Federated Generalized Learning on Non-IID Medical Imaging with Virtual Homogeneous Generation and Adversarial Domain Adaptation

21043007
YAN WENHAO
Department of Sciences and Informatics
Muroran Institute of Technology, Japan

January 15 2023

Acknowledgement

I first want to express my great appreciation to Prof. He Li for his excellent advice and diligent efforts to guide my research through my master career. Prof. He Li is always available with his valuable suggestions and guidance throughout my master study. I will never forget the revisions he made on each of my technical papers. Without his support from various aspects, I could not make it today.

I must unequivocally express my profound gratitude to Prof. Kaoru Ota, Prof. Mianxiong Dong, and Prof. Xu, for many suggestive comments to improve the quality of this dissertation.

I also would like to thank all the Emerging Networks and Systems Laboratory members, who were always there in both my study and life. It was a great time to be with you these years.

Various support from the Muroran Institute of Technology was of great importance for me to pursue my master studies. I especially would like to thank Mrs. Otani, as well as the other staffs for their always warm help during these years.

Finally, my deep gratitude goes to my family. I thank my parents for their love and support.

Abstract

Distributed machine learning has been widely applied to the predicament of data silos in recent years. Due to increasing privacy concerns, medical institutions are trying to train deep neural network models collaboratively by adopting Federated Learning (FL). However, the Not identically and independently distributed (Non-IID) medical images become the main challenge limiting the performance of existing FL approaches. Mainstream works focus on correcting client-specific drifts with a local adaptation, and it will cause the inability of the aggregated model to perform well on the samples from uninvolved domains. Hence, This thesis propose a novel framework named FedViDA (Federated Virtual Discriminative Adaptation) to solve data heterogeneity while keeping generalization ability. In particular, FedViDA enables the deep neural network models to extract bias-free feature maps across clients by adversarial training with a virtual homogeneous dataset. The virtual dataset is composed of representations of all clients, and it does not contain any private information so that it can be shared among the clients without limitation. After exhaustive experiments, we demonstrate that FedViDA outperforms existing generalized approaches in both accuracy and robustness.

Contents

List of Figures

List of Tables

1 Introduction

By involving more samples in gradient calculation to avoid overfitting, training deep neural networks with multi-clients collaboratively jointed shows its outstanding potential to enhance the performance of data-driven models on medical image analysis. However, due to the increasing concern of privacy preservation, data sharing is still banned in this and similar sensitive fields. In terms of existing frameworks of data protection and privacy laws, the *2016 General Data Protection Regulation (GDPR)* from European Union (EU) is the most recent and well-known example for its harsh restrictions on unauthorized or unlawful processing of private data. This practice is being followed up by other countries and regions worldwide. Towards this dilemma, Federated Learning (FL) is proposed by McMahan, Brendan, *et al.* [1] to cover both model performance and privacy protection.

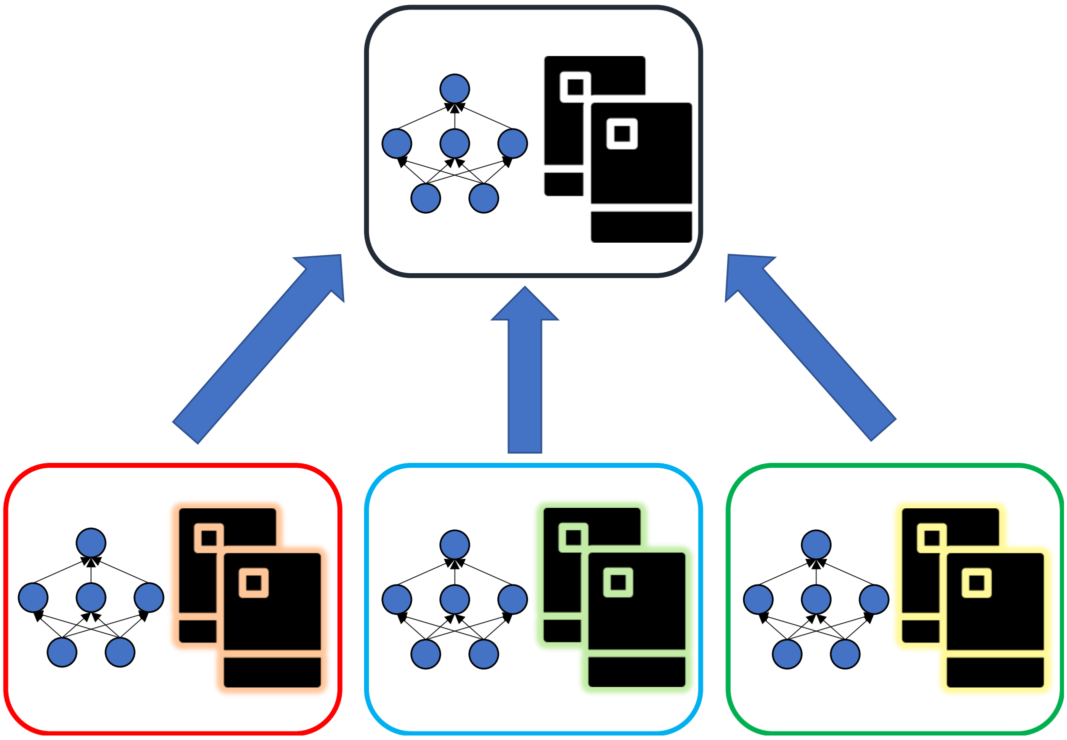


Figure 1: Overview of federated learning across clients

Federated learning allows a decentralized solution to model training, with different joining clients processing private data locally and aggregating their gradient updates globally. Any kind of actual data sharing will not take place throughout the whole training period. Always keeping local data only stored locally effectively avoids the tackling of data leakage and abuse. Meanwhile, conventional methods of data security like differential privacy or secure multi-party computation can also be taken in combination to protect the shared weights update from gradient leakage attacks [2], [3]. All these above mechanisms provide FL with unparalleled data security.

Despite FL achieving impressive progress in many privacy-sensitive fields. the non-independent and identically distributed (Non-IID) data is still a great challenge to model convergence and robustness, especially in real-world practice [4]. Such Non-IID issues happen for different causes in different specific scenarios. In the aspect of medical images, the style of histology images varies a lot due to different staining operations, dermatoscopic images can perform huge morphological differences due to shooting angles and other optical factors, and MRI images of different institutions also suffer from data heterogeneity associated with various scanners or imaging protocols. Some severe heterogeneity can easily cause drifts among joint training clients and then lead to consequences like unreliable convergence and weak performance [5].

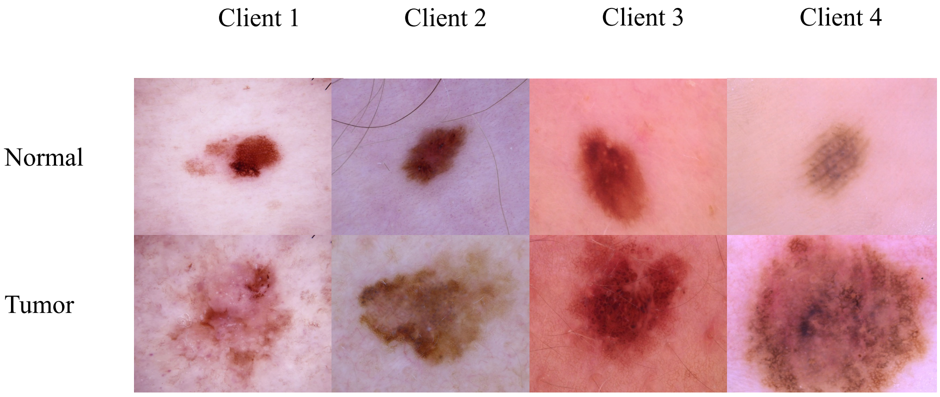


Figure 2: Examples of skin lesion images of normal and tumor tissues from 4 clients, showing large heterogeneity.

Actually, it has been demonstrated by previous works that such data heterogeneity introduces drifts causing local (client) and global (server) optimization to be inconsistent. Specifically, each client optimizes its own weights towards local optima instead of achieving overall solutions. And the inconsistency of local optima and global objective incite drifts across client updates. Subsequently, global drifts will be raised due to aggregating these mismatching local updates [6].

