸ for WSIs (disease) classification among multiple hospitals. Each client in FL is represented by a blue rectangle. Each client, first transforms their local WSIs into mosaics (sets of representative patches). The patches in each mosaic are converted to feature vectors using a DenseNet model⁹. Finally the sets of feature vectors are classified using a MEM model. A shared central MEM model is trained using FedAvg⁶ among multiple clients (mimicking hospitals). Furthermore, DP-SGD¹⁰ is used for training the central MEM model with strict privacy bounds.

the server for aggregation across clients, without disclosing local private data. Mathematically, FL can be formulated as:

$$\min_{\omega \in \mathbb{R}^d} f(\omega) \quad \text{with} \quad f(\omega) = \frac{1}{n} \sum_{i=1}^n f_i(\omega), \quad (1)$$

where $f(\omega)$ represents the total loss function over n clients, and $f_i(\omega)$ represents the loss function with respect to client i 's local data. The objective is to find weights ω that minimize the overall loss. McMahan et al.⁶ introduced federated averaging, or *FedAvg* (Algorithm 1), in which each client receives the current model ω_t from the server, and computes $\nabla f_i(\omega_t)$, the average gradient of the loss function over its local data. The gradients are used to update each client's model weights using stochastic gradient descent (SGD) as $\omega_{t+1}^i \leftarrow \omega_t - \eta \nabla f_i(\omega_t)$ according to the learning rate η . Next, the central server receives the updated weights ω_{t+1}^i from all participating clients and averages them to update the central model, $\omega_{t+1} \leftarrow \sum_{i=1}^n \frac{n_i}{n} \omega_{t+1}^i$, where n_i is the number of data points used by client i . To reduce communication costs, several local steps of SGD can be taken before communication and aggregation, however, this affects the convergence properties of FedAvg⁷.

Other methods for FL have also been proposed. Yurochkin et al.¹¹ proposed a Bayesian framework for FL. Claici et al.¹² used KL divergence to fuse different models. Much work has also been done to improve the robustness of FL algorithms. Pillutla et al.¹³ proposed a robust and secure aggregation oracle based on the geometric median using a constant number of calls to a regular non-robust secure average oracle. Andrychowicz et al.¹⁴ proposed a meta-learning approach to coordinate the learning process in client/server distributed systems by using a recurrent neural network in the central server to learn how to

optimally aggregate the gradients from the client models. Li et al.¹⁵ proposed a new framework for robust FL where the central server learns to detect and remove malicious updates using a spectral anomaly detection model, leading to targeted defense.

Federated Learning in Histopathology. FL is especially important for histopathology departments, as it facilitates collaboration among institutions without sharing private patient data. One prominent challenge when applying FL to medical images, and specifically histopathology, is the problem of *domain adaptation*. Since most hospitals have diverse imaging methods and devices, images from a group of hospitals will be markedly different, and machine learning methods risk overfitting to non-semantic differences between them. Models trained using FL can suffer from serious performance drops when applied to images from previously unseen hospitals³. Several recent works have explored applications of FL in histopathology, and grapple with this problem. Lu et al.¹⁶ demonstrated the feasibility and effectiveness of FL for a large-scale computational pathology studies. FedDG proposed by Liu et al.¹⁷ is a privacy-preserving solution to learn a generalizable FL model through an effective continuous frequency space interpolation mechanism across clients. Sharing frequency domain information enables the separation of semantic information from noise in the original images. Li et al.¹⁸ tackles the problem of domain adaptation with a physics-driven generative approach to disentangle the information about model and geometry from the imaging sensor.

Algorithm 1: FedAvg (or, Federated Averaging)⁶. There are n clients, B is the local minibatch size, E is the number of local epochs per communication round, η is the learning rate, and f_i is the local loss function.

