Towards Federated COVID-19 Vaccine Side Effect Prediction

Jiaqi Wang[1⋆], Cheng Qian[2], Suhan Cui[1], Lucas Glass[2], and Fenglong Ma[1]✉

[1] College of Information Sciences and Technology, The Pennsylvania State University
{jqwang,sxc6192,fenglong}@psu.edu
[2] Analytics Center of Excellence, IQVIA
alextoqc@gmail.com, Lucas.Glass@iqvia.com

Abstract. We propose FedCovid, a new federated learning system based on electronic health records (EHR), to predict COVID-19 vaccination side effects. Federated learning allows diverse data owners to work together to train machine learning models without sharing data, ensuring the privacy of EHR data. However, because EHR data is unique, directly using existing federated learning models may fail. The EHR data is diverse, with numerical and categorical characteristics as well as consecutive visits. Furthermore, each client's data size is unequal, and the data labels are skewed due to the small number of patients that experience serious side effects. We present an adaptive approach to fuse heterogeneous EHR data and apply data augmentation techniques working with a margin loss to overcome the data imbalance issue in the client model training to address both challenges simultaneously in FedCovid. We recommend that when the server is updated, the data size of each client be taken into account to lessen the impact of clients with small data volumes. Finally, in order to train a stable and successful federated learning model, we suggest a new ordinal training technique. Experiments on a real-world dataset reveal that the suggested model is effective at predicting COVID-19 vaccination adverse effects. The performance increases by 14.35%, 17.81%, and 129.36% on the F1 score, Cohen's Kappa, and PR-AUC, respectively, compared with the best baseline. [3]

Keywords: COVID-19 vaccination · Side effect prediction · Federated learning · Electronic health records

1 Introduction

The COVID-19 pandemic has led to 486,761,597 confirmed cases and 6,142,735 deaths globally as of April 1, 2022[4]. One of the preventive measures to reduce the chances of infection is getting vaccinated. There are three widely-applied

⋆ This work was done when Jiaqi Wang interned at IQVIA.
[3] The source code of the proposed FedCovid is available at https://github.com/JackqqWang/FedCovid.git
[4] https://covid19.who.int/

COVID-19 vaccines, i.e., Moderna, Pfizer-BioNTech, and Johnson & Johnson's Janssen. According to a recent report in [15], during September 22, 2021 to February 6, 2022, approximately 82.6 million U.S. residents aged $\leq$ 18 years had received COVID-19 vaccine doses. Although COVID-19 vaccines are safe and effective, some people may still have a few side effects after receiving the vaccines [31,3,25]. The common side effects include, but are not limited to, swelling, redness, fever, headache, tiredness, muscle pain, chills, and nausea. In fact, these symptoms are normal and are signs that the body is building immunity. A small number of people may experience serious health events after the COVID-19 vaccination, such as anaphylaxis [30], thrombosis with thrombocytopenia syndrome (TTS) [28], myocarditis and pericarditis [9], and Guillain-Barre syndrome (GBS) [27]. These rare yet serious side effects may cause death. Therefore, a challenging but practical question arises: *Is it possible to predict whether people will have COVID-19 vaccine side effects after their vaccination?*

To answer this question, the first challenge that we may face is what kinds of data can be used to learn the vaccine side effect predictor. Existing work shows that the side effects of the COVID-19 vaccine may be related to gender and underline diseases [10]. The Centers for Disease Control and Prevention (CDC) also points out that women over the age of 30-49 years should be aware of the increased risk of the TTS side effect[5]. Thus, the data used for predicting vaccine side effects should contain patient demographics and historical disease information. Fortunately, electronic health records (EHR) consist of patient demographics, historical visit records, and corresponding laboratory results, which have been commonly used for the medical predictive modeling task in recent years [5,20,21,19]. Each visit record includes multiple diagnosis codes, procedure codes, and medication codes. Each diagnosis code represents a disease, a symptom, or an abnormal finding. Therefore, these characteristics make EHR data suitable for being used for predicting the COVID-19 vaccine side effects.

Due to the privacy issue and the high sensitivity of EHR data, hospitals, health insurance companies, or medical research institutes usually do not allow others to share them with others. The second challenging issue is how to train an accurate predictive model when stakeholders do not share their own data. Towards this end, we propose to use an advanced technique in the machine learning field, i.e., *federated learning* (FL), which enables different clients to work cooperatively to learn a global model by only sharing model parameters, instead of sharing data with others [24,37]. In our case, a local client, e.g., a hospital, a research institute, or a data center in one state, trains its own model with the local patient EHR data. After that, selected clients only need to upload their model parameters to the server for the global model aggregation. After aggregation, the server will distribute the global model back to active clients. The active clients will then train their local models starting from the global model they received with their local data. During this iterative process, local clients collaborate to maintain a global model by acquiring concealed information

[5] `https://www.cdc.gov/coronavirus/2019-ncov/vaccines/safety/adverse-events.html`

from each client while maintaining data privacy. Although federated learning approaches such as FedAvg [24] have shown their effectiveness on the image datasets such as MNIST[6], CIFAR-10[7], and CIFAR-100[8], they may not work well on the EHR data.

