Preserving Patient Privacy while Training a Predictive Model of In-hospital Mortality

Pulkit Sharma, Farah E Shamout, David A Clifton
Department of Engineering Science
University of Oxford
{pulkit.sharma, farah.shamout, david.clifton}@eng.ox.ac.uk

Abstract

Machine learning models can be used for pattern recognition in medical data in order to improve patient outcomes, such as the prediction of in-hospital mortality. Deep learning models, in particular, require large amounts of data for model training. However, the data is often collected at different hospitals and sharing is restricted due to patient privacy concerns. In this paper, we aimed to demonstrate the potential of distributed training in achieving state-of-the-art performance while maintaining data privacy. Our results show that training the model in the federated learning framework leads to comparable performance to the traditional centralised setting. We also suggest several considerations for the success of such frameworks in future work.

1 Introduction

In recent years, deep learning has achieved state-of-the-art performance that surpasses human level performance across various tasks in computer vision, speech recognition, and natural language processing [1, 2, 3]. One of the key drivers of this success is the availability of large amounts of data to train the deep learning models. In applications where privacy is not a concern, datasets are usually curated in a central location and may even be publicly available for further research [4].

Although we are currently in the era of data-rich medicine, clinical data often exists in isolation due to several privacy reasons. Some of those issues include potential invasion of privacy, data misuse, or patient discrimination. Basic data anonymisation techniques, such as those listed in the guide published by the Personal Data Protection Commission in Singapore [5], are also at risk of de-anonymisation through reverse engineering. Due to the sensitivity of medical data, most existing clinical databases are curated for private use only [6], and only a few are publicly available [7]. This hinders the progress of developing deep learning frameworks using diverse datasets in order to improve patient outcomes.

In order to address such concerns, various governments have made initiatives to strengthen data privacy and security [8], such as the General Data Protection Regulation (GDPR) that was enforced in 2018 by the European Union. Privacy is also a key value of the Montreal Declaration for a Responsible Development of Artificial Intelligence (2018) [9]. These regulations make the use of traditional centralised machine learning models more challenging, where the data is collected from multiple parties and then processed on a central server.

In this paper, we test a privacy preserving framework for the task of in-hospital mortality prediction amongst patients admitted to the intensive care unit (ICU), as in related works [10]. We use federated learning (FL) [11, 12, 13], which involves training a global machine learning model using vital-signs data distributed across remote devices (e.g. in various hospitals), without having to share the data with a centralised server. The FL framework provides a solution for all the issues related to privacy,

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

locality and ownership of healthcare data [14]. Based on the results, we discuss the strengths and challenges that are unique to the potential of FL within healthcare and offer recommendations for relevant policy-making parties.

2 Problem Formulation

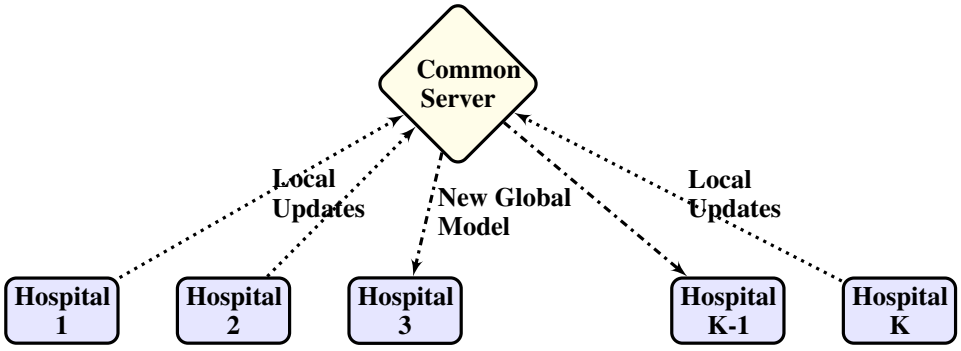

Figure 1: Schematic of the federated learning (FL) framework adopted for the in-hospital mortality prediction task. In order to preserve the privacy of clinical data, the model is trained in a distributed fashion: The hospitals periodically communicate the local updates with a common server to learn a global model. The common server incorporates the updates and sends back the parameters of the updated global model.

We aim to securely train a deep learning model, referred to thereafter as the global model, that can predict in-hospital mortality for ICU patients. The mortality prediction task is formulated as a binary classification problem, where the label indicates patient's death before hospital discharge. The block diagram depicting the proposed approach is shown in Figure 1. The data stored locally at different hospitals is used to estimate the local (model) updates which are communicated to a common server without sending the data. The common server then employs these updates to derive a better global model which is further used during testing at different hospitals.

To perform the secured model training, we assume that a set of hospitals $\mathcal{H} = \{\mathcal{H}_1, \ldots, \mathcal{H}_K\}$ participate in training a global machine learning model (also referred to as the training federation) with a common server S coordinating between them. Each hospital $\mathcal{H}_k$ stores its data $D_k = \{(x_1^k, y_1^k), (x_2^k, y_2^k), \ldots (x_{|D_k|}^k, y_{|D_k|}^k)\}$ locally and does not share it with S. In this work, x_i^k and y_i^k represent the data sample i and its corresponding label, respectively, at hospital k. $|D_k|$ represents the total number of data samples stored at hospital k. Those assumptions make the FL a suitable choice for machine learning training using clinical data in hospitals where data security and privacy are of utmost importance.