To overcome such a Non-IID predicament in FL, relevant researchers pay their attention to local and global perspectives respectively, corresponding to tackling the drift locally or globally. The client-side opinion holds a philosophy that the global aggregated model should

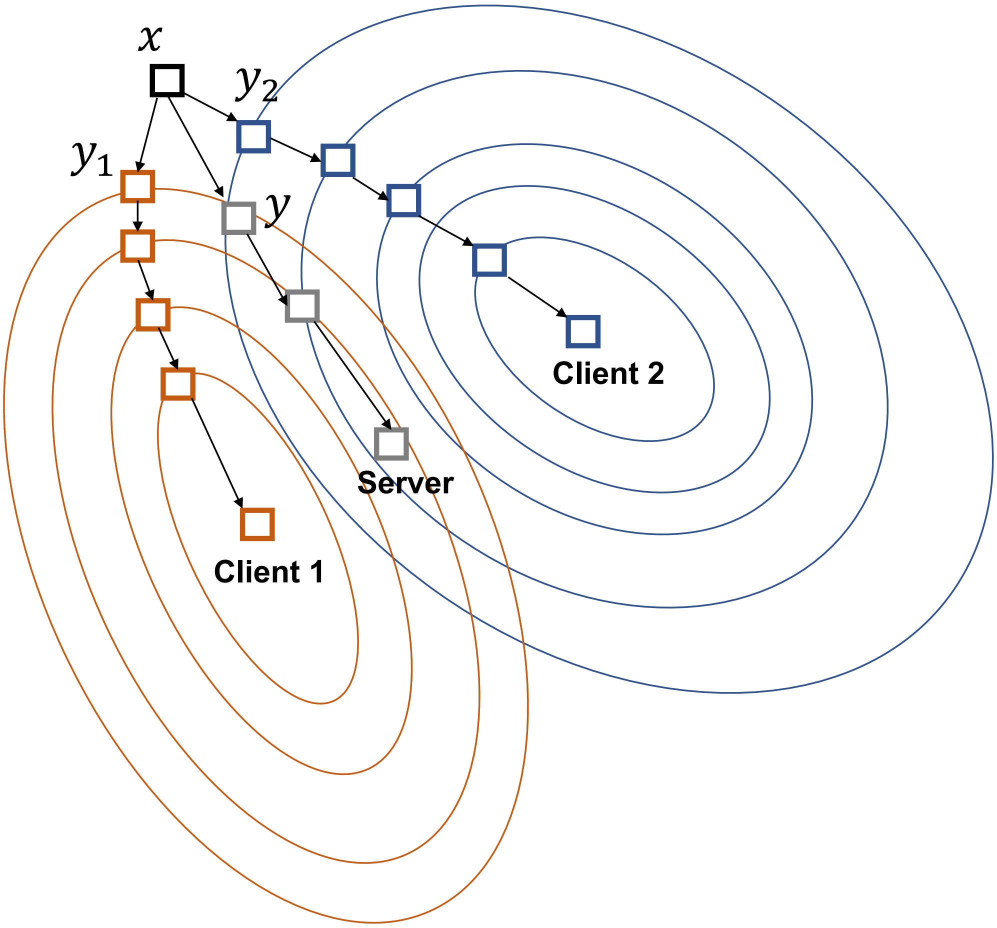


Figure 3: Loss visualization of two heterogeneous clients in FL

obey the local distribution by adapting to local deviations. For example, Karimireddy, *et al.* proposed an algorithm (SCAFFOLD) [7] using control variates to correct drifts during local updates. Li, *et al.* designed MOON [8], utilizing the similarity between model representations to correct the local training of individual parties. FedBN [9] from Li, *et al.* is the most representative of these client-side works. The Authors of FedBN proposed a simple but efficient idea that aggregating the weights except batch normalization layers can alleviate the feature shift gracefully. Respectively, the server-side opinion would devote their efforts towards global schemes such as aggregation strategy. As an example, adjusting aggregation weights is a widely-applied operation for many works with no doubt. Wang, *et al.* proposed a general framework named FedNova [10] as a normalized averaging method that eliminates objective inconsistency. Reddi, *et al.* [11] provided the federated version of several adaptive optimizers including Adagrad, Adam, and Yogi. And these modified optimizers can significantly improve the performance of FL. However, all the mentioned methods tackled data heterogeneity by taking a one-sided approach via either the local aspect or the global aspect. HarmoFL [12] from Jiang, *et al.* jointly addressed the coupled drifts without extra communication cost.

Although some of the current leading advanced FL algorithms can provide highly reliable prediction as well as experienced doctors, these progresses still have a long way to go before wide and practical implementation. Generally speaking, the algorithms with local calibration or adaptation outperform the others a lot, but a model with local components also simultaneously it is hard to perform well on those samples not participating in the training process. It can also be said that these models are weak in generalization. FL aims at training powerful models with jointed privacy information. it is in some ways a victory of cooperation in which Train a model collaboratively rather than by oneself and benefits more people. Hence, a model with local calibration will limit the beneficiary to patients from participating hospitals or institutions. And in most cases, an FL system will not be a project with low technical doorsill. It is not practical for most small medical facilities, especially those in many developing countries or regions to afford such technical costs of FL. However, those low-tech and backward institutions are the ones that most urgently need the assistance of AI and FL for their lack of advanced detection instruments, experienced doctors, and adequate accumulation of samples. Obviously, most local-side models cannot meet such down-to-earth demands. On the other hand, server-side models perform slightly better than the local-side ones on generalizability. But overall, they still underperform our expectations and also suffer from a non-negligible debilitation while handling samples outside training clients. Yet how to keep the pursuit of high precision while taking into account generalization capabilities still remains unclear.

In this thesis, we devote our efforts to solving data heterogeneity with feature extraction in end-to-end learning models. we consider minimizing the proportion of client-specific (local) features as the key point to overcome the Non-IID problem. In our opinion, solving feature skew in FL systems is actually a Domain Adaptation (DA) task among distributed clients without data access. DA generally solves highly generalized models by reducing the distribution divergences among domains, but it is impractical under the constraints of inaccessible data across clients. Considering the above reasons, we propose a new framework named FedViDA (Federated Virtual Discriminative Adaptation) to address this dilemma, FedViDA includes a federated generative adversarial network (GAN) architecture to generate virtual unlabeled samples with shared commonality among clients. This virtual dataset can be shared without any limitation cause it does not belong to real patients. Every client of the FedViDA system can execute a domain generation task, targeting local natural data respectively, with a common global source of generated virtual data. Then the classifier part of the models would be trained with homogeneous features. Under adaptive aggregation strategies on the feature extraction part and classifier part, the FedViDA algorithm can achieve comprehensive higher performance on samples from both existing sources and appended sources.

Most of our contributions can be summarized as follows:

- We raise a fundamental question about the generalizability of existing FL algorithms, especially those executing with local calibration. And then propose an idea to solve this problem through virtual datasets with no real private info but pervasive representations.

- To create virtual datasets with shared commonality among clients, we propose a distributed GAN architecture with local discriminators together with a common global adversary, trading off the authenticity and pervasiveness of generated samples.

- We design an adversarial domain adaptation mechanism on the classifying neural network, to immigrate the representations of virtual data to the features of natural data, to achieve a homogeneous convergence.

- After comprehensive experiments, we demonstrate the FedViDA algorithm outperforms existing FL approaches in generalizability and benefits robustness to some extent.

2 Preliminaries

2.1 Federated Learning

Federated learning is a recently proposed framework for distributed machine learning while protecting data privacy. Its ideas have been widely implemented in different fields such as medical AI, the Internet of Things, and recommendation systems.

FedAvg [1] has been recognized as the de facto algorithm of FL and most of the subsequent proposed methods can be considered its variants. In each round of FedAvg updates, the algorithm can be divided into four steps. First, the server shares the global model with the clients as initial weights. Second, the clients train their own models for certain epochs with local data. Third, the clients send back their updates. At last, the server aggregates the received weights based on the sample quantity of each client, to produce an updated global model for the next round.

In essence, FedAvg aims to solve a minimization of a global objective function $L_g(\omega)$:

$$\min L_g(\omega) = \sum_{k=1}^{K} p_k L_k(w) \tag{1}$$

where K means the number of clients in the FL system, k stands for the order of each client, p_k denotes the ratio of the k-th client's samples to the total samples of all clients so that it must satisfy $\sum_{k=1}^{K} p_k = 1$. $L_k(\omega)$ stands for the local objective function of each client, which is defined as:

$$L_k(\omega) = \mathbb{E}_{(x,y)\sim v(x,y)} L\left(f_k(x,\omega), y\right) \tag{2}$$

In equation (2), $v(x,y)$ stands for the joint distribution of data in client k, and f stands for the classifier model in client k. L is the cross-entropy loss function. Besides the basic iteration, FedAvg also offers some hyperparameters to control the aggregation strategy. Considering the efficiency issues when aggregating model parameters, most systems of FedAvg do not aggregate the weights of all clients at each update, but randomly select clients at a defined rate η:

$$\frac{|S_j|}{|S|} = \eta \tag{3}$$

with S_j being the set of select clients at round j and S being all clients.

What's more, to reduce communication costs and improve the robustness of aggregated gradients, E is usually set to represent the number of local updates. it can also be considered as an aggregation interval:

$$\omega_k^{j+1} \leftarrow \omega_k^j + \alpha \bigtriangledown L_k\left(\omega_k^j\right), j \bmod E \neq 0 \tag{4}$$

$$\omega^{j+1} = \sum_{k \in S_j} p_k \omega_k^j, j \bmod E = 0 \tag{5}$$

in which ω_k^j stands for k-th update of client k at epoch j, and α represents the learning rate. In general, the learning rate is consistent across clients.