First, **EHR data are heterogeneous**. As we mentioned before, EHR data contains not only demographic information but also visit information. The static demographics include discrete gender and numerical age. The visits are time-ordered sequential data, and each visit consists of a set of unordered discrete codes. Thus, how to automatically integrate these types of data is a challenge. Second, federated learning prevents each client from uploading its EHR data to the central server, and only allows each client to solely update the prediction model with its own data. However, **the size of EHR data stored for each client is unequal**. In other words, the EHR data are not distributed in a uniform and independent manner among customers. Each state in the United States is treated as a data center or client in our work. The amount of EHR data taken from each state varies due to the uneven distribution of the population throughout the 50 states. Clients with limited data may end up with an over-fitted model. Aggregating these "poor" client models on the server side may jeopardize the learning of the global predictive model. Third, our goal is to forecast the side effects of the COVID-19 vaccine. The patients who had side effects are labeled as positive cases, whereas those who did not are labeled as negative cases. According to existing research [31,3,25], only a small percentage of persons have side effects. This means that the number of positive cases should be smaller than that of negative cases in the real world. As a result, **the EHR data used for training the predictive model are imbalanced**.

To address these challenges simultaneously, in this paper, we propose a novel **Fed**erated learning framework (named `FedCovid`) for predicting **COVID**-19 vaccine side effects using EHR data extracted from the database of IQVIA[9]. In particular, to address the heterogeneous data challenge, we first map each type of data to a latent representation and then use the proposed adaptive fusion mechanism to obtain the aggregated patient representation. Moreover, to tackle the data imbalance issue, we propose to use the data augmentation technique to increase the number of positive patient representations and incorporate the metric or contrastive learning loss into the client model training. Finally, we designed an ordinary training strategy to deal with the Non-IID issue. In contrast to existing federated learning models such as FedAvg [24] to treat each client equally, we classify clients into two categories according to the amount of EHR data they have.We first train the clients with a larger size to obtain an initialized global model. After the global model becomes stable, we then allow the clients with a smaller amount of data to participate in the model training. In addition, we take the size of clients into consideration when aggregating the global model.

[6] `http://yann.lecun.com/exdb/mnist/`
[7] `https://www.cs.toronto.edu/~kriz/cifar.html`
[8] `https://www.cs.toronto.edu/~kriz/cifar.html`
[9] `https://www.iqvia.com/`

To sum up, the contributions of this work are listed as follows:

- To the best of our knowledge, we are the first to investigate the feasibility of using advanced machine learning techniques to predict COVID-19 vaccine side effects with EHR data.
- We propose a novel federated learning framework `FedCovid` to protect EHR data privacy, fuse different types of EHR data, handle the imbalance data issue, and tackle the Non-IID data distribution challenge simultaneously.
- We conducted extensive experiments to show the effectiveness and efficiency of the proposed framework compared with state-of-the-art baselines. Furthermore, we provide comprehensive results for hyperparameter exploration, ablation study, and convergence analysis.

2 Related Work

Since COVID-19 was declared as a worldwide pandemic, artificial intelligence (AI) has been applied to conduct related research, such as developing novel diagnostic approaches [34], drug discovery [35], spread monitor [14], and e-pharmacy supply chain optimization [23]. There are also several reviews [26,2,1] summarizing the roles of AI during the fight with COVID-19.

There are also several research studies applying federated learning (FL) techniques on COVID-19 related topics. In [13], the authors applied a GAN-augmented FL for COVID-image segmentation. In [8], a FL model was proposed to predict the future oxygen requirements of symptomatic patients with COVID-19 based on chest X-ray images. In [32], a model was trained using dispersed raw clinical data to predict death in COVID-19-infected hospitalized patients.

Current COVID-19-related FL research, however, has a number of limitations. (1) The majority of FL frameworks and models are designed for medical picture data solely, ignoring heterogeneous EHR data. (2) In several previous research, the present centralized machine learning approaches are simply embedded into the FL architecture. Such a simplistic mix overlooks the distributed paradigm's merits and limitations. (3) To our knowledge, no published research effort has investigated the COVID-19 vaccine side effect prediction utilizing distributed EHR data in a FL scenario, specifically to address the problems of imbalanced data and Non-IID concerns in the real-world setting.

3 COVID-19 Vaccine EHR Data

3.1 Dataset Overview

We extracted the EHR data from the health insurance claims database of IQVIA. Similar to other types of data [38,36], EHR data are **heterogeneous**, which include patients' age, gender, zip code, diagnosis codes within each visit, the vaccine brand, and a binary label of the side effects. In this extracted dataset, there are 6,526 patients with COVID-19 vaccinated. 1,097 of them have side

Table 1: Data statistics of the extracted EHR dataset.

Patient Count	6,526	Moderna	3,355
Positive Patient Count	1,097	Pfizer-BioNTech	2,159
Negative Patient Count	5,429	Janssen	1,012
Male	1,761	ICD Code Count	803
Female	4,765	State Count	29

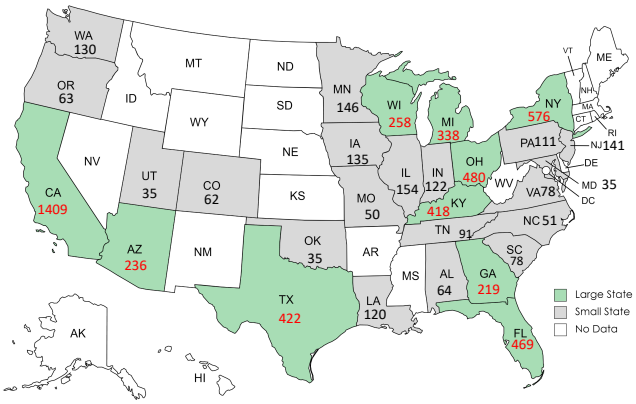

Fig. 1: Patient geographical distribution across states. The states marked in green color are the ones with the total number of data larger than 200.

effects who are labeled as 1, and 5429 of them have no side effects who are labeled as 0 on the record. The **imbalanced** label ratio is around 1:5 (# of positive labels : # of negative labels). The vaccine brands include Moderna, Pfizer-BioNTech, and Johnson & Johnson's Janssen. The number of patients with the brands of vaccines is 3,355, 2,159, and 1,012, respectively. The basic statistic of the dataset is shown in Table 1.

