Data-Free Black-Box Federated Learning via Zeroth-Order Gradient Estimation

Xinge Ma, Jin Wang*, Xuejie Zhang

School of Information Science and Engineering
Yunnan University
Kunming, China
maxinge@mail.ynu.edu.cn, {wangjin, xjzhang}@ynu.edu.cn

Abstract

Federated learning (FL) enables decentralized clients to collaboratively train a global model under the orchestration of a central server without exposing their individual data. However, the iterative exchange of model parameters between the server and clients imposes heavy communication burdens, risks potential privacy leakage, and even precludes collaboration among heterogeneous clients. Distillation-based FL tackles these challenges by exchanging low-dimensional model outputs rather than model parameters, yet it highly relies on a task-relevant auxiliary dataset that is often not available in practice. Data-free FL attempts to overcome this limitation by training a server-side generator to directly synthesize task-specific data samples for knowledge transfer. However, the update rule of the generator requires clients to share on-device models for white-box access, which greatly compromises the advantages of distillation-based FL. This motivates us to explore a data-free and black-box FL framework via Zeroth-order Gradient Estimation (FedZGE), which estimates the gradients after flowing through on-device models in a black-box optimization manner to complete the training of the generator in terms of fidelity, transferability, diversity, and equilibrium, without involving any auxiliary data or sharing any model parameters, thus combining the advantages of both distillation-based FL and data-free FL. Experiments on large-scale image classification datasets and network architectures demonstrate the superiority of FedZGE in terms of data heterogeneity, model heterogeneity, communication efficiency, and privacy protection.

Code — https://github.com/maxinge8698/FedZGE

Introduction

The remarkable success of deep neural networks is largely attributed to data-driven centralized training schemes, which consolidate data scattered across decentralized clients onto a central server to build high-performing models. However, growing data concerns and regulations impose stringent limitations on data collection and transmission, leading to isolated data silos where clients keep their data locally to ensure privacy and security. To break down this dilemma, federated learning (FL) (McMahan et al. 2017) has emerged as a privacy-preserving decentralized training paradigm that delivers a "centralized" model by periodically sharing the parameters of models locally trained by individual clients on their private data with a central server for aggregation. However, the iterative exchange of model parameters between the server and clients exposes FL to several inherent challenges: 1) *Degeneration to data heterogeneity* resulting from the significant divergence between model parameters caused by the non-independent and identically distributed (non-IID) data among clients (Zhu et al. 2021; Ma et al. 2022); 2) *Failure to model heterogeneity* when clients individually design personalized models to meet distinct specifications rather than replicating a homogeneous model architecture for parameter aggregation (Gao, Yao, and Yang 2022; Ye et al. 2023); 3) *Exorbitant communication overhead* that scales up with the number of model parameters hinders its application to larger models or client populations (Konečný et al. 2016; Almanifi et al. 2023); 4) *Vulnerability to privacy attacks* where the shared model parameters can be maliciously exploited to reconstruct clients' private data in a white-box manner (Lyu et al. 2022; Sharma and Marchang 2024).

To tackle these challenges, distillation-based FL (Li and Wang 2019; Chang et al. 2019; Hu et al. 2021; Itahara et al. 2021; Gong et al. 2021; Cheng et al. 2021; Sattler et al. 2022; Gong et al. 2022; Ma et al. 2023) extends knowledge distillation (KD) (Hinton, Vinyals, and Dean 2015) to the context of FL by exchanging model outputs associated with a public auxiliary dataset for knowledge transfer, offering a diverse range of benefits such as model-agnostic collaboration, lower communication overhead, and less exposure of sensitive information. However, a desirable auxiliary dataset is not always available since its construction requires careful deliberation and even prior knowledge about clients' private data to achieve satisfactory performance, which is inconsistent with the privacy-preserving nature of FL.

Instead of assuming a pre-determined auxiliary dataset, data-free FL (Zhu, Hong, and Zhou 2021; Zhang et al. 2022; Zhang, Wu, and Yuan 2022; Luo et al. 2023) eliminates the need for auxiliary data by training a server-side generator to directly synthesize task-specific data samples for knowledge transfer. However, the update rule of the generator depends on the availability of clients' local models, which involves backpropagating first through local models and then through the generator to yield the gradients. Consequently,

*Corresponding author.

Taxonomy	Method	Auxiliary Data	Model Heterogeneity	Black-box Access	Exchanged Information	
					Downlink	Uplink
parameter-based FL	FedAvg	-	✗	✗	model parameter	model parameter
distillation-based FL	MHAT	labeled	✔	✔	auxiliary data, model output	model output
	DS-FL	unlabeled	✔	✔	auxiliary data, model output	model output
data-free FL	FedGen	data-free	✗	✗	model parameter	model parameter
	FedFTG	data-free	✗	✗	model parameter	model parameter
	DFRD	data-free	✗	✗	model parameter	model parameter
	FedZKT	data-free	✔	✗	model parameter	model parameter
data-free black-box FL	FedZGE	data-free	✔	✔	synthetic data, model output	model output

Table 1: Comparison of FedZGE with related work.

it is inevitable for clients to share on-device models with the server for white-box access, which greatly compromises the advantages of distillation-based FL.

To combine the advantages of both distillation-based FL and data-free FL, this motivates us to explore a data-free and black-box FL framework called FedZGE, which leverages Zeroth-order Gradient Estimation (Nesterov and Spokoiny 2017) to complete the training of the generator in terms of fidelity, transferability, diversity, and equilibrium by estimating the gradients after flowing through on-device models in a black-box optimization manner, without involving any auxiliary data or sharing any model parameters. In summary, this paper makes the following contributions:

- We propose FedZGE, a data-free black-box FL framework that is granted the ability to launch data-free black-box knowledge transfer across heterogeneous clients via zeroth-order gradient estimation.
- Contrary to distillation-based FL approaches that rely on task-relevant auxiliary datasets, FedZGE trains a generator on the server to directly synthesize faithful, transferable, diverse, and balanced data samples for effective knowledge transfer in a data-free manner.
- Contrary to data-free FL approaches that require clients to share on-device models, FedZGE estimates the gradients after flowing through these on-device models in a black-box optimization manner to complete the training of the generator, without sharing any model parameters.
- Empirical experiments with extensive analysis demonstrate the superiority of FedZGE in terms of data heterogeneity, model heterogeneity, communication efficiency, and privacy protection.

