Federated Learning with Generative Content

Anonymous Author(s)
Affiliation
Address
email

Abstract

Federated learning (FL) enables leveraging distributed private data for model training in a collaborative and privacy-preserving way. However, the ubiquitous and notorious issue of data heterogeneity, where different data-owing clients hold heterogeneous datasets, significantly and fundamentally limits the performance of current FL methods. To address this issue, this paper explores a new direction, data-centric intervention, which directly enriches the clients' local data with generative content, fundamentally reducing the level of data heterogeneity. Following this idea, we propose a novel framework, federated learning with generative content (FedGC). FedGC is a simple-yet-effective framework, where each client leverages diverse generative data from advanced generative models and original private data to train its local model, all guided by strategies summarized and learned from our four-aspect analysis. FedGC offers two significant advantages: (1) FedGC mitigates data heterogeneity as the diverse generative data prevents each client from over-fitting its client-specific private data; and (2) FedGC contributes to better privacy preservation as the introduced generative data dilutes the concentration of sensitive data in the enriched dataset, mitigating the risk of memorizing private information. Empirical studies on 9 baselines and 7 datasets demonstrate that FedGC consistently and significantly improves task performance and privacy preservation.

1 Introduction

Federated learning (FL) is a privacy-preserving machine learning paradigm that enables multiple clients to collaboratively train a shared global model without directly sharing their raw data [1, 2]. With the world's increasing emphasis on data ownership and privacy [3, 4, 5, 6], FL has attracted significant attention [7, 8, 9] and has been applied to diverse real-world fields such as natural language processing [10], healthcare [11], and finance [12].

Despite multi-fold benefits of FL, data heterogeneity stands as a prominent and fundamental challenge in FL, significantly impacting FL's overall performance [1, 13, 14]. This heterogeneity arises inherently due to the diverse environments and preferences during the collection of clients' data. Consequently, it results in biased and divergent local model updates, posing difficulties in achieving a well-generalized aggregated global model capable of effectively addressing diverse data sources.

To address this issue, a series of works have been proposed, primarily focusing on *model-centric interventions* that operate within the space of model parameters [15]. On the client side, they regularize the distance between local and global model [16, 17], introduce control variates to correct local gradients [18], align the feature space [19, 20]. On the server side, they introduce momentum to update global model [21, 13], adjust the process of aggregating local models [22, 23], modify model initialization [24, 25]. Despite these efforts, such *model-centric interventions* do not directly confront heterogeneous data distributions, offering only palliative solutions to its adverse impacts.

In this paper, we explore a new direction, *data-centric intervention*, which directly operates on the clients' local data to fundamentally reduce the level of data heterogeneity. Specifically, given the fact that data heterogeneity roots from clients' potentially specific uniform data, advanced generative models [26, 27, 28] offer unprecedented opportunities to enrich clients' heterogeneous data with general and complementary generative content [29, 30]. This could facilitate more homogeneous local model updates and enhance the performance of the aggregated global model. Such *data-centric intervention* directly addresses the root cause: client-specific heterogeneous data, avoiding the problem in *model-centric interventions* where data heterogeneity will cause persistent harm to FL.

Following this idea, we propose a novel framework, Federated Learning with Generative Content (FedGC). In FedGC, each client uses an off-the-shelf generative model conditioned on task-related prompts to generate diverse data, which is utilized to supplement the originally client-specific (the root of data heterogeneity) data. The supplemented dataset can subsequently facilitate local model training by encouraging the local model to learn general and diverse patterns rather than the potentially biased and specific patterns of its private data. Given the advancements in generative models across various modalities, the FedGC framework is inherently applicable to diverse modalities, such as image and text. Moreover, we position FedGC as a comprehensive and adaptable framework, setting the stage for thorough investigation in various dimensions. Specifically, we identify and meticulously examine four pivotal dimensions: budget allocation, prompt design, generation guidance, and training strategy, which correspond to consideration of generation efficiency, data diversity, data fidelity, and training effectiveness respectively (Figure 1). For each dimension, we explore three feasible solutions and rigorously evaluate their effectiveness in enhancing model performance, ultimately identifying the most effective solution. For example, for better data fidelity, we propose real-data-guidance which generates data conditioned on both client's real data and task-related prompts.

Overall, our data-centric solution FedGC offers two fundamental advantages. (1) FedGC can significantly mitigate data heterogeneity as the diverse generative data prevents each client from over-fitting its client-specific private data. (2) FedGC can contribute to better privacy preservation as the introduced generative data dilutes the concentration of sensitive data in the enriched dataset, which mitigates the risk of memorizing private information.

To verify the effectiveness of FedGC and deepen understanding, we conduct a systematic empirical study from diverse perspectives, including compatibility with 9 FL baselines, 7 datasets, 2 modalities, and 3 data heterogeneity types. Extensive experiments reveal three significant findings: 1) our data-centric intervention that adds generative data is a more direct, concise, and effective solution to tackle data heterogeneity, than many model-centric interventions that may involve sophisticated designs; 2) FedGC can enhance both privacy preservation and performance of FL; 3) the generative data is not necessary to fully resemble real data yet can implicitly reduce data heterogeneity and model divergence that lead to enhanced performance.

Our contributions are as follows:

1. We propose FedGC, a new, simple yet effective data-centric FL framework that handles data heterogeneity from a new perspective: generating diverse data to supplement private real data.
2. We summarize four critical and worth-exploring dimensions in FedGC, explore three feasible solutions for each, rigorously evaluate their effectiveness, and identify the most effective solution.
3. We provide a systematic empirical study on FedGC framework, showing its effectiveness for enhancing both performance and privacy preservation under data heterogeneity and providing new insights for future works through several interesting experimental findings.

2 Related Work

Federated learning (FL) enables multiple clients to collaboratively train a global model without sharing raw data [1], which has attracted much attention due to its privacy-preserving property [8, 2]. Data heterogeneity is one representative challenge in FL that significantly limits the FL's performance [13, 31]. Addressing this, many methods are proposed to mitigate its adverse effects from the perspective of model-centric interventions. (1) On client-side intervention [32, 17, 33], FedProx [16] and SCAFFOLD [18] propose to conduct model-level correction such as regularizing ℓ_2 distance between local and global model and introducing a control variate to correct gradient of local model. MOON [20] and FedDecorr [34] propose to regularize feature space. (2) On server-side intervention [35, 14, 36], FedNova [22] and FedDisco [37] propose to modify aggregation weights to obtain better-aggregated model. Some explore the effects of model initialization [24, 25]. FedAvgM [13] and FedOPT [21] introduce momentum to improve the aggregated global model. Unlike these model-centric methods that still fundamentally suffer from data heterogeneity, our FedGC framework focuses on data-centric improvement, which mitigates heterogeneity of the distributed real data by complementing it with diverse generative data. Besides, our FedGC framework is orthogonal to these methods, allowing seamless integration within our framework.

Generative models have demonstrated remarkable performance across multiple domains such as large language models [38, 27, 39] for language generation and diffusion models [40, 26, 41] for image generation. Though these models can generate high-quality data for general cases, the generated data is not sufficient to train a well-perform model due to its incapability of representing real data [42], especially for uncommon cases such as medical tasks [43, 44]. Recently, [45] shows the importance of data diversity for image classification tasks. Some recent works explore the effectiveness of generative models in pre-training in FL [46, 47]. In this paper, we systematically explore the potential of using generative models to directly assist FL on private downstream tasks (both image and text). Based on our FedGC, we verify that despite failing to fully represent real data, generated data can still contribute to improving the performance of FL under heterogeneous private data.

3 Federated Learning with Generative Content

In this section, we introduce our proposed framework FedGC, which leverages generative content to tackle the issue of data heterogeneity in FL. Based on FedGC, we explore four aspects to better study the effects of generative content in FL and explore three solutions for each aspect.

3.1 FedGC Framework Overview

Our FedGC follows the standard FedAvg [1] framework, encompassing of four iterative phases: global model broadcasting, local model training, local model uploading, and global model aggregation. Our goal is to generate diverse data to supplement private data to facilitate local model training. Though the data generation can be handled by either the server or the client (also see Appendix E), we focus on the latter considering communication cost [2] and flexibility, which avoids additional communication cost required for server-to-client transmitting generative data, and enables using the local data as prior to generate more task-specific data. Thus, we focus on local model training, which is decomposed into: data generation and local model training. Specifically, in FedGC, we 1) design to generate diverse data, 2) merge the generative and private dataset, and 3) train the local model, where the first two are required for only once; see Figure 1 for the overview. Note that FedGC is versatile across modalities, while here we focus on two most common modalities: image and text.

3.2 Data Generation in FedGC

On the designs for data generation in FedGC framework, we consider the following criteria: generation efficiency, data diversity, and data fidelity. Following the criteria, we explore three crucial aspects, including budget allocation, prompt design, and generation guidance, and propose three representative solutions as candidates for each aspect. Without loss of generality, we use the text-guided latent diffusion model [26] to generate images based on prompts for image task, and an LLM [38] to generate texts based on prompts for text task.