The dataset also provides geographic visualization via the zip codes. Based on the zip code information, patients are from 29 states. However, the **data distribution of states** is extremely **unequal**. There are 1,409 patients from CA, while there are only 35 patients in MD, OK, and UT in the dataset. We highlight the 10 states with more than 200 patients in green and visualize the data with geological information in Figure 1. There are 19 states where the data is less than 200 patients, which raises a **small data challenge**. When we do global model aggregation for federated learning, how we treat the models trained by the small clients appropriately will be a new practical challenge for the COVID-19 vaccine side effect prediction task.

3.2 Training and Test Data Construction

As it is not a benchmark dataset with a well-established training and test split, we will introduce how we create our training and test datasets. To keep as much

Table 2: Training and testing data statistics.

Training		Testing	
# Patient	5,006	# Patient	1,520
# Positive Patient	879	# Positive Patient	218
# Negative Patient	4,127	# Negative Patient	1,302

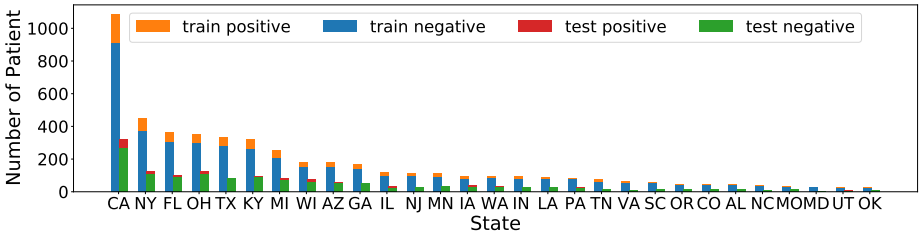

Fig. 2: Training and test data label ratio for each state.

of the original information as possible, we split the data based on the geological information and label distribution. To preserve data privacy, we treat each state as an individual client in our framework. For each state, we randomly sample 80% data for training and 20% data for testing on positive labels and negative labels accordingly.

After that, we keep the training data of each state locally for each client to train the local model. We merge the test data from each state into a large dataset for testing the performance of the global model. In such a way, we preserve the data privacy for each state without sharing patients' data for model training. On the other hand, we construct the training and test data while preserving as much of the geologically similar label distribution as possible.The basic statistics of the training and test data are shown in Table 2. The label ratio of training and test data from different states is visualized in Figure 2.

4 Task & Notation

In this paper, we focus on a real-world application scenario where each state holds its patients' EHR data and cooperates with other states' data to obtain a COVID-19 vaccine side effect prediction model. Assume that we have K clients or state data centers, and the EHR dataset on the k-th client is denoted as $\mathcal{D}_k = \{X_i^k, y_i^k\}_{i=1}^{N_k}$, where X_i^k represents the EHR data of the i-th patient in the k-th client, y_i^k is the corresponding binary label, and N_k is the number of patient EHR data stored in the k-th client.

As we mentioned before, EHR data are heterogeneous, and $X_i^k := \{Z_i^k; a_i^k; V_i^k\}$, where Z_i^k is the categorical feature set including gender g_i^k and vaccine brand b_i^k, a_i^k is the numerical feature age, and V_i^k is the time-ordered visit information. $V_i^k = \{x_{i,1}^k, x_{1,2}^k, \cdots, x_{i,M_i}^k\}$, where $x_{i,m}^k$ represents the medical

Table 3: Notations table.

Symbol	Definition and description
$\mathcal{D}_k$	The set of dataset on the k-th client
X_i^k	The EHR record of patient i at client k
$y_i^k \in \{0, 1\}$	Vaccine side effect label of patient i on the k-th client
g_i^k	Gender of patient i on the the k-th client
a_i^k	Age of patient i on the the k-th client
b_i^k	Vaccine brand information of patient i on the k-th client
$x_{i,m}^k$	Medical code of patient i at visit m on the the k-th client
K	The number of clients
B	The number of active/selected clients

code set that patient i received at visit m, and M_i denotes the number of visits of patient i.

There are 29 states in our dataset, which are treated as 29 clients in our FL framework. The goal of this paper is to jointly train client models $[\mathbf{w}_1, \cdots, \mathbf{w}_K]$ using the data $\{\mathcal{D}_k\}_{k=1}^K$ stored in all clients, where $K = 29$. Furthermore, we consider the challenges of local model training and global model aggregation raised by the imbalanced labels, Non-IID issue, and small data. We summarize the key notations used in the following sections in Table 3.

5 Methodology

5.1 Model Overview

Figure 3 shows the overview of the proposed federated learning framework `FedCovid`, which mainly contains the local update and the server update. During the local update, each client k will use the local training data $\mathcal{D}_k$ to update the model parameter $\mathbf{w}_k$. In particular, we propose to learn each patient's embedding by aggregating multiple types of EHR data via an adaptive fusion mechanism. Furthermore, to address the imbalance issue, we propose augmenting the embeddings for the positive patients. Finally, a hybrid fusion loss is used to train the local model $\mathbf{w}_k$. After the local update, active client parameters $[\mathbf{w}_1, \cdots, \mathbf{w}_B]$ will be uploaded to the server. In the server update, the global model $\mathbf{w}_g$ is obtained by aggregating $[\mathbf{w}_1, \cdots, \mathbf{w}_B]$ as well as taking the contribution score β_k of each local model $\mathbf{w}_k$. Note that we first use the clients with larger size to learn the warm-up global model $\mathbf{w}_g$, and then all the clients will be added into the model learning. This new ordinal training strategy aims to alleviate the small data issue. Next, we show the details of each component of the proposed `FedCovid` framework.

