2022 44th Annual International Conference of
the IEEE Engineering in Medicine & Biology Society (EMBC)
Scottish Event Campus, Glasgow, UK, July 11-15, 2022

2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) | 978-1-7281-2782-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/EMBC48229.2022.9871742

Privacy-preserving Model Training for Disease Prediction Using Federated Learning with Differential Privacy

Amol Khanna[1], Vincent Schaffer[2], Gamze Gürsoy*[,3], and Mark Gerstein*[,4]

***Abstract*— Machine learning is playing an increasingly critical role in health science with its capability of inferring valuable information from high-dimensional data. More training data provides greater statistical power to generate better models that can help decision-making in healthcare. However, this often requires combining research and patient data across institutions and hospitals, which is not always possible due to privacy considerations. In this paper, we outline a simple federated learning algorithm implementing differential privacy to ensure privacy when training a machine learning model on data spread across different institutions. We tested our model by predicting breast cancer status from gene expression data. Our model achieves a similar level of accuracy and precision as a single-site non-private neural network model when we enforce privacy. This result suggests that our algorithm is an effective method of implementing differential privacy with federated learning, and clinical data scientists can use our general framework to produce differentially private models on federated datasets. Our framework is available at https://github.com/gersteinlab/idash20FL.**

I. INTRODUCTION

Machine learning and pattern recognition have been important tools in biomedical and clinical research to identify meaningful patterns between clinical measurements and human disease [1], [2]. For example, models can predict whether a patient has a certain type of cancer using gene expression values as features [3], [4]. In the clinical setting, machine learning allows for quick diagnosis of unclear cases and provides information regarding the relationship of important features with disease.

To make meaningful and accurate predictions, machine learning algorithms must be trained on large numbers of samples [5]. Obtaining a large enough sample size at a single medical institution is often not possible; therefore, data from multiple sites must be used. This creates concerns regarding the privacy of training data [6]. Especially in the clinical setting, institutional policies may not allow outside parties to access medical data, whether or not the data is de-identified. These restrictions can even be set at the country or continent level. For example, the General Data Protection Regulation in the European Union prohibits hospitals and institutions from sharing personal data with third parties [7].

Federated learning is a technique to train a shared neural network on data kept at different sites [8]. In federated learning, each site locally trains a network and combines this network with networks trained at other sites. This approach ensures that data owners from each site cannot access others' data. Federated learning has been shown to be useful for genomic and medical research, especially when dealing with large amounts of data from different sites [9], [10]. However, this decentralized training mechanism has privacy issues. Studies have shown that neural networks tend to memorize training data, and trained networks can be reverse-engineered to determine samples used to train the network [11], [12]. To remedy this problem, differential privacy can be used when sites share model parameters.

Differential privacy is a method to ensure that single data points in a dataset cannot be identified with the output of summary statistics [13]. This is achieved by adding controlled noise to outputs. A differentially private mechanism ensures that summary statistics will not significantly change whether an individual's data is present or not, and thus ensures that the individual's participation to the data cannot be inferred from the summary-level output. Mathematically, differential privacy is defined as the following: let ϵ be a non-negative number and $\mathcal{A}$ be a randomzied algorithm taking a dataset as input. $\mathcal{A}$ is ϵ-differentially private if $\mathbb{P}[\mathcal{A}(D_1)] \leq e^{\epsilon} \cdot \mathbb{P}[\mathcal{A}(D_2)]$, where D_1 and D_2 are datasets which differ by only a single datapoint. This equation implies that the results of an algorithm will not change significantly whether or not an individual is in the dataset. Therefore, an algorithm which satisfies this equation protects the individual's privacy since the algorithm's output will not reveal information specific to the individual. In practice, this is achieved by adding noise from a Laplacian or Gaussian distribution to the results of the algorithm. Note that small ϵ values correspond to higher privacy.

Our model uses differential privacy to apply noise to neural network weight parameters after training at each site. This ensures that users cannot reverse-engineer network weights to obtain information about members of the training dataset.