2.2 Generative Adversarial Network

As a representative of unsupervised generative models based on game theory, GANs [14][22][29][30] can create virtual data that look like real ones. A typical GAN model consists of two deep neural networks: generator (G) and discriminator (D). Generator networks map randomly sampled noise to generate similar data by approaching the distribution of those real ones, and discriminator networks learn to predict whether the inputting data is real or fake.

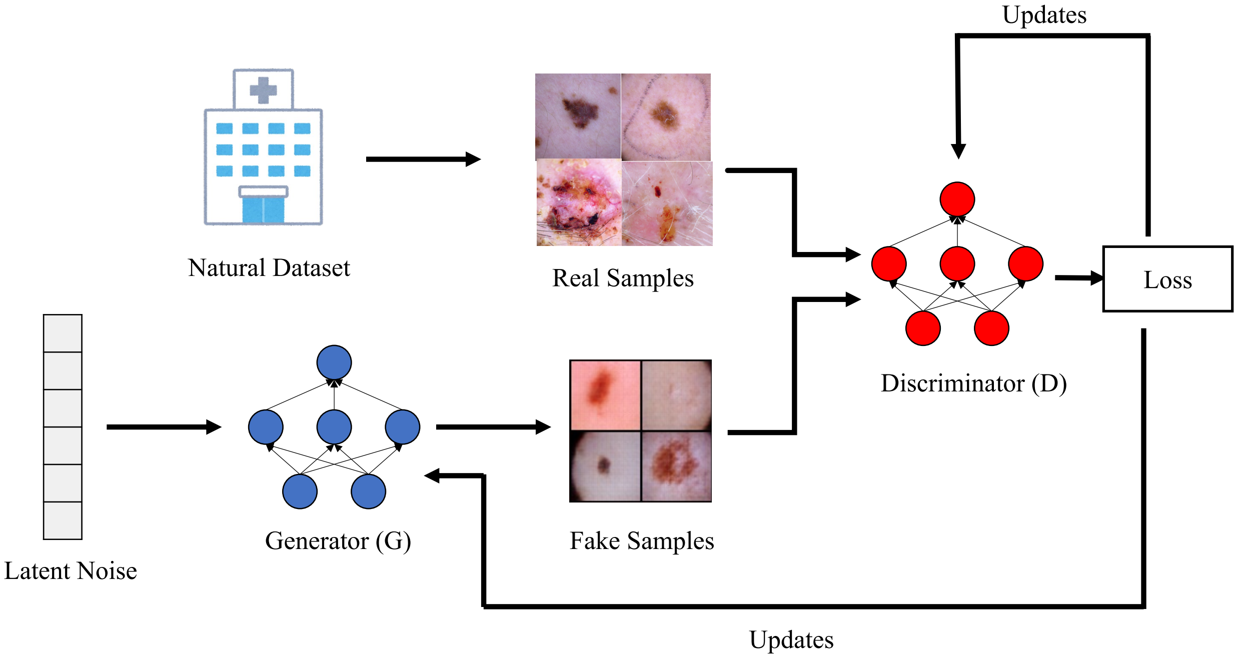


Figure 4: Basic architecture of GANs

Following the convention, x represents the real data, while υ_d stands for the real natural distribution, which is also the learning target of G. z is used to describe the latent vector as well as random noise, and υ_z is usually denoted as certain common sampling distribution such as *Gaussian distribution.* $G(z, \theta_g)$ and $D(x, \theta_d)$ are differentiable functions of the generator and discriminator model, in which θ_g and θ_d stand for parameters of these two models. During training, the output prediction y of the discriminator according to different sources can be expressed as:

$$y_{real} = D(x, \theta_d) \tag{6}$$

$$y_{fake} = D(G(z, \theta_g), \theta_d) \tag{7}$$

As for the objective function, the job of the discriminator network is to tell how an input image is realistic. it is actually not too much different from common classification networks. And the generator network seeks to increase the probability of confusing the discriminator. These two neural networks will form a competitive relationship, we would generally refer to this process as *Adversarial Training.* The joint objective function can be summarized as:

$$\begin{aligned}\min_{G}\max_{D} V_{GANs}(G,D) =& \mathbb{E}_{z\sim P_z}\left[\log\left(1-D\left(G\left(z\right)\right)\right)\right] \\ &+ \mathbb{E}_{X\sim P_d}\left[\log\left(D\left(x\right)\right)\right]\end{aligned} \tag{8}$$

where $\mathbb{E}$ stands for the inputting batch consisting of natural samples and generated ones.

2.3 Domain Adaptation

Domain adaptation [13] is an emerging crossing field associated with machine learning and transfer learning. its aim is to solve the problem when training data and testing data are from heterogeneous distributions. In mainstream works of DA, most of the methods aim to find a domain-invariant feature extraction ϕ, by minimizing the discrepancy between source domain distribution υ_s and target domain distribution υ_t. The risk of a solved hypothesis h is commonly bounded as:

$$\varepsilon^t\left(h\right) \leq \varepsilon^s\left(h\right) + divergence_H\left(\upsilon_t, \upsilon_s\right) + \lambda_H + C \tag{9}$$

where $\varepsilon\left(h\right)$ stands for risk of hypothesis model h on the specific domain, and λ_H denotes the complexity of the hypothesis space H for the learning task among the domains.

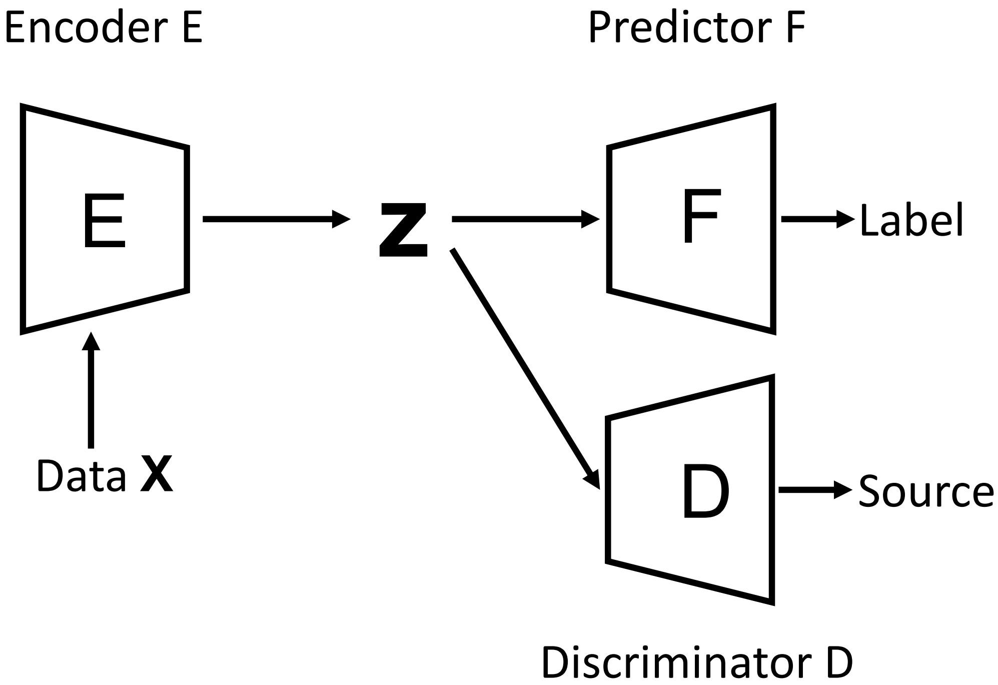


Figure 5: A Typical Adversarial Domain Adaptation Method

2.4 Analysis

In the FL scenarios, heterogeneous clients can be considered as different domains. However, due to the limitations of FL for cross-client data access, it is not practical to measure or reduce the discrepancy between domains. Hence, we introduce virtual data generated by GANs as a mediator of measuring discrepancy. Instead of minimizing the discrepancy between every two clients, we propose a new idea to make each client calibrate its natural feature distribution homogeneous with the shared virtual feature distribution.

We regard virtual distribution as the source domain and natural distribution as the target domain. Assume that we have K domains (clients) in total. For those domains T_i, we define a global aggregated feature extraction ϕ so that the risk of each domain T_i can be bounded as:

$$\begin{aligned} \varepsilon^{T_i}\left(\phi\right) \leq & \varepsilon^{v}\left(\phi\right) + \lambda_{\phi} + C \\ & + divergence_{\phi}\left(\upsilon_{T_i}\left(\phi\left(x\right)\right), \upsilon_{v}\left(\phi\left(x\right)\right)\right) \end{aligned} \tag{10}$$

in which υ_v stands for the generated virtual distribution, and ε^v denotes the risk of υ_v.

3 Virtual Homogeneous Generation

The key point to enhancing the generalization ability of FL algorithms with a virtual dataset is calibrating the natural feature distribution and the virtual feature distribution. We consider the virtual dataset as the source domain [15], therefore we hope the virtual dataset can preserve as many global homogeneous features as possible and not contain client-specific features. In this work of FedViDA, we design a federated GAN architecture with local discriminators and a global adversary to implement homogeneous generation.