Budget allocation for efficiency. Though, (1) the process of data generation is just one-shot and (2) FedGC does not compromise on the two first-order concerns in FL: communication cost and privacy [2], it still costs some computation budget in exchange for algorithm utility [48]. Thus, it is essential to design efficient strategies to allocate the generation budget (i.e., the total number of generative samples, denoted as M) to each client and label. To achieve this, we design three allocation strategies. (1) The equal allocation strategy allocates the budget equally to each client and each category, which is the simplest and most general allocation strategy. That is, each client can generate $\frac{M}{KC}$ data samples for each category. (2) Inverse allocation strategy allocates the budget inversely to each client according to its number of data samples. Specifically, each client k can generate $\frac{M\cdot(N_{max}-N_k)}{C\cdot\sum_i(N_{max}-N_i)}$ samples for each category, where N_{max} denotes the maximum number in

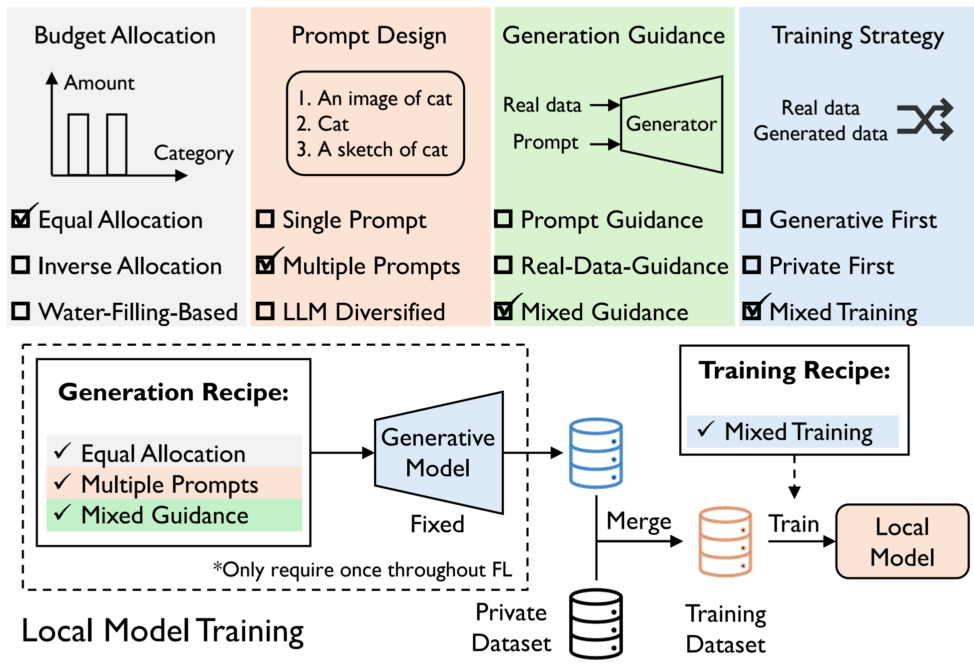


Figure 1: Overview of the designs of FedGC on client side. Above, we summarize four crucial aspects that are worth exploring and propose three solutions for each aspect. Below is the pipeline of local training, where each client first generates data based on the generation recipe, then merges the generative and private dataset, and finally trains the local model based on the training recipe.

$\{N_i\}_i$. (3) Water-filling-based: each client can generate $\frac{M}{K}$ samples in total, and apply water filling algorithm to allocate samples to each category [49].

Prompt design for diversity. Data diversity plays a key role in learning a generalized model in many domains such as image [50] and text [51]. To increase the diversity, it is essential to design appropriate prompts since they directly guide the process of generation. For image task, we consider three diversity levels. (1) Single prompt, where we use "a photo of {class}" [52]. (2) Multiple prompts, where we consider diverse formats such as "{class}". (3) LLM-based diversified prompts, where we instruct an LLM such as ChatGPT to diversify the prompts. For text generation, we only design one prompt since advanced LLMs are sufficient to generate diverse content.

Generation guidance for diversity and fidelity. Finally, we feed the prompts to the generative models for generation. Besides designing prompts, we randomly set the guidance scale for diffusion models [26] (or non-zero temperature for LLMs) to enhance the data diversity.

(Prompt-Only Guidance) However, data diversity may not be sufficient to ensure improving model training, while data fidelity is also a critical factor. For cases where the domain gap between the generative and real data is too large, the benefits of increasing diversity may be outweighed by the negative effects of the domain gap, leading to degraded performance [42].

(Real-Data Guidance) To alleviate this issue, we propose a new real-data-guided generation approach, which conditions data generation on both real data and prompts. For image task, unlike the original text-guided generation that starts from a random Gaussian noise at latent space $\mathbf{z}_T^1$ [26], we propose to inject information of real data into the starting noise. Specifically, we first use the auto-encoder to encode the real image $\mathbf{x}$ to latent representation $\mathbf{z}$, then add some Gaussian variation to obtain a new $\mathbf{z}_T^2$, which substitutes $\mathbf{z}_T^1$ as the starting point; see illustration in Figure 8. This enriched latent representation, infused with real data insights, enables the generative model to produce outputs closely resembling real data, optimizing the trade-off between diversity and fidelity. For text task, see illustration in Figure 9 using an off-the-shelf large language models (LLMs), such as Llama2-70B-Chat [39] and ChatGPT. Please see more detailed illustrations in Appendix A.

Table 1: Experiments on two heterogeneity types, four datasets, two heterogeneity levels, and nine baselines. Test accuracy (%) averaged over three trials is reported. FedGC consistently and significantly brings performance gain over baselines across diverse settings.

Baseline	H-Type	Label Level				Feature Level				**Avg.**
	Dataset	CIFAR-10		EuroSAT		PACS		VLCS		**Acc.**
	H-Level	High	Low	High	Low	High	Low	High	Low	**Δ**
FedAvg	Vanilla	61.25	75.88	53.82	75.59	38.67	49.13	48.00	44.74	**+16.41**
	+ FedGC	74.50	79.73	74.83	84.46	71.89	75.64	56.51	60.82	
FedAvgM	Vanilla	60.83	74.40	50.91	72.80	25.71	44.42	49.00	48.05	**+18.39**
	+ FedGC	73.84	78.90	73.48	84.87	72.14	73.29	56.46	60.27	
FedProx	Vanilla	64.02	75.62	59.61	73.20	36.37	48.22	50.60	46.89	**+15.54**
	+ FedGC	74.36	79.25	73.04	84.76	72.94	75.48	58.42	60.62	
SCAFFOLD	Vanilla	63.98	78.79	52.72	76.80	34.62	51.98	50.50	51.05	**+14.36**
	+ FedGC	73.96	80.29	69.48	81.04	75.19	76.14	58.27	60.97	
MOON	Vanilla	63.40	75.43	52.67	70.02	32.87	51.23	44.69	45.74	**+17.80**
	+ FedGC	74.02	79.82	73.69	86.06	72.24	74.89	57.52	60.22	
FedDecorr	Vanilla	64.14	76.19	63.74	69.57	34.92	41.77	44.89	46.39	**+15.67**
	+ FedGC	73.94	78.16	69.93	81.30	71.19	74.59	57.01	60.87	
FedDyn	Vanilla	56.14	80.50	67.09	83.67	38.72	55.38	52.76	52.66	**+13.38**
	+ FedGC	73.47	83.42	71.96	87.02	75.34	79.04	61.77	61.92	
FedSAM	Vanilla	56.96	74.28	54.13	68.59	37.72	46.22	48.35	45.74	**+18.37**
	+ FedGC	73.73	78.45	71.43	84.06	74.09	76.49	59.42	61.27	
FedDisco	Vanilla	61.06	75.98	56.24	70.46	35.57	48.32	51.35	45.79	**+16.21**
	+ FedGC	74.65	80.01	69.15	84.22	73.34	75.09	57.62	60.37	

(Mixed Guidance) Furthermore, given that certain clients may lack data samples from specific categories, we propose a mixed guidance strategy. Specifically, for a given budget $N_{k,c}$ for client k in category c, (1) if client k possesses samples from category c, it generates $N_{k,c}/2$ samples using text-only guidance and $N_{k,c}/2$ samples with real-data guidance; (2) in the absence of samples for client k from category c, it generates all the $N_{k,c}$ samples using text-only guidance. This approach effectively addresses category omissions and refines the trade-off between diversity and fidelity.

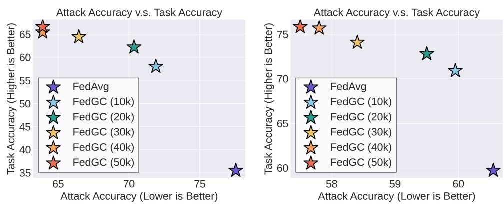
3.3 Local Model Training in FedGC

By choosing generation recipe from the three aspects above, we can generate data using the generative model to assist local model training. Given the generative dataset $\mathcal{D}_g$ and the private dataset $\mathcal{D}_p$, there could be diverse training strategies such as sequential training (optimizing on the two datasets sequentially) and mixed training (optimizing on the mixed dataset).

We find that the mixed training strategy is the most effective despite its simplicity (Table 6). Thus, we directly merge the two datasets as the final new training dataset $\mathcal{D}_m$, based on which we train the local model with the same training manner protocol as other FL methods. Specifically, at the t-th FL communication round, each client k first receives the global model $\boldsymbol{\theta}^t$ and re-initializes its local model with $\boldsymbol{\theta}^t$. Then, each client conducts model training based on the merged dataset $\mathcal{D}_m$ for several optimization steps. Finally, each client k obtains its local model $\boldsymbol{\theta}_k^t$, which is subsequently sent to the server for model aggregation ($\boldsymbol{\theta}^{t+1} := \sum_k p_k \boldsymbol{\theta}_k^t$, where $p_k = N_k / \sum_i N_i$ is the relative dataset size). Note that this process is orthogonal to local training algorithm, which can be SGD-based training [1], proximity-based training [16] or control-variate-based training [18].

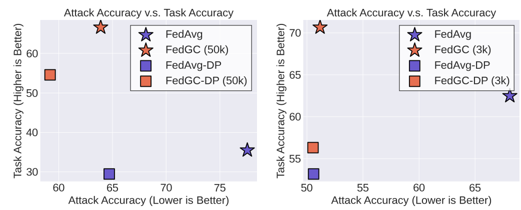
4 Experiments

Experimental setups. Our experiments focus on two most common modalities: image and text. For image tasks, we consider two types of data heterogeneity, including label heterogeneity and



(a) CIFAR-10k (b) CIFAR-50k

Figure 2: FedGC achieves both better task accuracy and privacy preservation (lower attack accuracy). More generative data contributes to higher task accuracy and better privacy preservation.



(a) CIFAR-10k-DP (b) Sentiment140-3k-DP

Figure 3: FedGC achieves both better task accuracy and privacy preservation (lower attack accuracy). FedGC with differential privacy (DP) achieves good privacy-utility trade-off.

feature heterogeneity. For label heterogeneity, we consider a natural image dataset CIFAR-10 [53], a satellite image dataset EuroSAT [54], and a medical image dataset HAM10000 [55], where we allocate the original training dataset to clients based on the frequently used strategy in FL: Dirichlet distribution [56]. The parameter β controls the level of heterogeneity, where we denote 0.05 as high and 0.1 as low. For feature heterogeneity, we consider PACS [57] and VLCS [58], where we allocate training dataset of each domain to several clients according to Dirichlet distribution. This captures both the properties of feature- and label-level heterogeneity. For text datasets, we consider Sentiment140 from LEAF benchmark [59] (naturally allocated) and Yahoo! Answers [60] (split by Dirichlet distribution). We use ResNet-20 [61] for image task and LSTM for text task [59]. We set the number of communication rounds as 100. See more details in Section C.