Server Executes

```

initialize  $\omega_0$ ;
for each round  $t = 0, 1, \dots$  do
  for each client  $i = 0, \dots, n-1$  in parallel
     $\omega_{t+1}^i \leftarrow \text{ClientUpdate}(i, \omega_t)$ 
   $\omega_{t+1} \leftarrow \sum_{i=1}^n \frac{n_k}{n} \omega_{t+1}^k$ 

```

ClientUpdate(i, ω): // Run on client i

```

for each local epoch  $e$  from  $0, \dots, E-1$  do
  for each minibatch  $b$  of size  $B$  do
     $\omega_{e+1} \leftarrow \omega_e - \eta \nabla f_i(\omega_e; b)$ 
return  $\omega_E$  to server

```

Differential Privacy. While FL attempts to provide privacy by keeping private data on client devices, it does not provide a meaningful privacy guarantee. Updated model parameters are still sent from the clients to a centralized server, and these can contain private information¹⁹, such that even individual data points can be reconstructed²⁰. *Differential privacy* (DP) is a formal framework for quantifying the privacy that a protocol provides²¹. The core idea of DP is that privacy should be viewed as a resource, something that is used up as information is extracted from a dataset. The goal of private data analysis is to extract as much useful information as possible while consuming the least privacy. To formalize this concept, consider a *database* $\mathcal{D}$, which is simply a set of datapoints, and a probabilistic function M acting on databases, called a *mechanism*. The mechanism is said to be (ϵ, δ) -*differentially private* if for all subsets of possible outputs $\mathcal{S} \subseteq \text{Range}(M)$, and for all pairs of databases $\mathcal{D}$ and $\mathcal{D}'$ that differ by one element,

$$\Pr[M(\mathcal{D}) \in \mathcal{S}] \leq \exp(\epsilon) \Pr[M(\mathcal{D}') \in \mathcal{S}] + \delta. \quad (2)$$

When both ϵ and δ are small positive numbers, Equation 2 implies that the outcomes of M will be almost unchanged in distribution if one datapoint is changed in the database. In other words, adding one patient's data to a differentially private study will not affect the outcomes, with high probability.

The advantage of DP is that it is quantitative. It yields a numerical guarantee on the amount of privacy that can be expected, in the stochastic sense, where lower ϵ and δ implies that the mechanism preserves more privacy. The framework also satisfies several useful properties. When multiple DP-mechanisms are composed, the total operation is also a DP-mechanism with well defined ϵ and δ ²². Also, once the results of a DP-mechanism are known, no amount of post-processing can change the (ϵ, δ) guarantee²³. Hence, while FL alone does not guarantee privacy, we can apply FL in conjunction with DP to give rigorous bounds on the amount of privacy afforded to clients and patients who participate in the collaboration.

The simplest way to create a DP-mechanism is by adding Gaussian noise to the outcomes of a deterministic function with bounded sensitivity²⁴. This method can be used in the context of training a machine learning model by clipping the norm of gradients to bound them, then adding noise, a process called *differentially private stochastic gradient descent* (DP-SGD)¹⁰. McMahan et al.²⁵ applied this at scale to FL.

Figure 2. Illustration of a sample WSI and its mosaic extracted using the approach in Kalra et al.²⁶

Differential Privacy for Medical Imaging. Past works have noted the potential solution DP provides for machine learning in the healthcare domain. Kaissis et al.¹ surveyed privacy-preservation techniques to be used in conjunction with machine learning, which were then implemented for classifying chest X-rays and segmenting CT scans^{27,28}. In histopathology, Lu et al.¹⁶ reported DP guarantees for a neural network classifier trained with FL, following Li et al.²⁹. Their treatment involved adding Gaussian noise to trained model weights, however, neural networks weights do not have bounded sensitivity making their DP guarantee vacuous. A meaningful guarantee would require clipping the model weights before adding noise. We propose the more standard approach of DP-SGD, which clips gradient updates and adds noise, for use in histopathology.

Multiple Instance Learning (MIL). MIL is a type of supervised learning approach which uses a set of instances known as a bag. Instead of individual instances having an associated label, only the bag as a whole has one³⁰. MIL is thus a natural candidate for learning to classify WSIs which must be broken into smaller representations due to size limitations. Permutation invariant operators for MIL were introduced by Tomczak et al.³¹ and successfully applied to digital pathology images. Isle et al.³² used MIL for digital pathology and introduced a different variety of MIL pooling functions, while Sudarshan et al.³³ used MIL for histopathological breast cancer image classification. Graph neural networks (GNNs) have been used for MIL applications because of their permutation invariant characteristics. Tu et al.³⁴ showed that GNNs can be used for MIL, where each instance acts as a node in a graph. Adnan et al.³⁵ demonstrated an application of graph convolution neural networks to MIL in digital pathology and achieved state of the art accuracy on a lung sub cancer classification task.