Related Work

Parameter-based Federated Learning

Parameter-based FL represented by FedAvg (McMahan et al. 2017) is a classical FL framework where a central server distributes the global model to individual clients for local training on their respective data, and then the resulting model parameters are sent back for aggregation. Despite its great potential in privacy-sensitive domains (Liang, Pan, and Ming 2021; Liu et al. 2022; Li et al. 2023), parameter-based FL still faces challenges such as data heterogeneity, model heterogeneity, communication efficiency, and privacy leakage (Li et al. 2020; Zhang et al. 2021; Kairouz et al. 2021). In response to these challenges, numerous variants of FedAvg have emerged (Tan et al. 2022; Alam et al. 2022; Bibikar et al. 2022; Noble, Bellet, and Dieuleveut 2022). Unfortunately, they still rely on model parameter exchange and thus lack the ability to address all these issues simultaneously.

Distillation-based Federated Learning

Distillation-based FL (Mora et al. 2022; Wu et al. 2023; Qin et al. 2024) tackles these challenges by leveraging KD to aggregate local models in a black-box manner. However, it relies heavily on a public auxiliary dataset to synchronize the knowledge produced by these models. Earlier work constructed a task-relevant labeled (Li and Wang 2019; Hu et al. 2021; Cheng et al. 2021) or unlabeled (Chang et al. 2019; Itahara et al. 2021; Sattler et al. 2022) dataset with a distribution similar to that of clients' private data as auxiliary data. However, this may explicitly expose clients' private data. Recent efforts (Gong et al. 2021, 2022; Ma et al. 2023) have relaxed this restriction to a cross-domain unlabeled dataset, whereas finding a desirable auxiliary dataset remains challenging in practice.

Data-Free Federated Learning

Inspired by the recent success of data-free KD (Lopes, Fenu, and Starner 2017; Fang et al. 2019; Micaelli and Storkey 2019; Fang et al. 2022), data-free FL equips the server with a generator to directly synthesize the data samples necessary for knowledge transfer. Most existing approaches (Zhu, Hong, and Zhou 2021; Zhang et al. 2022; Luo et al. 2023) are essentially fine-tuning methods that train the generator using the ensemble of local models to further improve the performance of the preliminary global model aggregated from FedAvg. To accommodate heterogeneous clients, recent work (Zhang, Wu, and Yuan 2022) adopts them as data-free methods that transfer knowledge from local models to the global model without parameter aggregation. However, sharing clients' local models with the server is still essential in these frameworks since the training of the generator depends on the availability of local models, which contradicts the benefits of distillation-based FL. A detailed comparison of FedZGE with related work is provided in Table 1.

Methodology

Problem Statement

Consider a FL setting consisting of a central server and K decentralized clients. Each client $k \in \{1, 2, ..., K\}$ locally hosts a private dataset $\mathcal{D}_k = \{(x_i^k, y_i^k)\}_{i=1}^{N_k}$ and an on-device model f_k parameterized by θ_k, where $x_i^k \in \mathbb{R}^d$ and $y_i^k \in \{1, 2, ..., \mathcal{C}\}$ are the i-th sample and its corresponding label, $\mathcal{C}$ is the number of classes, and N_k is the local sample size. The goal is to learn a global model f parameterized by θ on the server through multiple rounds of collaborative training across clients while keeping their private data decentralized.

To overcome the limitations imposed by model parameter exchange in FL, distillation-based FL resorts to a publicly available auxiliary dataset $\mathcal{D}_p = \{x_i^p\}_{i=1}^{N_p}$ for black-box model aggregation via KD. Specifically, in each communication round, the server selects a random set of available clients $\mathcal{K} \subseteq \{1, 2, ..., K\}$ based on a sampling fraction ε to participate in the FL training, where $|\mathcal{K}| = \lceil \varepsilon \cdot K \rceil$. Each participating client $k \in \mathcal{K}$ performs local training to update its local model f_k based on its own private data $\mathcal{D}_k$ via the following local optimization objective:

$$\mathcal{L}_k(\mathcal{D}_k; \theta_k) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}_k} \left[\mathrm{CE}(f_k(x; \theta_k), y)\right], \quad (1)$$

where $\mathrm{CE}(\cdot)$ is the cross-entropy loss used to measure training error for a classification task. Subsequently, it downloads the auxiliary data $x_p \sim \mathcal{D}_p$ from the server and sends back the output $f_k(x_p; \theta_k)$ of its local model f_k on x_p. The server then aggregates these local outputs into the ensemble output $\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(x_p; \theta_k)$ through weighted averaging, which assembles knowledge from other peer clients and thus can be used to distill the global model to preserve the knowledge of local models via the following global distillation objective:

$$\mathcal{L}_f(\mathcal{D}_p; \theta) = \mathop{\mathbb{E}}_{x_p\sim\mathcal{D}_p} \left[\mathrm{KL}(\sigma(\sum_{k\in\mathcal{K}} \tfrac{N_k}{N} f_k(x_p; \theta_k); \tau) || \sigma(f(x_p; \theta); \tau))\right], \quad (2)$$

where $N = \sum_{k\in\mathcal{K}} N_k$ is the cumulative sample size, $\sigma(\cdot; \tau)$ is a softmax function with temperature τ to control the distributional smoothness, and $\mathrm{KL}(\cdot||\cdot)$ is the Kullback-Leibler divergence (Kullback and Leibler 1951) used to measure the discrepancy between the global model and the ensemble of local models. However, a task-relevant auxiliary dataset may not always be available in practice.

Instead, data-free FL introduces a conditional generator G parameterized by θ_G on the server to capture the training space of clients' local models in a data-free manner. Specifically, taking a d_z-dimensional random noise z sampled from a standard Gaussian distribution $\mathcal{N}(0, \mathbf{1})$ as the input and a random label $\hat{y}$ sampled from a uniform distribution $\mathcal{U}(1, \mathcal{C})$ as the conditional context, the conditional generator G is trained to synthesize a categorical data $\hat{x} = G(z, \hat{y}; \theta_G)$ by minimizing the cross-entropy loss between the ensemble output and the random label, as follows:

$$\mathcal{L}_G^{fid}(\hat{x}; \theta_G) = \mathop{\mathbb{E}}_{\substack{\hat{x}\sim G(z,\hat{y};\theta_G)\\ \hat{y}\sim\mathcal{U}(1,\mathcal{C})}} \left[\mathrm{CE}(\sum_{k\in\mathcal{K}} \tfrac{N_k}{N} f_k(\hat{x}; \theta_k), \hat{y})\right], \quad (3)$$

which allows $\hat{x}$ to be categorized to one specific class $\hat{y}$ with a high probability, thereby approximating clients' private data under label $\hat{y}$ from a global perspective. Meanwhile, the generated synthetic data sample $\hat{x}$ is used to transfer knowledge from local models to the global model, as follows:

$$\mathcal{L}_f(\hat{x}; \theta) = \mathop{\mathbb{E}}_{\hat{x}\sim G(z,\hat{y};\theta_G)} \left[\mathrm{KL}(\sigma(\sum_{k\in\mathcal{K}} \tfrac{N_k}{N} f_k(\hat{x}; \theta_k); \tau) || \sigma(f(\hat{x}; \theta); \tau))\right]. \quad (4)$$

The generator and the global model are alternately trained on the server. However, the update rule of the generator depends on the availability of local models and requires clients to upload their local models to the server for white-box access, which greatly compromises the advantages of distillation-based FL. Our proposed FedZGE aims to address this issue.

