Original Paper

Federated Learning on Clinical Benchmark Data: Performance Assessment

Geun Hyeong Lee[1], MS; Soo-Yong Shin[1,2,3], PhD

[1]Department of Digital Health, Samsung Advanced Institute for Health Sciences & Technology, Sungkyunkwan University, Seoul, Republic of Korea

[2]Big Data Research Center, Samsung Medical Center, Seoul, Republic of Korea

[3]Department of Intelligent Precision Healthcare Convergence, Sungkyunkwan University, Suwon, Republic of Korea

Corresponding Author:
Soo-Yong Shin, PhD
Department of Digital Health
Samsung Advanced Institute for Health Sciences & Technology
Sungkyunkwan University
115 Irwon-ro
Gangnam-gu
Seoul, 06355
Republic of Korea
Phone: 82 2 3410 1449
Email: sy.shin@skku.edu

Abstract

Background: Federated learning (FL) is a newly proposed machine-learning method that uses a decentralized dataset. Since data transfer is not necessary for the learning process in FL, there is a significant advantage in protecting personal privacy. Therefore, many studies are being actively conducted in the applications of FL for diverse areas.

Objective: The aim of this study was to evaluate the reliability and performance of FL using three benchmark datasets, including a clinical benchmark dataset.

Methods: To evaluate FL in a realistic setting, we implemented FL using a client-server architecture with Python. The implemented client-server version of the FL software was deployed to Amazon Web Services. Modified National Institute of Standards and Technology (MNIST), Medical Information Mart for Intensive Care-III (MIMIC-III), and electrocardiogram (ECG) datasets were used to evaluate the performance of FL. To test FL in a realistic setting, the MNIST dataset was split into 10 different clients, with one digit for each client. In addition, we conducted four different experiments according to basic, imbalanced, skewed, and a combination of imbalanced and skewed data distributions. We also compared the performance of FL to that of the state-of-the-art method with respect to in-hospital mortality using the MIMIC-III dataset. Likewise, we conducted experiments comparing basic and imbalanced data distributions using MIMIC-III and ECG data.

Results: FL on the basic MNIST dataset with 10 clients achieved an area under the receiver operating characteristic curve (AUROC) of 0.997 and an F1-score of 0.946. The experiment with the imbalanced MNIST dataset achieved an AUROC of 0.995 and an F1-score of 0.921. The experiment with the skewed MNIST dataset achieved an AUROC of 0.992 and an F1-score of 0.905. Finally, the combined imbalanced and skewed experiment achieved an AUROC of 0.990 and an F1-score of 0.891. The basic FL on in-hospital mortality using MIMIC-III data achieved an AUROC of 0.850 and an F1-score of 0.944, while the experiment with the imbalanced MIMIC-III dataset achieved an AUROC of 0.850 and an F1-score of 0.943. For ECG classification, the basic FL achieved an AUROC of 0.938 and an F1-score of 0.807, and the imbalanced ECG dataset achieved an AUROC of 0.943 and an F1-score of 0.807.

Conclusions: FL demonstrated comparative performance on different benchmark datasets. In addition, FL demonstrated reliable performance in cases where the distribution was imbalanced, skewed, and extreme, reflecting the real-life scenario in which data distributions from various hospitals are different. FL can achieve high performance while maintaining privacy protection because there is no requirement to centralize the data.

(J Med Internet Res 2020;22(10):e20891) doi: 10.2196/20891

KEYWORDS

federated learning; medical data; privacy protection; machine learning; deep learning

Introduction

Background

Traditional machine learning and deep learning require a centralized dataset to train a model. Therefore, such methods not only require data transfer to collect data from many devices, people, or institutions but also have a high computational cost because they must be trained on large datasets. When collecting privacy-sensitive data such as medical data, privacy protection is a major hurdle. Centralized databases are the main targets of hacking attacks, and therefore the risk of a data breach is severely increased [1,2]. Moreover, data centralization increases the risk of reidentification of deidentified data because of the increased data size [3].

To reduce the computational cost, Google proposed a method known as federated learning (FL), which uses the computational cores in mobile devices [4-6]. In FL, training is performed at the individual client level, and then the local weights of each client are sent to the server. The server collects the updated local weights and calculates the new global weights. Subsequently, the client downloads the global weights from the server and continues the training process. Since its first use in mobile apps [7-9], many researchers have been studying and improving FL in various fields [10-14]. In particular, studies on heterogeneity of data [4,15], robust optimization [16-20], and security methods such as differential privacy and secure multiparty computation have also been conducted with an FL approach [12,21,22]. Research on FL has also been conducted in the medical field [10,13,19]. In particular, studies have been conducted using electronic medical records and brain tumor data [23-25]. However, the application of FL to real medical data has not been sufficiently studied.