Method

Our proposed method (*local* to each client) consists of two steps, *bag preparation* and *Multiple-Instance Learning (MIL)*. In the first step, we extract multiple patches from the full-resolution WSI and create a mosaic (set) of patches. In the second step, we formulate the representation learning of WSIs as a set learning problem by applying a MEM model, an attention based MIL algorithm proposed by Kalra et al.⁸. The MEM model is locally trained through DP-SGD to provide quantitative privacy bounds, and the local MEM models are centrally aggregated through FedAvg. In this section, we discuss the bag preparation step and MIL. An overview of the proposed method is visualized in Figure 1.

Bag Preparation. A patch selection method proposed by Kalra et al.²⁶ is used to extract representative patches (called *mosaics*) from each WSI. A sample WSI and its mosaic is illustrated in Figure 2. The steps involved in creations of a mosaic are: (i) removal of non-tissue regions using colour thresholding; (ii) grouping the remaining tissue-containing patches into a pre-set number of categories through a clustering algorithm; and (iii) randomly selecting a portion of all clustered patches (e.g., 10%) within each cluster, yielding a *mosaic*. The mosaic is transformed into a bag $X = \{x_1, \dots, x_n\}$ for MIL, where x_i is the feature vector of the i^{th} patch, obtained through a pre-trained feature extractor network. We use a DenseNet model for the feature extractor⁹. Each patch in the mosaic has size 1000×1000 pixels at 20x magnification (0.5 mpp resolution).

MIL Method. We used the MEM model proposed by Kalra et al.⁸ to get a single vector representation of all feature vectors of patches in a mosaic. MEM consists of memory units composed within a memory block. A *memory block* is the main component of MEM and produces a permutation invariant representation from a input sequence. Multiple memory blocks can be stacked together for modeling complex relationships and dependencies in set data. The memory block is made of memory units and a bijective transformation unit shown in Figure 3. A memory unit transforms an input sequence into an attention vector. A higher attention value represents a higher “importance” of the corresponding element of the input sequence. Essentially, it captures the

Figure 3. Schematic of a MEM model used for the classification of WSI mosaics. X is an input sequence containing a number n of f -dimensional vectors. (a) The memory block is a sequence-to-sequence model that takes X and returns another sequence $\hat{X}$. The output $\hat{X}$ is a permutation-invariant representation of X . A bijective transformation model (an autoencoder) converts the input X to a permutation-equivariant sequence C . The weighted sum of C is computed over different probability distributions p_i from memory units. The hyper-parameters of a memory block are i) the dimensions of the bijective transformation h , and ii) the number of memory units m . (b) The memory unit has A_i , a trainable embedding matrix that transforms elements of X to a d -dimensional space (memories). The output p_i is a probability distribution over the input X , also known as attention. The memory unit has a single hyper-parameter d , i.e. the dimension of the embedding space⁸ (* represents learnable parameters).

relationships among different elements of the input. Multiple memory units enable the memory block to capture many complex dependencies and relationships among the elements. Each memory unit consists of an embedding matrix A_i that transforms a f -dimensional input vector x_j to a d -dimensional memory vector u_{ij} according to

$$u_{ij} = \rho(x_j A_i), \quad (3)$$

where ρ is some non-linearity. The memory vectors are stacked to form a matrix $U_i = [u_{i0}, \dots, u_{in}]$ of shape $(n \times d)$. The relative degree of correlations among the memory vectors are computed using cross-correlation followed by a column-wise softmax and then taking a row-wise average,

$$\begin{aligned} S_i &= \text{column-wise-softmax}(U_i U_i^T), \\ p_i &= \text{row-wise-average}(S_i). \end{aligned} \quad (4)$$

The p_i is the final output vector ($1 \times n$) from the i^{th} memory unit U_i , as shown in Figure 3. The purpose of each memory unit is to embed feature vectors into another space that could correspond to a distinct “attribute” or “characteristic” of instances. The cross correlation of the calculated attention vectors highlights instances which are highly suggestive of those attributes. Memory vectors are non-normalized as the magnitude may play an important role during the cross correlation.