The training procedure relies on efficient communication between the central server S and the distributed hospitals. The objective is to minimise the global loss function to estimate the global (G) parameters $\mathbf{w}^G \in \mathbb{R}^d$ of the global model without directly accessing the data stored at the hospitals, where d represents the number of parameters of the model.

The common server S first broadcasts the global model $\mathbf{w}_t^G$ to a subset of non-identically-distributed hospitals $\mathcal{H}^- \subset \mathcal{H}$ at time t. A local loss is then optimised over the local data D_k at every node k in $\mathcal{H}^-$ to estimate the local parameter vector $\mathbf{w}_{t+1}^k$. The various hospitals $\mathcal{H}^-$ then send their computed model parameters to the common server S, which aggregates the findings to estimate $\mathbf{w}^G$ of the final model. This consists of a weighted mean of all the local models to obtain an updated global model $\mathbf{w}_{t+1}^G$ as:

$$\mathbf{w}_{t+1}^G = \sum_{k=1}^{|\mathcal{H}^-|} p_{t+1}^k \mathbf{w}_{t+1}^k. \tag{1}$$

Algorithm 1 A summary of the FL framework to compute the global model at common server using data stored locally at different hospitals. Functions ModelUpdate and LocalTestAccuracy are executed locally on the k^{th} hospital. Variable a_t is an estimation of the global accuracy at time t.

Input: $\mathbf{w}_t^G$ a_t
Output: $\mathbf{w}_{t+1}^G$ a_{t+1}

broadcast $\mathbf{w}_t^G$ to hospitals in $\mathcal{H}^-$
for each hospital $k \in \mathcal{H}^-$ do
    $\mathbf{w}_{t+1}^k \leftarrow ModelUpdate(k, \mathbf{w}_t^G)$
    $\mathbf{p}_{t+1}^k \leftarrow \frac{|D_k|}{\sum_k |D_k|}$
end for
$\tilde{\mathbf{w}}_{t+1}^G \leftarrow \sum_{k=1}^{|\mathcal{H}^-|} p_{t+1}^k \mathbf{w}_{t+1}^k$
for each hospital $k \in \mathcal{H}$ do
    $a_{t+1}^k \leftarrow LocalTestAccuracy(k, \tilde{\mathbf{w}}_{t+1}^G)$
end for
$a_{t+1} \leftarrow weighted\ average\ of\ a_{t+1}^k\ \forall\ k \in \mathcal{H}$
while $a_{t+1} < a_t$
    $\tilde{\mathbf{w}}_{t+1}^G \leftarrow \mathbf{w}_t^G$
    $a_{t+1} \leftarrow a_t$
end while
$\mathbf{w}_{t+1}^G \leftarrow \tilde{\mathbf{w}}_{t+1}^G$

For the sake of simplicity, we drop the time dimension and consider only one time instance as:

$$\mathbf{w}^G = \sum_{k=1}^{|\mathcal{H}^-|} p^k \mathbf{w}^k, \tag{2}$$

where $p^k \in [0, 1]$ represents the weights associated with each hospital k such that $\sum_{k=1}^{|\mathcal{H}^-|} p^k = 1$. The term p^k specifies the relative impact of each hospital and we adopt a general choice of $p^k = \frac{|D_k|}{D}$, where $D = \sum_k |D_k|$ is the total number of samples across the k hospitals. It is worth noting that incorporating the information about data at different hospitals in the hospital-dependent value p^k is important for computing an efficient federated model.

We iterate through the described training procedure across different hospitals until convergence or some stopping criterion. Algorithm 1 describes the proposed FL setup, where the goal is to estimate model parameters at time $t+1$; i.e. $\mathbf{w}_{t+1}^G$ given $\mathbf{w}_t^G$. It has to be noted that at each step the model can be updated locally at each hospital in $k \in \mathcal{H}^-$. However this model is evaluated using the test data at all the hospitals; i.e. $k \in \mathcal{H}$. Accuracy a_t (on test data $k \in \mathcal{H}$) can be used as metric of evaluation while updating the global model where a model is updated if and only if $a_{t+1} \geq a_t$.

3 Experimentation

3.1 Dataset

The proposed FL framework is evaluated for the task of predicting in-hospital mortality using data obtained from the publicly available MIMIC-III database [7]. The total number of patient admissions for the benchmark mortality prediction task were 21,138, where the variables collected in the first 48-hour window were used as input features [15]. The variables included vital-sign data, such as heart rate and temperature, and details can be found in [7]. The number of time stamped observations in first 48 hours varied per patient episode. Hence, we used hand-engineered features as described in [16]. For each vital sign, six different sample statistic features were computed on seven time-series: maximum, minimum, mean, standard deviation, skew and number of measurements. The seven time-series included the full time-series, the first/last 10% of time, first/last 25% of time, first/last 50% of time. The features were then scaled into the range [-1,1] before feeding them into the classifier.