We aimed to develop a general framework for privacy-preserving federated learning which can be easily implemented by clinical data scientists to solve a variety of clinical prediction problems. To this end, we adopted the publicly-available TensorFlow-Privacy package [14] for differential

[1]Department of Biomedical Engineering, Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218, USA
[2]Department of Computer Science, Yale University, New Haven, CT 06520, USA
[3]Department of Biomedical Informatics, Columbia University, New York, NY 10032, USA; New York Genome Center, New York, NY 10013, USA gamze.gursoy@columbia.edu
[4]Program in Computational Biology and Bioinformatics, Department of Molecular Biophysics and Biochemistry, Department of Computer Science, Department of Statistics and Data Science, Yale University, New Haven, CT 06250, USA mark@gersteinlab.org
*Corresponding Author

privacy. Note that TensorFlow-Privacy does not provide the option of choosing the ϵ privacy level and instead takes a "noise multiplier" parameter for Laplacian or Gaussian noise, which can then be converted into ϵ. However, for users who are not experts in differential privacy, it is not clear which noise multiplier corresponds to what ϵ value. To address this issue, our generalized framework is capable of taking epsilon values as input. It then performs neural network training in federated setting following established deep learning techniques including learning rate decay and an early stopping criterion [15]. We developed a model using this framework to predict breast cancer status from gene expression data to demonstrate that this framework can produce accurate predictions with high precision while ensuring the privacy of individuals in the training dataset.

II. METHODS

Our algorithm operates on two levels: the server level and the client level. At the server level, it performs the following. Since TensorFlow-Privacy cannot accept an ϵ value and instead expects a noise multiplier, we implemented a simple linear search function which iterates through equally spaced noise multipliers until finding a noise multiplier which produces the desired ϵ value. The server is then responsible for setting and updating the hyperparameters and architecture of the neural networks and storing the current averaged, or federated, neural network. It performs federated iterations, in which it sends this neural network along with hyperparameters to each client, receives trained weights back from each client, and averages the weights it received to produce a new federated weights for the network [16]. This average is performed by weighting each set of neural network weights by the fraction of total samples which the client holds. The server also receives accuracy metrics for the previously sent federated model from each network. If the federated model has not improved for some set number of iterations, the server will return the best-performing federated model. Until then, the server will continue perform federated iterations.

The clients are responsible for receiving the current federated weights and hyperparameters from the server. Prior to training, the clients compute the accuracy of the neural network with federated weights on a held-out validation and test set, and send these accuracy metrics to the server. After this, the clients train the network further based on locally stored data. They then add differentially-private noise to the trained weights before sending back them to the server.

Figure 1 consists of detailed flowcharts of the server and client processes.

To test our framework, we used data provided by the iDASH 2020 competition [17]. This data contained the gene expression of 17,814 genes taken from 61 normal and 529 tumor samples. The samples were split among two clients in four different ways. The splits are listed in Table 1. Independent and identically distributed (IID) splits indicate that the proportion of normal to tumor training data on both clients was roughly the same, and equal splits indicate that the overall number of examples on each client was roughly the same.

III. RESULTS

After the data was split, each client randomly chose 10% of its data as a validation set and another 10% of its data as a test set. Finally, to prevent inherent bias towards either condition, each client oversampled its normal training data to meet the number of tumor training samples.

After finishing this initial process, differentially-private federated training began. $\epsilon = \{1, 5, 10, \ldots, 45, 50\}$ were used. While it is known that $\epsilon > 5$ does not provide significant privacy benefits, high values of ϵ were tested to observe patterns over a wide range of ϵ values.

TABLE I
DATA SPLIT CONDITIONS

Label	Client 1		Client 2	
	Normal	Tumor	Normal	Tumor
IID, Equal	31	264	30	265
IID, Unequal	15	132	46	397
Non-IID, Equal	14	281	46	248
Non-IID, Unequal	14	397	46	132

Table 2 indicates the model hyperparameters and Figure 2 demonstrates the network architecture for our setting. Our network consisted of an input layer for 17,814 genes, followed by a 20% dropout to improve model generalization. We then fed the remaining values into layers of 100, 10, and 1 nodes. The 100 and 10 layers consisted of rectified linear unit (ReLU) activation functions, while the 1 layer consisted of a logistic activation function to produce binary output.

TABLE II
MODEL HYPERPARAMETERS

Hyperparameter	Value
Early Stopping Patience	10
Batch Size	32
Client Epochs	20
Initial Learning Rate	0.0005
Server Learning Rate Decay	0.95
Delta	$\frac{1}{17814}$
ℓ_2 Norm Clip	10

Since the data splits and differential privacy were a random process, we ran the model 20 times for each data split and ϵ condition to collect metrics and measure performance. Figure 3 demonstrates the accuracy for different ϵ values in different cases. The model maintains a median accuracy above 0.975 for all data split and ϵ cases, and actually achieves a median accuracy of 1.00 for all ϵ cases when the data is IID and equally distributed.