4.1 Main Results

FedGC significantly improves the FL performance under data heterogeneity. In Table 1, we show experimental results on image modality on two heterogeneity types (label-level and feature-level heterogeneity), two datasets for each type (CIFAR-10, EuroSAT, PACS, and VLCS), and two heterogeneity levels for each dataset. From the table, we see that (1) incorporating baseline in our FedGC framework can consistently and significantly improve the performance of baseline across diverse settings. (2) FedGC is extremely helpful when the heterogeneity level is relatively high, convincingly supporting our motivation of introducing generative data to mitigate the effects of data heterogeneity. Specifically, based on FedAvg, FedGC brings 21.01 absolute accuracy improvement under a high heterogeneity level on EuroSAT and 12.26 absolute accuracy improvement on average.

FedGC is compatible with existing FL methods. From Table 1, we also see that FedGC consistently and significantly brings performance gain across 6 different baselines, including FedAvg, FedAvgM, FedProx, SCAFFOLD, MOON, and FedDecorr. For example, FedGC averagely brings 12.68 absolute accuracy improvement to SCAFFOLD [18]. This demonstrates the compatibility and universality of our proposed FedGC framework.

FedGC achieves better performance and privacy preservation at the same time. Figure 2 explores the effectiveness of different amounts of generative data, where we use image dataset CIFAR-10 as examples. Figure 3 explores differential privacy (DP) technique, where we consider image dataset CIFAR-10 and text dataset Sentiment140. To measure privacy preservation, we use a simple membership inference attack method based on loss evaluation [62, 63] to evaluate attack accuracy; see more details in Appendix F.1. Lower attack accuracy indicates better privacy preservation. From the figures, we clearly see that our FedGC framework can not only improve the performance under data heterogeneity, but also enhance privacy preservation. This observation accords with our expectation that the generative data can dilute the concentration of real sensitive data, which mitigates the risk of memorizing private information. This explanation can be further verified by Figure 2 since (1) as the number of generated samples increases, FedGC achieves lower attack accuracy (better privacy preservation). (2) When the number of real training samples is smaller, i.e., from 50k (Figure 2(b)) to 10k (Figure 2(a)), we see a much larger reduction in attack accuracy and improvement in task accuracy, since the ratio of private data samples in the whole dataset is lowered. We also compare FedAvg, FedGC, FedAvg with differential privacy (FedAvg-DP), and FedGC with differential privacy

Table 2: Increasing number of generated samples makes FedAvg [1] prevail.

No. Gen.	0	100	200	500	1000	2000	5000	10000	20000	50000
FedAvg	61.25	63.67	66.21	67.13	66.98	66.28	71.65	**74.50**	**76.93**	**76.39**
FedProx	**64.02**	66.47	67.40	67.05	68.55	69.19	**72.10**	**74.36**	**76.81**	**76.73**
SCAFFOLD	**63.98**	**69.05**	**71.33**	**71.55**	**71.33**	**70.04**	70.34	73.96	74.88	73.98

Table 3: Different budget allocation strategies of FedGC applied on baselines. Equal allocation is preferred for effectiveness and simplicity.

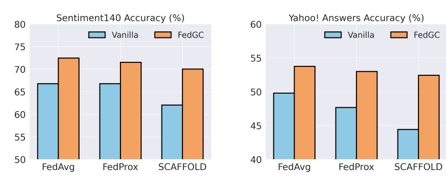
Baseline	**Equal**	Inverse	Water
FedAvg	**74.50**	68.10	71.26
FedProx	**74.36**	68.51	72.23
SCAFFOLD	73.96	73.94	**74.43**

Table 4: Different prompt designs of FedGC applied on baselines. The design of multiple prompt formats is preferred for its effectiveness, diversity, and simplicity.

Baseline	No-GC	Single	**Multiple**	LLM
FedAvg	27.06	50.53	**54.08**	41.32
FedProx	29.12	50.48	**53.03**	40.82
SCAFFOLD	28.56	54.13	**58.53**	45.87

(FedGC-DP) regarding the trade-off between performance and privacy preservation in Figure 3. From the figure, we clearly see that FedAvg-DP enhances privacy preservation while dramatically compromising on task performance compared with FedAvg. In contrast, FedGC can enhance both metrics compared with FedAvg; while FedGC-DP outperforms FedAvg-DP with a clear gap in both metrics. See experiments with deep gradient leakage [64] in Appendix F.2.

FedGC is general across modalities. In Figure 4, we report the performance of FedGC in text modality. We consider two datasets, Sentiment140 and Yahoo! Answers, consisting of 1000 and 100 clients, respectively. Here, we use ChatGPT as the generative model. Note that we use ChatGPT as an example just for the simplicity of our implementation and without loss of generality we can use other open-source LLMs locally. We apply equal budget allocation and single prompt. For real-data-guidance, we take advantage of LLM's few-shot learning ability by giving several real examples in the context [65]. From the figure, we see that FedGC consistently and significantly brings performance gain to all baselines. This experiment verifies that our proposed FedGC framework has the potential to generalize well to diverse modalities.



(a) Sentiment140 (b) Yahoo! Answers

Figure 4: Results on two text datasets. Our proposed FedGC consistently and significantly brings improvement.

Applicability to diverse scenarios. We also (1) consider scenarios where the server handles data generation in Appendix E; (2) consider scenarios where only some clients are capable of generating data in Appendix I; (3) experiment under different heterogeneity levels in Appendix J; (4) experiment on partial client participation scenarios in Appendix K.

4.2 Design Analysis

This section analyzes the effectiveness of different designs in FedGC.

Generating more data could make FedAvg prevail. In Table 2, we explore the effects of number of generated samples on FL's performance, where 0 denotes vanilla FL baseline. Experiments are conducted on CIFAR-10 ($\beta = 0.05$). From the table, we have an interesting finding: (1) when the number of generated samples is relatively small (0$\sim$2000), FedGC can enlarge the gap between standard FedAvg and the method (SCAFFOLD) that is specifically designed for addressing data heterogeneity; (2) however, as the number continues to grow, the situation is reversed that the basic FL method FedAvg prevails. This finding suggests that apart from carefully designing FL algorithm, it is also a promising direction to explore the greater potential from the perspective of generative data.

Equal allocation is a preferred allocation strategy for its effectiveness and simplicity. Data generation inevitably introduces computation overhead, therefore it is meaningful to explore an efficient allocation strategy given fixed generation budget. In Table 3, we compare different budget

Table 5: Different generation guidance of FedGC applied on baselines on medical dataset. Mixed guidance is the best.

Baseline	T-G	TR-G	**Mixed**
FedAvg	51.91	42.38	**56.67**
FedProx	51.43	44.76	**56.19**
SCAFFOLD	56.67	49.52	**58.57**

Table 6: Different training strategies of FedGC applied on baselines. Generated data can only exhibit its efficacy when combined with real data. Mix training is the best.

Baseline	Pri.	Gen.	P2G	G2P	**Mixed**
FedAvg	60.77	41.85	67.06	67.11	**73.99**
FedProx	63.62	40.93	67.23	69.04	**73.69**
SCAFFOLD	65.00	43.45	66.73	69.50	**75.79**

allocation strategies on CIFAR-10, including equal allocation, inverse allocation, and water-filling-based allocation. Experiments show that equal allocation contributes to better performance for both FedAvg and FedProx, and comparable performance compared with water-filling-based allocation for SCAFFOLD. Considering effectiveness and simplicity, we prefer equal allocation strategy.

Multiple prompts lead to better performance, while LLM-based diversification might be unnecessary. Prompts play an important role in the diversity and quality of generated data. It is thus essential to explore different prompt designs. In Table 4, we explore multiple prompt designs on PACS dataset, including using one single prompt format, multiple prompt formats and prompts generated by another LLM. PACS contains significant label-level and feature-level variations, making it an apt choice for this exploration. We compare baseline without FedGC, FedGC with single, multiple, and LLM-based prompts. From the table, (1) we see that FedGC incorporated with all the prompt designs improves the performance of baselines (see improvement over the No-GC column). (2) We see that multiple prompts consistently and significantly perform better, while LLM-based prompts perform ordinarily. This may result from the fact that the scene descriptions from the LLM are usually complicated, causing multifaceted patterns in one sample, thereby complicating model training. Overall, we prefer using multiple prompts for its effectiveness, diversity, and simplicity.

Mixed guidance contributes to higher performance for rare tasks. Pure text-driven prompts cannot control the generative models to generate data that resembles real data; therefore, it would be essential to consider various generation guidances. This is especially critical for rare tasks, such as medical analysis, where the off-the-shelf generative models might fail to generate photorealistic data given simple textual guidance. In Table 5, we compare different generation guidance designs on a medical dataset HAM10000 [55]. The reason for choosing this dataset is that the diffusion model [26] fails to correctly understand medical prompts [66], which helps support our claim more convincingly. We consider three designs, including text-guided generation (T-G), our proposed data generation with guidance of text and real data (TR-G), and the mixed usage of T-G and TR-G. These experiments convey three interesting findings: (1) even though the diffusion model fails to generate data that visually agrees with real data, the generated data still contributes to enhancing the performance of FL (see improvement from Pri. to T-G). (2) TR-G itself fails to bring performance gain, which may result from the limited diversity and incapability to generate for missing classes. (3) Mixing these two strategies contributes to consistently and significantly better performance.

Mixed training is the most effective training strategy. In Table 6, we compare different training strategies on CIFAR-10, including training only on the private dataset (Pri.), training only on the generative dataset (Gen.), sequential training with private dataset first (P2G), sequential training with generative dataset first (G2P), and mixed training. Experiments show that 1) generative data itself fails to ensure training, indicating that there is a gap between generative data and real private data. 2) However, when using generative data together with real private data, we see consistent performance gain compared to training on private data. This indicates that despite the incapability of fully representing real data, the generative data still contributes to improving training by increasing diversity. 3) Mixed training consistently and significantly achieves better performance.

4.3 Mechanism Analysis

This section analyzes how FedGC contributes to enhanced performance.

FedGC reduces data heterogeneity. In Figure 5, we explore the effects of FedGC on data heterogeneity from the perspective of data. To measure the data heterogeneity, we first extract the features of data for each client using a pre-trained ResNet-18 [61], average the features, and compute the

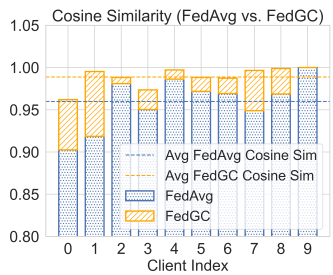


Figure 5: FedGC increases similarity between local datasets.

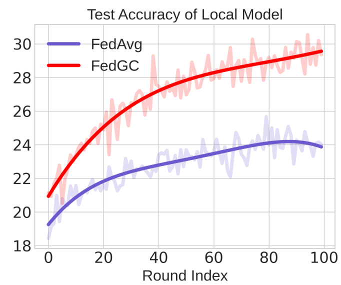


Figure 6: FedGC better preserves local models' generality.