FL can be used to resolve privacy issues and mitigate the risk of a data breach in clinical information, since transfer and centralization of data are not required. Privacy protection is particularly beneficial for medical data analysis, since medical data represent some of the most sensitive types of personal data. To protect patients' privacy, deidentification methods have typically been applied [26-28]. However, data centralization is required for both deidentifying data and evaluating the risk of reidentification. If the data are centralized, the risk of a data breach is increased. Moreover, when deidentifying the dataset, the direct or indirect identifiers in the medical data must be determined. This is challenging because of the lack of clear guidelines. The Health Insurance Portability and Accountability Act in the United States provides clear deidentification guidance; it defines 18 types of protected health information to be removed [29]. However, many researchers and social activists claim that this guidance should be revised to enhance privacy protection [30]. In contrast, FL does not require the centralization of raw data. As a result, even the FL developers cannot access the raw data. Therefore, FL can solve privacy or deidentification issues that occur when using clinical data.

Objectives

The aim of this study was to assess the performance of FL on three benchmark datasets: the Modified National Institute of Standards and Technology (MNIST) dataset, Medical Information Mart for Intensive Care-III (MIMIC-III) dataset, and PhysioNet Electrocardiogram (ECG) dataset. We also verified FL in environments that simulate real-world data distributions by modifying the MNIST, MIMIC-III, and ECG datasets.

Methods

FL Code and Server

FL is supported by several open-source projects, including TensorFlow Federated in TensorFlow 2.0 [31], PySyft [32,33], and Federated AI Technology Enabler [34,35]. However, there are limitations in using these libraries. First, most of these libraries only support a single server and not a network environment. Therefore, there is no control process for data communication. Second, as a prototype, the necessary features were not fully implemented to handle a complex dataset. For future research using real clinical data from hospitals, we implemented our own client-server version of FL using Python. The implemented server code is available on the FL_Server repository [36] and the client code is available on the FL_Client repository [37]. The MNIST dataset analyzed during the current study is available in the Keras package in the TensorFlow framework. Additionally, the original code used to generate and preprocess the MIMIC-III experiment used in this study referred to the mimic3-benchmarks repository [38]. The original MIMIC-III dataset analyzed during this study is available on the PhysioNet repository [39]. The ECG dataset analyzed during this study is available on the 2017 PhysioNet/CinC Challenge website [40]. The model and environment assessed in this study refer to Hannun et al [41].

The FL server was developed using the Django framework and Python in Amazon Web Services (AWS). The server provides several application programming interfaces (APIs) for communication with a client, as shown in Table 1, and performs federated averaging (FedAVG) [4], which calculates the weighted averages. FedAVG is a widely used optimization algorithm that calculates the average value when the local weights collected from the client reach a specific level. The implemented code was deployed and managed in AWS Beanstalk, which was continuously monitored during the training process.

Table 1. Application programming interface calls provided by the server.

Method	URL	Parameter	Description	Return
GET	/round	N/A[a]	Request current round	Number
GET	/weight	N/A	Request global weight	List
PUT	/weight	List	Update local weight	N/A

[a]N/A: Not applicable.

Client

The client consists of three components. The first is the local learning component, which builds a suitable model for the dataset during the learning phase. The second is the communication component, which updates local weights according to the results of local training (the first component) on the server and downloads the global weights from the server. The third is the performance measure component, in which the performance of each client is measured using the downloaded global weights. The implemented code was deployed on an AWS EC2 instance. We used the specifications of g4dn.xlarge with the NVIDIA T4 Tensor core GPU for the Amazon instance.

Communications

Client–server communication for FL was implemented based on the process described by McMahan et al [42]. However, the implemented code exhibits some differences. The communication assumes that all clients (hospitals) are always powered (as is the case for a typical computer but not for a mobile device) and that their online status is maintained by a wired network connection. In addition, rather than selecting clients via an eligibility criterion from multiple client pools (thousands or millions), the code was implemented to manage a predefined fixed number of clients. In other words, all clients could participate in each round.

A schematic diagram of the FL client–server communication is shown in Multimedia Appendix 1. In brief, the client decides whether to participate in the current round through the API. If it has already participated (sending local weights to the server), it waits to participate in the next round. The server waits for the client's weight updates and ensures that no clients are eventually dropped. All communications are performed through the API provided by the server. The monitoring system is used to continuously observe system abnormalities.

Datasets

MNIST

The MNIST dataset, which consists of digit handwriting images, contained 70,000 samples (including 60,000 for training and 10,000 for testing). The basic model was a simple artificial neural network with an input layer, one hidden layer with 128 units with a rectified linear unit activation function, and an output layer. The hyperparameters for training were set as follows: batch size 32, maximum 1000 epochs, and early stopping. Stochastic gradient descent was used as an optimizer [43].

For FL, we used 10 individual clients to best mimic a real environment. We modified the datasets and hyperparameters of the learning algorithms. The datasets were modified considering differences in the distribution of medical data between hospitals. Hyperparameters were adjusted for training in each client. The proposed approach was evaluated on the MNIST dataset in four different experiments.