In summary, a memory block is a sequence-to-sequence model, i.e., it transforms a given input sequence $X = x_1, \dots, x_n$ to another representative sequence $\hat{X} = \hat{x}_1, \dots, \hat{x}_m$. A memory block contains m memory units, each of which takes sequential data as an input and generates an attention vector. These attention vectors are subsequently used to compute the final output sequence. By design, the output sequence is invariant to element-wise permutations of the input sequence as needed for MIL.

Experiments and Discussion

We validated the performance of FL for the classification of histopathology images using a simulated distributed environment and also using real-world hospital data. Previous studies have mostly experimented with a fixed number of clients having similar distributions of data^{1,16,36}. Since real-world data is not necessarily IID, it is important to study the effect of non-IID data on the performance of FL, specifically *FedAvg*. Furthermore, we provide a privacy analysis of the method through the differential privacy framework, suggesting that FL can outperform non-collaborative training while maintaining a strong privacy guarantee.

In the *first experiment series*, we vary the number of clients, with each client representing one hospital. To make our simulated environment better approach the non-IID real-world data, each client can have a different number of patients and a

Data Distribution	Number of Clients n	Accuracy		
		Without FL	With FL	Centralized
IID	4	0.731 ± 0.03	0.824 ± 0.02	0.848 ± 0.02
	8	0.620 ± 0.06	0.780 ± 0.05	
	16	0.570 ± 0.03	0.726 ± 0.06	
	32	0.527 ± 0.02	0.641 ± 0.09	
Non IID	4	0.682 ± 0.10	0.824 ± 0.01	0.848 ± 0.02
	8	0.561 ± 0.08	0.823 ± 0.05	
	16	0.524 ± 0.03	0.750 ± 0.06	
	32	0.520 ± 0.03	0.550 ± 0.20	

Table 1. Evaluation on different data distributions. Centralized accuracy denotes the accuracy when the data is centralized. The accuracy without FL is the mean and standard deviation of accuracy values across multiple clients without any collaboration. The accuracy with FL is the mean and standard deviation of the central model trained at the end of FL evaluated on each client dataset.

Dataset Type	Source Hospital (Clients)	LUAD Images	LUSC Images	Total
Train/Test	International Genomics Consortium	189	78	267
	Indivumed	94	117	211
	Asterand	90	117	207
	Johns Hopkins	121	78	199
External	Christiana Healthcare	169	54	223
	Roswell Park	35	75	110
	Princess Margaret Hospital (Canada)	0	52	52

Table 2. Source hospitals for test/train and external dataset and their data distribution.

different distribution of cancer sub-types. In the *second experiment series*, we calculate the privacy bound of differentially private FL using real-world hospital data. We used the available attributes in TCGA to divide the dataset across the tissue origin site (hospital) and created four client datasets as shown in Table 2.

Lung Cancer Dataset – LUAD vs LUSC Classification. Lung Adenocarcinoma (LUAD) and Lung Squamous Cell Carcinoma (LUSC) are two main sub-types of non-small cell lung cancer (NSCLC) that account for 65-70% of all lung cancers³⁷. An automated classification of these two main sub-types of NSCLC is a crucial step to assist pathologists for more informed diagnoses^{37,38}. We obtained 2,580 hematoxylin and eosin (H&E) stained WSIs of lung cancer from TCGA³⁹, comprising about two TB of data. The images were split into two groups of 1,806 training, and 774 testing samples WSIs⁸. We transformed each raw image into a mosaic²⁶, and then into a bag of features X using a pre-trained DenseNet⁹. From the data, we carried out two experiment series by varying the parameters of FedAvg, or by varying the data distributions across clients. These experiment series are discussed as follows.