Reconstruction of Generator Training

Although the training objective in Eq. (3) ensures the fidelity of the generated synthetic data samples, they may not be sufficiently transferable, diverse, and balanced due to the lack of explicit intervention in the training process of the generator. Considering that less useful information is transferred from model outputs than from model parameters, it is imperative to improve the quality of generated synthetic data samples so that the global model is able to capture more useful information from local models to better exploit the unique advantages of KD. Therefore, we systematically reconstruct the training of the generator in terms of fidelity, transferability, diversity, and equilibrium.

Improvement of Transferability. To effectively transfer knowledge from local models to the global model, one solution is to improve the transferability of synthetic data samples. Enlightened by the success of generative adversarial networks (GANs) (Goodfellow et al. 2014; Mirza and Osindero 2014; Radford, Metz, and Chintala 2015), we introduce an adversarial loss to encourage the generator G to generate difficult data samples for the training of the global model by maximizing the disagreement between the global model f and the ensemble of local models $\{f_k\}_{k\in\mathcal{K}}$, as follows:

$$\mathcal{L}_G^{adv}(\hat{x}; \theta_G) = \mathop{\mathbb{E}}_{\hat{x}\sim G(z,\hat{y};\theta_G)} \left[-\mathrm{KL}(\sigma(\sum_{k\in\mathcal{K}} \tfrac{N_k}{N} f_k(\hat{x}; \theta_k); \tau) || \sigma(f(\hat{x}; \theta); \tau))\right]. \quad (5)$$

Meanwhile, the generated synthetic data samples need to ensure knowledge consistency between the global model and local models by minimizing the disagreement, as shown in Eq. (4). As a result, the goals of G and f are to maximize and minimize the disagreement between f and $\{f_k\}_{k\in\mathcal{K}}$, respectively, with the following adversarial game:

$$\min_{\theta} \max_{\theta_G} \mathop{\mathbb{E}}_{\hat{x}\sim G(z,\hat{y};\theta_G)} \left[\mathrm{KL}(\sigma(\sum_{k\in\mathcal{K}} \tfrac{N_k}{N} f_k(\hat{x}; \theta_k); \tau) || \sigma(f(\hat{x}; \theta); \tau))\right]. \quad (6)$$

Promotion of Diversity. The conditional generator G may experience the mode collapse problem (Mao et al. 2019) during training since it tends to learn from the conditional context that determines the main content while ignoring the latent noise that induces diverse outputs, resulting in data samples that are often generated from only a single or a few modes of the distribution. To mitigate this problem, we add a diversity loss to the generator training to promote the diversity and dispersion of the synthetic data samples $\{\hat{x}_i\}_{i=1}^B$ generated in each batch over the data space, as follows:

$$\mathcal{L}_G^{div}(\hat{x}; \theta_G) = \mathop{\mathbb{E}}_{\hat{x}\sim G(z,\hat{y};\theta_G)} \left[e^{\frac{1}{B*B} \sum_{i,j\in\{1,2,...,B\}} -||\hat{x}_i - \hat{x}_j||_2 * ||z_i - z_j||_2}\right], \quad (7)$$

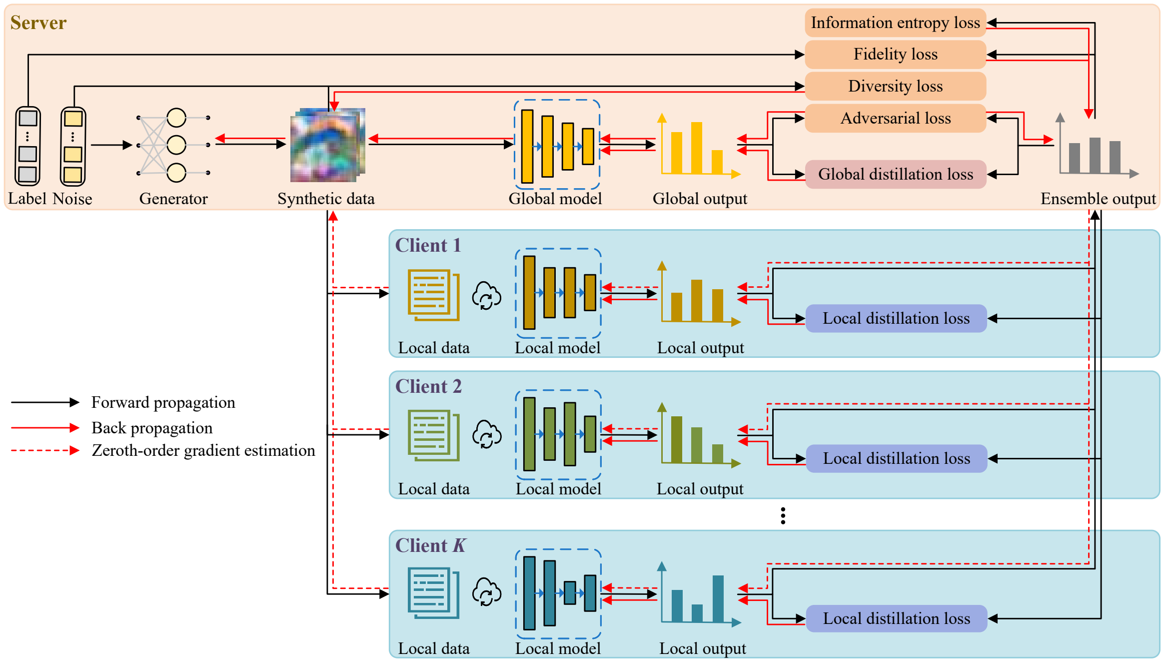


Figure 1: Overview of the proposed FedZGE framework.

where B is the batch size. This explicitly maximizes the distance between the generated data and the corresponding latent noises to increase the chances of generating data from different modes and thus synthesizing diverse data samples.