We first evaluated the basic performance of the FL. Ten clients randomly selected 600 images from the basic dataset. We continued the process for up to 500 rounds and observed the results. For the imbalanced FL experiment, each client used different sizes of randomly selected data, ranging from 1 to 600, for training (ie, one client used 36 data points and another client used 537 data points). However, other environments such as hyperparameters and the number of rounds were the same as set in the basic FL experiment. In addition, the MNIST dataset was split into single-digit groups, ranging from 0 to 9. Each of the 10 numbers was assigned to 10 different clients. Consequently, each client had a single digit instead of 10. This modified MNIST simulated an extremely skewed data distribution. Each client randomly selected 600 images from a dataset with a single digit for training. The simple artificial neural network used in the basic model was also used in these experiments. The hyperparameters were set as follows: 5 epochs and a batch size of 10. We continued the process for up to 3000 rounds and observed the results. For evaluation, a model was created with the latest updated global weights using 10,000 test samples. Finally, we conducted an extension of the modified MNIST FL that represents a skewed distribution. Each client was trained on data with an imbalanced and skewed distribution. Hence, each client was trained only on a single digit using a randomly selected sample.

MIMIC-III

The MIMIC-III dataset is a clinical dataset related to human health information, including demographics, vital signs, laboratory tests, and medications from intensive care units. MIMIC-III data were preprocessed using a state-of-the-art (SOTA) benchmark [44]. In this case, FL experiments with three individual clients were performed to predict in-hospital mortality, which is a classification problem that predicts death within the first 48 hours of an intensive care unit stay. After preprocessing the MIMIC-III dataset using the method described by Harutyunyan et al [44], the dataset contained 21,139 samples (including 17,903 for training and 3236 for testing). The basic model was a standard long short-term memory (LSTM) with reference to the benchmark [44]. The LSTM was chosen with 16 hidden units, depth 2, dropout 0.3, time step 1.0, batch size 8, and an adaptive moment estimation (ADAM) optimizer.

For FL, randomly chosen samples from the original dataset were divided into 3 datasets without duplication and assigned

to each client. This simulates having data from three different institutions. The same basic LSTM was used, and hyperparameters were set as follows: 2 epochs and a batch size of 4. We continued the process for up to 30 rounds and observed the results.

For the basic FL experiment, each client was trained on a subset of data that were split into three parts with the same data size without duplication. For the imbalanced FL experiment, all data were split into 50%, 30%, and 20% without duplication, and one subset was assigned to each client.

ECG

The 2017 PhysioNet/CinC Challenge ECG dataset was used in this study [40]. This target problem is a multiclassification problem that classifies four signals: atrial fibrillation, normal sinus rhythm, alternative rhythm, and noisy using a single short ECG signal. The total data size is 8528 single-lead ECG data points. The dataset was divided into 90% training data (7676) and 10% test data (852). For traditional learning, a convolution neural network with 34 layers based on Hannun et al [41] was applied to the ECG dataset. The hyperparameters were chosen with a batch size of 32 and an ADAM optimizer.

For FL, randomly chosen samples from the original dataset were divided into 3 datasets without duplication and assigned to 3 clients. The same model was used, and hyperparameters were set as follows: 3 epochs and a batch size of 16. We continued the process for up to 30 rounds and observed the results.

For the basic FL experiment, each client was trained on a subset of data that were split into three parts with the same data size without duplication. For the imbalanced FL experiment, all data were split into 50%, 30%, and 20% without duplication, and a subset was assigned to each client.

Evaluation

During training, we monitored the FL accuracy to evaluate performance. If the accuracy did not improve during the round, we completed the FL. Finally, we chose the best model and conducted bootstrapping to determine if there were significant differences between the experiments.

In all experiments, the area under the receiver operating characteristic curve (AUROC) score and F1-score were used as performance metrics. In addition, we evaluated the confusion matrix, precision, recall, or area under the precision recall curve (AUPRC) for comparison with the performance of the SOTA method. We calculated the 95% CIs and resampled the test set *K* times (for MNIST and ECG, *K* was 100, whereas for MIMIC-III, *K* was 10,000).

Results

MINST

The proposed approach was evaluated on the MNIST dataset for five different cases (as described in the Methods). Table 2 presents the values of the AUROC and F1-score for each case, and Multimedia Appendix 2 presents the confusion matrix for each case.

Table 2. Comparison of the experimental results for the five different MNIST cases described in the Methods.[a]

Experiments	AUROC[b] (95% CI)	F1-score (95% CI)	Precision (95% CI)	Recall (95% CI)
CML[c]	0.999 (0.999-0.999)	0.981 (0.978-0.983)	0.981 (0.972-0.989)	0.981 (0.971-0.989)
Basic FL[d]	0.997 (0.996-0.998)	0.946 (0.941-0.950)	0.945 (0.929-0.959)	0.945 (0.930-0.959)
Imbalanced FL	0.995 (0.994-0.995)	0.921 (0.917-0.927)	0.920 (0.904-0.937)	0.920 (0.903-0.937)
Skewed FL	0.992 (0.991-0.993)	0.905 (0.899-0.911)	0.905 (0.885-0.922)	0.904 (0.885-0.920)
Imbalanced and skewed FL	0.990 (0.989-0.991)	0.891 (0.884-0.896)	0.890 (0.869-0.909)	0.889 (0.868-0.908)

[a]All experiments used the same model and hyperparameters. All results are presented with a 95% CI by resampling the validation task 100 times.