Experiment Series 1 – Effect of Number of Clients and Data Distributions. We studied the effect of IID and non-IID distributions on the performance of FedAvg by randomly dividing the training images without replacement among different clients (hospitals). We also varied the number of clients (n) while keeping the total number of images fixed. FL achieves superior performance for both IID and non-IID distributions of data compared to non-collaborative training. FL performs comparably to centralized training for reasonably sized datasets ($n = 4, 8$). Interestingly, FL can achieve slightly better accuracy when trained on a non-IID data distribution. Results are summarized in Table 1 and Figure 4. The number of training samples for each client model is in Figure 5.

We compared the performance with and without FedAvg for each setting. In total we tested 16 experimental settings in Table 1. In each of the experiments, the server model trained using FedAvg outperformed the models trained using local client datasets, showing the advantage of collaboration. As the total dataset is divided into smaller partitions for more clients, both client and server model performances deteriorate. We visualize the relative improvement of FedAvg in Table 1.

Experiment Series 2 – Real-world Setting. In the second experiment series, we considered the effect of distributional differences from different source hospitals, and a requirement to preserve privacy. Histopathology images can differ greatly,

Figure 4. Comparison of the mean accuracy across clients versus the accuracy of the central model trained with FL for the fabricated clients (not the real hospitals). The accuracy is computed on two types of data distribution settings across clients—IID and Non-IID.

Source Hospital	Non-collaborative Training		FL Training		Combined Training	
	Test	External	Test	External	Test	External
International Genomics Consortium	0.654	0.631	0.823 ± 0.01	0.707 ± 0.01	0.839 ± 0.01	0.768 ± 0.003
Indivumed	0.648	0.556				
Asterand	0.709	0.701				
John Hopkins	0.681	0.600				

Table 3. Evaluation of collaborative and non-collaborative learning on Test and External Datasets using DP-SGD, achieving privacy parameter $\epsilon = 2.90$ for $\delta = 0.0001$. For FL and Combined training we report the mean accuracy and standard deviation across the client’s test datasets. On the external dataset we ran the experiments using three random initializations, and report the mean accuracy and standard deviation across them.

among others depending on the staining and imaging protocols of the source hospital. We selected seven hospitals from the TCGA dataset, four to act as clients in FL, and an additional three to provide externally collected data for model robustness testing. The distribution of images by hospital is described in Table 2. For each of the four clients, we divided their available images in an 80 : 20 ratio for training and internal testing datasets, respectively. Then we combined the images from the remaining three hospitals into a single external validation dataset to study the effects of distributions shifts on FedAvg.

We compared the performance of privacy-preserving FL training with both centralized training and non-collaborative training. In the FL training, the four hospitals act as clients collaborating to train one central model. Performance is evaluated on each client’s internal test set, as well as the external validation set. For comparison, we train a single model on the combined (centralized) training datasets which gives an upper bound on what could be achieved in the absence of privacy regulations. Finally, in the non-collaborative setting each client hospital trains their own model on only their own training dataset. We used DP-SGD to train the FL and combined models and computed the privacy guarantees (ϵ , δ) using a Rényi DP accountant⁴⁰. It was observed that the MEM model was sensitive to DP-SGD hyper parameters. We used a vectorized Adam optimizer⁴¹ with the following hyper-parameter values¹⁰: epochs = 180, training set size = 705, batch size = 32, gradient clipping norm = 1.0, Gaussian noise standard deviation = 4.0, number of minibatches = 32, learning rate = 2×10^{-5} .

As shown in Table 3, FL training achieves strong privacy bounds ($\epsilon = 2.90$ at $\delta = 0.0001$) with better performance than non-collaborative training, comparable to centralized training. This demonstrates that FL could be effectively used in clinical settings to ensure data privacy with no significant degradation in performance. Results are shown in Table 3. FedAvg achieves comparable performance to centralized training without explicitly sharing private data with strong privacy guarantees. Due to distribution shifts, accuracy decreases on external validation for both Federated Learning and centralized training. Therefore, we experimentally demonstrate the Federated Learning can be used for medical image analysis in real-world setting without explicitly sharing data, while achieving similar performance to centralized training with data sharing.