3.1 Classic GAN

Before introducing our work, let us make some explanations about the classic generative adversarial networks. As instructed in Section II, the classic GAN consists of two competitive neural networks, the generator G and the discriminator D [18], playing a two-player game to achieve a dynamic balance on the value function $V(G, D)$ as:

$$\begin{aligned}\min_G \max_D V_{GANs}(G, D) =& \mathbb{E}_{z\sim P_z}\left[\log\left(1 - D\left(G\left(z\right)\right)\right)\right] \\ &+ \mathbb{E}_{X\sim P_d}\left[\log\left(D\left(x\right)\right)\right]\end{aligned} \tag{11}$$

where z stands for the latent vector under the pre-defined random sampling distribution P_z, and P_{data} represents the natural data distribution. During adversarial training, the generator network is learning to produce real-like samples to confuse the discriminator network, while the goal of the discriminator network is to splitting generated samples from natural ones. This adversarial training process will lead to the distribution of generator networks approximating the real distribution in the training dataset.

3.2 FedGAN

The FedGAN framework was proposed by Rasouli, et al [21] to address privacy-preserving limitations while training GANs on distributed datasets. The FedGAN system set an aggregation server for calculating the weighted average of parameters collected from client generators and client discriminators, and broadcast aggregated parameters to clients at the start of a new iteration.

In addition, the authors make an experimental validation that FedGAN can provide similar performance as centralized GANs while protecting local privacy and reducing communication costs.

3.3 Motivation of FedViDA-GAN

Although FedGAN [21] can provide enough performance in most distributed training scenarios, we note that it still faces a major bottleneck caused by Non-IID distributed data. The natural data on each client are not exactly equivalent to data on other clients. Due to the heterogeneous distribution, The discriminator on each client is not actually learning to make "real or fake" judgments, but learning to figure out whether the input data is from the local source. it will cause feature drifts problem when aggregating, and also not conducive to our need to retain features of commonality on heterogeneous clients.

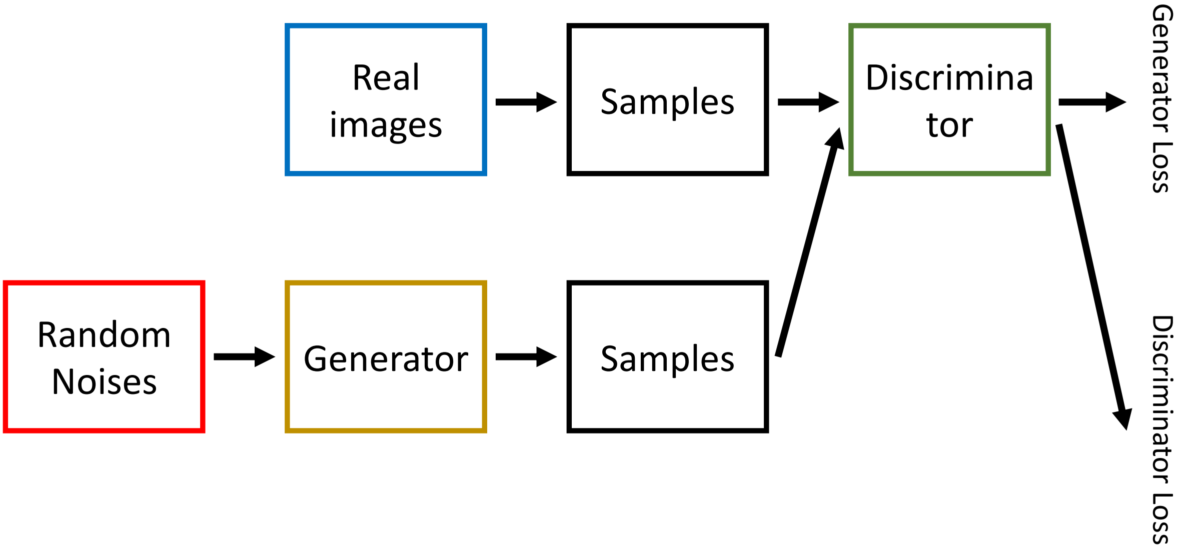


Figure 6: Backpropagation in Discriminator Training in Classic GANs

3.4 Mathematical Formulation of FedViDA-GAN

The key to going over the bottleneck caused by Non-IID data is avoiding the localization tendencies of discriminators. Based on this idea, we propose a multi-discriminator GAN architecture, where a local generator not only works with a local discriminator as usual but also forms an adversarial training process with a global discriminator for domain prediction. Therefore, the generator would avoid learning client-specific features but across-client distribution to confuse the global adversary.

Given K heterogeneous clients, and initialize K separate generator-discriminator pairs (G_i, D_i), their local value function is the same as that of classical GANs:

$$\begin{aligned} V\left(G_i, D_i\right) = & \mathbb{E}_{z\sim P_z}\left[\log\left(1 - D_i\left(G_i\left(z\right)\right)\right)\right] \\ & + \mathbb{E}_{X\sim P_d}\left[\log\left(D_i\left(x\right)\right)\right] \end{aligned} \tag{12}$$

We further introduce a global adversary (named domain discriminator) whose goal is to identify which generator generated the synthetic sample. The loss of the domain discriminator can then be defined with cross-entropy function as:

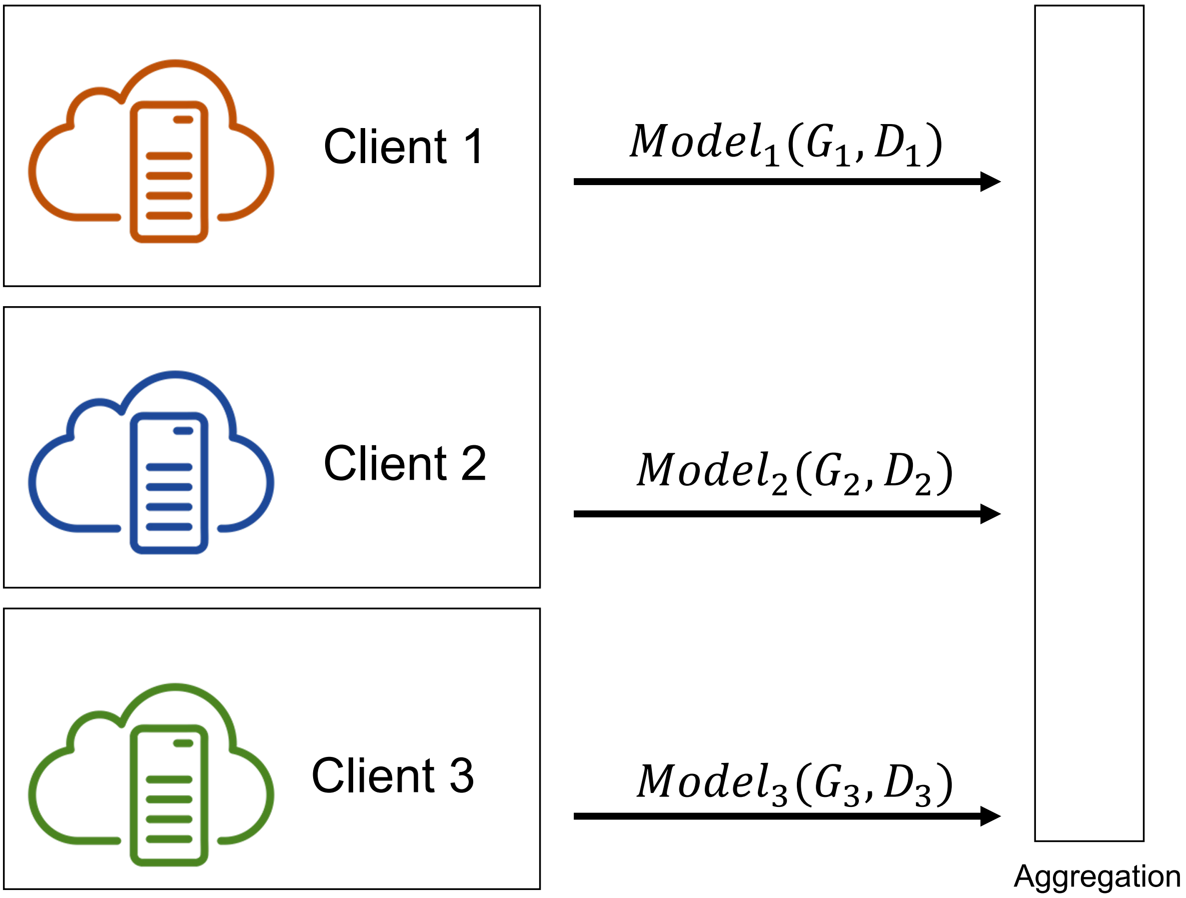


Figure 7: The Architecture of FedGAN

$$R(D_d) = \mathbb{E}_{z\sim p_z}[\log D_d^i G_i(z)] \tag{13}$$

where $D_d(x) = \left(D_d^1(x), D_d^2(x), \cdots, D_d^K(x)\right)$ stand for the softmax embedding of the domain discriminator output representing the probability of x to be generated by the generator G_i from domain i, It is also satisfying the following equation.