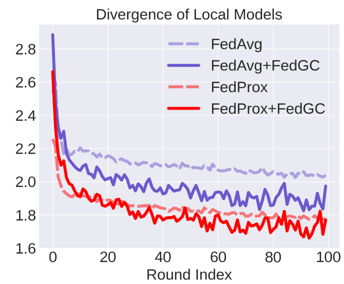


Figure 7: FedGC implicitly reduces model divergence.

pair-wise cosine similarity among the averaged features of all clients. Figure 5 shows the pair-wise similarity using Client 9 as reference. From the figure, we see that FedGC can significantly increase the similarity between datasets of two clients, verifying that FedGC can contribute to mitigating data heterogeneity. We also report ℓ_2 distance as metric and results on PACS in Appendix H.

FedGC alleviates over-fitting local data distribution. In Figure 6, we compare the averaged test accuracy of local models on the global test dataset. From the figure, we can see a clear accuracy gap between our FedGC and the baseline FedAvg. (1) This indicates that our proposed FedGC can encourage each client to preserve the capability on the global general task, rather than overly fit the local specific task (local data distribution). (2) This also helps explain why the generative data can bring performance gain even though they may fail to resemble real data.

FedGC implicitly reduces model divergence. In Figure 7, we visualize the local model divergence along with the round increases. Specifically, at each round, we compute the ℓ_2 difference between each local model and the aggregated global model [16] and report the averaged difference. From the figure, we see that FedGC consistently and significantly reduces the model divergence of local models under severe heterogeneity level ($\beta = 0.05$). This result well supports the claim that FedGC is a pleasant FL framework for tackling the issue of data heterogeneity since it has been shown that data heterogeneity leads to larger model divergence and thus mediocre performance empirically [16] and theoretically [15, 8].

Generated data is diverse, but may not be similar to real data. In Figure 11, we visualize the real data and generated data on EuroSAT [54]. We notice that the generated data samples do not always closely We notice that the generated data samples do not always closely resemble real images, indicating the gap between generative data and real private data (at least visually). Yet, their inclusion still improves the FL's performance under data heterogeneity, which may result from two perspectives. (1) The generative data might act as a form of data augmentation, which potentially introduces variations that are not covered by the original dataset. (2) The generative data diversify the dataset, which serves as a form of implicit regularization, preventing the model from over-fitting to the potentially biased private local data. Please refer to more details and discussions in Appendix G. We also provide an initial exploration of filtering mechanism in Appendix L.

5 Conclusions

This paper focuses on the notorious issue of data heterogeneity in FL. We propose a new data-centric FL framework termed FedGC, which leverages diverse generative data to promote FL under heterogeneous private data. FedGC is a comprehensive and adaptable framework, where we investigate four pivotal dimensions adn conclude several appropriate designs that contribute to better performance of FedGC. We conduct extensive experiments with 9 baselines, 7 datasets, and 2 modalities, showing that our FedGC can consistently and significantly improves the task performance and privacy preservation of FL. Overall, our FedGC, as a data-centric solution, represents a paradigm shift from the conventional model-centric solutions, which well aligns with the current trends in the field of AI and could open up new possibilities for AI applications. Appendix B shows more detailed conclusions.

Limitations. Despite putting much effort into diversifying the experimental settings, there are still cases not covered. For example, we only explore one diffusion model and LLM respectively. There could be future works to explore the effects of different generative models.

References

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[2] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[3] Anastasios Dosis and Wilfried Sand-Zantman. The ownership of data. *The Journal of Law, Economics, and Organization*, 39(3):615–641, 2023.

[4] General Data Protection Regulation. General data protection regulation (gdpr). *Intersoft Consulting, Accessed in October*, 24(1), 2018.

[5] Stuart L Pardau. The california consumer privacy act: Towards a european-style privacy regime in the united states. *J. Tech. L. & Pol'y*, 23:68, 2018.

[6] W Nicholson Price and I Glenn Cohen. Privacy in the age of medical big data. *Nature medicine*, 25(1):37–43, 2019.

[7] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[8] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

[9] Zheng Xu, Yanxiang Zhang, Galen Andrew, Christopher Choquette, Peter Kairouz, Brendan Mcmahan, Jesse Rosenstock, and Yuanbo Zhang. Federated learning of gboard language models with differential privacy. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 629–639, 2023.

[10] Rui Ye, Wenhao Wang, Jingyi Chai, Dihan Li, Zexi Li, Yinda Xu, Yaxin Du, Yanfeng Wang, and Siheng Chen. Openfedllm: Training large language models on decentralized private data via federated learning. *arXiv preprint arXiv:2402.06954*, 2024.

[11] Georgios A Kaissis, Marcus R Makowski, Daniel Rückert, and Rickmer F Braren. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*, 2(6):305–311, 2020.

[12] David Byrd and Antigoni Polychroniadou. Differentially private secure multi-party computation for federated learning in financial applications. In *Proceedings of the First ACM International Conference on AI in Finance*, pages 1–9, 2020.

[13] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

[14] Ziqing Fan, Yanfeng Wang, Jiangchao Yao, Lingjuan Lyu, Ya Zhang, and Qi Tian. Fedskip: Combatting statistical heterogeneity with federated skip aggregation. In *2022 IEEE International Conference on Data Mining (ICDM)*, pages 131–140. IEEE, 2022.

[15] Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H Brendan McMahan, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, et al. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917*, 2021.

[16] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.

[17] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2020.

[18] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

[19] Rui Ye, Zhenyang Ni, Chenxin Xu, Jianyu Wang, Siheng Chen, and Yonina C Eldar. Fedfm: Anchor-based feature matching for data heterogeneity in federated learning. *arXiv preprint arXiv:2210.07615*, 2022.

[20] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10713–10722, 2021.

[21] Sashank J Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2020.

[22] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020.

[23] Divyansh Jhunjhunwala, Shiqiang Wang, and Gauri Joshi. Fedexp: Speeding up federated averaging via extrapolation. In *The Eleventh International Conference on Learning Representations*, 2022.

[24] John Nguyen, Jianyu Wang, Kshitiz Malik, Maziar Sanjabi, and Michael Rabbat. Where to begin? on the impact of pre-training and initialization in federated learning. In *The Eleventh International Conference on Learning Representations*, 2022.

[25] Hong-You Chen, Cheng-Hao Tu, Ziwei Li, Han Wei Shen, and Wei-Lun Chao. On the importance and applicability of pre-training for federated learning. In *The Eleventh International Conference on Learning Representations*, 2022.

[26] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[27] OpenAI. Gpt-4 technical report, 2023.

[28] OpenAI. Video generation models as world simulators. `https://openai.com/research/video-generation-models-as-world-simulators`, 2024. Accessed: 2024-04-22.

[29] Weiming Zhuang, Chen Chen, and Lingjuan Lyu. When foundation model meets federated learning: Motivations, challenges, and future directions. *arXiv preprint arXiv:2306.15546*, 2023.

[30] Yuhang Li, Xin Dong, Chen Chen, Jingtao Li, Yuxin Wen, Michael Spranger, and Lingjuan Lyu. Is synthetic image useful for transfer learning? an investigation into data generation, volume, and utilization. *arXiv preprint arXiv:2403.19866*, 2024.

[31] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2019.

[32] Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. In *International Conference on Machine Learning*, pages 18250–18280. PMLR, 2022.

[33] Ziqing Fan, Jiangchao Yao, Bo Han, Ya Zhang, Yanfeng Wang, et al. Federated learning with bilateral curation for partially class-disjoint data. *Advances in Neural Information Processing Systems*, 36, 2024.

[34] Yujun Shi, Jian Liang, Wenqing Zhang, Vincent YF Tan, and Song Bai. Towards understanding and mitigating dimensional collapse in heterogeneous federated learning. *arXiv preprint arXiv:2210.00226*, 2022.

[35] Zexi Li, Tao Lin, Xinyi Shang, and Chao Wu. Revisiting weighted aggregation in federated learning with neural networks. In *International Conference on Machine Learning*, pages 19767–19788. PMLR, 2023.

[36] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243*, 2020.

[37] Rui Ye, Mingkai Xu, Jianyu Wang, Chenxin Xu, Siheng Chen, and Yanfeng Wang. Feddisco: Federated learning with discrepancy-aware collaboration. *arXiv preprint arXiv:2305.19229*, 2023.

[38] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[39] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[40] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, pages 16784–16804. PMLR, 2022.

[41] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[42] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and XIAOJUAN QI. Is synthetic data from generative models ready for image recognition? In *The Eleventh International Conference on Learning Representations*, 2022.

[43] Gunther Eysenbach et al. The role of chatgpt, generative language models, and artificial intelligence in medical education: a conversation with chatgpt and a call for papers. *JMIR Medical Education*, 9(1):e46885, 2023.

[44] Pedro Celard, EL Iglesias, JM Sorribes-Fdez, Rubén Romero, A Seara Vieira, and L Borrajo. A survey on deep learning applied to medical images: from simple artificial neural networks to generative models. *Neural Computing and Applications*, 35(3):2291–2323, 2023.

[45] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 769–778, 2023.

[46] Tuo Zhang, Tiantian Feng, Samiul Alam, Mi Zhang, Shrikanth S Narayanan, and Salman Avestimehr. Gpt-fl: Generative pre-trained model-assisted federated learning. *arXiv preprint arXiv:2306.02210*, 2023.

[47] Shanshan Wu, Zheng Xu, Yanxiang Zhang, Yuanbo Zhang, and Daniel Ramage. Prompt public large language models to synthesize data for private on-device applications. *arXiv preprint arXiv:2404.04360*, 2024.

[48] Xiaojin Zhang, Yan Kang, Kai Chen, Lixin Fan, and Qiang Yang. Trading off privacy, utility and efficiency in federated learning. *arXiv preprint arXiv:2209.00230*, 2022.

[49] John G Proakis. *Digital communications*. McGraw-Hill, Higher Education, 2008.

[50] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[51] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[52] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[53] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[54] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

[55] Philipp Tschandl, 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):1–9, 2018.

[56] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *International Conference on Learning Representations*, 2020.

[57] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13025–13032, 2020.

[58] Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1657–1664, 2013.

[59] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

[60] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

[61] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[62] Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. How does data augmentation affect privacy in machine learning? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10746–10753, 2021.

[63] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pages 5558–5567. PMLR, 2019.

[64] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019.