5.2 Local Update: Patient Representation Learning

Patient EHR data contains categorical, numerical, and sequential information. For each type of information, we need to map it to a latent vector representation.

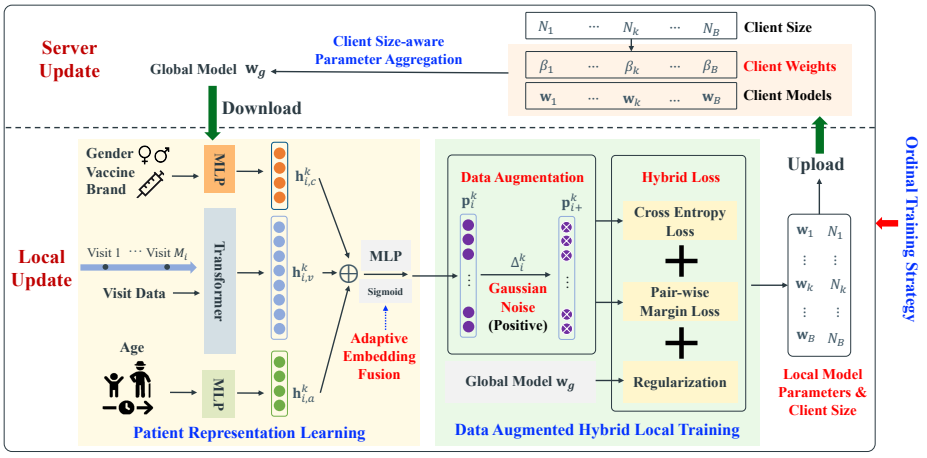

Fig. 3: Overview of the proposed `FedCovid` model.

Embedding Numerical and Categorical Features We first handle patients' demographic information, including age, gender, and COVID-19 vaccine brand. We treat the age information $a_i k$ as a numerical feature. For the gender g_i^k and COVID-19 vaccine brand information b_i^k, we treat them as the categorical features. We feed these two kinds of features into multi-layer perceptrons (MLP_a and MLP_c) to learn the latent representations for patient i at client k as given by Eq. (1) as follows:

$$\mathbf{h}_{i,a}^k = \text{MLP}_a(a_i^k); \quad \mathbf{h}_{i,c}^k = \text{MLP}_c(g_i^k, b_i^k). \tag{1}$$

Embedding Sequential Visit Data EHR data also contain the time-ordered sequential visit information $V_i^k = \{x_{i,1}^k, x_{1,2}^k, \cdots, x_{i,M_i}^k\}$. Several approaches [5,20,21,19,22] are proposed to embed the visit data built upon long short-term memory network (LSTM) [16], bidirectional LSTM (Bi-LSTM)[29], convolutional neural network (CNN) [17], and Transformer [33]. Using these backbone models, we can learn the visit embedding as follows:

$$\mathbf{h}_{i,v}^k = \mathcal{M}_b\left(V_i^k\right), \tag{2}$$

where $\mathcal{M}_b$ denotes the backbone approach used for embedding the visit data.

Adaptive Embedding Fusion The three latent embeddings are obtained from different types of data and models. Here we design an embedding fusion approach to combine the three embeddings in an adaptive approach via a gated linear unit (GLU) [7]. We first concatenate these embeddings as $\mathbf{h}_i^k = [\mathbf{h}_{i,a}^k, \mathbf{h}_{i,c}^k, \mathbf{h}_{i,v}^k]$ and then map $\mathbf{h}_i^k$ to a new representation as follows:

$$\mathbf{h}_i^{k'} = \mathbf{W}_i^k \mathbf{h}_i^k, \tag{3}$$

where $\mathbf{W}_i^k$ is a learnable weight matrix. We then learn a weight for each element in $\mathbf{h}_i^{k\prime}$ via a Sigmoid function, i.e.,

$$\boldsymbol{\phi}_i^k = \text{sigmoid}(\mathbf{h}_i^{k\prime}). \tag{4}$$

Finally, the element-wise multiplication $\circ$ is used to generate the patient representation as follows:

$$\mathbf{p}_i^k = \boldsymbol{\phi}_i^k \mathbf{h}_i^{k\prime}. \tag{5}$$

5.3 Local Update: Data Augmented Hybrid Local Training

Using Eq. (5), we can fuse different types of EHR data together to learn an aggregated patient representation, which can be directly used for prediction. However, as mentioned before, there is another challenge for our setting – imbalanced data. To address this problem, we propose using data augmentation techniques to balance the data, as well as a margin loss to differentiate between positive and negative patient representation learning.

EHR Data Augmentation Data augmentation approaches have been widely-used for image classification tasks such as rotating, flipping, or mixup technique [4], and natural language processing tasks, e.g., example interpolation techniques and model-based techniques [12]. However, EHR data is heterogeneous, with categorical features, numerical features, and discrete EHR sequence data, making it difficult to directly add small noise to the raw data. To address this issue, we implement the augmentation on the learned embeddings via Eq. (5) rather than the raw input X_i^k. The assumption is that if the patients are similar to each other, then the learned patient representations should also be similar.