Equilibrium of Distribution. Uniformly sampling of random labels alone does not guarantee that the generated synthetic data samples are balanced since the overall predicted probability of ensemble outputs is higher on some classes and lower on the others (Chen et al. 2019; Zhang, Chen, and Lyu 2023). To this end, we train the generator to maximize the information entropy loss between the synthetic data samples generated in each batch, inducing them to be uniformly distributed across classes. Specifically, we calculate the frequency distribution of these synthetic data samples $\{\hat{x}_i\}_{i=1}^B$ belonging to each class $c \in \{1, 2, ..., \mathcal{C}\}$ from their corresponding ensemble outputs $\{\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(\hat{x}_i;\theta_k)\}_{i=1}^B$, as follows:

$$\mathbf{p} = \frac{1}{B}\sum_{i=1}^{B} \sigma(\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(\hat{x}_i;\theta_k)), \tag{8}$$

where $\sigma(\cdot)$ is a standard softmax function. The resulting probability distribution $\mathbf{p} = [p_1, p_2, ..., p_{\mathcal{C}}] \in \mathbb{R}^{\mathcal{C}}$ symbolizes the probability that each batch of synthetic data samples will be generated on each class. By maximizing the information entropy loss, as follows:

$$\mathcal{L}_G^{info}(\hat{x};\theta_G) = -H(\mathbf{p}) = \sum_{c=1}^{\mathcal{C}} p_c \log(p_c), \tag{9}$$

each element p_c in $\mathbf{p}$ will be equal to $\frac{1}{\mathcal{C}}$ when the loss reaches the maximum, which implies that the generator can generate data from each class with approximately the same probability, resulting in a balanced batch of synthetic data samples.

Overall Training Objective. By combining the above three losses with the original training loss (*i.e.*, fidelity loss $\mathcal{L}_G^{fid}(\hat{x};\theta_G)$ in Eq. (3)), the generator can be encouraged to synthesize faithful, transferable, diverse, and balanced data samples for effective knowledge transfer, as follows:

$$\mathcal{L}_G(\hat{x};\theta_G) = \mathcal{L}_G^{fid} + \beta_1\mathcal{L}_G^{adv} + \beta_2\mathcal{L}_G^{div} + \beta_3\mathcal{L}_G^{info}, \tag{10}$$

where β_1, β_2, and β_3 are the tunable scaling factors.

Generator Training via Zeroth-Order Gradient Estimation

As shown in Figure 1, $\mathcal{L}_f(\hat{x};\theta)$ in Eq. (4) and $\mathcal{L}_G(\hat{x};\theta_G)$ in Eq. (10) are two loss functions that we wish to minimize over the d_f- and d_G-dimensional optimization variables $\theta \in \mathbb{R}^{d_f}$ and $\theta_G \in \mathbb{R}^{d_G}$, respectively. Accordingly, training the generator G requires using gradient descent to minimize the loss function $\mathcal{L}_G(\hat{x};\theta_G)$ with respect to θ_G, as follows:

$$\theta_G \leftarrow \theta_G - \eta_G\nabla_{\theta_G}\mathcal{L}_G, \tag{11}$$

where η_G is the learning rate of the generator. Similarly, training the global model f requires minimizing the loss function $\mathcal{L}_f(\hat{x};\theta)$ with respect to θ, as follows:

$$\theta \leftarrow \theta - \eta\nabla_\theta\mathcal{L}_f, \tag{12}$$

where η is the learning rate of the global model. In the global model training, it is natural to access the parameters of the global model on the server for backpropagation. However, in the generator training, according to the chain rule, it requires backpropagating through local models on individual clients to compute the derivatives of the loss function $\mathcal{L}_G(\hat{x};\theta_G)$ with respect to θ_G, as follows:

$$\nabla_{\theta_G}\mathcal{L}_G = \frac{\partial\mathcal{L}_G}{\partial\theta_G} = \frac{\partial\mathcal{L}_G}{\partial(\sum\limits_{k\in\mathcal{K}}\frac{N_k}{N}f_k(\hat{x};\theta_k))} \times \frac{\partial(\sum\limits_{k\in\mathcal{K}}\frac{N_k}{N}f_k(\hat{x};\theta_k))}{\partial\theta_G}. \tag{13}$$

Existing data-free FL approaches assume that clients must share their local models with the server for white-box access to perform the gradient-based first-order optimization for the generator training. However, it is not desirable due to communication burdens and privacy concerns. Instead, treating these local models as on-device black boxes and sharing only their model outputs is consistent with distillation-based FL. Unfortunately, this prevents the computation of first-order gradients for backpropagation. To bridge this gap, we complete the generator training from the perspective of black-box optimization via zeroth-order gradient estimation (Nesterov and Spokoiny 2017).

However, directly estimating the gradients of the loss function $\mathcal{L}_G(\hat{x};\theta_G)$ with respect to the generator parameters $\theta_G \in \mathbb{R}^{d_G}$ is expensive and inefficient considering millions of parameters accompany the generator (Liu et al. 2018). To overcome the dimensionality curse, we instead choose to estimate the gradients of $\mathcal{L}_G(\hat{x};\theta_G)$ with respect to the synthetic data $\hat{x} \in \mathbb{R}^d$ produced by the generator G since it has a significantly lower dimensionality and is independent of the scaling of the generator, *i.e.*, $d \ll d_G$, as follows:

$$\nabla_{\theta_G}\mathcal{L}_G = \frac{\partial \mathcal{L}_G}{\partial \theta_G} = \frac{\partial \mathcal{L}_G}{\partial \hat{x}} \times \frac{\partial \hat{x}}{\partial \theta_G}, \tag{14}$$

where the second term can be directly computed by performing backpropagation through G, as follows:

$$\frac{\partial \hat{x}}{\partial \theta_G} = \frac{\partial G(z,\hat{y};\theta_G)}{\partial \theta_G}. \tag{15}$$

However, the computation of the first term still requires access to the parameters of local models, as follows:

$$\frac{\partial \mathcal{L}_G}{\partial \hat{x}} = \frac{\partial \mathcal{L}_G}{\partial(\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(\hat{x};\theta_k))} \times \frac{\partial(\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(\hat{x};\theta_k))}{\partial \hat{x}}. \tag{16}$$