Finally, we compared our results against two benchmarks: the same neural network architecture with neither federation nor differential privacy and the same neural network architecture with federation but without differential privacy. Figure 4 and Figure 5 show that differential privacy with federation did not produce a large change in precision-recall or receiver operating curve (ROC) metrics.

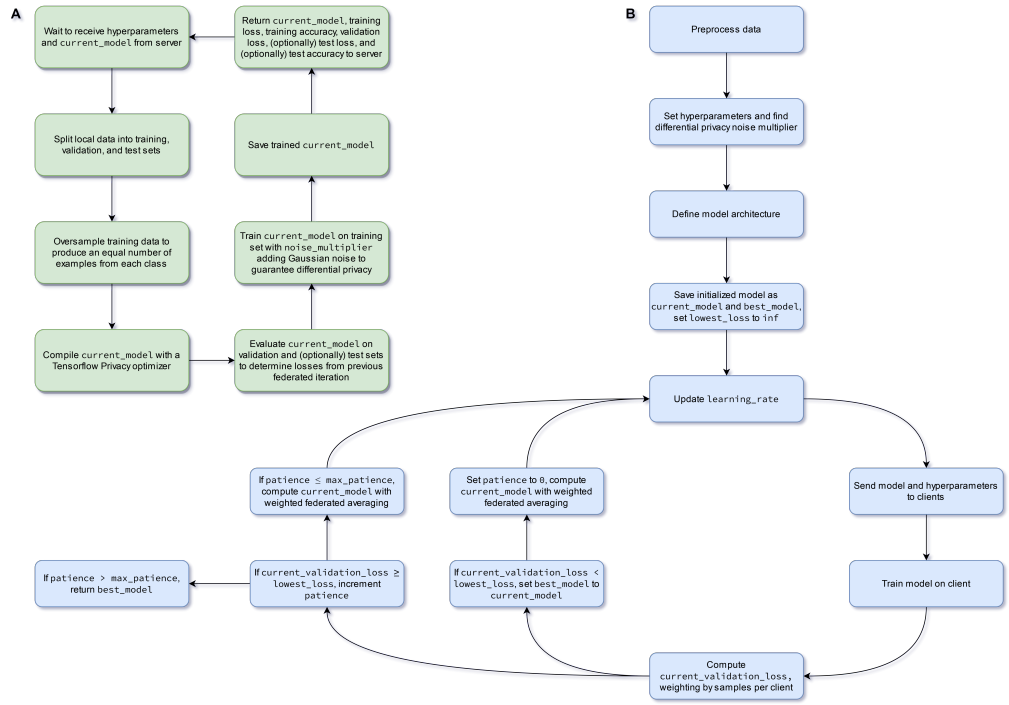

Fig. 1. Flowcharts of the client (A) and server (B) algorithms.

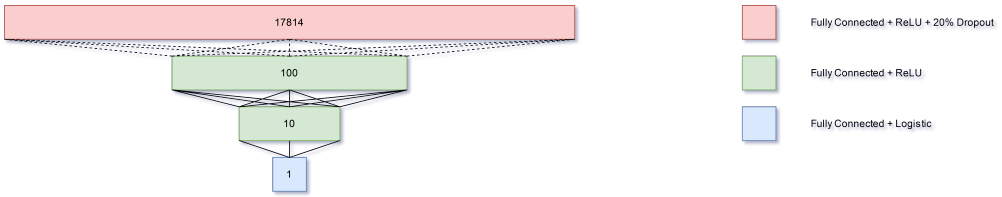

Fig. 2. Diagram of the network architecture used when testing on the breast cancer dataset.