[b]AUROC: area under the receiver operating characteristic curve.

[c]CML: centralized traditional machine-learning method.

[d]FL: federated learning.

Centralized machine learning (CML) is a baseline training method that was used as a control group. CML achieved an AUROC of 0.999 and an F1-score of 0.981. For basic FL, the AUROC and F1-score were 0.997 and 0.946, respectively. The initial performance of the basic FL was fairly high, with an accuracy of approximately 0.800, which continually improved (Multimedia Appendix 3A).

Imbalanced FL was designed to reflect a realistic clinical data distribution. As described in the Methods section, each client had a different training data size. Interestingly, the performance of imbalanced FL was significantly superior, with an AUROC and F1-score of 0.995 and 0.921, respectively. The initial performance was rather poor, as expected. However, after several rounds of processing, the performance rapidly improved to reach an accuracy of 0.900, after which the performance improvement was slow (Multimedia Appendix 3B).

Skewed FL assumed an extreme case. Each client had only one digit from 0 to 9, thereby simulating a situation in which each hospital has a unique subpopulation of patients without overlaps. The final AUROC and F1-score were 0.992 and 0.905, respectively. As expected, the initial performance was poor;

however, it rapidly improved after the initial rounds (Multimedia Appendix 3C).

The most extreme case was designed by combining an imbalanced and a skewed dataset. In this experiment, the AUROC and F1-score were 0.990 and 0.891, respectively. Similar to the skewed FL, the initial performance was very poor, but it rapidly improved after the initial rounds (Multimedia Appendix 3D).

Additionally, the precision and recall results for each digit class classification in each experiment are presented in Multimedia Appendices 4-8.

MIMIC-III

The proposed approach was evaluated on the MIMIC-III dataset in two different cases to compare the performance with a reported benchmark. FL experiments were performed on three individual clients. Apart from the AUROC and F1-score, we also refer to the AUPRC, which is reported in the benchmark [44]. The results are presented in Table 3 and in Multimedia Appendices 9 and 10.

SOTA performance was achieved by executing the codes provided in Harutyunyan

[38]. FL achieved an AUROC, F1-score, and AUPROC comparable with those of the SOTA method. The imbalanced FL experiment, as an extension of the basic MIMIC-III FL, also achieved AUROC, F1-score, and AUPRC comparable with those of SOTA (Table 3).

Table 3. Comparison results of MIMIC-III.[a]

Experiments	AUROC[b] (95% CI)	F1-score (95% CI)	AUPRC[c] (95% CI)	Precision (95% CI)	Recall (95% CI)
SOTA[d]	0.857 (0.837-0.875)	0.944 (0.938-0.950)	0.505 (0.451-0.558)	0.973 (0.967-0.979)	0.773 (0.907-0.927)
Basic FL[e]	0.850 (0.830-0.869)	0.944 (0.938-0.950)	0.483 (0.427-0.537)	0.975 (0.969-0.980)	0.797 (0.906-0.926)
Imbalanced FL	0.850 (0.829-0.869)	0.943 (0.937-0.949)	0.481 (0.426-0.535)	0.981 (0.976-0.986)	0.714 (0.897-0.918)

[a]All results are presented with a 95% CI by resampling 10,000 times.

[b]AUROC: area under the receiver operating characteristic curve.

[c]AUPRC: area under the precision-recall curve.

[d]SOTA: state of the art.

[e]FL: federated learning.

ECG

The proposed approach was evaluated on the ECG database using two different methods to compare the performance with a reported benchmark [41]. The results are presented in Table 4 and Multimedia Appendices 11-14.

Benchmark results were achieved using the code available on github [45]. The AUROC and F1-score of both basic and imbalanced FL were comparable with those of the benchmark (Table 4).

Table 4. Comparison results for the electrocardiogram dataset.[a]

Experiments	AUROC[b] (95% CI)	F1-score (95% CI)	Precision (95% CI)	Recall (95% CI)
Benchmark	0.954 (0.930-0.978)	0.814 (0.655-0.910)	0.820 (0.672-0.943)	0.814 (0.640-0.936)
Basic FL[c]	0.938 (0.860-0.978)	0.807 (0.651-0.931)	0.823 (0.645-0.942)	0.795 (0.660-0.925)
Imbalanced FL	0.943 (0.883-0.977)	0.807 (0.635-0.902)	0.830 (0.650-0.935)	0.788 (0.626-0.905)

[a]All results are presented with a 95% CI by resampling 100 times.

[b]AUROC: area under the receiver operating characteristic curve.

[c]FL: federated learning.