Conclusions

There is a vast reserve of knowledge in mass archives of clinical data held by hospitals which remains mostly untapped due to many confidentiality and privacy concerns. In this work, we proposed *differentially private federated learning* as a potential

method for learning from decentralized medical data such as histopathology images. Federated learning allows training models without explicitly sharing patient data and thus mitigates some confidentiality and privacy issues associated with clinical data. Differential privacy supplements this with quantitative bounds on the amount of privacy provided. We demonstrated the efficacy of federated learning (FedAvg) with simulated real-world data, using both IID and non-IID data distributions. Private federated learning achieves a comparable result compared to conventional centralized training, and hence it could be considered for distributed training on medical data.

References

1. Kaissis, G. A., Makowski, M. R., Rückert, D. & Braren, R. F. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence* **2**, 305–311 (2020). URL <https://www.nature.com/articles/s42256-020-0186-1>. Number: 6 Publisher: Nature Publishing Group.
2. Aggarwal, R. *et al.* Diagnostic accuracy of deep learning in medical imaging: a systematic review and meta-analysis. *NPJ digital medicine* **4**, 1–23 (2021).
3. Dehkharghanian, T. *et al.* Biased data, biased ai: Deep networks predict the acquisition site of tcga images. *under review* (2021).
4. Sheller, M. J. *et al.* Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports* **10**, 12598 (2020). URL <https://www.nature.com/articles/s41598-020-69250-1>. Number: 1 Publisher: Nature Publishing Group.
5. Weinstein, J. N. *et al.* The cancer genome atlas pan-cancer analysis project. *Nature genetics* **45**, 1113–1120 (2013).
6. McMahan, B., Moore, E., Ramage, D., Hampson, S. & y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282 (PMLR, 2017).
7. Li, X., Huang, K., Yang, W., Wang, S. & Zhang, Z. On the Convergence of FedAvg on Non-IID Data. In *International Conference on Learning Representations* (2020). URL <https://openreview.net/forum?id=HJxNANvtDS>.
8. Kalra, S., Adnan, M., Taylor, G. & Tizhoosh, H. R. Learning permutation invariant representations using memory networks. In *European Conference on Computer Vision*, 677–693 (Springer, 2020).
9. Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700–4708 (2017).
10. Abadi, M. *et al.* Deep Learning with Differential Privacy. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (2016). URL <http://dx.doi.org/10.1145/2976749.2978318>.
11. Yurochkin, M. *et al.* Bayesian Nonparametric Federated Learning of Neural Networks. *arXiv:1905.12022 [cs, stat]* (2019). URL <http://arxiv.org/abs/1905.12022>. ArXiv: 1905.12022.
12. Claici, S., Yurochkin, M., Ghosh, S. & Solomon, J. Model Fusion with Kullback–Leibler Divergence. *arXiv:2007.06168 [cs, stat]* (2020). URL <http://arxiv.org/abs/2007.06168>. ArXiv: 2007.06168.
13. Pillutla, K., Kakade, S. M. & Harchaoui, Z. Robust Aggregation for Federated Learning. *arXiv:1912.13445 [cs, stat]* (2019). URL <http://arxiv.org/abs/1912.13445>. ArXiv: 1912.13445.
14. Andrychowicz, M. *et al.* Learning to learn by gradient descent by gradient descent (2016). [1606.04474](https://arxiv.org/abs/1606.04474).
15. Li, S., Cheng, Y., Wang, W., Liu, Y. & Chen, T. Learning to Detect Malicious Clients for Robust Federated Learning (2020). [2002.00211](https://arxiv.org/abs/2002.00211).
16. Lu, M. Y. *et al.* Federated learning for computational pathology on gigapixel whole slide images. *arXiv preprint arXiv:2009.10190* (2020).
17. Liu, Q., Chen, C., Qin, J., Dou, Q. & Heng, P.-A. FedDG: Federated Domain Generalization on Medical Image Segmentation via Episodic Learning in Continuous Frequency Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1013–1023 (2021).
18. Li, D., Kar, A., Ravikumar, N., Frangi, A. F. & Fidler, S. Fed-Sim: Federated Simulation for Medical Imaging (2020). [2009.00668](https://arxiv.org/abs/2009.00668).
19. Bhowmick, A., Duchi, J., Freudiger, J., Kapoor, G. & Rogers, R. Protection against reconstruction and its applications in private federated learning. *arXiv preprint arXiv:1812.00984* (2018).
20. Melis, L., Song, C., De Cristofaro, E. & Shmatikov, V. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, 691–706 (IEEE, 2019).