Since the number of positive patients is much smaller than that of negative ones, we only need to increase the number of positive cases to make these two classes balanced. In particular, we add a noise vector Δ_i^k generated from a Gaussian distribution with parameters $\{\mu, \sigma\}$ to the learned positive patient embeddings via Eq. (5), where μ is the mean value and σ is the standard deviation for the Gaussian distribution, i.e., $\hat{\mathbf{p}}_{i+}^k = \mathbf{p}_{i+}^k + \Delta_i^k$. Due to the 1:5 ratio of positive and negative labels in our dataset, for each positive data, we will add four randomly generated noise vectors, respectively.

Hybrid Local Training Loss Let $\hat{\mathbf{P}}_+^k$ represent the representation matrix of the augmented positive data, $\mathbf{P}^k = [\mathbf{P}_+^k, \mathbf{P}_-^k]$ denote the real data representation matrix, where $\mathbf{P}_+^k$ represents the matrix of the real positive data and $\mathbf{P}_-^k$ is the matrix of the real negative data. Using $\hat{\mathbf{P}}_+^k$ and $\mathbf{P}^k$, we can directly train our local model using the cross entropy (CE) loss. To avoid the influence of noise, we will assign different weights to the loss terms of the real data and the augmented data as follows:

$$\mathcal{L}_c^k = \frac{1}{N_k}\text{CE}(f(\mathbf{P}^k), \mathbf{y}^k) + \frac{\lambda_c}{N_k^+}\text{CE}(f(\hat{\mathbf{P}}_+^k), \mathbf{y}_+^k), \tag{6}$$

where λ_c is a hyperparameter, $\mathbf{y}^k = [\mathbf{y}^k_+, \mathbf{y}^k_-]$ is the ground truth label vector of all real data, $\mathbf{y}^k_+$ is the positive label vector, $\mathbf{y}^k_-$ is the negative label vector, and N^+_k is the total number of augmented data.

To further learn the distinguishable patient representations, we also add a pair-wise margin loss to $\mathcal{L}^k_c$ as follows:

$$\mathcal{L}^k_m = \frac{1}{N_k + N^+_k} \sum_{i=1}^{N_k+N^+_k} \max(d(\tilde{\mathbf{p}}^k_i, \bar{\mathbf{p}}^k_{j+}) - d(\tilde{\mathbf{p}}^k_i, \mathbf{p}^k_{j'-}) + \delta, 0), \tag{7}$$

where $d(\cdot,\cdot)$ is the Euclidean distance function, $\tilde{\mathbf{p}}^k_i \in \{\mathbf{P}^k, \hat{\mathbf{P}}^k_+\}$ presents any data representation (i.e., the anchor sample), $\bar{\mathbf{p}}^k_{j+} \in \{\mathbf{P}^k_+, \hat{\mathbf{P}}^k_+\}$ is any positive real or augmented representation, $\mathbf{p}^k_{j'-} \in \mathbf{P}^k_-$ is a negative patient representation, and δ is the predefined margin value.

These two loss terms $\mathcal{L}^k_c$ and $\mathcal{L}^k_m$ all consider to update the local parameters based on the data. However, when the amount of data on the k-th local client is extremely small, only optimizing these two terms may cause the overfitting problem. To avoid this issue, we add an extra regularization term, which forces the local parameters $\mathbf{w}_k$ to be as close as the global model $\mathbf{w}_g$, i.e., $\|\mathbf{w}_k - \mathbf{w}_g\|^2$. In such a way, we can obtain the final hybrid loss as follows:

$$\mathcal{L}_k = \mathcal{L}^k_c + \lambda_m \mathcal{L}^k_m + \frac{\lambda_w}{N_w} \|\mathbf{w}_k - \mathbf{w}_g\|^2, \tag{8}$$

where λ_m and λ_w are trade-off hyperparameters, and N_w is the number of model parameters. Using Eq. (8), we can learn the local parameter set $\mathbf{w}_k$ and then upload it to the server side.

5.4 Server Update: Client Size-aware Aggregation

At each communication round, the server side will receive B client models $[\mathbf{w}_1, \cdots, \mathbf{w}_B]$. In general, we can follow FedAvg [24] to directly average them to obtain the global model $\mathbf{w}_g$. As we discussed before, the data size of each local client is unequally. The client with small size may not learn an accurate model by optimizing Eq. (8), and the average operation may destroy the learning of $\mathbf{w}_g$.

To avoid this problem, we propose to upload the size of each client and quantify the contribution of each client according to its size. The larger size, the more reliable, and the greater weight. Let β_k denote the contribution weight of the k-th client, which is defined as follows:

$$\beta_k = \frac{\log(N_k)}{\sum_{i=1}^{B} \log(N_i)}. \tag{9}$$

Using $[\beta_1, \cdots, \beta_B]$, we can obtain the updated global model as follows:

$$\mathbf{w}_g = \frac{1}{B} \sum_{k=1}^{B} \beta_k * \mathbf{w}_k. \tag{10}$$

$\mathbf{w}_g$ will be downloaded to each selected or active client for the next round local model training. This procedure will iteratively run until the model converges or achieves the maximum number of communication round.

5.5 Ordinal Training Strategy

As shown in Figure 2, most of clients only contain a small number of data and they have a higher probability to be selected if we use traditional federated learning training strategy. This may lead to a bad global model learning. To address this issue, we propose to divide the clients into two groups according to their size. We first train the model with the larger size clients. This stage can be considered as model warmup or initialization. After we get the initialized model $\mathbf{w}_g$, we then allow smaller clients to join the training. In particular, we lower the number of epochs and learning rates when training their local models compared with those used for the larger ones. This straightforward training strategy tries to make the negative effect caused by the smaller clients as low as possible.