$$\sum_{i=1}^{K} D_d^i(x) = 1 \tag{14}$$

Hence, the complete objective function consisting of $V(G_i, D_i)$ and $R(D_d)$ of FedViDA-GAN can be define as:

$$\sum_{i=1}^{K} \left\{ \mathbb{E}_{x\sim P_d^i}[\log D_i(x)] + \mathbb{E}_{z\sim P_z} \log[1 - D_i(G_i(x))] + \lambda \mathbb{E}_{z\sim P_z} \log\left[D_d^i(G_i(z))\right] \right\} \tag{15}$$

in which the p_d^i the natural data distribution of client i (domain i) for $i = 1, \cdots, K$. P_z is the pre-defined sampling distribution of random noise. λ is the hyperparameter to keep

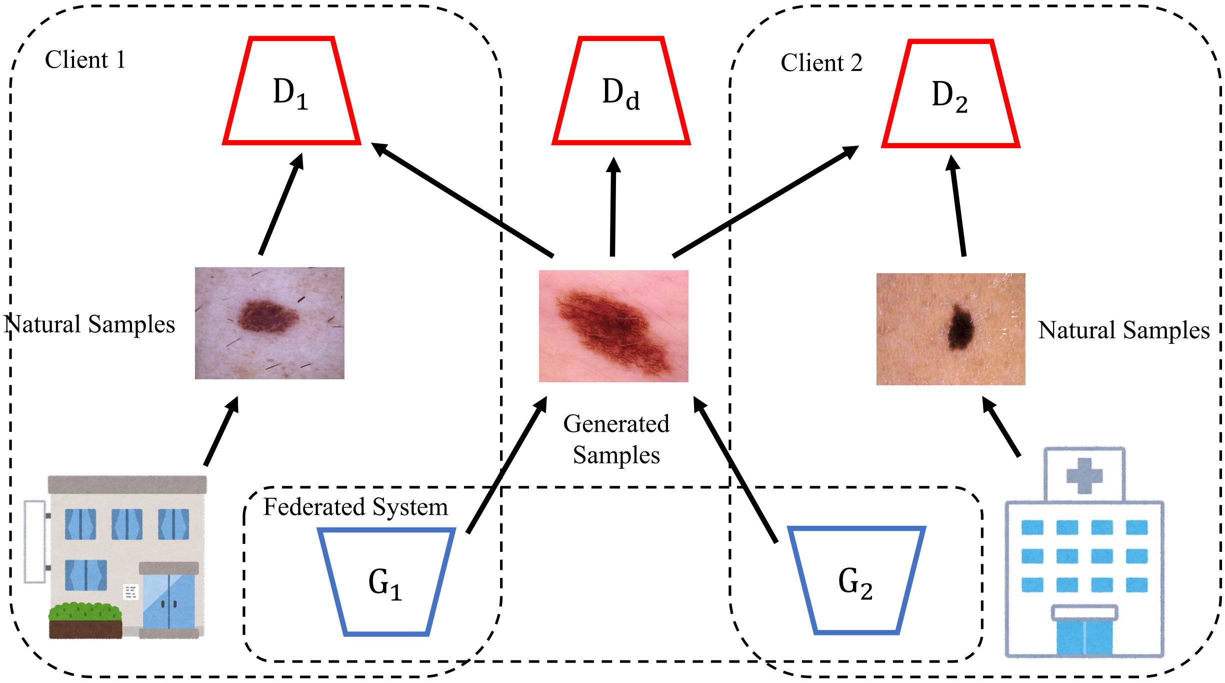


Figure 8: The FedViDA-GAN architecture

a balance between fidelity and generalization. And it is obvious that the complete value function can be summarized as the form:

$$\sum_{i=1}^{K} \overbrace{V\left(G_i, D_i\right)}^{fidelity} + \overbrace{\lambda R\left(D_d\right)}^{generalization} \tag{16}$$

Here in this simplified formula, the first term V is to optimize the generating quality, and the second term λR is to improve generalization. Overall, the optimization of FedViDA-GAN can be defined as:

$$\min_{\{G_i\}_{i=1}^{K}} \max_{D_d} \max_{\{D_i\}_{i=1}^{K}} \sum_{i=1}^{K} V\left(G_i, D_i\right) + \lambda R\left(D_d\right) \tag{17}$$

In addition, at each round of aggregation, the server also collects metadata produced by each client generator besides local model parameters. The metadata consists of an equal number of samples from all the client generators. We initialize the federated model with aggregated client parameters and retrain this model on a generalization with all client generations to obtain a bias-free model. This process can be repeated between each aggregating and the next round of broadcasting.

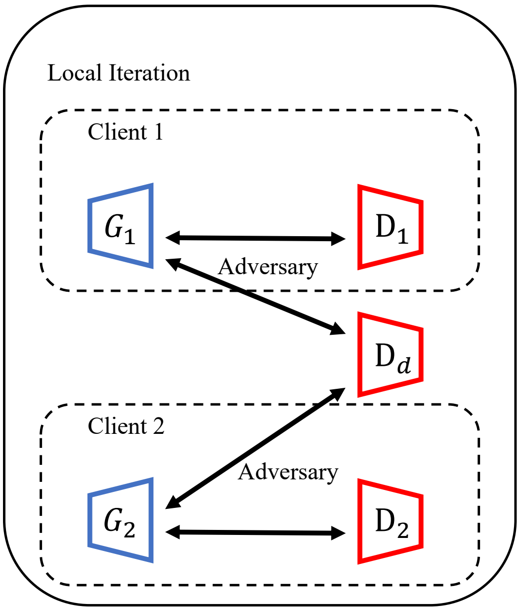


Figure 9: A Representative Round in FedViDA-GAN (Local Iteration)

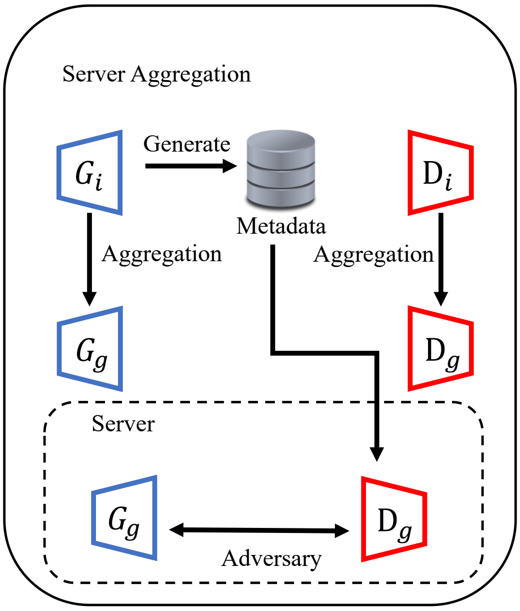


Figure 10: A Representative Round in FedViDA-GAN (Server Iteration)

4 Adversarial Domain Adaptation

In this section, we introduce an adversarial representation learning approach for our following domain adaptation task between natural data on distributed clients and virtual generated data. This approach aims to find a feature extraction ϕ, to obtain features that cannot disseminate between the target domain (the real) and source domain (the fake), Among which we can assure that the obtained feature containing no client-specific information due to that the generated contains only characteristics of commonality [17][19][27].

Our approach implements our generalization goal in the training process of common deep classifying models architecture that is trained on the labeled natural data and the unlabeled virtual data. we do not use any generated labels to avoid the appearance of error accumulation. As our training progresses, the approach promotes the emergence of features that are discriminating with the main classification task while being indiscriminate with respect to the heterogeneity between the clients. This approach does not limit the structure of the backbone classification networks and can be trained with a common backpropagation scheme and usual gradient-descent-based optimizers.

4.1 Main Idea

The most important goal of our approach is to learn a model that can do similar works on both the natural data and the generated data, and ignore the differences between these two domains. If each client i can obtain a generalizing feature extraction ϕ_i that can output homogeneous features on local natural data and shared virtual data. We can aggregate all the client feature extraction $\{\phi_i\}_{i=0}^{K}$ to obtain a federated generalizing model. Due to the aggregated model extracting only all-client features instead of client-specific features from heterogeneous samples, it is expected to achieve an outperforming generalization ability on the data from new appended domains (clients).

4.2 Mathematical Formulation of Deep Classification

As each computation of classification on client i, the action of the deep model of our main task $\mathbb{M}_i$ can be defined as:

$$\mathbb{M}_i\left(\omega_i, X_i\right) = \sigma\left(\omega_i^{\sigma}\right) \circ \phi\left(\omega_i^{\phi}, X_i\right) \tag{18}$$

where X_i means the local dataset of client i, σ and ϕ denotes the classifier part and feature extraction part of $\mathbb{M}_k$, $\circ$ stands for function composition between ϕ and σ, ω_i denotes the whole weights parameter of $\mathbb{M}_k$, accordingly, $\omega_i^{\phi} \subseteq \omega_i$ and $\omega_i^{\sigma} \subseteq \omega_i$ are subsets of ω_i that denotes parameters of the feature extraction part and the classifier part respectively.