21. Dwork, C., McSherry, F., Nissim, K. & Smith, A. Calibrating noise to sensitivity in private data analysis. In Halevi, S. & Rabin, T. (eds.) *Theory of Cryptography*, 265–284 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006).
22. Dwork, C., Rothblum, G. N. & Vadhan, S. Boosting and differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, 51–60 (IEEE, 2010).
23. Dwork, C. & Roth, A. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* **9**, 211–407 (2014). URL <https://doi.org/10.1561/04000000042>.
24. Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I. & Naor, M. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *Advances in Cryptology (EUROCRYPT 2006)*, vol. 4004 of *Lecture Notes in Computer Science*, 486–503 (Springer Verlag, 2006). URL <https://www.microsoft.com/en-us/research/publication/our-data-ourselves-privacy-via-distributed-noise-generation/>.
25. McMahan, H. B., Ramage, D., Talwar, K. & Zhang, L. Learning Differentially Private Recurrent Language Models. In *International Conference on Learning Representations* (2018).
26. Kalra, S. *et al.* Yottixel – An Image Search Engine for Large Archives of Histopathology Whole Slide Images. *Medical Image Analysis* **65**, 101757 (2020). URL <https://www.sciencedirect.com/science/article/pii/S1361841520301213>.
27. Kaissis, G. *et al.* End-to-end privacy preserving deep learning on multi-institutional medical imaging. *Nature Machine Intelligence* **3**, 473–484 (2021).
28. Ziller, A. *et al.* Medical imaging deep learning with differential privacy. *Scientific Reports* **11**, 1–8 (2021).
29. Li, X. *et al.* Multi-site fMRI analysis using privacy-preserving federated learning and domain adaptation: ABIDE results. *Medical Image Analysis* **65**, 101765 (2020). URL <https://www.sciencedirect.com/science/article/pii/S1361841520301298>.
30. Carbonneau, M.-A., Cheplygina, V., Granger, E. & Gagnon, G. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition* **77**, 329–353 (2018). URL <https://linkinghub.elsevier.com/retrieve/pii/S0031320317304065>.
31. Tomczak, J. M., Ilse, M. & Welling, M. Deep Learning with Permutation-invariant Operator for Multi-instance Histopathology Classification. *Workshop on Bayesian Deep Learning at 31st Conference on Neural Information Processing Systems* (2017). URL <http://arxiv.org/abs/1712.00310>.
32. Ilse, M., Tomczak, J. & Welling, M. Attention-based deep multiple instance learning. In *International conference on machine learning*, 2127–2136 (PMLR, 2018).
33. Sudharshan, P. *et al.* Multiple instance learning for histopathological breast cancer image classification. *Expert Systems with Applications* **117**, 103–111 (2019). URL <https://linkinghub.elsevier.com/retrieve/pii/S0957417418306262>.
34. Tu, M., Huang, J., He, X. & Zhou, B. Multiple instance learning with graph neural networks. *ICML 2019 Workshop on Learning and Reasoning with Graph-Structured Representations* (2019). URL <http://arxiv.org/abs/1906.04881>.
35. Adnan, M., Kalra, S. & Tizhoosh, H. R. Representation learning of histopathology images using graph neural networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 4254–4261 (2020).
36. Chang, K. *et al.* Distributed deep learning networks among institutions for medical imaging. *Journal of the American Medical Informatics Association* **25**, 945–954 (2018).
37. Zappa, C. & Mousa, S. A. Non-small cell lung cancer: current treatment and future advances. *Translational lung cancer research* **5**, 288 (2016).
38. Graham, S. *et al.* Classification of lung cancer histology images using patch-level summary statistics. In *Medical Imaging 2018: Digital Pathology*, vol. 10581, 1058119 (International Society for Optics and Photonics, 2018).
39. Tomczak, K., Czerwińska, P. & Wiznerowicz, M. The Cancer Genome Atlas (TCGA): an immeasurable source of knowledge. *Contemporary oncology* **19**, A68 (2015).
40. Mironov, I. Rényi differential privacy. *2017 IEEE 30th Computer Security Foundations Symposium (CSF)* (2017). URL <http://dx.doi.org/10.1109/CSF.2017.11>.
41. Subramani, P., Vadivelu, N. & Kamath, G. Enabling fast differentially private sgd via just-in-time compilation and vectorization (2020). [2010.09063](https://arxiv.org/abs/2010.09063).