6 Experiment

6.1 Experiment Setup

Dataset In our experiments, we use the dataset that is introduced in Section 3.

Baselines We use the following federated learning approaches as baselines:

- **FedAvg** [24] is the classical baseline. Active local clients train their own models and upload the model parameters to the server. The server averages the parameters of local models and re-distributes the updated global model back to active clients for the next round local training.
- **FedProx** [18] adds a reference loss in local training for each client to measure the distances between the local model and the global model, which constrains the local personalized optimization process not to drift excessively.
- **Per-FedAvg** [11] is a personalized federated learning algorithm inspired by meta learning to find an initial shared model that can be easily adapted to local datasets within limited steps of updates.

Implementation Details We implement all models with Pytorch on Ubuntu 20.04 with NVIDIA RTX A6000 GPU. We leverage the training and testing datasets constructed in Section 3.2. The hyperparameters δ, λ_c, λ_m, and λ_w in the loss function Eq. (8) are set to $\frac{1}{5}$, $\frac{1}{2}$, $\frac{1}{6}$, and $\frac{1}{3}$, respectively.

The total communication round is 400, where we set the warmup round as 200 to train the clients' models with larger clients (i.e., CA, NY, FL, OH, TX, KY, MI in Figure 2). We set the learning rate as 0.001 at the warmup stage and 0.01 after the warmup stage. For the small clients, we set the learning rate as 0.001 after the 200 communication round when they are selected to contribute

Table 4: Performance comparison

Setting	Algorithm	F1 Score	Cohen's Kappa	PR-AUC
Central Training	CNN	0.4855	0.4279	0.4270
	Transformer	0.4680	0.3842	0.4382
Federated Training	FedAvg	0.4081	<u>0.3138</u>	<u>0.1376</u>
	FedProx	<u>0.4083</u>	0.3129	0.1368
	Per-FedAvg	0.3722	0.2669	0.1361
	`FedCovid`	**0.4669**	**0.3697**	**0.3156**

to the model updates. Baselines do not use the ordinal training strategy, they treat all client equally and use the same learning rate 0.001. In this paper, we apply Transformer as $\mathcal{M}_b$ in Eq. (2) to embed the visit data. In particularly, we employ a two layer Transformer with hidden dimension of 16 and number of heads 8, and apply max-pooling to the output sequence to get the EHR latent embedding. All approaches use Adam as the optimizer, except for Per-FedAvg that uses the SGD optimizer.

6.2 Performance Evaluation

We conduct experiments on the dataset introduced in Section 3.2 to validate the proposed approach and baselines. Since the dataset is imbalanced, we use F1 score, Cohen's Kappa, and Area Under the Precision-Recall Curve (PR-AUC) as the evaluation metrics following [6]. We report the average values of the last 10 rounds of the test results at the server side in Table 4.

To explore the performance upper bound of the federated setting, in this experiment, we also put all the training data together to train a prediction model in the central training setting. We use CNN and Transformer as $\mathcal{M}_b$ to embed the visit data. The network structure of Transformer is the same as that of `FedCovid`. For the CNN model, we use a 1D CNN with kernel size 3 and step size 1. The output channel dimension is set to 2, and we apply a flatten operation to get the visit latent embedding. In Table 4, we can observe that the performance of central training-based approaches is better than that of federated learning approaches.

In the federated setting, FedAvg and FedProx have similar performance, which demonstrates that the reference loss in FedProx may not work for the clients with small size. Due to the unique challenges of the EHR datasets as we discussed in Section 3, the personalized federated learning approach Per-FedAvg does not outperform FedAvg and FedProx. We can also observe that the proposed `FedCovid` achieves the best performance in terms of three metrics. Compared with the best performance of baselines (with underline in Table 4), the performance of our proposed `FedCovid` model increases 14.35%, 17.81%, and 129.36% on F1 score, Cohen's Kappa, and PR-AUC, respectively.

Table 5: Ablation study

Approach	F1	Cohen's Kappa	PR-AUC
EHR Concatenation in Section 5.2	0.4365	0.3356	0.2832
CE Loss Only in Section 5.3	0.4150	0.2775	0.2204
Average Aggregation in Section 5.4	0.4486	0.3093	0.2996
Normal Federated Training in Section 5.5	0.4306	0.3266	0.2817
`FedCovid`	**0.4669**	**0.3697**	**0.3156**

6.3 Ablation Study

In the proposed `FedCovid` model, we design several novel mechanisms. To investigate the contribution of each component, we conduct the following ablation study and the results are shown in Table 5. To validate the benefit of the proposed adaptive EHR fusion mechanism in Section 5.2, we use the simple **EHR concatenation** operation to learn patient representation. **CE Loss Only** aims to validate the power of data augmentation and the margin loss for handling the imbalance issue in Section 5.3. The approach of **Average Aggregation** is to prove the usefulness of the proposed client size-aware aggregation in Section 5.4. The goal of **Normal Federated Training** is to show the advantage of ordinal training strategy proposed in Section 5.5.

From the results listed in Table 5, we can observe that compared with the proposed `FedCovid`, the performance of all comparison approaches drops, especially for the CE Loss Only. However, they all outperform the best baselines in Table 4. These results can clearly confirm that each mechanism used in `FedCovid` is necessary and essential to improve the prediction performance. The contribution descending order in boosting performance is (1) data augmented hybrid loss for training client model, (2) ordinal training strategy, (3) adaptive EHR fusion, and (4) client size-wise model aggregation.