For each input sample $x_{i,j} \in X_i$, its latent embedding of features will be computed as follows:

$$f_{i,j} = \phi\left(\omega_i^{\phi}, x_{i,j}\right) \tag{19}$$

Then the obtained $f_{i,j}$ will be fed into the classifier part σ to calculate class-wise probability scores $\overline{y_{i,j}}$ for class prediction as:

$$\overline{y_{i,j}} = \sigma\left(w_i^\sigma, f_{i,j}\right) \tag{20}$$

At last, the prediction $\overline{y_{i,j}}$ and its natural class $y_{i,j}$ will be computed with the loss function such as the cross-entropy function to solve a backward gradient for optimization to minimize the prediction difference.

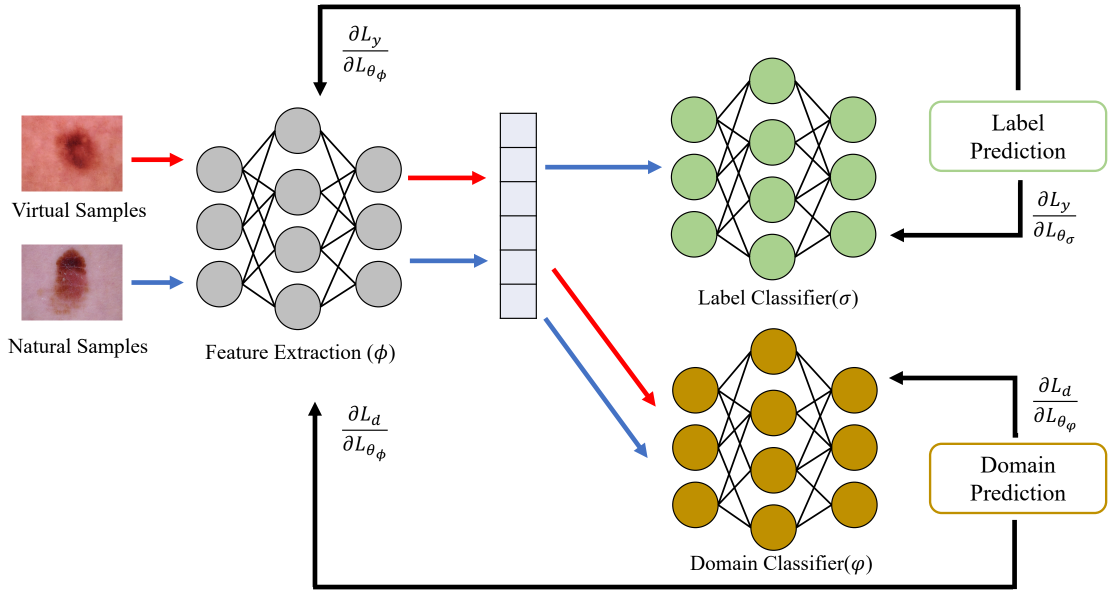
4.3 Adversarial Domain Adaptation

To achieve a homogeneous classification with both natural samples and generated samples, we append a parallel domain classifier, to work as an adversary toward the original feature extraction. The training determines whether the input features are from natural or virtual samples, forcing the feature extractor to focus on extracting features consistent with each other. the formulation of training process of client i can be summarized as:

$$\mathbb{M}_i\left(\omega_i, X_i\right) = \sigma\left(\omega_i^\sigma\right) \circ \phi\left(\omega_i^\phi, X_i\right) \tag{21}$$

$$\mathbb{A}_i\left(\omega_i', \left(X_i, \overline{S}\right)\right) = \varphi_i\left(\omega_i^{\varphi_i}\right) \circ \phi\left(\omega_i^\phi, \left(X_i, \overline{S}\right)\right) \tag{22}$$

in which $\mathbb{A}_i$ stands for the adversarial training process on client i, $\overline{S}$ denotes the shared virtual dataset across clients. φ_i stands for the domain predictor and $\omega_i^{\varphi_i}$ denotes the parameter of it. ω_i' is the new universe set of parameters of the model.



$\frac{\partial L_y}{\partial L_{\theta_\phi}}$
Virtual Samples
Natural Samples
Feature Extraction (ϕ)
Label Classifier(σ)
Label Prediction
$\frac{\partial L_y}{\partial L_{\theta_\sigma}}$
$\frac{\partial L_d}{\partial L_{\theta_\phi}}$
$\frac{\partial L_d}{\partial L_{\theta_\varphi}}$
Domain Prediction
Domain Classifier(φ)

Figure 11: The FedViDA domain adaptation architecture

As can be seen in the above formulation, we do not feed the unlabeled virtual samples into the main learning task because the performance on natural samples is actually the ultimate goal. The adversary with virtual samples is essentially just a calibration of the feature extraction networks to extract more generalized feature vectors for classification.

During the federated progress, only the main classifier σ and feature extraction ϕ will be aggregated [26][28]. The domain classifiers φ_i should keep being client-specific due to that the divergences between the shared virtual distribution and natural distribution on each client are not consistent.

5 Experiment

In this section, we comprehensively make an evaluation of our framework to demonstrate that training via domain adaptation with a homogeneous virtual source can benefit model generalization greatly. Our framework FedViDA (Federated Virtual Discriminative Adaptation) can achieve not only higher generalizing ability but also a more stable performance with the datasets of heterogeneous feature distribution.

5.1 Dataset Instructions and Experimental Settings

We decide to use the famous challenging public skin lesion dataset HAM10000 [24] (Human Against Machine with 10000 training images), which collects over 10000 dermatoscopic images from different institutions. These medical samples are acquired and stored by different modalities. It brings multiple heterogeneities to these samples. The cases comprise a representative and comprehensive selection of each significant pigmented lesion diagnostic category, which can be separated into seven imbalanced subsets. Details of the sample categories are shown in the following Table 1.

Table 1: Summary of HAM10000 dataset

Source	**License**	**Number of samples each category**						
		akiec	bcc	bkl	df	mel	nv	vasc
Rosendahl	CC BY-NC 4.0	295	296	490	30	342	803	3
ViDIR Legacy	CC BY-NC 4.0	0	5	10	4	67	350	3
ViDIR Current	CC BY-NC 4.0	32	211	475	51	680	1832	82
ViDIR Molemax	CC BY-NC 4.0	0	2	124	30	24	3720	54

Considering that the samples of three of these categories are too few, we only use the four most populated. They are:

- Lesion 0: Melanocytic nevi, abbreviated as *nv*, it is the most popular skin lesion with a collection of 6705 samples.
- Lesion 1: Melanoma, abbreviated as *mel*, contains 1113 samples in total.
- Lesion 2: Benign keratosis, abbreviated as *bkl*, contains 1099 samples.
- Lesion 3: Basal cell carcinoma, abbreviated as *bcc*, contains 514 samples.

Among these four most populated lesions, melanocytic nevi (nv) and benign keratosis (bkl) are normal benign whereas lesions of little risk, while melanoma (mel) and basal cell carcinoma (bcc) are usually signs of various kinds of skin cancer.

Therefore, we label these samples with two classes according to their potential risk:

- Class 0: Samples of benign lesions, contains *nv* and *bkl*.
- Class 1: Samples of potential skin cancer, consisting of *mel* and *bcc*.

Here we have constructed a binary classification task in a distributed system with both feature heterogeneity and imbalanced label space, to perform an evaluation of the performance of our algorithm. And it is also an extreme enough challenge for most FL algorithms.

The main goal of this work is to address the generalization bottle of FL, therefore we will not split out the testing dataset from the client within FL training. Instead, we will execute the test task on a new client outside FL systems.

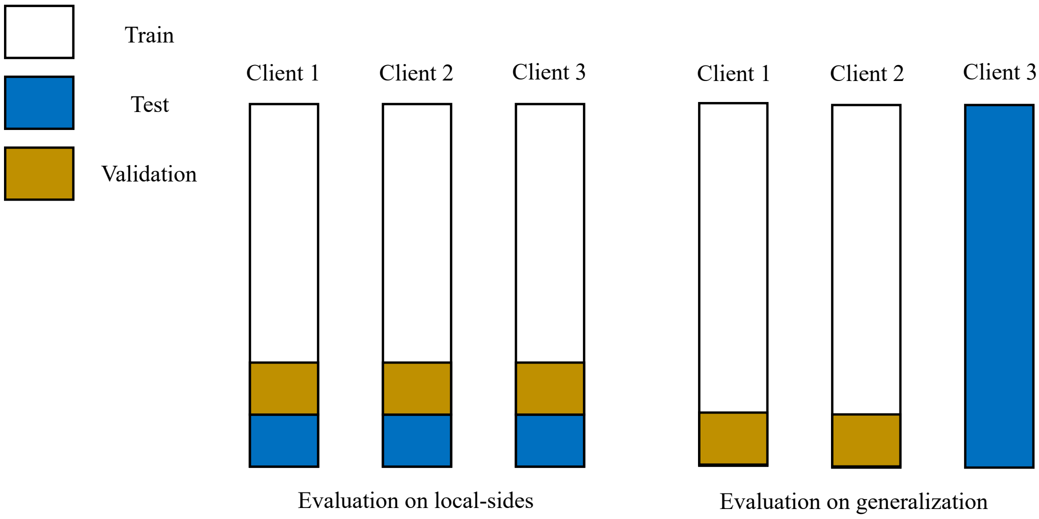


Figure 12: Evaluation on local-sides and generalization

5.2 Backbone Networks

Like other FL algorithms, FedViDA can be launched with most mainstream neural networks. we use a deep neural network of VGG-16 [23][25] as our computing model for clients and train it over 100 epochs with a pre-defined aggregation frequency. We use the cross-entropy function to calculate the loss of this multi-class classification error and update client models with a Stochastic Gradient Descent (SGD) optimizer at a learning rate of 10^{-3}.