To mimic the FL framework described in Section 2, we distributed the training and testing data amongst virtual workers using the coMind federate learning toolkit [17]. This virtual set up mirrored the realistic scenario such that the data is located physically across different hospitals.

Table 1: Comparison of the proposed FL methods with the standard setup. LR-ORG/MLP-ORG and LR-FL/MLP-FL represents logistic regression/multi-layer perceptron classifier trained in normal and FL setup.

	LR-ORG	LR-FL	MLP-ORG	MLP-FL
AUROC	0.8152	0.7890	0.7925	0.7769
AUPRC	0.4030	0.3659	0.3900	0.3504

3.2 Results

The primary metrics that we used for evaluation are the area under the receiver operator characteristic curve (AUROC) and area under the precision-recall curve (AUPRC). Logistic regression (LR) and a multi-layer perceptron (MLP) classifier were employed in the traditional training procedure such that the training data exists at a central server, denoted as ORG, and in the FL setup.

The architecture of MLP here was modelled based on [18] which consists of a single layer with 50 nodes between the input and output. The networks were trained with a cross-entropy loss and Adam optimiser [19] using a batch size of 8 for 100 epochs.

The FL setup was simulated with a common virtual server and two local workers (clients/hospitals) with around 1,700 rounds of communication before deriving the final model. The train-test split was the same as in [15] and we split both the train and test data equally among the two local workers.

The results in form of AUROC and AUPRC are shown in Table 1, where LR-ORG/MLP-ORG and LR-FL/MLP-FL represent the classifiers trained in the traditional and FL setups. It can be observed that there is a slight decrease in performance in the case of the FL setup, as opposed to the traditional setting.

4 Summary and Future Directions

The performance of the models trained with FL is comparable to training with centralised data. This demonstrates the potential of distributed learning in training machine learning models for clinical tasks. The decline in performance needs further investigation, which could involve further experimentation with the hyper-parameters. While there may be a potential trade-off between privacy and performance, it is evident that FL-based models can generalise for other outcome prediction tasks in healthcare, since it performs well for the in-hospital mortality prediction task.

Improving data privacy can have a trickle-down effect on other ethical related issues, such as fairness and diversity. With improved data privacy, data owners would be more comfortable in utilising their data for machine learning research by not sharing the data directly. As machine learning models are trained using larger and potentially more diverse datasets, the performance of the models would also improve. Better performing models can then be deployed in clinical trials and eventually improve patient outcomes.

Achieving such long term benefits requires the participation of various stakeholders. First, research efforts must focus on mitigating the limitations of FL and related frameworks. For example, the inaccessibility of the data by the central server compromises model interpretability, which highly relies on examining the original inputs and their respective outputs. This requires further discussion between the parties involved with model development; i.e. the researchers and participating data owners, such as hospitals. Secondly, the tested framework requires setting up an effective infrastructure across hospitals and research institutions. This infrastructure depends on several resources that must be allocated by involved parties, which include communication, regulation, and funding.

Privacy-preserving models are vital for the development of machine learning research in the healthcare domain. Future work should focus on improving the performance of those frameworks. Improved performance is also a key consequence of training robust models using diversified and large datasets, while protecting the privacy of the patient - the most important stakeholder.

References

[1] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[2] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, Nov. 2012.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[4] Awesomedata. awesomedata/awesome-public-datasets, Aug 2019.

[5] Personal Data Protection Commission Singapore. Guides to basic data anonymization techniques. Technical Report January, 2018.

[6] Benjamin Shickel, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis. *IEEE Journal of Biomedical and Health Informatics*, pages 1–16, 2017.

[7] Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3, 2016.

[8] Vito Walter Anelli, Yashar Deldjoo, Tommaso Di Noia, and Antonio Ferrara. Towards effective device-aware federated learning, 2019.

[9] Montreal declaration for responsible ai, 2018.

[10] Farah E Shamout, Tingting Zhu, Pulkit Sharma, Peter J Watkinson, and David A Clifton. Deep interpretable early warning system for the detection of clinical deterioration. *IEEE journal of biomedical and health informatics*, 2019.

[11] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1273–1282, 2017.

[12] Jakub Konečný, Brendan McMahan, and Daniel Ramage. Federated optimization:distributed optimization beyond the datacenter, 2015.

[13] Jakub Konečný, H. Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence, 2016.

[14] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečný, Stefano Mazzocchi, H. Brendan McMahan, Timon Van Overveldt, David Petrou, Daniel Ramage, and Jason Roselander. Towards federated learning at scale: System design, 2019.

[15] Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data, 2017.

[16] Zachary Chase Lipton, David C. Kale, Charles Elkan, and Randall C. Wetzel. Learning to diagnose with LSTM recurrent neural networks. In *4th International Conference on Learning Representations (ICLR)*, 2016.

[17] coMindOrg. comindorg/federated-averaging-tutorials, Mar 2019.

[18] Anna-Lena Popkes, Hiske Overweg, Ari Ercole, Yingzhen Li, José Miguel Hernández-Lobato, Yordan Zaykov, and Cheng Zhang. Interpretable outcome prediction with sparse bayesian neural networks in intensive care, 2019.

[19] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.