Discussion

Principal Findings

When comparing the performances of CML and FL in basic MNIST experiments, both the AUROC and F1-score were high. Unexpectedly, when using an imbalanced dataset, FL delivered good performance with only small differences (AUROC and F1-score of 0.003 and 0.035, respectively). When using a skewed dataset, FL also yielded remarkable results with respect to both the AUROC and F1-score. When comparing the confusion matrices for experiments with four datasets (ie, normal, imbalanced, skewed, and a combination of two distributions), FL showed some deterioration in performance for visually similar numbers (eg, 3 vs 5; 4 vs 9). Even in the basic MNIST classification, the performance was relatively poor in these cases. However, this problem was not related to the small sizes of the training datasets. When we monitored the size of the training datasets for each client, the dataset for class 5 was not small. Moreover, depending on the experiment, the

datasets for class 1 or 7 could be small, but superior classification performance was nevertheless achieved. This trend was maintained in the experiments with basic FL and imbalanced FL using the MIMIC-III dataset.

The FL experiments using MIMIC-III also exhibited good and competitive performance compared to a benchmark that has been trained on CML. The experimental results of in-hospital mortality using the MIMIC-III dataset, which is a well-known dataset with real clinical data, also showed good performance. This experiment was performed by splitting the randomly selected MIMIC-III data into three parts (ie, from the perspective of each institution, learning one-third of the total data). However, the performances of FL and CML were almost the same, with only a 0.005 difference in AUROC detected compared with the SOTA performance reported by Harutyunyan et al [44]. Before the experiments, we expected that the performance of FL would be slightly inferior to that of CML because FL uses a distributed dataset instead of a centralized dataset. Nevertheless, no significant difference was found in well-known evaluation indicators such as accuracy, sensitivity, precision, and F1-score (except for AUROC). Experimental results with an imbalanced dataset were very similar to those of basic FL. Therefore, an individual client may only use a small amount of data for training in FL, and the results will be similar to those achieved when all available data are used for training.

The FL experiments using ECG data also exhibited good and competitive performance compared to CML. This experiment was performed by splitting the randomly selected ECG data into three parts (ie, from the perspective of each institution, learning on one-third of the total data) with each using different data distributions.

However, the performance of FL and CML was not significantly different. Experimental results with an imbalanced dataset were very similar to those for basic FL. As shown in Multimedia Appendices 12-14, the noisy case was shown to have relatively low performance in precision and recall. This is because the data size for training was only 3% of the total size. However, the other classes performed well, such as atrial fibrillation, normal sinus rhythm, and alternative rhythm.

The performance of FL was verified using three datasets with changed data distributions: imbalanced (with disproportionally represented classes) and skewed (the distribution of the target variable was different) to imitate real-world medical data. As a result, FL was comparable to CML. During the initial rounds, only a relatively small amount of data was used on each client instead of an ensemble; therefore, the performance of FL was significantly inferior to that of CML. However, in the subsequent rounds, the performance of FL (with respect to AUROC and F1-score) became similar to that of CML. Typically, medical centers have datasets with very different distributions, and our results demonstrate that FL is suitable for real-world medical datasets without requiring data centralization.

One reason for the comparable performance of FL might be that the weight updates and the process of FedAVG could have a similar effect in mini-batches [46-48] and ensembles [49]. In FL, each client trains on a relatively small dataset and then transfers the local weights to the server. The server then collects the local weights and updates the global weights that reflect all of the data through FedAVG. Subsequently, the round is repeated to improve the global weights. Hence, individual clients are an element of a mini-batch, and FedAVG is similar to ensemble processing. When implementing FL, we used the widely known FedAVG aggregation method [5], but this does not guarantee the best choice. To solve this problem, many researchers have studied aggregation methods that can work well with abnormal distributions, robust aggregation, and efficient communication such as FedProx [16], FSVRG [17], CO-OP [18], LoAdaBoost FedAVG [19], and RFA [20]. Hyperparameter selection also requires further research.

In addition, many researchers have studied methods to reduce communication costs. First, it has been suggested to reduce the communication round through methods such as client selection, peer-to-peer, and local update [11-13]. Second, a method such as sparsification, subsampling, or quantization has been suggested to reduce the communication message size [12,13]. Third, the asynchronous update method in traditional parallel computation methods can be applied.

FL can be used to build medical artificial intelligence apps by protecting patient privacy. Although the data themselves are not exposed or gathered in the central repository in FL, these data can nevertheless be guessed during the aggregation process in the network [12]. Therefore, other privacy preservation methods such as differential privacy, secure multiparty computation, and homomorphic encryption [11,12,21,22] might be necessary to protect privacy from diverse up-to-date privacy attack methods.

In future studies, we plan to use the proposed FL methods in real clinical datasets rather than benchmark datasets. First, we will try to improve the FL framework based on the results from this study. We will then compare the performance of a breast cancer recurrence prediction model using data from two different medical centers in Korea.