6.4 Convergence Analysis

Figure 4 show the performance changes with regards to each communication round. We can observe that the F1 score also increases dramatically at the beginning and then become stable until 200 communication round. In this warmup stage, we use clients with larger size to train the global model. After the 200th communication round, the performance sharply increases again. This shows that even using the small size of client data, `FedCovid` can still boost the performance can make the model converge.

6.5 Hyperparameter Sensitivity Analysis

In this subsection, the number of communication rounds for warm-up is very important. To investigate the affect of this parameter on the performance change, we conduct the following experiment. Let γ controls the warmup round for

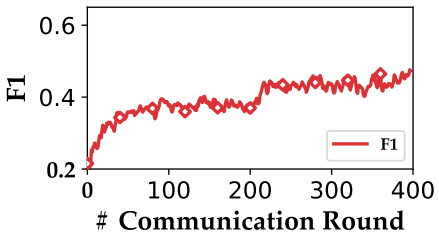

Fig. 4: Model convergence

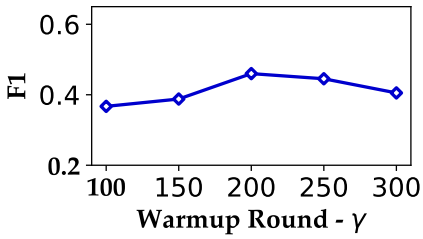

Fig. 5: Hyperparameter analysis

the large states. Ideally, with the increase of γ, model performance will first increases and then decreases, as there is a trade-off between a well-trained global model and the generalization. To validate this assumption, we alter γ as $\{100, 150, 200, 250, 300\}$, and the results are shown in Figure 5. We observe that the performance increases first and then decreases with the increase of the warmup round. The reason is that the warmup stage lasts too long, which makes the global model not able to capture enough information from the small states given a fixed communication round and further affects the generalization of the global model. This observation is in accord with our assumption.

7 Conclusion

In this study, we propose `FedCovid`, a new federated learning model for predicting COVID-19 vaccination side effects. As far as we know, this is the first work to apply a federated learning framework using EHR data to predict COVID-19 side effects. `FedCovid` solves the following challenges caused by EHR data, including EHR data heterogeneity issue, label imbalanced problem, and client size difference challenge, in a single framework. We conduct experiments on a real world EHR dataset provided by IQVIA. Experimental results show that the proposed `FedCovid` outperforms baselines in terms of three different metrics, including F1 score, Cohen's Kappa, and PR-AUC. An ablation study demonstrates that all designed mechanisms are useful to improve the prediction performance. Finally, the model insight analysis shows the convergence and hyperparameter sensitivity of the proposed `FedCovid` model.

References

1. Abiodun, K.M., Awotunde, J.B., Aremu, D.R., Adeniyi, E.A.: Explainable ai for fighting covid-19 pandemic: Opportunities, challenges, and future prospects. In: Computational Intelligence for COVID-19 and Future Pandemics, pp. 315–332. Springer (2022)
2. Almars, A.M., Gad, I., Atlam, E.S.: Applications of ai and iot in covid-19 vaccine and its impact on social life. In: Medical Informatics and Bioimaging Using Artificial Intelligence, pp. 115–127. Springer (2022)

3. Borriello, A., Master, D., Pellegrini, A., Rose, J.M.: Preferences for a covid-19 vaccine in australia. Vaccine **39**(3), 473–479 (2021)
4. Chlap, P., Min, H., Vandenberg, N., Dowling, J., Holloway, L., Haworth, A.: A review of medical image data augmentation techniques for deep learning applications. Journal of Medical Imaging and Radiation Oncology **65**(5), 545–563 (2021)
5. Choi, E., Bahadori, M.T., Schuetz, A., Stewart, W.F., Sun, J.: Doctor ai: Predicting clinical events via recurrent neural networks. In: MLHC. pp. 301–318 (2016)
6. Cui, L., Biswal, S., Glass, L.M., Lever, G., Sun, J., Xiao, C.: Conan: complementary pattern augmentation for rare disease detection. In: AAAI. pp. 614–621 (2020)
7. Dauphin, Y.N., Fan, A., Auli, M., Grangier, D.: Language modeling with gated convolutional networks. In: Proceedings of ICML. pp. 933–941. PMLR (2017)
8. Dayan, I., Roth, H.R., Zhong, A., Harouni, A., Gentili, A., Abidin, A.Z., Liu, A., Costa, A.B., Wood, B.J., Tsai, C.S., et al.: Federated learning for predicting clinical outcomes in patients with covid-19. Nature medicine **27**(10), 1735–1743 (2021)
9. Diaz, G.A., Parsons, G.T., Gering, S.K., Meier, A.R., Hutchinson, I.V., Robicsek, A.: Myocarditis and pericarditis after vaccination for covid-19. Jama **326**(12), 1210–1212 (2021)
10. Elnaem, M.H., Mohd Taufek, N.H., Ab Rahman, N.S., Mohd Nazar, N.I., Zin, C.S., Nuffer, W., Turner, C.J.: Covid-19 vaccination attitudes, perceptions, and side effect experiences in malaysia: Do age, gender, and vaccine type matter? Vaccines **9**(10), 1156 (2021)
11. Fallah, A., Mokhtari, A., Ozdaglar, A.: Personalized federated learning: A meta-learning approach. arXiv preprint arXiv:2002.07948 (2020)
12. Feng, S.Y., Gangal, V., Wei, J., Chandar, S., Vosoughi, S., Mitamura, T., Hovy, E.: A survey of data augmentation approaches for nlp. arXiv:2105.03075 (2021)
13. Georgiadis, A., Babbar, V., Silavong, F., Moran, S., Otter, R.: St-fl: Style transfer preprocessing in federated learning for covid-19 segmentation. arXiv (2022)
14. Gupta, A., Gharehgozli, A.: Developing a machine learning framework to determine the spread of covid-19. Available at SSRN 3635211 (2020)
15. Hause, A.M., Baggs, J., Marquez, P., Myers, T.R., Su, J.R., Blanc, P.G., Baumblatt, J.A.G., Woo, E.J., Gee, J., Shimabukuro, T.T., et al.: Safety monitoring of covid-19 vaccine booster doses among adultsâ€"united states, september 22, 2021–february 6, 2022. Morbidity and Mortality Weekly Report **71**(7), 249 (2022)
16. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
17. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)
18. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems **2**, 429–450 (2020)
19. Luo, J., Ye, M., Xiao, C., Ma, F.: Hitanet: Hierarchical time-aware attention networks for risk prediction on electronic health records. In: KDD. pp. 647–656 (2020)
20. Ma, F., Chitta, R., Zhou, J., You, Q., Sun, T., Gao, J.: Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks. In: KDD. pp. 1903–1911 (2017)
21. Ma, F., Gao, J., Suo, Q., You, Q., Zhou, J., Zhang, A.: Risk prediction on electronic health records with prior medical knowledge. In: KDD. pp. 1910–1919 (2018)
22. Ma, F., Wang, Y., Xiao, H., Yuan, Y., Chitta, R., Zhou, J., Gao, J.: A general framework for diagnosis prediction via incorporating medical code descriptions. In: BIBM. pp. 1070–1075. IEEE (2018)