[65] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[66] Amirhossein Kazerouni, Ehsan Khodapanah Aghdam, Moein Heidari, Reza Azad, Mohsen Fayyaz, Ilker Hacihaliloglu, and Dorit Merhof. Diffusion models for medical image analysis: A comprehensive survey. *arXiv preprint arXiv:2211.07804*, 2022.

[67] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[68] Liping Yi, Wang Gang, and Liu Xiaoguang. QSFL: A two-level uplink communication optimization framework for federated learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 25501–25513. PMLR, 17–23 Jul 2022.

[69] Jakub Konecnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 8, 2016.

[70] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019.

[71] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in neural information processing systems*, 33:16937–16947, 2020.

[72] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pages 265–284. Springer, 2006.

[73] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[74] Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. Opacus: User-friendly differential privacy library in PyTorch. *arXiv preprint arXiv:2109.12298*, 2021.

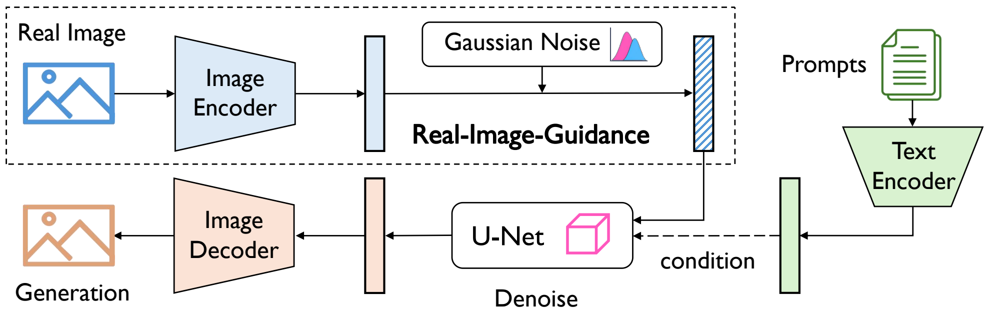


Figure 8: Real-data-guidance for image generation based on diffusion model. The real-data-guidance method involves 4 steps: (1) initializing latent features with real-image data, (2) adding controlled noise, (3) denoising with text features, and (4) generating new images using the decoder.

Table 7: Obtaining LLM-based prompts for generating images using diffusion models. Instructions for generating scene descriptions (i.e., prompts for diffusion models) given a class name using ChatGPT. Here, we provide an example on the dog category of PACS dataset.

System Prompt: You are an AI assistant that helps people find information. **User Prompt:** Please help me come up with scene descriptions that contain a dog while not containing an elephant, giraffe, guitar, horse, house, person. For example: [“A dog is running on the grass”, “A dog is sleeping on the floor”] Please generate 10 samples in the format of a list. Remember: each description should be within 10 words.

A More Illustration of FedGC

For the prompts conditioned on the latent diffusion model, we show the LLM-based prompts for generating images in Table 7. In detail, we instruct ChatGPT through System Prompt and User Prompt, to help us create text samples containing the corresponding class name for image generation. Utilizing ChatGPT’s rich imagination of scenarios and the diversity of text styles, we can achieve a diversity of prompts. Therefore, it helps Stable-diffusion to generate diverse and more realistic pictures.

For generation guidance beyond prompts, we show the real-data guidance for image generation using diffusion models in Figure 8. First of all, the latent features are meticulously initialized using actual real-image data. Subsequently, controlled noise is introduced into the latent representations, which serves to perturb and diversify the features while maintaining the underlying structure. Following this, with conditioned prompts, we denoise this combined feature using U-Net [67]. Finally, passing through the image decoder, we obtain generated images.

We show the real-data-guidance for text generation using ChatGPT in Figure 9. Please note that using ChatGPT is just an example and without loss of generality we can also use many open-source LLMs such as Llama2 [39]. Compared to prompts containing class num, here we instruct the LLM to imitate the theme and content of the corresponding text and directly expand the amount of text data. In our illustrative examples shown in Figure 9, we simulate real-world data scenarios by incorporating four actual instances and generating an additional set of four synthetic instances. In this experimental setup, we task the LLM with the generation of data that exhibits diverse patterns akin to those found in authentic real data. Furthermore, we guide the LLM to produce two distinct samples for each distinct label category, fostering a balanced and representative dataset.

System Prompt:

Assistant is an intelligent chatbot designed to help users generate similar data. Users will provide a few real samples and the Assistant will generate data that follows the pattern of real samples. This is a binary dataset on sentiment analysis, where 0 denotes negative and 1 denotes positive.

Instructions:
1. Generate two samples with label 0 and two samples with label 1, try to make the content diverse
2. Should have a similar pattern of users' data.

User Prompt:

```
**Data: {example_input_1}, Label: {example_label_1}**
**Data: {example_input_2}, Label: {example_label_2}**
**Data: {example_input_3}, Label: {example_label_3}**
**Data: {example_input_4}, Label: {example_label_4}**
```

Generate two samples with label 0 and two samples with label 1.
In the format of Data: {}, Label: {}. Each sample should start with ** and end with **.

Figure 9: Real-data-guidance for text generation using LLMs. Real data is modeled in the examples, where we provide four real examples and generate four new examples. We instruct the LLM to generate diverse data that has a similar pattern to real data. We also instruct the LLM to generate two samples for each label.

B Detailed Conclusions

This work introduces a new data-centric federated learning (FL) solution named FedGC, which leverages diverse generative content to address the notorious data heterogeneity issue in FL. Unlike previous works on data heterogeneity issue that focus on model-level optimization yet largely overlook the root cause of the issue: data itself, our FedGC targets this core aspect directly by enriching client datasets with generative content. FedGC is a simple yet effective framework, which merely introduces a one-shot data generation process compared to standard FL framework. Specifically, in FedGC, each client generates a series of diverse data based on off-the-shelf advanced generative models to enrich its potentially biased private data, then trains its local model on this enriched dataset. We further explore FedGC from four pivotal dimensions including budget allocation, prompt design, generation guidance, and training strategy, and conclude several appropriate designs that contribute to better performance of FedGC.

The advantages of FedGC are three folds. (1) FedGC enhances FL's performance under data heterogeneity. Since the diverse generative content can help enrich clients' potentially biased and heterogeneous data, clients' data would be enriched to be more general and homogeneous, therefore directly and fundamentally reducing the heterogeneity level. (2) FedGC contributes to better privacy preservation of FL. Since the diverse generative data dilutes the concentration of real, sensitive data in the enriched dataset, it naturally mitigates the model's memorization of private data. (3) FedGC is compatible with standard FL infrastructure without extra changes to the training phase, making it simple to deploy in real-world applications. Additionally, in the future, as generative models become increasingly powerful, our FedGC can also grow stronger in tandem. Technically, FedGC, as a data-centric solution, represents a paradigm shift from the conventional model-centric solutions, which well aligns with the current trends in the field of AI. This could potentially opens up new possibilities for AI applications in areas where data collection is challenging or ethically sensitive, such as in medical or personal domains. Broadly, by mitigating data heterogeneity and enhancing privacy, FedGC sets the stage for more powerful and socially responsible AI development, fostering greater trust among users and increase their willingness to participate in AI-enabled systems.

We conduct extensive experiments on 7 datasets, 2 modalities, and 9 FL baselines to verify the effectiveness of our FedGC framework. The results demonstrate that FedGC not only brings consistent performance gain to all FL baselines on all settings by mitigating the data heterogeneity level, but also enhances privacy preservation by mitigating the risk of memorization. In comparison to the standard privacy-preserving FL method FedAvg-DP that compromises utility for privacy, FedGC

Table 8: Number of clients for each dataset.

Dataset	CIFAR-10	EuroSAT	PACS	VLCS	HAM10000	Sentiment	Yahoo!
Client Number	10	10	20	20	10	1000	100

Table 9: Performance comparison between local training with generative content and our FedGC.

Method	CIFAR-High	CIFAR-Low	EuroSAT-High	EuroSAT-Low
Local+GC	46.89	50.47	24.87	35.48
FedGC	**74.50**	**79.93**	**74.83**	**84.46**

with differential privacy strikes a significantly better privacy-utility balance. Additionally, we conduct experiments to determine the most suitable designs in FedGC; and to shed light on why FedGC could bring such huge benefits.

C Implementation Details

We list the number of clients for each dataset in Table 8.The number of iterations for local model training is 200 and uses SGD as the optimizer with a batch size of 64. The learning rate is set to 0.01 [20, 37]. We use ResNet-20 [61] for image task and LSTM for text task [59].

Our experiments were conducted on a machine equipped with an NVIDIA GeForce RTX 3090 GPU with 24 GB of VRAM. However, when training without differential privacy, most experiments only cost less than 2GB of VRAM. The generative model we use can run with a GPU with only 8GB of VRAM. Experiments with differential privacy on text dataset need 20GB of VRAM since the client number is large.

D The Necessity of Federated Setting

Even though we have reach a massive boost on performance with FedGC, we can't determine whether the boost comes from generative model itself or the mitigation data heterogeneity. In other words, whether local training with generative content can still achieve similar results?

To find whether the federated setting is a must, we conduct experiments local training with generative content and federated training with local contents respectively on CIFAR-10 and EuroSAT. The results are shown in Table 9. We can see that local training (without FL) with generative content performs significantly worse.

In fact, we propose FedGC to with a focus on data heterogeneity. Data heterogeneity is a representative and common issue in federated setting while in a non-federated setting there is no definition of data heterogeneity. The generative data can significantly mitigate the level of data heterogeneity and the issue of overfitting, which promotes the performance of FL.

E Discussion about Generation and Communication Cost

In our framework, the data generation can be handled by either the server or the client. Here, different from the main text, we focus on the former where the communication cost should be considered.

However, even in such case, the communication cost is quite low. Here, we provide a detailed example on launching FedGC on SCAFFOLD on CIFAR-10 in Table 10. From the table, we can see that FedGC can achieve significantly higher performance than the baseline while introducing minor additional communication cost. Besides, we only introduce some downlink cost rather than uplink cost, and it is commonly known that the uplink is slower at least five times than the downlink [68, 69]. Specifically, FedGC can achieve 5.07% absolute accuracy improvement while only introducing 0.007% additional communication cost.

Table 10: Communication cost per client and accuracy in cases where we use cloud generation.

Method	SCAFFOLD	FedGC-100	FedGC-200	FedGC-1000	FedGC-10000
Downlink Cost (B)	215,777,600	+30,720	+61,400	+307,200	+3,072,000
Uplink Cost (B)	215,777,600	+0	+0	+0	+0
Total Cost (B)	431,555,200	+30,720	+61,400	+307,200	+3,072,000
Additional Cost (%)	-	+0.007%	+0.014%	+0.071%	+0.712%
Accuracy	63.98	+5.07%	+7.35%	+7.35%	+9.98%

Table 11: Accuracy comparison between FedGC and SCAFFOLD when keeping FedGC with less communication cost.