To solve Eq. (16), we leverage zeroth-order gradient estimation that interacts with the loss function $\mathcal{L}_G(\hat{x};\theta_G)$ by simply sending inputs along a set of random direction variables (*i.e.*, $\hat{x}$ and $\{\hat{x}+\epsilon u_i\}_{i=1}^q$) to on-device local models and receiving the corresponding model outputs (*i.e.*, $\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(\hat{x};\theta_k)$ and $\{\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(\hat{x}+\epsilon u_i;\theta_k)\}_{i=1}^q$) to approximate its first-order gradients with respect to the synthetic data $\hat{x}$ through finite differences, as follows:

$$\frac{\partial \mathcal{L}_G}{\partial \hat{x}} \approx \tilde{\nabla}_{\hat{x}}\mathcal{L}_G = \frac{1}{q}\sum_{i=1}^{q} \frac{d(\mathcal{L}_G(\hat{x}+\epsilon u_i;\theta_G) - \mathcal{L}_G(\hat{x};\theta_G))}{\epsilon} u_i, \tag{17}$$

where d is the dimensionality of the synthetic data $\hat{x}$, u_i is a randomized perturbation variable drawn from a standard Gaussian distribution $\mathcal{N}(0,\mathbf{1})$, $\epsilon > 0$ is a smoothing parameter that controls the perturbation size, q is the number of random perturbation directions used to acquire finite differences. Consequently, we can compute a biased estimation $\tilde{\nabla}_{\theta_G}\mathcal{L}_G$ as an approximation for the true gradients $\nabla_{\theta_G}\mathcal{L}_G$, as follows:

$$\nabla_{\theta_G}\mathcal{L}_G \approx \tilde{\nabla}_{\theta_G}\mathcal{L}_G = \tilde{\nabla}_{\hat{x}}\mathcal{L}_G \times \frac{\partial G(z,\hat{y};\theta_G)}{\partial \theta_G}, \tag{18}$$

which can then be used to complete the update of the generator through gradient descent, as follows:

$$\theta_G \leftarrow \theta_G - \eta_G \tilde{\nabla}_{\theta_G}\mathcal{L}_G. \tag{19}$$

As a result, the prerequisite of sharing model parameters between the server and clients is eliminated by simply downloading the synthetic data and its perturbations from the server to clients and uploading the corresponding model outputs from clients to the server. It is worth noting that clients only need to perform forward propagation in this process, and thus no excessive computational burden is imposed.

Mitigation of Data Heterogeneity

Only refining the global model may not adequately address the challenge of data heterogeneity since the non-IID data distribution among clients is not mitigated. To this end, we leverage the balanced synthetic data samples and the corresponding ensemble outputs as additional supervision for local distillation, aiming to smooth the non-IID data distributions among local models and narrow their representation gaps towards global generalization. Specifically, after transferring knowledge from local models to the global model on the server, the ensemble output also needs to be transmitted back to clients, with which each client can further regulate its local model via the following local distillation objective:

$$\mathcal{L}'_k(\hat{x};\theta_k) = \mathop{\mathbb{E}}_{\hat{x}\sim G(z,\hat{y};\theta_G)}\left[\mathrm{KL}(\sigma(\sum_{k\in\mathcal{K}} \frac{N_k}{N} f_k(\hat{x};\theta_k);\tau)||\sigma(f_k(\hat{x};\theta_k);\tau))\right]. \tag{20}$$

In this way, the central server can coordinate decentralized clients to collaboratively train a global model by aggregating their distributed knowledge without involving auxiliary data or sharing on-device models. See Appendix A for detailed algorithmic procedures of the proposed FedZGE framework.

Experiments

Experimental Setup

Datasets. The evaluation is conducted on two image classification datasets commonly used in FL research: 1) **CIFAR-10** (Krizhevsky 2009); 2) **CIFAR-100** (Krizhevsky 2009). To simulate data heterogeneity among clients, we follow prior work (Hsu, Qi, and Brown 2019) to heterogeneously partition the training set of each dataset among clients using a Dirichlet distribution $Dir(\alpha)$, where α is a concentration parameter that controls the degree of non-IID, with smaller values indicating more heterogeneous data distribution. To ensure reliable performance evaluation, we run the experiments three times with different random seeds and report the average accuracy with standard deviation of the global model on the original test set. See Appendix B for more dataset statistics and partitioning details.

Baselines. We comprehensively compare FedZGE with three types of FL frameworks: 1) **parameter-based FL approaches**, including FedAvg (McMahan et al. 2017), which aggregates knowledge through iterative model parameter exchange; 2) **distillation-based FL approaches**, including MHAT (Hu et al. 2021) and DS-FL (Itahara et al. 2021), which rely on a labeled and an unlabeled auxiliary dataset, respectively, to iteratively exchange model outputs for knowledge transfer; 3) **data-free FL approaches**, including FedGen (Zhu, Hong, and Zhou 2021), FedFTG (Zhang et al. 2022), DFRD (Luo et al. 2023), and FedZKT (Zhang, Wu, and Yuan 2022), which still require clients to share their local models for data-free knowledge transfer.

Method	ResNet-18			ResNet-34			ResNet-50		
	α=1	α=0.1	Comm	α=1	α=0.1	Comm	α=1	α=0.1	Comm
FedAvg	80.42±0.34	68.90±0.51	83.31	81.36±0.19	69.48±0.25	158.62	82.01±0.16	70.12±0.41	175.30
MHAT	77.66±0.18	65.17±0.36	5.76	78.29±0.24	66.06±0.36	5.76	78.83±0.09	66.53±0.19	5.76
DS-FL	77.35±0.23	64.75±0.29	5.76	78.06±0.33	65.83±0.60	5.76	78.57±0.30	66.30±0.27	5.76
FedGen	81.29±0.31	69.97±0.40	90.92	82.33±0.12	71.48±0.43	166.23	82.96±0.17	72.10±0.27	182.91
FedFTG	81.44±0.17	70.37±0.37	83.31	82.93±0.30	72.07±0.19	158.62	83.38±0.20	72.50±0.54	175.30
DFRD	81.78±0.11	70.86±0.24	83.31	83.10±0.22	72.31±0.25	158.62	83.59±0.12	72.72±0.36	175.30
FedZKT	81.20±0.16	70.06±0.32	83.31	82.64±0.17	71.78±0.24	158.62	83.21±0.31	72.28±0.51	175.30
FedZGE	**82.19±0.20**	**71.35±0.28**	**63.17**	**83.40±0.12**	**72.65±0.18**	**63.17**	**83.78±0.21**	**72.94±0.25**	**63.17**

Table 2: Test accuracy (%) and communication overhead (GB) on the CIFAR-10 dataset under the homogeneous FL setting.

Method	ResNet-18			ResNet-34			ResNet-50		
	α=1	α=0.1	Comm	α=1	α=0.1	Comm	α=1	α=0.1	Comm
FedAvg	56.87±0.40	41.63±0.55	83.65	58.24±0.35	42.11±0.43	158.97	58.89±0.16	43.04±0.50	176.68
MHAT	53.12±0.31	38.71±0.36	6.10	54.00±0.09	39.34±0.22	6.10	54.76±0.28	40.11±0.44	6.10
DS-FL	52.71±0.28	38.25±0.19	6.09	53.65±0.13	39.08±0.24	6.09	54.23±0.36	39.79±0.37	6.09
FedGen	57.42±0.15	42.84±0.35	91.30	58.67±0.27	43.45±0.35	166.61	59.06±0.13	44.09±0.29	184.32
FedFTG	57.76±0.16	43.03±0.41	83.65	58.86±0.16	43.74±0.19	158.97	59.54±0.17	44.47±0.18	176.68
DFRD	58.00±0.25	43.20±0.21	83.65	59.39±0.23	43.96±0.26	158.97	59.93±0.26	44.51±0.33	176.68
FedZKT	57.58±0.32	42.93±0.14	83.65	58.50±0.14	43.59±0.31	158.97	58.74±0.19	44.16±0.51	176.68
FedZGE	**58.41±0.11**	**43.66±0.23**	**65.18**	**59.67±0.25**	**44.24±0.32**	**65.18**	**60.02±0.17**	**44.67±0.36**	**65.18**

Table 3: Test accuracy (%) and communication overhead (GB) on the CIFAR-100 dataset under the homogeneous FL setting.