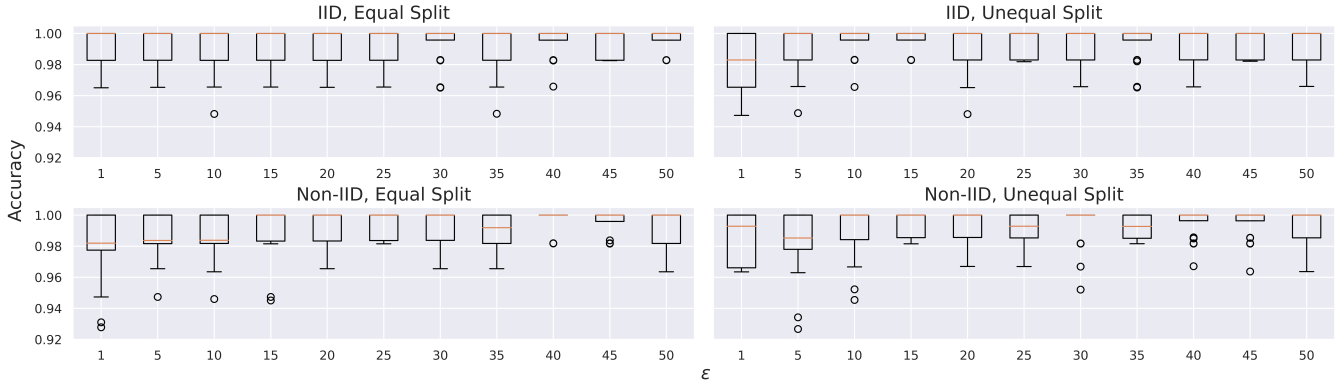

Fig. 3. Accuracy metrics on test data for different data split and ϵ cases.

IV. DISCUSSION

In this work, we intended to develop a simple framework for differentially private machine learning and test the framework on a clinical dataset. The above results indicate that our framework can produce high accuracy and near-ideal precision-recall metrics. These results, along with the ease of implementing our framework, suggest that this technique might way for clinical data scientists and bioinformatics researchers to ensure privacy when training a federated neural network. We also recognize that ϵ may not be an easy parameter to navigate for scientists with no privacy background. Therefore, in the future, combining federated learning with differential privacy with recommendation systems on privacy budget would be beneficial [18].

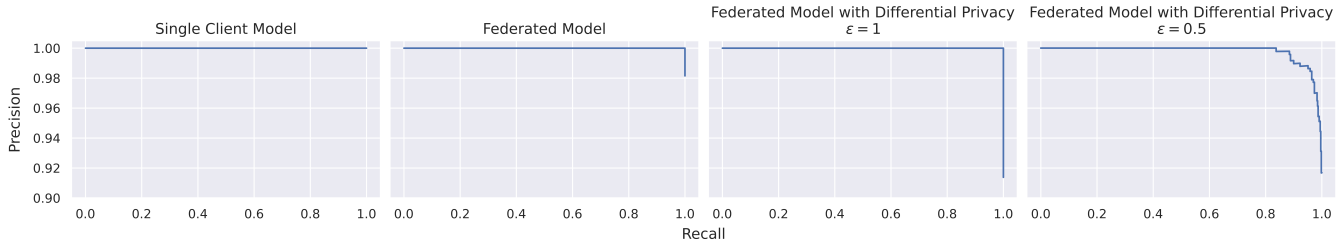

Fig. 4. Precision-recall curves for the non-federated, federated, and differentially private federated cases. The differentially private federated cases were generated with a non-IID and unequal data split since this split best represents real-world data.

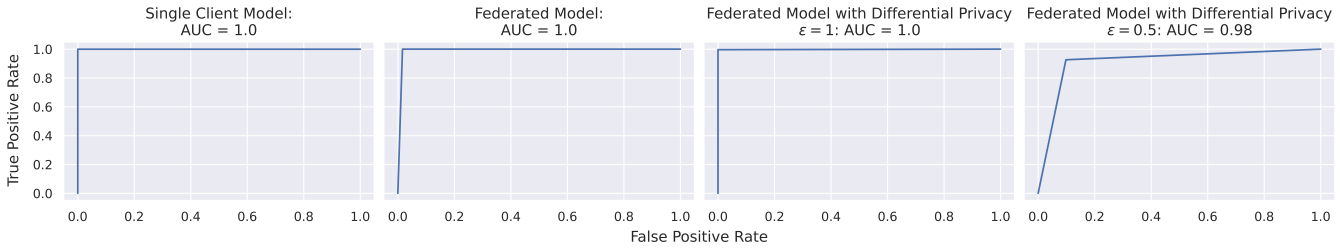

Fig. 5. Receiver operating characteristic (ROC) curves with area under curve (AUC) metrics for the non-federated, federated, and differentially private federated cases. The differentially private federated cases were generated with a non-IID and unequal data split.