Method	SCAFFOLD	FedGC-100	FedGC-200	FedGC-1000	FedGC-10000
Total Cost (B)	431,555,200	427,270,368	427,301,048	427,546,848	430,311,648
Accuracy	63.98%	69.05%	71.33%	71.33%	73.96%

For further comparation when considering communication cost, we keep the communication cost less than baselines by reducing the communication rounds (i.e., 1-2 rounds reduction) for FedGC in Table 11. From the table, we see that even with less communication cost, FedGC still significantly outperforms the baseline.

F Privacy

F.1 Membership Inference Attack

To measure the privacy preservation of FedAvg and FedGC, we carry out a simple membership inference attack based on loss evaluation, as [63] has shown that it is reasonable to use the loss of the model to infer membership. We consider a scene where an attacker who has a tiny amount of training data can get the global model and wants to figure out whether a similar datum (i.e. also a photo of an airplane) has been used to train the model or not. During the attack, the attacker feeds its few data to the global model and trains a binary classifier based on the loss of each training-used and not-training-used datum.

We conduct our experiment on CIFAR-10 dataset. In the training process, we set the client number to 10 and the Dirichlet distribution parameter to $\beta = 0.1$. We also discard data augmentations (i.e. flipping and cropping) for more clear comparisons. In the main body, we compare both task accuracy and attack accuracy, as shown in Figure 2.

We also compare the attack accuracy at the point when FedAvg and FedGC achieve similar task accuracy in Table 12. From the table, we see a much more significant reduction in privacy leakage (i.e., much lower attack accuracy). This is reasonable as FedGC can accelerate the convergence speed, which means FedGC requires fewer steps of optimization on the sensitive private data to achieve the same.

F.2 Deep Gradient Leakage

In Figure 2 and Figure 3, we show that FedGC can significantly alleviate the risk of membership inference attack. Here, we further evaluate the level of privacy preservation before and after introducing generative content via deep gradient leakage [70, 71]. We run two experiments for FedAvg [1] and our FedGC respectively. For FedAvg, two real images are used for training while for FedGC, one real image and one generative image are used for training. We report the results in Figure 10 and see that FedGC mitigates the risks of one real image being recovered. Though the rightmost image is recovered in FedGC, it does not raise privacy concerns as the image is generative rather than real.

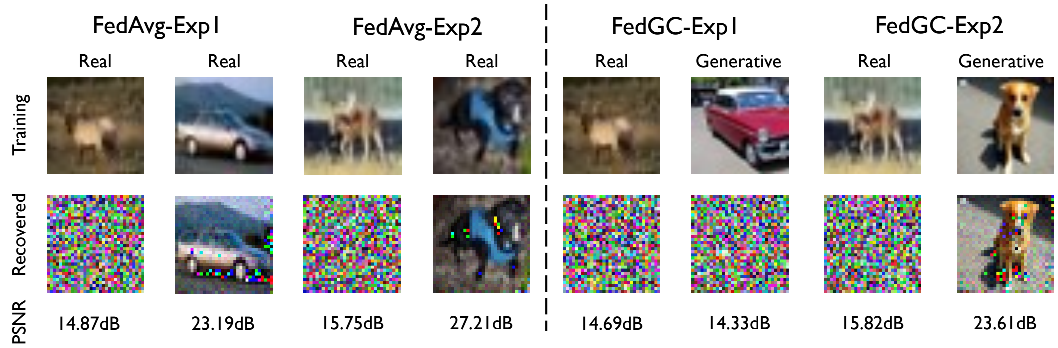


Figure 10: Evaluation of privacy preservation by DLG [70]. Results show that FedGC mitigates the risks of privacy leakage.

Table 12: Membership inference attack accuracy comparisons when FedAvg and FedGC achieve similar task accuracy. We consider two scenarios where the total number of clients' real samples is 50k and 10k, respectively. We also explore the effects of using different number of generated samples. FedGC can reduce privacy leakage to a very low level (since random guess is 50%) while maintaining task accuracy at the same time.

Number of Real Samples		50k		10k	
Accuracy		Task	Attack	Task	Attack
No. of Generated Samples	0	59.71	60.55	35.48	77.55
	10k	61.65	52.05	35.97	52.80
	20k	62.49	51.20	39.18	52.85
	30k	61.82	51.95	39.40	52.50
	40k	60.38	51.20	37.17	52.75
	50k	62.49	51.60	38.68	52.35

F.3 More Details about Differential Privacy

Differential privacy (DP) [72] has become a widely accepted framework for ensuring privacy in statistical analyses. With the help of DP, we can implement computation on large datasets and keep individual data points indistinguishable at the same time, which protects individual's privacy.

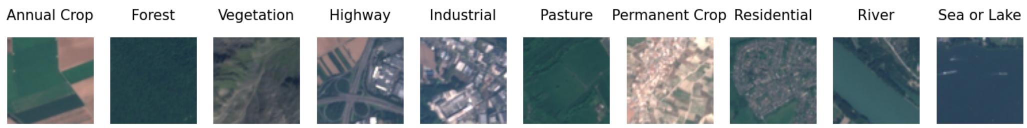
We use privacy parameters ϵ and δ to formally define DP. Specifically, a randomized mechanism $M : \mathcal{D} \to \mathcal{R}$ is (ϵ, δ)-differentially private for $\epsilon > 0$ and $\delta \in [0, 1)$ if for any two neighboring datasets $D, D' \in \mathcal{D}$ differing by at most one entry and for any subset of outputs $R \subseteq \mathcal{R}$ it holds that

$$\mathbb{P}(M(D) \in R) \leq \exp(\epsilon)\mathbb{P}(M(D') \in R) + \delta.$$

Differentially Private Stochastic Gradient Descent (DP-SGD) [73] is a DP algorithm that trains a neural network using sensitive data modified from SGD. In DP-SGD, per-sample-gradients are clipped and Gaussian noise is added to the clipped gradients.

In our experiments, we use a commonly used library Opacus [74] to implement DP-SGD, ensuring sample-level DP. Opacus uses a parameter called 'noise_multiplier' to change the noise level, which represents the ratio of the standard deviation of the Gaussian noise to the ℓ_2-sensitivity of the function to which the noise is added. It uses another parameter called 'max_grad_norm' to clip the gradients, which means the maximum norm of the per-sample gradients.

For experiments on image dataset CIFAR-10, we set noise_multiplier to 0.1 and max_grad_norm to 2, when using text dataset Sentiment140, we set noise_multiplier to 0.5 and max_grad_norm to 2. As shown in Figure 3, FedGC with differential privacy (DP) achieves good privacy-utility trade-off with privacy guarantee.



(a) EuroSAT: real data samples

(b) EuroSAT: generated similar samples

(c) EuroSAT: generated dissimilar samples

Figure 11: Visualization of real and generated data. (a) Visualization of real data samples from the EuroSAT dataset. (b) Visualization of generated data samples that are more aligned with the corresponding semantic or real data. (c) Visualization of generated data samples that are not aligned with the corresponding semantic or real data.

G Visualization of Real and Generated Data

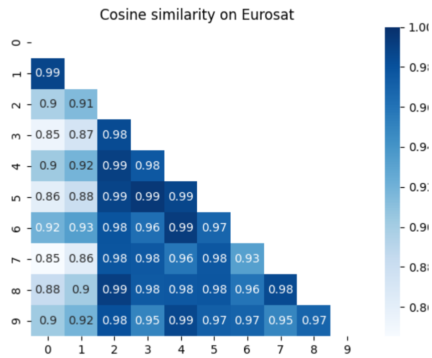
Generated data is diverse, but may not be similar to real data. We notice that the generated data samples do not always closely resemble real images, indicating the gap between generative data and real private data (at least visually). Yet, their inclusion still improves the FL's performance under data heterogeneity, which may result from two perspectives. (1) The generative data might act as a form of data augmentation, which potentially introduces variations that are not covered by the original dataset. (2) The generative data diversify the dataset, which serves as a form of implicit regularization, preventing the model from over-fitting to the potentially biased private local data.

We visualize the real data and generated data on EuroSAT [54] in Figure 11. For the uncommon and detailed satellite images in EuroSAT [54], the quality of the data generated by the diffusion models varies. From the naked eye, the data generated by some diffusion can capture the semantic information brought by the label very well. For example, the generated images with the label "River" as guidance do contain rivers, but hard to achieve a similar satellite style to actual images. Although the gap between generated and actual data definitely exists, generated data obviously improves specific task performance, which is demonstrated by our extensive experiments.

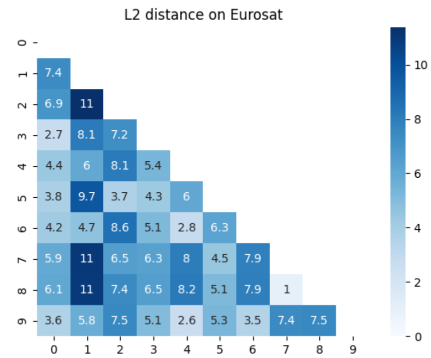
H FedGC Mitigates Data Heterogeneity

We visualize the cosine similarity and ℓ_2 distance of features on EuroSAT and PACS in Figure 12 and Figure 13 respectively. We measure the discrepancy among local data in clients on the feature level, using 2 metrics: cosine similarity and ℓ_2 distance. To be specific, we calculate the average features with pre-trained ResNet-18 [61] on each client in turn, and then measure the indicators between all pairs of clients.

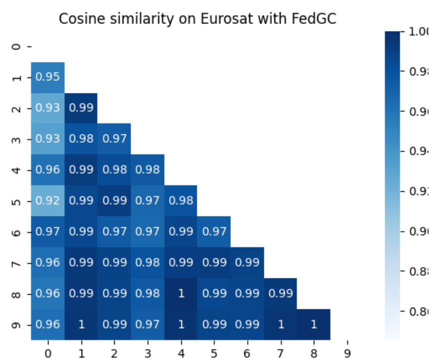
Results in the figures manifest that after applying FedGC, the cosine similarity and ℓ_2 distance among client pairs separately increase and decrease. In other words, local data possessed by clients are more homogeneous than before. FedGC efficiently mitigates data heterogeneity by generating corresponding data on the client side. From the feature respective, we show the latent reason for significant performance improvement brought by FedGC.