Conclusions

Our experiments demonstrated the potential of FL in terms of performance and data protection, which is important for dealing with sensitive medical data. Specifically, in FL, only weights are transferred, and the participants are unaware of each other's local datasets. This can prevent personal information leaks. In addition, the proposed approach can be used to supplement existing approaches and to avoid problems that may occur during the deidentification process. The future direction of research is to use FL for actual medical data through collaborations with multiple institutions. Tasks such as expanding the client–server version of FL and improving communication will be expected to be important for the application of FL in real-world medical data with multiple institutions.

Acknowledgments

This work was supported by the Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korean government (MSIT) (2018-0-00861, Intelligent SW Technology Development for Medical Data Analysis), and by the Fourth Stage of Brain Korea 21 Project (Department of Intelligent Precision Healthcare Convergence) in 2021.

Conflicts of Interest

None declared.

Multimedia Appendix 1

Client–server communication logic.
[PDF File (Adobe PDF File), 324 KB-Multimedia Appendix 1]

Multimedia Appendix 2

Confusion matrices for the MNIST experiments.
[PDF File (Adobe PDF File), 303 KB-Multimedia Appendix 2]

Multimedia Appendix 3

Accuracy changes for each round of MNIST federated learning (FL) experiments.
[PDF File (Adobe PDF File), 353 KB-Multimedia Appendix 3]

Multimedia Appendix 4

Each digit class classification results of precision and recall in a centralized machine learning (CML) experiment using the MNIST dataset.
[PDF File (Adobe PDF File), 230 KB-Multimedia Appendix 4]

Multimedia Appendix 5

Each digit class classification result of precision and recall in the basic federated learning (FL) experiment using the MNIST dataset.
[PDF File (Adobe PDF File), 230 KB-Multimedia Appendix 5]

Multimedia Appendix 6

Each digit class classification result of precision and recall in the imbalanced federated learning (FL) experiment using the MNIST dataset.
[PDF File (Adobe PDF File), 229 KB-Multimedia Appendix 6]

Multimedia Appendix 7

Each digit class classification result of precision and recall in the skewed federated learning (FL) experiment using the MNIST dataset.
[PDF File (Adobe PDF File), 229 KB-Multimedia Appendix 7]

Multimedia Appendix 8

Each digit class classification result of precision and recall in an imbalanced and skewed federated learning (FL) experiment using the MNIST dataset.
[PDF File (Adobe PDF File), 249 KB-Multimedia Appendix 8]

Multimedia Appendix 9

Area under the receiver operating characteristic curve, which is the result of the in-hospital mortality prediction for each experiment.
[PDF File (Adobe PDF File), 255 KB-Multimedia Appendix 9]

Multimedia Appendix 10

Confusion matrix of federated learning (FL) to predict in-hospital mortality using MIMIC-III. (A) Basic FL. (B) Imbalanced FL.
[PDF File (Adobe PDF File), 227 KB-Multimedia Appendix 10]

Multimedia Appendix 11

Confusion matrices for the ECG experiments.
[PDF File (Adobe PDF File), 251 KB-Multimedia Appendix 11]

Multimedia Appendix 12

Each class classification result of precision and recall in a centralized machine learning (CML) experiment using the ECG dataset.
[PDF File (Adobe PDF File), 229 KB-Multimedia Appendix 12]

Multimedia Appendix 13

Each class classification result of precision and recall in the basic federated learning (FL) experiment using the ECG dataset.
[PDF File (Adobe PDF File), 229 KB-Multimedia Appendix 13]

Multimedia Appendix 14

Each class classification result of precision and recall in the imbalanced federated learning (FL) experiment using the ECG dataset.
[PDF File (Adobe PDF File), 229 KB-Multimedia Appendix 14]