Method	CIFAR-10			CIFAR-100		
	α=1	α=0.1	Comm	α=1	α=0.1	Comm
FedAvg	-	-	-	-	-	-
MHAT	78.50±0.17	66.30±0.28	5.76	54.33±0.17	39.77±0.21	6.10
DS-FL	78.23±0.09	65.95±0.35	5.76	54.81±0.31	39.63±0.45	6.10
FedGen	-	-	-	-	-	-
FedFTG	83.09±0.13	72.24±0.24	159.00	59.31±0.24	44.09±0.10	160.07
DFRD	83.21±0.22	72.52±0.11	159.00	59.52±0.19	44.26±0.19	160.07
FedZKT	83.00±0.26	71.94±0.45	142.70	58.59±0.11	43.90±0.27	143.46
FedZGE	**83.60±0.14**	**72.81±0.33**	**63.17**	**59.73±0.21**	**44.44±0.30**	**65.18**

Table 4: Test accuracy (%) and communication overhead (GB) on the CIFAR-10 and CIFAR-100 datasets under the heterogeneous FL setting.

Settings. The experiments are performed under two distinct FL settings with the key hyperparameters α=$\{1, 0.1\}$, K=10, ε=1, T=100, B=500, and q=10: 1) **homogeneous FL setting**, where clients are forced to replicate homogeneous local models with the same architecture as the global model. We employ three types of network architectures to explore the effects of model scaling, including ResNet-18, ResNet-34, and ResNet-50 (He et al. 2016); 2) **heterogeneous FL setting**, where clients are allowed to independently design heterogeneous local models. We employ ResNet-50 as the global model and allocate ResNet-18, ResNet-34, and ResNet-50 as the local models to clients in a ratio of 3:3:4. See Appendix C for full implementation details.

Experimental Results

Robustness to Data Heterogeneity. Tables 2 and 3 report the test accuracy and communication overhead of different FL approaches on the CIFAR-10 and CIFAR-100 datasets under the homogeneous FL setting, respectively. A notable observation is that the performance of all FL approaches deteriorates dramatically as the degree of data heterogeneity among clients increases. However, FedZGE significantly outperforms distillation-based FL approaches and achieves comparable performance to data-free FL approaches at different levels of data heterogeneity as well as different scales of network architectures, demonstrating its strong robustness in data heterogeneous FL scenarios.

Compatibility to Model Heterogeneity. Table 4 reports the test accuracy and communication overhead of different FL approaches on the CIFAR-10 and CIFAR-100 datasets under the heterogeneous FL setting. We can observe that FedZGE consistently outperforms distillation-based FL approaches with superior performance and overwhelms data-free FL approaches with less communication overhead, without sharing any model parameters, suggesting that FedZGE is more advantageous in model heterogeneous FL scenarios.

Alleviation to Communication Overhead. The formulations to calculate communication overhead for different FL methods are listed in Appendix D. As shown in Tables 2, 3

Method	CIFAR-10			CIFAR-100		
	α=1	α=0.1	Comm	α=1	α=0.1	Comm
FedZGE	**83.78±0.21**	**72.94±0.25**	**63.17**	**60.02±0.17**	**44.67±0.36**	**65.18**
FedZGE (q=1)	83.10±0.30	72.08±0.22	11.50	58.81±0.14	43.85±0.50	12.00
FedZGE (q=5)	83.44±0.26	72.33±0.33	34.46	59.37±0.20	44.40±0.41	35.64
FedZGE (q=20)	83.88±0.17	73.00±0.40	120.57	60.08±0.07	44.76±0.32	124.26
w/ $\mathcal{L}_G^{fid}$	82.90±0.08	72.01±0.13	63.17	59.04±0.11	43.97±0.27	65.18
w/o $\mathcal{L}_G^{adv}$	83.54±0.18	72.70±0.37	63.17	59.58±0.14	44.39±0.48	65.18
w/o $\mathcal{L}_G^{div}$	83.70±0.26	72.82±0.21	63.17	59.74±0.25	44.58±0.24	65.18
w/o $\mathcal{L}_G^{info}$	83.58±0.10	72.76±0.46	63.17	59.70±0.18	44.44±0.25	65.18
w/o $\mathcal{L}'_k$	83.47±0.24	72.68±0.36	63.17	59.46±0.28	44.30±0.33	65.18

Table 5: Ablation study of different FedZGE variants on the CIFAR-10 and CIFAR-100 datasets using ResNet-50.

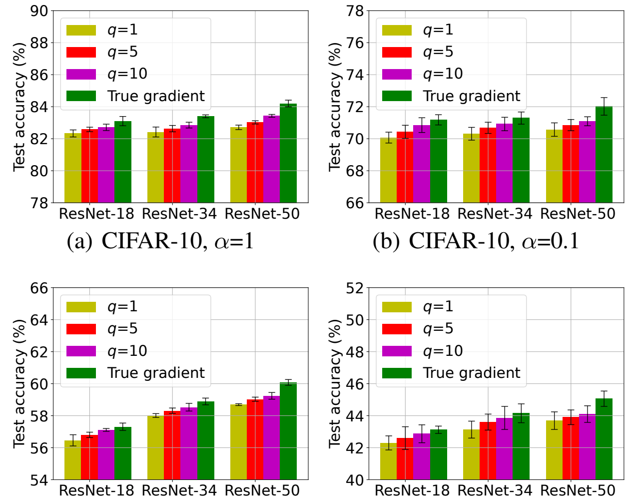
and 4, the communication overhead of FedZGE is independent of model size, and consistently lies between distillation-based FL approaches and data-free FL approaches. In contrast to data-free FL that sacrifices the benefits of black-box access and communication-efficient collaboration offered by distillation-based FL in exchange for improved performance, FedZGE retains the low communication overhead of distillation-based FL and the high performance of data-free FL simultaneously, serving as a compromise between them to provide a more effective and efficient FL solution.