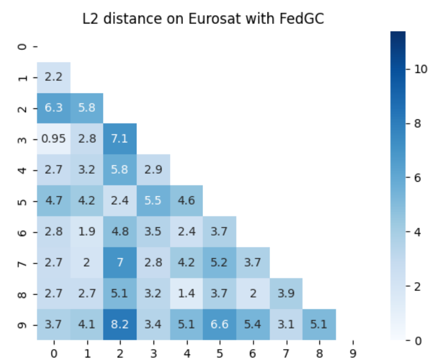
(a) cosine: FedAvg



(c) ℓ_2: FedAvg



(b) cosine: FedGC



(d) ℓ_2: FedGC

Figure 12: Feature cosine similarity and ℓ_2 distance heatmap among 10 clients on EuroSAT. We calculate the two metrics on average data features among clients using the pre-trained ResNet-18 [61]. FedGC enhances the feature similarity and closes their distance, which effectively mitigates the feature-level heterogeneity on EuroSAT.

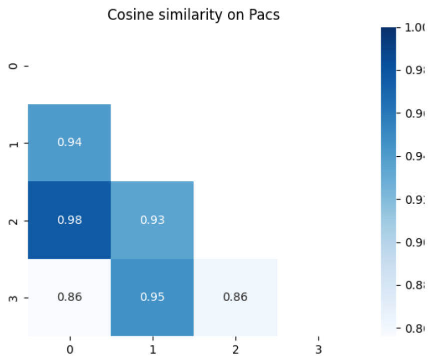
I FedGC with Partial Clients Capable of Generation

Our proposed FedGC framework is also applicable in cases where not every client has the capability to generate data. Here, we experiment on CIFAR-10 under two different heterogeneity levels. In Table 13, we compare vanilla baseline with no generative data, FedGC where all clients can generate data, and FedGC where only half of the clients can generate data.

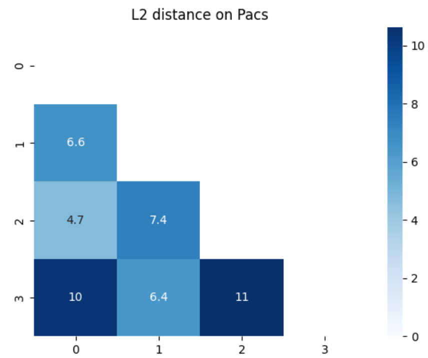
From the table, we see that (1) our proposed FedGC can consistently and significantly achieve the best performance despite the amount of generation-capable clients. (2) Surprisingly, we find that under low heterogeneity level, when applied to SCAFFOLD [18], FedGC with few generation-capable clients even performs better. This interesting finding demonstrates that our framework may be further improved by more fine-grained designs regarding who is responsible for data generation and the volume of data to be generated.

J FedGC under Different Heterogeneity Levels

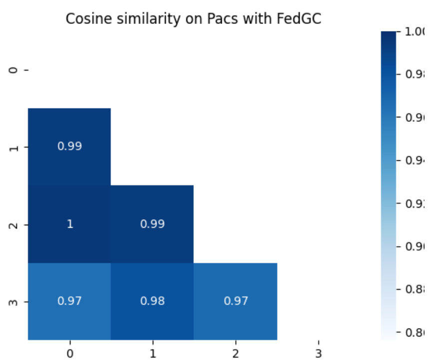
Here, we conduct experiments of three baselines including FedAvg, FedProx, and SCAFFOLD, with different heterogeneity levels on CIFAR-10. The Beta β stands for the hyper-parameter in the Dirichlet distribution. As β increases in [0.05, 0.07, 0.1, 0.3, 0.5, 1.0, 5.0], the data heterogeneity level reduces. Illustrated in Figure 14, we can observe that (1) FedGC consistently outperforms these three algorithms in all different data heterogeneity levels. (2) As the heterogeneity level increases, the accuracy improvement brought by FedGC significantly elevates, which showcases the reliability of FedGC to mitigate heterogeneity, one of the intricate issues in FL.



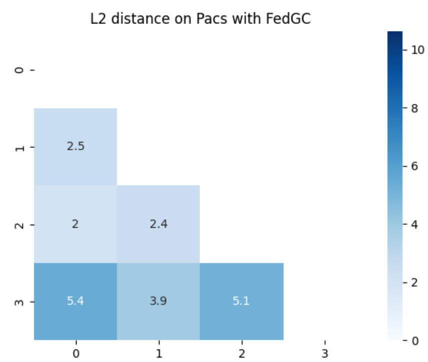
(a) cosine: FedAvg



(c) ℓ_2: FedAvg



(b) cosine: FedGC



(d) ℓ_2: FedGC

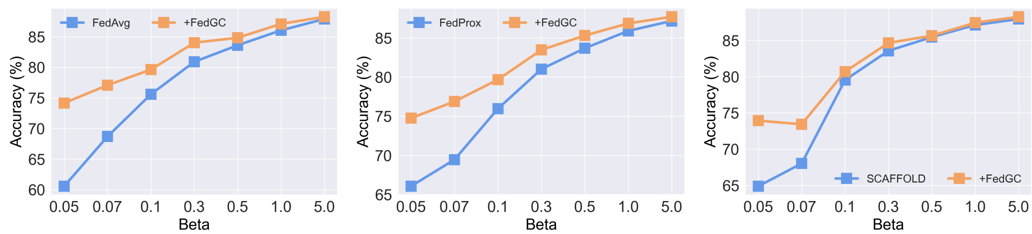
Figure 13: Feature cosine similarity and ℓ_2 distance heatmap among 4 clients on PACS. We calculate the two metrics on average data features among clients using the pre-trained ResNet-18. FedGC enhances the feature similarity and closes their distance, which effectively mitigates the feature-level heterogeneity on PACS.

Table 13: Experiments of a scene in which partial clients are capable of generation. 1k/50% indicates only half of the clients are capable of generation. However, FedGC still significantly outperforms the baseline with no generative data.

H-Level		High			Low	
Generation	No	1k/100%	1k/50%	No	1k/100%	1k/50%
FedAvg	60.77	73.99	71.53	71.57	79.73	77.45
FedProx	63.62	73.69	72.65	75.76	79.25	79.23
SCAFFOLD	65.00	75.75	73.28	78.74	80.29	81.27

K FedGC for Partial Client Participation Scenarios

Here, we conduct experiments of three baselines including FedAvg, FedProx, and SCAFFOLD on CIFAR-10 with Dirichlet distribution parameter $\beta = 0.1$. Specifically, we set the communication round to 200, local iteration number to 100, and try different client number and participation rate. As illustrated in Table 14, we can observe that FedGC still significantly outperforms the baseline with no generated data under each circumstance.



(a) FedAVg (b) FedProx (c) SCAFFOLD

Figure 14: Performance comparisons between vanilla baseline and baseline in FedGC framework under different heterogeneity levels on CIFAR-10. Beta (β) is the hyper-parameter in Dirichlet distribution. As the heterogeneity level increases (Beta decreases), the improvement brought by FedGC becomes more significant. This indicates that FedGC can effectively alleviate the issue of data heterogeneity.

Table 14: Experiments of a scene in which only partial clients participate in training each round. We conduct experiments on three different total client numbers and several different participation rates. For example, client 200 and participation rate 5% means randomly selecting 10 clients to participate in training each round. In each case, FedGC still significantly outperforms the baseline with no generative data.

Baseline	Client	200			100		50	
	Participation	5%	10%	20%	10%	20%	10%	20%
FedAvg	Vanilla	53.62	60.00	65.76	56.53	57.69	55.90	63.33
	+ FedGC	68.93	74.06	75.74	74.16	74.26	75.34	77.20
FedProx	Vanilla	53.93	59.95	64.53	56.74	59.54	56.36	65.66
	+ FedGC	70.23	73.79	75.07	74.39	74.05	75.47	77.47
SCAFFOLD	Vanilla	60.41	68.02	70.15	65.03	68.12	65.73	72.42
	+ FedGC	71.65	74.83	77.54	74.38	76.26	72.74	77.56

L Global-model-based Data Filtering

We propose global-model-based data filtering, where each client conducts data filtering on the client side according to the received global model before local model training. Specifically, to determine which data to filter, a client feeds its generated data to the global model to evaluate the loss value for each data sample. Then, each client selects the top $x\%$ data (we set $x = 90$ here) and mixes the selected generated data with its real data.

Furthermore, since the global model might perform drastically differently on different categories, simply selecting according to the loss of all data samples may result in imbalanced filtering. That is, this could make global model filter out most of the samples where it performs poorly. Addressing this, we further propose category-wise data filtering based on global model, which filers the same ratio of data for each category.

Here, we perform experiments on EuroSAT dataset with two heterogeneity levels in Table 15. Vanilla denotes FedAvg itself, No F denotes FedGC without filtering, F@50 denotes filtering from round 50, F@50-C denotes category-wise filtering. From the table, we see that (1) under a high heterogeneity level, F@75 contributes to higher performance than No F, even with only 90% of data at final rounds. (2) Category-wise filtering generally performs better than unified filtering, indicating its effectiveness. (3) Nevertheless, such filtering technique can not always ensure performance improvement, calling for more future work. The performance drop could result from reduced number of data samples and ineffective filtering.

Overall, here we just provide an initial attempt to consider the potential of data filtering. We believe more future works could be proposed to better filter the generated data such that we could use the generated data more efficiently.

Table 15: Experiments of global-model-based data filtering. We conduct our initial attempt on EuroSAT dataset with two heterogeneity types ($\beta = 0.05$ and $\beta = 0.1$ denote high and low heterogeneity level respectively). F@50 means start filtering after 50 communication rounds and C means filtering by each class.