We choose VGG-16 for our client based on two considerations. First, as a classical and well-analyzed deep convolutional neural network model, the boundary between the feature extraction part and the classifier part is very clear and well-proved. Otherwise, the split of those two parts would become something like a hyperparameter and make too many uncontrollable variables in the comparison. Second, the experiments in FedBN works have proved that batch normalization (BN) layers may cause inferiority for FL models. Choosing VGG models can help avoid this uncontrollable risk due to no BN layers contained.

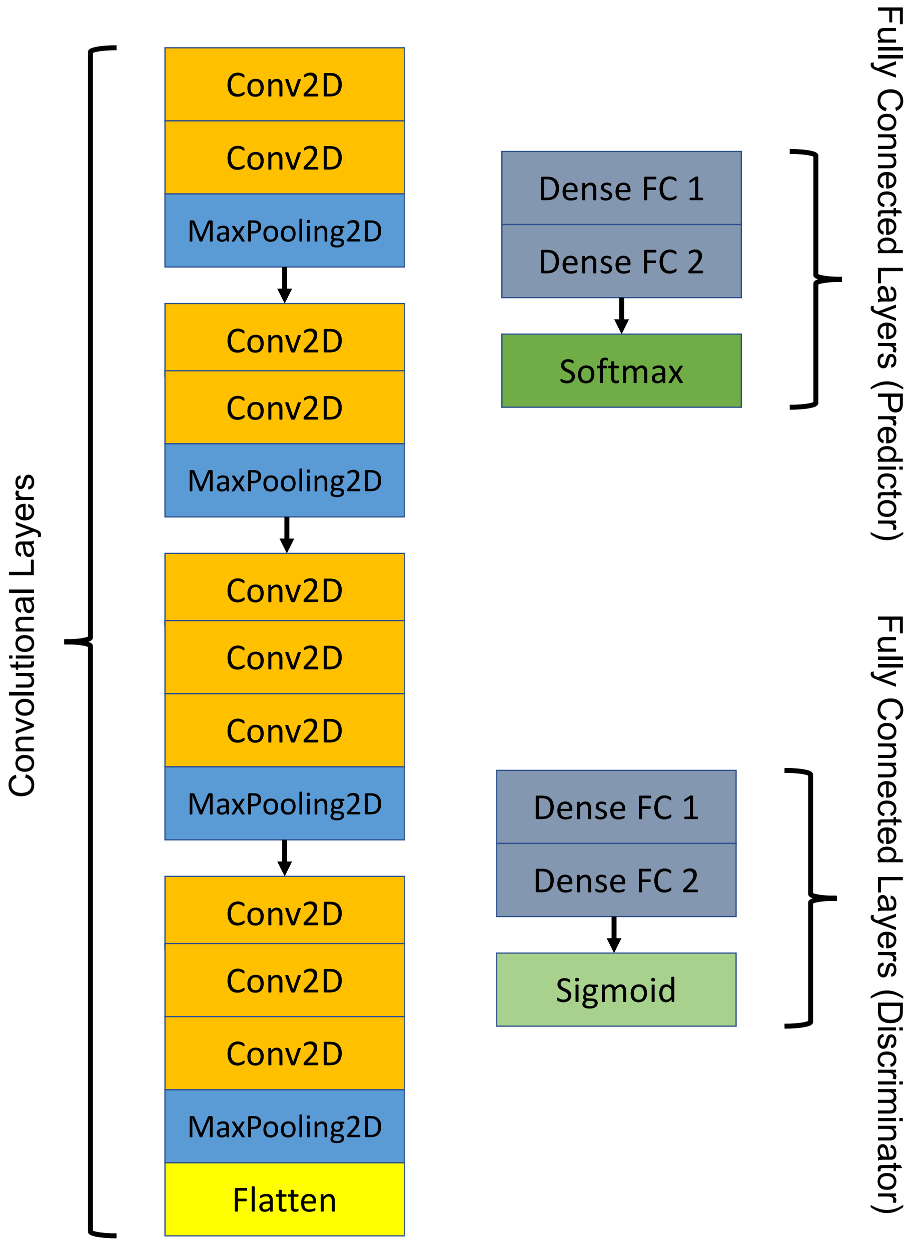


Figure 13: The Structure of VGG16 Network and Domain Adaptation Discriminator

5.3 Comparison with the SOTAs

We compare our proposed method with several leading state-of-the-art (SOTA) FL models in the aspects of solving Non-IID feature drifts while keeping generalizing performance. For the local-side model, **Fedprox** is taken into comparison. Meanwhile, **FedNova** is proposed as a general respective model to tackle global drifts. **FedAvg** is also included as the pioneering work and milestone of FL.

Table 2: Results for skin lesion image classification accuracy of FL methods

Methods	Accuracy on the testing client (%)			
	Rosendahl	ViDIR Current	ViDIR Molemax	Average
FedAvg (PMLR2017)	78.91%	70.62%	78.98%	76.17%
FedProx (MLSys2020)	78.52%	72.9%	79.31%	76.91%
FedNova (NeurIPS2020)	77.92%	76.14%	79.34%	77.80%
FedViDA (Ours)	**80.22%**	**80.17%**	**83.01%**	**81.13%**

For the skin lesion image classification accuracy reported in Table 2, due to the experiments we designed paying more attention to the generalization ability of FL models. Those server-side algorithms clearly have a greater advantage compared to those local-side ones. **FedProx** only achieve little improvements towards **FedAvg**. However, our method (**FedViDA**) can achieve higher accuracy than other FL algorithms and always has similar performance for our aggressive elimination of distributed personality while retaining commonality.

The performances of FedViDA compared with other mainstream algorithms on each client are shown in Fig 14 to Fig 16.

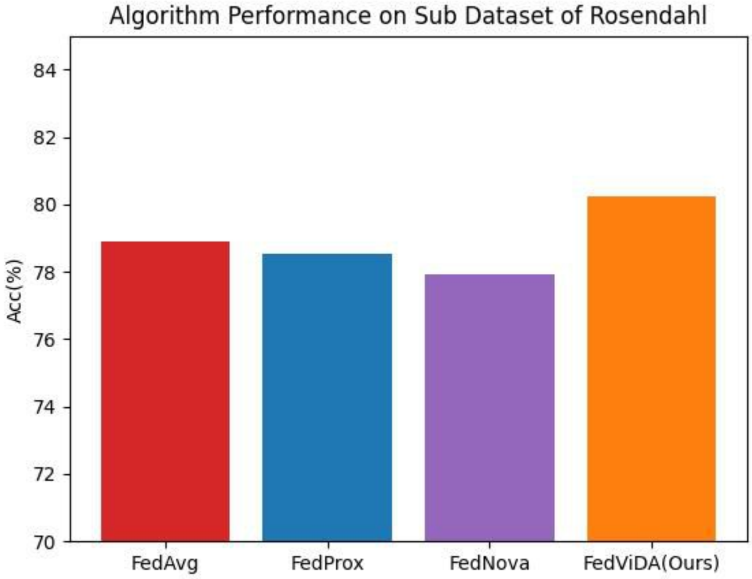


Figure 14: The Performance of FedViDA compared with other mainstream algorithms on the client of "Rosendahl"

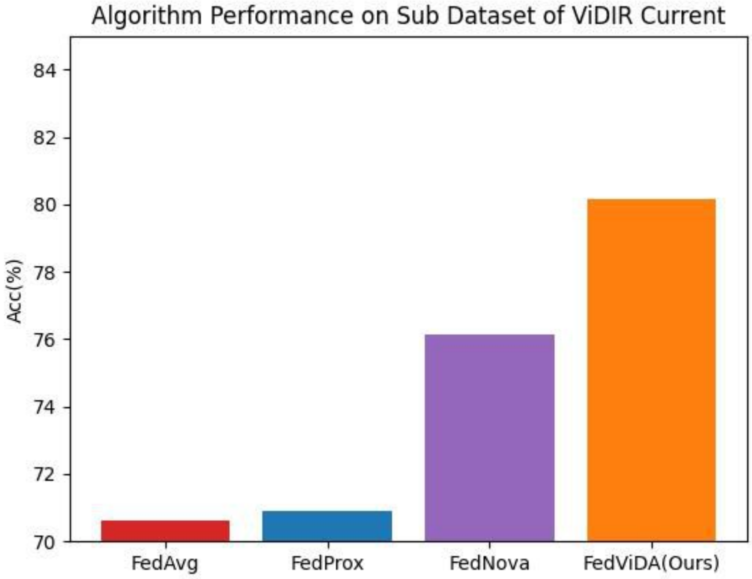


Figure 15: The Performance of FedViDA compared with other mainstream algorithms on the client of "ViDIR Current"