Figure 5. Visualisation of IID and non-IID distribution of data among client models

Acknowledgements

We would like to thank the Ontario Government for awarding an ORF-RE grant for this project (Ontario Research Fund Research Excellence). The first author is supported by a Vector Institute internship. The basic research of the corresponding author leading to this work has been supported by The Natural Sciences and Engineering Research Council of Canada (NSERC) through multiple Discovery Grants. For access to computational and storage facilities, we would like to thank Compute Canada.

Author Contributions

MA designed and conducted the experiments, and wrote the first draft of the paper. SK helped in designing the datasets, figures experimental setup, and provided important edits to the paper. JCC was involved in the discussion, contributed to the differential privacy analysis of the method, reviewed the code, and provided important edits to the paper. GWT and HRT were involved in the discussions of the approach, and provided critical feedback to the paper. HRT guided the external validations with TCGA images.

Data availability

The publicly available dataset of 30,072 WSIs from TCGA³⁹ (Genomic Data Commons GDC) was used for conducting this study.

List of Figures

1	The proposed federated learning algorithm to train a MEM model ⁸ for WSIs (disease) classification among multiple hospitals. Each client in FL is represented by a blue rectangle. Each client, first transforms their local WSIs into mosaics (sets of representative patches). The patches in each mosaic are converted to feature vectors using a DenseNet model ⁹ . Finally the sets of feature vectors are classified using a MEM model. A shared central MEM model is trained using FedAvg ⁶ among multiple clients (mimicking hospitals). Furthermore, DP-SGD ¹⁰ is used for training the central MEM model with strict privacy bounds.	2
2	Illustration of a sample WSI and its mosaic extracted using the approach in Kalra et al. ²⁶	4
3	Schematic of a MEM model used for the classification of WSI mosaics. X is an input sequence containing a number n of f -dimensional vectors. (a) The memory block is a sequence-to-sequence model that takes X and returns another sequence $\hat{X}$. The output $\hat{X}$ is a permutation-invariant representation of X . A bijective transformation model (an autoencoder) converts the input X to a permutation-equivariant sequence C . The weighted sum of C is computed over different probability distributions p_i from memory units. The hyper-parameters of a memory block are i) the dimensions of the bijective transformation h , and ii) the number of memory units m . (b) The memory unit has A_i , a trainable embedding matrix that transforms elements of X to a d -dimensional space (memories). The output p_i is a probability distribution over the input X , also known as attention. The memory unit has a single hyper-parameter d , i.e. the dimension of the embedding space ⁸ (* represents learnable parameters).	5
4	Comparison of the mean accuracy across clients versus the accuracy of the central model trained with FL for the fabricated clients (not the real hospitals). The accuracy is computed on two types of data distribution settings across clients—IID and Non-IID.	7
5	Visualisation of IID and non-IID distribution of data among client models	10

List of Tables

1 Evaluation on different data distributions. Centralized accuracy denotes the accuracy when the data is centralized. The accuracy without FL is the mean and standard deviation of accuracy values across multiple clients without any collaboration. The accuracy with FL is the mean and standard deviation of the central model trained at the end of FL evaluated on each client dataset. 6

2 Source hospitals for test/train and external dataset and their data distribution. 6

3 Evaluation of collaborative and non-collaborative learning on Test and External Datasets using DP-SGD, achieving privacy parameter $\epsilon = 2.90$ for $\delta = 0.0001$. For FL and Combined training we report the mean accuracy and standard deviation across the client’s test datasets. On the external dataset we ran the experiments using three random initializations, and report the mean accuracy and standard deviation across them. 7