Recently, other solutions to privacy-preserving federated learning have been proposed [19], [20]. These solutions employ principles of homomorphic encryption and secure multi-party computation to reduce the amount of noise required to achieve privacy. Since these solutions add less noise, they produce more accurate results. However, homomorphic encryption and secure multi-party computation migh introduce large overheads to the system due to high computational complexity [21], [22]. Our solution is promising in that it requires no special infrastructure unless the suggested deep learning framework requires GPUs. Additionally, clinical data scientists with limited knowledge of differential privacy can easily implement our solution since it uses a publicly available package, which makes using differential privacy much less difficult. In fact, implementing differential privacy without a package requires significant knowledge of differential privacy theory, which most clinical data scientists do not have. However, note that the privacy budgets in differential privacy needs to be set by a knowledgeable user to prevent potential privacy leakages.

REFERENCES

[1] M. A. Myszczynska, P. N. Ojamies, A. M. B. Lacoste, et al., Applications of machine learning to diagnosis and treatment of neurodegenerative diseases. Nat. Rev. Neurol., vol. 16, pp. 440 - 456, 2020.

[2] J. G. Richens, C. M. Lee, and S. Johri, Improving the accuracy of medical diagnosis with causal machine learning. Nat. Commun., vol. 11, no. 3923, 2020.

[3] T. I. Lee and R. A. Young, Transcriptional regulation and its misregulation in disease, Cell, vol. 152, no. 6, pp. 1237-1251, 2013.

[4] M. Mostavi, Y. Chiu, Y. Huang, et al., Convolutional neural network models for cancer type prediction based on gene expression, BMC Med. Genomics, vol. 13, no. 44, 2020.

[5] D.M. Goldenholz, H. Sun, W. Ganglberger, M.B. Westover, Sample Size Analysis for Machine Learning Clinical Validation Studies. MedRxiv, 2021

[6] E. De Cristofaro, An Overview of Privacy in Machine Learning. arXiv, 2020.

[7] European Commision, 2018 Reform of EU Data Protection Rules.

[8] J. Konečný, B. McMahan, and D. Ramage, Federated optimization: distributed optimization beyond the datacenter, arXiv, 2015.

[9] K. D. Mandl, et al., The genomics research and innovation network: creating an interoperable, federated, genomics learning system, Genet. Med., vol. 22, pp. 371-380, 2020.

[10] B. Pfitzner, N. Steckhan, and B. Arnich, Federated learning in a medical context: a systematic literature review, ACM Trans. Internet Technol., vol. 21, pp. 1 - 21, 2021.

[11] A. Narayanan and V. Shmatikov, Robust de-anonymization of large datasets (how to break anonymity of the Netflix prize dataset), arXiv, 2008.

[12] N. Carlini, C. Liu, U. Erlingsson, et al., The secret sharer: evaluation and testing unintended memorization in neural networks, arXiv, 2019.

[13] C. Dwork and A. Roth, The Algorithmic Foundations of Differential Privacy, in Foundations and Trends in Theoretical Computer Science, vol. 9, nos. 3 - 4, pp. 211 - 407.

[14] H. B. McMahan, et al., A general approach to adding differential privacy to iterative training procedures, arXiv, 2019.

[15] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, pp. 162 - 481, 2016.

[16] H. B. McMahan, E. Moore, D. Ramage, et al., Communication-efficient learning of deep networks from decentralized data, arXiv, 2017.

[17] Mendeley Data, Gene Expression Profiles of Breast Cancer.

[18] J. Lee and C. Clifton, How Much Is Enough? Choosing for Differential Privacy., n: Lai, X., Zhou, J., Li, H. (eds) Information Security. ISC 2011. Lecture Notes in Computer Science, vol 7001. Springer, Berlin, Heidelberg.

[19] M. Hao, et al., Towards efficient and privacy-preserving federated deep learning, in Conf. Rec. 2019 Int. Conf. Communications.

[20] S. Truex, et al., A hybrid approach to privacy-preserving federated learning, in Proc. 12th AMC Workshop on Art. Intel. and Sec, 2019.

[21] Z. Ni and R. Wang, Performance evaluation of secure multi-party computation on heterogenous nodes, arXiv, 2020.

[22] N. Kucherov, M. Derbayin, and M. Babenko, Homomorphic encryption methods review, in IEEE Conf. Rus. Young Res. in Elec. Eng., pp. 370 - 373, 2020.