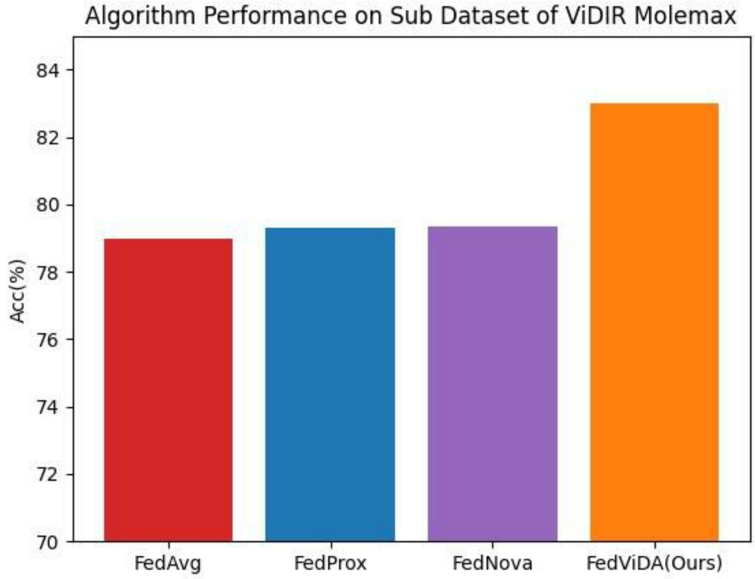


Figure 16: The Performance of FedViDA compared with other mainstream algorithms on the client of ”ViDIR Molemax”

At last, the overall performance of FedViDA compared with other mainstream algorithms on each client is shown in Fig 17.

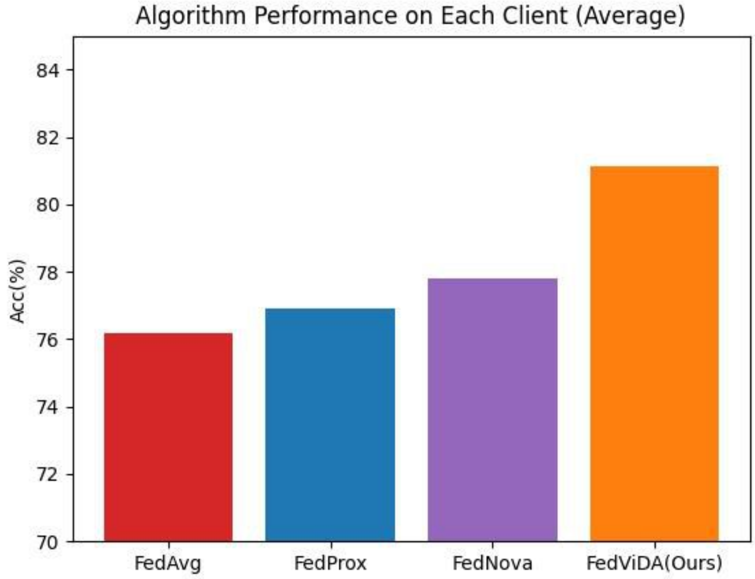


Figure 17: The Average Performance of FedViDA compared with other mainstream algorithms on clients

6 Conclusion

This thesis proposes a novel federated learning framework named FedViDA, which implements domain adaptations among no-data-sharing clients, to solve the feature drifts caused by the Non-IID distribution of medical imaging from various institutions. Due to the limitation of data access, we innovatively introduced the method of constructing virtual datasets by a distributed generation adversarial network architecture and using it as the shared source for domain generalization. In our method, domain generalization can be well finished without any real data sharing across clients. Compared to the existing methods of local calibrations, our method provides a much higher generalization ability and a steady performance in most real-world medical applications.

References

[1] McMahan, Brendan, et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

[2] Wang, Hongyi, et al. "Attack of the tails: Yes, you really can backdoor federated learning." Advances in Neural Information Processing Systems 33 (2020): 16070-16084

[3] Bagdasaryan, Eugene, et al. "How to backdoor federated learning." International Conference on Artificial Intelligence and Statistics. PMLR, 2020.

[4] Reisizadeh, Amirhossein, et al. "Robust federated learning: The case of affine distribution shifts." Advances in Neural Information Processing Systems 33 (2020): 21554-21565.

[5] Pfitzner, Bjarne, Nico Steckhan, and Bert Arnrich. "Federated learning in a medical context: a systematic literature review." ACM Transactions on Internet Technology (TOIT) 21.2 (2021): 1-31.

[6] Zhao, Yue, et al. "Federated learning with non-iid data." arXiv preprint arXiv:1806.00582 (2018).

[7] Karimireddy, Sai Praneeth, et al. "Scaffold: Stochastic controlled averaging for federated learning." International Conference on Machine Learning. PMLR, 2020.

[8] Li, Qinbin, Bingsheng He, and Dawn Song. "Model-contrastive federated learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[9] Li, Xiaoxiao, et al. "Fedbn: Federated learning on non-iid features via local batch normalization." arXiv preprint arXiv:2102.07623 (2021).

[10] Wang, Jianyu, et al. "Tackling the objective inconsistency problem in heterogeneous federated optimization." Advances in neural information processing systems 33 (2020): 7611-7623.

[11] Wang, Yujia, Lu Lin, and Jinghui Chen. "Communication-Efficient Adaptive Federated Learning." arXiv preprint arXiv:2205.02719 (2022).

[12] Jiang, Meirui, Zirui Wang, and Qi Dou. "Harmofl: Harmonizing local and global drifts in federated learning on heterogeneous medical images." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 36. No. 1. 2022.

[13] Wang, Mei, and Weihong Deng. "Deep visual domain adaptation: A survey." Neurocomputing 312 (2018): 135-153.

[14] Goodfellow, Ian, et al. "Generative adversarial networks." Communications of the ACM 63.11 (2020): 139-144.

[15] Mukherjee, Sumit, et al. ”privGAN: Protecting GANs from membership inference attacks at low cost to utility.” Proc. Priv. Enhancing Technol. 2021.3 (2021): 142-163.

[16] Ganin, Yaroslav, et al. ”Domain-adversarial training of neural networks.” The journal of machine learning research 17.1 (2016): 2096-2030.

[17] Li, Yitong, et al. ”Storygan: A sequential conditional gan for story visualization.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[18] Mirza, Mehdi, and Simon Osindero. ”Conditional Generative Adversarial Nets.” arXiv e-prints (2014): arXiv-1411.

[19] Radford, Alec, Luke Metz, and Soumith Chintala. ”Unsupervised representation learning with deep convolutional generative adversarial networks.” arXiv preprint arXiv:1511.06434 (2015).

[20] Rasouli, Mohammad, Tao Sun, and Ram Rajagopal. ”Fedgan: Federated generative adversarial networks for distributed data.” arXiv preprint arXiv:2006.07228 (2020).

[21] Mugunthan, Vaikkunth, et al. ”Bias-Free FedGAN: A Federated Approach to Generate Bias-Free Datasets.” arXiv preprint arXiv:2103.09876 (2021).

[22] Pang, Yutian, and Yongming Liu. ”Conditional generative adversarial networks (CGAN) for aircraft trajectory prediction considering weather effects.” AIAA Scitech 2020 Forum. 2020.

[23] Simonyan, Karen, and Andrew Zisserman. ”Very deep convolutional networks for large-scale image recognition.” arXiv preprint arXiv:1409.1556 (2014).

[24] Tschandl, Philipp, Cliff Rosendahl, and Harald Kittler. ”The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions.” Scientific data 5.1 (2018): 1-9.

[25] Guan, Qing, et al. ”Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study.” Journal of Cancer 10.20 (2019): 4876.

[26] Li, Xukun, et al. ”Identifying disaster damage images using a domain adaptation approach.” Proceedings of the 16th International Conference on Information Systems for Crisis Response And Management. 2019.

[27] Zhang, Changchun, and Qingjie Zhao. ”Attention guided for partial domain adaptation.” Information Sciences 547 (2021): 860-869.

[28] Luo, Zimeng, et al. ”Deep unsupervised domain adaptation for face recognition.” 2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018). IEEE, 2018.

[29] Zhang, Han, et al. ”Self-attention generative adversarial networks.” International conference on machine learning. PMLR, 2019.

[30] Yue, Yunpeng, et al. ”Generation of High-Precision Ground Penetrating Radar Images Using Improved Least Square Generative Adversarial Networks.” Remote Sensing 13.22 (2021): 4590.