Resistance to Privacy Leakage. Distillation-based FL and data-free FL require transmitting real data samples and model parameters between the server and clients, respectively, which may raise privacy-related concerns since a malicious attacker could exploit this information to reconstruct clients' private data. Instead, FedZGE transmits synthetic data samples that capture only high-level feature patterns, which are incomprehensible to humans and express the global characteristics of private data, thereby providing stronger privacy guarantees than previous FL approaches. To illustrate this, we provide a visual comparison between the synthetic data and the private data on the CIFAR-10 and CIFAR-100 datasets in Appendix E.

Ablation Study

Effects of Loss Function Design. To explore the contribution of different auxiliary losses to the generator training, we conduct a leave-one-out test to show the results after removing individual losses (denoted as w/o $\mathcal{L}_G^{adv}$, w/o $\mathcal{L}_G^{div}$, and w/o $\mathcal{L}_G^{info}$, respectively) and all losses (denoted as w/ $\mathcal{L}_G^{fid}$). Moreover, we remove local distillation losses from clients (denoted as w/o $\mathcal{L}'_k$) to explore the role they play. As shown in Table 5, it can observed that using only $\mathcal{L}_G^{fid}$ to train the generator leads to poor performance, and the absence of a single loss leads to significant performance degradation, suggesting that each component contributes to the improvement of the generator. Furthermore, removing local distillation losses leads to more pronounced performance degradation in scenarios with severe data heterogeneity, highlighting its effectiveness in mitigating data heterogeneity.

Effects of Perturbation Directions. To examine the effects of the number of random perturbation directions q on



(a) CIFAR-10, α=1 (b) CIFAR-10, α=0.1 (c) CIFAR-100, α=1 (d) CIFAR-100, α=0.1

Figure 2: Ablation study of FedZGE and FedZGE with true gradients on the CIFAR-10 and CIFAR-100 datasets.

FedZGE, we sample different q={1, 5, 20} for zeroth-order gradient estimation. As shown in Table 5, it can be observed that a larger value of q leads to a more accurate gradient estimation but increases the communication overhead. Therefore, there is a trade-off between test accuracy and communication overhead for FedZGE. Empirically, adopting q=10 is the best practice since it performs comparably or even better than other baselines with less communication overhead.

Effects of Gradient Estimation. To understand how the gradient estimation affects the performance of FedZGE, we modify FedZGE for analytical purposes by allowing clients to upload their local models to the server for white-box access to obtain the true gradient. As shown in Figure 2, It can be observed that the performance of FedZGE is close to that when the true gradient of the generator is available. However, it is closely related to the size of the local models to be estimated due to the higher the complexity of the black-box network architecture, the larger the difference between the estimated gradient and the true gradient. More experimental results and analysis can be found in Appendix F.

Conclusion

To combine the advantages of both distillation-based FL and data-free FL, this paper introduces FedZGE, a data-free and black-box FL framework that leverages Zeroth-order Gradient Estimation to complete the training of the generator in terms of fidelity, transferability, diversity, and equilibrium, without involving any auxiliary data or sharing any model parameters. Extensive experiments on large-scale image classification datasets and network architectures demonstrate the superiority of FedZGE in terms of data heterogeneity, model heterogeneity, communication efficiency, and privacy protection. Future work will explore the application of data-free black-box FL to more challenging textual domains with discrete input spaces and large foundation models.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant Nos.61966038 and 62266051, and the Postgraduate Research and Innovation Foundation of Yunnan University under Grant No.KC-23236531. The authors would like to thank the anonymous reviewers for their constructive comments.

References

Alam, S.; Liu, L.; Yan, M.; and Zhang, M. 2022. Fedrolex: Model-heterogeneous federated learning with rolling sub-model extraction. In *Proceedings of the 36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, 29677–29690.

Almanifi, O. R. A.; Chow, C. O.; Tham, M. L.; Chuah, J. H.; and Kanesan, J. 2023. Communication and computation efficiency in federated learning: A survey. *Internet of Things*, 22: 100742.

Bibikar, S.; Vikalo, H.; Wang, Z.; and Chen, X. 2022. Federated dynamic sparse training: Computing less, communicating less, yet learning better. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI 2022)*, 6080–6088.

Chang, H.; Shejwalkar, V.; Shokri, R.; and Houmansadr, A. 2019. Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. arXiv:1912.11279.

Chen, H.; Wang, Y.; Xu, C.; Yang, Z.; Liu, C.; Shi, B.; Xu, C.; Xu, C.; and Tian, Q. 2019. Data-free learning of student networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV 2019)*, 3514–3522.

Cheng, S.; Wu, J.; Xiao, Y.; Liu, Y.; and Liu, Y. 2021. Fedgems: Federated learning of larger server models via selective knowledge fusion. arXiv:2110.11027.

Fang, G.; Mo, K.; Wang, X.; Song, J.; Bei, S.; Zhang, H.; and Song, M. 2022. Up to 100x faster data-free knowledge distillation. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI 2022)*, 6597–6604.

Fang, G.; Song, J.; Shen, C.; Wang, X.; Chen, D.; and Song, M. 2019. Data-free adversarial distillation. arXiv:1912.11006.

Gao, D.; Yao, X.; and Yang, Q. 2022. A survey on heterogeneous federated learning. arXiv:2210.04505.

Gong, X.; Sharma, A.; Karanam, S.; Wu, Z.; Chen, T.; Doermann, D.; and Innanje, A. 2021. Ensemble attention distillation for privacy-preserving federated learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV 2021)*, 15056–15066.

Gong, X.; Sharma, A.; Karanam, S.; Wu, Z.; Chen, T.; Doermann, D.; and Innanje, A. 2022. Preserving privacy in federated learning with ensemble cross-domain knowledge distillation. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI 2022)*, 11891–11899.

Goodfellow, I. J.; Pouget-Adadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Proceedings of the 28th Conference on Neural Information Processing Systems (NeurIPS 2014)*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, 770–778.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. arXiv:1503.02531.

Hsu, T.-m. H.; Qi, H.; and Brown, M. 2019. Measuring the effects of non-identical data distribution for federated visual classification. arXiv:1909.06335.

Hu, L.; Yan, H.; Li, L.; Pan, Z.; Liu, X.; and Zhang, Z. 2021. Mhat: An efficient model-heterogenous aggregation training scheme for federated learning. *Information Sciences*, 560: 493–503.

Itahara, S.; Nishio, T.; Koda, Y.; Morikura, M.; and Yamamoto, K. 2021. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-iid private data. *IEEE Transactions on Mobile Computing*, 22(1): 191–205.