References

1. A Study on Global Data Leaks in 2018. InfoWatch Analytics Center. URL: https://infowatch.com/sites/default/files/report/analytics/Global_Data_Breaches_2018.pdf [accessed 2020-10-13]
2. Varunkumar K, Prabakaran M, Kaurav A, Chakkaravarthy S, Thiyagarajan S, Venkatesh P. Various Database Attacks and its Prevention Techniques. Int J Eng Trends Technol 2014 Mar 25;9(11):532-536. [doi: 10.14445/22315381/ijett-v9p302]
3. Emam KE, Dankar FK, Vaillancourt R, Roffey T, Lysyk M. Evaluating the Risk of Re-identification of Patients from Hospital Prescription Records. Can J Hosp Pharm 2009 Jul;62(4):307-319 [FREE Full text] [doi: 10.4212/cjhp.v62i4.812] [Medline: 22478909]
4. Konečný J, McMahan H, Yu F, Richtárik P, Suresh A, Bacon D. Federated learning: strategies for improving communication efficiency. arXiv preprint 2016:1610.05492 [FREE Full text]
5. McMahan H, Moore E, Ramage D, Hampson S. Communication-efficient learning of deep networks from decentralized data. arXiv preprint 2016:1602.05629 [FREE Full text]
6. Brendan H. Federated Learning: Collaborative Machine Learning without Centralized Training Data. Google AI Blog. 2017. URL: https://ai.googleblog.com/2017/04/federated-learning-collaborative.html [accessed 2020-10-13]
7. Hard A, Rao K, Mathews R, Ramaswamy S, Beaufays F, Augenstein S. Federated learning for mobile keyboard prediction. arXiv preprint 2018:1811.03604 [FREE Full text]
8. Yang T, Andrew G, Eichner H, Sun H, Li W, Kong N. Applied federated learning: Improving google keyboard query suggestions. arXiv preprint 2018:1812.02903 [FREE Full text]
9. Chen M, Mathews R, Ouyang T, Beaufays F. Federated learning of out-of-vocabulary words. arXiv preprint 2019:1903.10635 [FREE Full text]
10. Rieke N, Hancox J, Li W, Milletarì F, Roth HR, Albarqouni S, et al. The future of digital health with federated learning. NPJ Digit Med 2020;3:119. [doi: 10.1038/s41746-020-00323-1] [Medline: 33015372]
11. Kairouz K, McMahan H. Advances and Open Problems in Federated Learning. arXiv preprint 2019:1912.04977 [FREE Full text]
12. Li T, Sahu AK, Talwalkar A, Smith V. Federated Learning: Challenges, Methods, and Future Directions. IEEE Signal Process Mag 2020 May;37(3):50-60. [doi: 10.1109/msp.2020.2975749]
13. Jie X, Benjamin S, Glicksberg, Chang S, Peter W, Jiang B, et al. Federated learning for healthcare informatics. arXiv preprint 2019:1911.06270 [FREE Full text]
14. Qinbin L, Zeyi W, Bingsheng H. A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection. arXiv preprint 2019:1907.09693 [FREE Full text]
15. Zhao Y, Meng L, Liangzhen L, Naveen S, Damon C, Vikas C. Federated Learning with Non-IID Data. arXiv preprint 2018:1806.00582 [FREE Full text]
16. Li T, Sahu A, Zaheer M, Sanjabi M, Talwalkar A, Smith V. Federated Optimization in Heterogeneous Networks. arXiv preprint 2018:1812.06127 [FREE Full text]
17. Konečný J, McMahan H. Federated Optimization: Distributed Machine Learning for On-Device Intelligence. arXiv preprint 2016:1610.02527 [FREE Full text]
18. Wang Y. Cooperative Machine Learning from Mobile Devices Masters Thesis. University of Alberta. 2017. URL: https://era.library.ualberta.ca/items/7d680f04-7987-45c5-b9cd-4fe43c87606f [accessed 2020-10-15]