23. Mariappan, M.B., Devi, K., Venkataraman, Y., Lim, M.K., Theivendren, P.: Using ai and ml to predict shipment times of therapeutics, diagnostics and vaccines in e-pharmacy supply chains during covid-19 pandemic. The International Journal of Logistics Management (2022)
24. 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. pp. 1273–1282. PMLR (2017)
25. Mohamed, K., Rzymski, P., Islam, M.S., Makuku, R., Mushtaq, A., Khan, A., Ivanovska, M., Makka, S.A., Hashem, F., Marquez, L., et al.: Covid-19 vaccinations: The unknowns, challenges, and hopes. Journal of medical virology **94**(4), 1336–1349 (2022)
26. Napolitano, F., Xu, X., Gao, X.: Impact of computational approaches in the fight against covid-19: an ai guided review of 17 000 studies. Briefings in bioinformatics **23**(1), bbab456 (2022)
27. Rahimi, K.: Guillain-barre syndrome during covid-19 pandemic: an overview of the reports. Neurological Sciences **41**(11), 3149–3156 (2020)
28. Schultz, N.H., Sørvoll, I.H., Michelsen, A.E., Munthe, L.A., Lund-Johansen, F., Ahlen, M.T., Wiedmann, M., Aamodt, A.H., Skattør, T.H., Tjønnfjord, G.E., et al.: Thrombosis and thrombocytopenia after chadox1 ncov-19 vaccination. New England journal of medicine **384**(22), 2124–2130 (2021)
29. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE transactions on Signal Processing **45**(11), 2673–2681 (1997)
30. Shimabukuro, T.T., Cole, M., Su, J.R.: Reports of anaphylaxis after receipt of mrna covid-19 vaccines in the usâ€"december 14, 2020-january 18, 2021. Jama **325**(11), 1101–1102 (2021)
31. Sprent, J., King, C.: Covid-19 vaccine side effects: The positives about feeling bad. Science immunology **6**(60), eabj9256 (2021)
32. Vaid, A., Jaladanki, S.K., Xu, J., Teng, S., Kumar, A., Lee, S., Somani, S., Paranjpe, I., De Freitas, J.K., Wanyan, T., et al.: Federated learning of electronic health records to improve mortality prediction in hospitalized patients with covid-19: Machine learning approach. JMIR medical informatics **9**(1), e24207 (2021)
33. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NeurIPS **30** (2017)
34. Wang, Y., Hu, M., Li, Q., Zhang, X.P., Zhai, G., Yao, N.: Abnormal respiratory patterns classifier may contribute to large-scale screening of people infected with covid-19 in an accurate and unobtrusive manner. arXiv preprint arXiv:2002.05534 (2020)
35. Zhavoronkov, A., Zagribelnyy, B., Zhebrak, A., Aladinskiy, V., Terentiev, V., Vanhaelen, Q., Bezrukov, D.S., Polykovskiy, D., Shayakhmetov, R., Filimonov, A., et al.: Potential non-covalent sars-cov-2 3c-like protease inhibitors designed using generative deep learning approaches and reviewed by human medicinal chemist in virtual reality (2020)
36. Zhou, Y., He, J.: A randomized approach for crowdsourcing in the presence of multiple views. In: ICDM. pp. 685–694. IEEE Computer Society (2017)
37. Zhou, Y., Wu, J., Wang, H., He, J.: Adversarial robustness through bias variance decomposition: A new perspective for federated learning. arXiv (2020)
38. Zhou, Y., Ying, L., He, J.: Multic^2: an optimization framework for learning from task and worker dual heterogeneity. In: SDM. pp. 579–587. SIAM (2017)