Kairouz, P.; McMahan, H. B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A. N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; D'Oliveira, R. G.; Eichner, H.; El Rouayheb, S.; Evans, D.; Gardner, J.; Garrett, Z.; Gascón, A.; Ghazi, B.; Gibbons, P. B.; Gruteser, M.; Harchaoui, Z.; He, C.; He, L.; Huo, Z.; Hutchinson, B.; Hsu, J.; Jaggi, M.; Javidi, T.; Joshi, G.; Khodak, M.; Konecní, J.; Korolova, A.; Koushanfar, F.; Koyejo, S.; Lepoint, T.; Liu, Y.; Mittal, P.; Mohri, M.; Nock, R.; Özgür, A.; Pagh, R.; Qi, H.; Ramage, D.; Raskar, R.; Raykova, M.; Song, D.; Song, W.; Stich, S. U.; Sun, Z.; Suresh, A. T.; Tramèr, F.; Vepakomma, P.; Wang, J.; Xiong, L.; Xu, Z.; Yang, Q.; Yu, F. X.; Yu, H.; and Zhao, S. 2021. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1-2): 1–210.

Konečný, J.; McMahan, H. B.; Yu, F. X.; Richtárik, P.; Suresh, A. T.; and Bacon, D. 2016. Federated learning: Strategies for improving communication efficiency. arXiv:1610.05492.

Krizhevsky, A. 2009. Learning multiple layers of features from tiny images.

Kullback, S.; and Leibler, R. 1951. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1): 79–86.

Li, D.; and Wang, J. 2019. Fedmd: Heterogenous federated learning via model distillation. arXiv:1910.03581.

Li, T.; Sahu, A. K.; Talwalkar, A.; and Smith, V. 2020. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3): 50–60.

Li, Y.; Yu, H.; Zeng, Y.; and Pan, Q. 2023. Hfsa: A semi-asynchronous hierarchical federated recommendation system in smart city. *IEEE Internet of Things Journal*, 10(21): 18808–18820.

Liang, F.; Pan, W.; and Ming, Z. 2021. Fedrec++: Lossless federated recommendation with explicit feedback. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI 2021)*, 4224–4231.

Liu, S.; Kailkhura, B.; Chen, P. Y.; Ting, P.; Chang, S.; and Amini, L. 2018. Zeroth-order stochastic variance reduction for nonconvex optimization. In *Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*.

Liu, Z.; Chen, Y.; Zhao, Y.; Yu, H.; Liu, Y.; Bao, R.; Jiang, J.; Nie, Z.; Xu, Q.; and Yang, Q. 2022. Contribution-aware federated learning for smart healthcare. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI 2022)*, 12396–12404.

Lopes, R. G.; Fenu, S.; and Starner, T. 2017. Data-free knowledge distillation for deep neural networks. arXiv:1710.07535.

Luo, K.; Wang, S.; Fu, Y.; Li, X.; Lan, Y.; and Gao, M. 2023. Dfrd: Data-free robustness distillation for heterogeneous federated learning. In *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS 2023)*, 17854–17866.

Lyu, L.; Yu, H.; Ma, X.; Chen, C.; Sun, L.; Zhao, J.; Yang, Q.; and Yu, P. S. 2022. Privacy and robustness in federated learning: Attacks and defenses. *IEEE Transactions on Neural Networks and Learning Systems*, 35(7): 8726–8746.

Ma, X.; Liu, J.; Wang, J.; and Zhang, X. 2023. Fedid: Federated interactive distillation for large-scale pretraining language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*, 8566–8577.

Ma, X.; Zhu, J.; Lin, Z.; Chen, S.; and Qin, Y. 2022. A state-of-the-art survey on solving non-iid data in federated learning. *Future Generation Computer Systems*, 135: 244–258.

Mao, Q.; Lee, H. Y.; Tseng, H. Y.; Ma, S.; and Yang, M. H. 2019. Mode seeking generative adversarial networks for diverse image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, 1429–1437.

McMahan, H.; Moore, E.; Ramage, D.; Hampson, S.; and Arcas, B. A. y. 2017. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS 2017)*, 1273–1282.

Micaelli, P.; and Storkey, A. 2019. Zero-shot knowledge transfer via adversarial belief matching. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*.

Mirza, M.; and Osindero, S. 2014. Conditional generative adversarial nets. arXiv:1411.1784.

Mora, A.; Tenison, I.; Bellavista, P.; and Rish, I. 2022. Knowledge distillation for federated learning: A practical guide. arXiv:2211.04742.

Nesterov, Y.; and Spokoiny, V. 2017. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2): 527–566.

Noble, M.; Bellet, A.; and Dieuleveut, A. 2022. Differentially private federated learning on heterogeneous data. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS 2022)*, 10110–10145.

Qin, L.; Zhu, T.; Zhou, W.; and Yu, P. S. 2024. Knowledge distillation in federated learning: A survey on long lasting challenges and new solutions. arXiv:2406.10861.

Radford, A.; Metz, L.; and Chintala, S. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434.

Sattler, F.; Marban, A.; Rischke, R.; and Samek, W. 2022. Cfd: Communication-efficient federated distillation via soft-label quantization and delta coding. *IEEE Transactions on Network Science and Engineering*, 9(4): 2025–2038.

Sharma, A.; and Marchang, N. 2024. A review on client-server attacks and defenses in federated learning. *Computers and Security*, 140: 103801.

Tan, Y.; Long, G.; Ma, J.; Liu, L.; Zhou, T.; and Jiang, J. 2022. Federated learning from pre-trained models: A contrastive learning approach. In *Proceedings of the 36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, 19332–19344.

Wu, Z.; Sun, S.; Wang, Y.; Liu, M.; Jiang, X.; Li, R.; and Gao, B. 2023. Knowledge distillation in federated edge learning: A survey. arXiv:2301.05849.

Ye, M.; Fang, X.; Du, B.; Yuen, P. C.; and Tao, D. 2023. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Computing Surveys*, 56(3): 1–44.

Zhang, C.; Xie, Y.; Bai, H.; Yu, B.; Li, W.; and Gao, Y. 2021. A survey on federated learning. *Knowledge-Based Systems*, 216: 106775.

Zhang, J.; Chen, C.; and Lyu, L. 2023. Ideal: Query-efficient data-free learning from black-box models. In *Proceedings of the 11th International Conference on Learning Representations (ICLR 2023)*.

Zhang, L.; Shen, L.; Ding, L.; Tao, D.; and Duan, L. Y. 2022. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, 10174–10183.

Zhang, L.; Wu, D.; and Yuan, X. 2022. Fedzkt: Zero-shot knowledge transfer towards resource-constrained federated learning with heterogeneous on-device models. In *2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS 2022)*, 928–938.

Zhu, H.; Xu, J.; Liu, S.; and Jin, Y. 2021. Federated learning on non-iid data: A survey. *Neurocomputing*, 465: 371–390.

Zhu, Z.; Hong, J.; and Zhou, J. 2021. Data-free knowledge distillation for heterogeneous federated learning. In *Proceedings of the 38th International Conference on Machine Learning (ICML 2021)*, 12878–12889.