19. Huang L, Yin Y, Fu Z, Zhang S, Deng H, Liu D. LoAdaBoost: loss-based AdaBoost federated machine learning with reduced computational complexity on IID and non-IID intensive care data. arXiv preprint 2018:1811.12629 [FREE Full text]
20. Pillutla K, Kakade S, Harchaoui Z. Robust Aggregation for Federated Learning. arXiv preprint 2019:1912.13445 [FREE Full text]
21. Abadi M, Chu A, Goodfellow I, McMahan H, Mironov I, Talwar K, et al. Deep Learning with Differential Privacy. 2016 Presented at: CCS '16: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security; 2016; Vienna. [doi: 10.1145/2976749.2978318]
22. Pfohl Stephen R, Andrew M, Katherine H. Federated and Differentially Private Learning for Electronic Health Records. arXiv preprint 2019:1911.05861 [FREE Full text]
23. Li W. Privacy-Preserving Federated Brain Tumour Segmentation. : Springer, Cham; 2019 Presented at: Machine Learning in Medical Imaging. MLMI 2019; 2019; Shenzhen. [doi: 10.1007/978-3-030-32692-0_16]
24. Sheller MJ, Reina GA, Edwards B, Martin J, Bakas S. Multi-Institutional Deep Learning Modeling Without Sharing Patient Data: A Feasibility Study on Brain Tumor Segmentation. 2019 Presented at: International MICCAI Brain Lesion Workshop; 2018; Granada p. 92-104 URL: http://europepmc.org/abstract/MED/31231720 [doi: 10.1007/978-3-030-11723-8_9]
25. Brisimi TS, Chen R, Mela T, Olshevsky A, Paschalidis IC, Shi W. Federated learning of predictive models from federated Electronic Health Records. Int J Med Inform 2018 Apr;112:59-67 [FREE Full text] [doi: 10.1016/j.ijmedinf.2018.01.007] [Medline: 29500022]
26. El Emam K, Dankar FK, Issa R, Jonker E, Amyot D, Cogo E, et al. A globally optimal k-anonymity method for the de-identification of health data. J Am Med Inform Assoc 2009;16(5):670-682 [FREE Full text] [doi: 10.1197/jamia.M3144] [Medline: 19567795]
27. Taira RK, Bui AA, Kangarloo H. Identification of patient name references within medical documents using semantic selectional restrictions. Proc AMIA Symp 2002:757-761 [FREE Full text] [Medline: 12463926]
28. Thomas SM, Mamlin B, Schadow G, McDonald C. A successful technique for removing names in pathology reports using an augmented search and replace method. Proc AMIA Symp 2002:777-781 [FREE Full text] [Medline: 12463930]
29. Guidance Regarding Methods for De-identification of Protected Health Information in Accordance with the Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule. HHS.gov. Washington, DC: U.S. Department of Health & Human Services; 1996. URL: https://www.hhs.gov/hipaa/for-professionals/privacy/special-topics/de-identification/index.html [accessed 2020-10-13]
30. Nass SJ, Levit LA, Gostin LO. Beyond the HIPAA Privacy Rule: Enhancing Privacy, Improving Health Through Research. Washington, DC: National Academies Press; 2009.
31. TensorFlow Federated: Machine Learning on Decentralized Data. TensorFlow. URL: https://www.tensorflow.org/federated [accessed 2020-10-13]
32. Ryffel T, Trask A, Dahl M, Wagner B, Mancuso J, Rueckert D. A generic framework for privacy preserving deep learning. arXiv preprint 2018:1811.04017 [FREE Full text]
33. OpenMined. URL: https://www.openmined.org/ [accessed 2020-10-13]
34. An Industrial Grade Federated Learning Framework. FATE. URL: https://fate.fedai.org/ [accessed 2020-10-13]
35. FederatedAI. GitHub. URL: https://github.com/FederatedAI/FATE [accessed 2020-10-13]
36. FL_Server. GitHub. URL: https://github.com/bmiskkuedu/FL_Server [accessed 2020-10-13]
37. FL_Client. GitHub. URL: https://github.com/bmiskkuedu/FL_Client [accessed 2020-10-13]
38. Harutyunyan H. MIMIC-III benchmarks. GitHub. 2018. URL: https://github.com/YerevaNN/mimic3-benchmarks [accessed 2020-10-13]
39. Johnson A, Pollard T, Mark R. MIMIC-III Clinical Database. Physionet. URL: https://physionet.org/content/mimiciii/1.4/ [accessed 2020-10-13]
40. AF Classification from a Short Single Lead ECG Recording - The PhysioNet Computing in Cardiology Challenge 2017. PhysioNet. 2017. URL: https://physionet.org/content/challenge-2017/1.0.0/ [accessed 2020-10-13]
41. Hannun AY, Rajpurkar P, Haghpanahi M, Tison GH, Bourn C, Turakhia MP, et al. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nat Med 2019 Jan;25(1):65-69 [FREE Full text] [doi: 10.1038/s41591-018-0268-3] [Medline: 30617320]
42. Bonawitz K, Eichner H, Grieskamp W, Huba D, Ingerman A, Ivanov V. Towards Federated Learning at Scale: System Design. arXiv preprint 2019:1902.01046 [FREE Full text]
43. Ruder S. An overview of gradient descent optimization algorithms. arXiv preprint 2016:1609.04747 [FREE Full text]
44. Harutyunyan H, Khachatrian H, Kale DC, Ver Steeg G, Galstyan A. Multitask learning and benchmarking with clinical time series data. Sci Data 2019 Jun 17;6(1):96. [doi: 10.1038/s41597-019-0103-9] [Medline: 31209213]
45. Awni ECG. GitHub. URL: https://github.com/awni/ecg [accessed 2020-10-13]
46. Li M, Zhang T, Chen Y, Smola AJ. Efficient mini-batch training for stochastic optimization. 2014 Presented at: KDD '14: The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2014; New York p. 661-670. [doi: 10.1145/2623330.2623612]

47. Keskar N, Mudigere D, Nocedal J, Smelyanskiy M, Tang P. On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima. arXiv preprint 2016:1609.04836 [FREE Full text]
48. Masters D, Luschi C. Revisiting Small Batch Training for Deep Neural Networks. arXiv preprint 2018:1804.07612 [FREE Full text]
49. Dietterich T. Ensemble Methods in Machine Learning. : Springer; 2000 Presented at: International Workshop on Multiple Classifier Systems. MCS 2000; 2000; Berlin, Heidelberg. [doi: 10.1007/3-540-45014-9_1]

Abbreviations

ADAM: adaptive moment estimation
API: application programming interface
AUPRC: area under the precision-recall curve
AUROC: area under the receiver operating characteristic curve
AWS: Amazon Web Services
CML: centralized machine learning
ECG: electrocardiogram
FedAVG: federated averaging
FL: federated learning
LSTM: long short-term memory
MIMIC-III: Medical Information Mart for Intensive Care III
MNIST: Modified National Institute of Standards and Technology
SOTA: state of the art

Edited by G Eysenbach; submitted 01.06.20; peer-reviewed by JH Yoon, M Abdalla; comments to author 22.06.20; revised version received 16.08.20; accepted 02.10.20; published 26.10.20

Please cite as:
Lee GH, Shin SY
Federated Learning on Clinical Benchmark Data: Performance Assessment
J Med Internet Res 2020;22(10):e20891
URL: http://www.jmir.org/2020/10/e20891/
doi: 10.2196/20891
PMID: 33104011