Heterogeneity Level	Vanilla	No F	F@50	F@75	F@50-C	F@75-C
High	53.82	74.83	72.96	74.93	73.50	74.20
Low	75.59	84.46	83.82	83.83	84.19	83.83

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We list our main claims and contributions accurately point by point in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We provide a discussion of our limitations in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: Our paper does not include theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide our experiment setting details in Section 4 and Appendix C. For all conclusions we draw, we provide necessary information to reproduce experimental results in our analysis. Our code is also available and open source at public repository `https://anonymous.4open.science/r/FedGC` for anyone who want to reproduce our experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: All data and generative models we use is publicly available. The complete code for our experiments in the manuscript is accessible at public repository `https://anonymous.4open.science/r/FedGC`. Additionally, we provide detailed set-ups and descriptions in the guidance file of our repository.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide our experiment setting details in Section 4 and Appendix C. For all conclusions we draw, we provide necessary information to reproduce experimental results in our analysis. Our code is also available and open source at public repository `https://anonymous.4open.science/r/FedGC` for anyone who want to reproduce our experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not provide information about statistical significance of the experiments yet in main table due to the limit of space. However, we run the experiments three times and average the results (e.g. in Table 1).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the compute resources and environment to reproduce our experiments in Appendix C

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We strictly conform with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We provide discussion about our societal impacts in Appendix B.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This paper mainly focus on technical advancement, and the datasets and generative models we used are all publicly available. Thus, the paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We cite the original paper of all datasets and generative models we use in Section 4. For bits of others' code, we list the source and license.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: Our paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

Official Review of Submission6501 by Reviewer MURY

Official Review Reviewer MURY 13 Jul 2024, 06:33 (modified: 25 Sept 2024, 23:53) Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer MURY

Revisions

Summary:
The paper proposes a method FedGC to tackle data heterogeneity in federated systems. The main solution revolves around each client over-sampling selected classes using generative models. There are two methods proposed: a prompt based and a data based approach.

Soundness: 2: fair
Presentation: 3: good
Contribution: 2: fair
Strengths:

1. Data heterogeneity is an important problem to solve in federated learning context. The paper acknowledges that the proposed method utilizes more computational cost and provides a budget allocation method.
2. The paper is well-written and easy to follow.

Weaknesses:

1. The originality of this method is limited. The core idea seems to be to use generative models to over-sample minority classes. There have been methods proposed earlier using generative models in federated learning. [1][2].
2. Data augmentation techniques like image based data augmentations are also popular techniques to oversample. More experiments needs to be conducted to see how this method compares to typical augmentation techniques as baselines.
3. It is not clear what the motivation to participate in federated learning is if clients can use generative models to get access to more data. Moreover, if clients already have access to foundational models and can use them without any privacy constraints, why is federated learning needed? For example, if a client has access to a multi-modal LLM, then they can write a prompt to classify the image directly. For this paper, a LLM based classification baseline is needed, as the proposed solution already assumes LLMs can be used.

[1] https://arxiv.org/pdf/2306.16064 [2] https://www.sciencedirect.com/science/article/abs/pii/S0743731524000807

Questions:

1. For table 9, could the authors explain the exact procedure used to get the results? How many samples are being generated ? Is it prompt based generation or real-data based generation?
2. How are the privacy restrictions of FL not violated? In section 3.2, the authors mention the use of ChatGPT. There have been various reports of privacy concerns with chatGPT and that chatGPT can store user data (https://www.forbes.com/sites/kateoflahertyuk/2024/05/17/chatgpt-4o-is-wildly-capable-but-it-could-be-a-privacy-nightmare/).

Limitations:
The limitation of privacy is not addressed here. Please see the above questions.

Flag For Ethics Review: No ethics review needed.
Rating: 3: Reject: For instance, a paper with technical flaws, weak evaluation, inadequate reproducibility and/or incompletely addressed ethical considerations.
Confidence: 4: You are confident in your assessment, but not absolutely certain. It is unlikely, but not impossible, that you did not understand some parts of the submission or that you are unfamiliar with some pieces of related work.
Code Of Conduct: Yes

Rebuttal by Authors

Rebuttal Authors (Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 07 Aug 2024, 06:45 (modified: 07 Aug 2024, 20:58) Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors Revisions

Rebuttal:
Thanks for your time and suggestions. Here are our detailed replies to your questions.

W1: The originality of this method is limited. The core idea seems to be to use generative models to over-sample minority classes. There have been methods proposed earlier using generative models in federated learning. [1][2].

Response: Thanks for pointing out these two related works. However, we would like to emphasize two main things: (1) the reviewer's understanding of our core idea is not accurate; (2) our method is significantly different from the referred methods. Here is the evidence.

Our core idea is not to over-sample minority classes, but rather, to generate diverse content to mitigate data heterogeneity, which treats majority and minority classes equally. This can be clearly verified by our experimental results in Table 3 and we report here again for convinience. From the table, we can see that equal (treating all classes equally) allocation contributes to better performance than inverse and water strategies (tends to allocate more generative data to minority classes). We hope that this result can convince the reviewer of what our true core idea is.

[**Table R1.** Experiments with different class allocation strategies]

	Equal	Inverse	Water
Acc	**74.50**	68.10	71.26

Our paper is significantly different from the recommended papers. We would like to emphasize that generative model itself is a big topic that has been researched for decades. It would be too harsh to reject our paper simply because some existing papers are also related to generative models, since our paper is a totally different one.

[Methodology] Our paper proposes mixing generated and real data during local model training at client side while the server and client share information via model parameters. While in [1], client shares text embeddings to the server, who generates data and trains the global model; in [2], client shares generators and latent vectors to the server, who conducts reconstruction and trains the global model.

[Advantage] Our method follows the classical framework of FedAvg, making it naturally compatible with mature techniques such as secure aggregation and differential privacy (see results in Figure 2,3); while whether [1,2] will do so is unclear since their frameworks do not follow FedAvg anymore. Therefore, our work can be easily and safely deployed in practice.

W2: More experiments needs to be conducted to see how this method compares to typical augmentation techniques as baselines.

Response: Thanks for the suggestions. Actually, our implementations of baselines have included several typical data augmentations (e.g., RandomCrop and RandomHorizontalFlip) in image data and we will include this implementation detail in the revision. Here, we additionally provide the results of FedAvg without data augmentation. From the table, we see that FedGC significantly outperforms both baselines.

[**Table R2.** Comparisons with FedAvg with and without data augmentation]

	CIFAR-High	CIFAR-Low
FedAvg without data augmentation	51.03	60.20
FedAvg	61.25	75.88
FedGC	**74.50**	**79.73**

W3: Motivation of federated learning when clients can use generative models.

Response: Thanks for your valuable output. We would like to respond from three perspectives.

(1) **Clients' motivation to participate in FL.** We have conducted the experiments in Table 9 (reported below for convinience) by comparing local training with generative content and federated learning with generative content. From the table, we see that our FedGC still brings significant benefits compared to clients' training with local and generative data.

[**Table R3.** Comparisons with local training and FL with generatative content]

Method	CIFAR-High	CIFAR-Low
Local+GC	46.89	50.47
FedGC	**74.50**	**79.73**

(2) **Our idea does not necessarily require client's access to large models.** We have provided detailed discussions in Section E showing that generating data can be conducted either at client or server side (except for real data guidance for rare tasks).

(3) **The cost of inferring small models is significantly lower than large models, meaning that it is better to deploy small models if they could work** Even if clients have access to large generative models, it only takes several steps of inference of the generative models to generate data to facilitate FL on small models. Once the FL process concludes, clients can use the small models for their tasks rather than the large models, which require significantly lower inference cost in the long term (our small model only has 0.016% parameters compared to a 8B-size LLM).

Q1: Details about table 9.

Response: For CIFAR-10 (50000 real images in total), in FedGC, there are 10 clients, each with 1000 generated images (since we are using the equal allocation). In Local+GC, each client trains its own model using its own real images with 1000 generated images. Both of them are prompt-based generation.

Q2: Will using ChatGPT affect privacy?

Response: Thanks for this comment. Our framework is orthogonal with the choice of generative models. Using ChatGPT is just an example and one can use any open-source LLMs instead (please note that we have discussed this in Line 238-241).

To further verify this point, we now replace the ChatGPT with the open-source Llama3-8B-Instrcut model on Sentiment140 dataset. From the following table, we can clearly see that since Llama3-8B-Instruct has better performance than ChatGPT in following instruction, it even contributes to better performance.

[**Table R4.** Results of using different LLMs]

Method	FedAvg	FedGC (ChatGPT)	FedGC (Llama3-8B-Instruct)
Accuracy	66.76	72.45	74.97

Overall, we hope that our responses can fully address your concerns and will be grateful for any feedback.

Official Comment by Authors

Official Comment Authors (Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 11 Aug 2024, 17:37

Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer MURY

Comment:
Dear Reviewer:

Thanks again for the comments. We have now provided more clarifications, explanations, and experiments to address your concerns. Specifically, we:

- clarify our contributions and originality.
- provide experimental results to compare with baselines with and without data augmentation, show clients' motivation to participate FL, and show the compatibility of our method with other LLMs.

Please kindly let us know if anything is unclear. We truly appreciate this opportunity to improve our work and shall be most grateful for any feedback you could give to us.

Replying to Official Comment by Authors

Thanks for the Rebuttal

Official Comment Reviewer MURY 12 Aug 2024, 08:54 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer MURY

Comment:
Thanks for addressing most of my comments. However, my key reason for not accepting the paper is this: In order for a generative model to generate good datapoints of some dataset (say Sentiment140), it already should have a pretty good understanding of that dataset, hence should be a good classifier as well. Please refer to https://arxiv.org/html/2304.04339v2, here they show chatGPT is an impressive zero-shot sentiment analyzer. This makes me question the motivation of this work. Why would a client not use chatGPT (or any other LLM) directly for sentiment analysis (few-shot), but instead use it to generate more data and train in a federated setting? I do not see numbers to back this claim.

Replying to Thanks for the Rebuttal

Official Comment by Authors

Official Comment Authors (Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 12 Aug 2024, 12:07

Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer MURY

Comment:
Dear Reviewer:

Thanks for the reply. We kindly remind you that you may be ignoring the **huge gap of inference cost** between our federated learning model and the large generative models (our small model **only has 0.016%** parameters compared to a 8B-size LLM), which exactly shows why federated learning on small models is necessary.

Sadly, the reason that you are using to reject our paper can be universally used to reject tons of accepted papers. For example,

- MobileLLM [1] trains sub-billion (e.g., 350M) parameter language models for the device-side usage, which performs worse than ChatGPT. This model has 4.4% parameters compared to a 8B-size LLM. Then, according to your criterion, why do we need MobileLLM if we already have ChatGPT.
- WizardLM (7B/13B/70B) [2] distills knowledge from ChatGPT to Llama models, which also performs worse than its teacher model or GPT-4. Then, according to your criterion, why do we need WizardLM if we already have ChatGPT/GPT-4.

There are so many cases where we have a stronger but larger models at hand but still endeavor for training smaller models with the help of stronger and larger models, especially for the field of knowledge distillation [3].

Similarly, in our case, the large generative models are **only required for several inference times** (e.g., 1-10 times for Sentiment140) during the training time of small models. After the training time, the large generative models are **no longer needed** and we can deploy the small models for applications. Since real-time application is a long-term issue (maybe used every day), applying small models would require **significantly less inference cost** compared to applying large models.

We sincerely hope that the reviewer can seriously think about our response and look forward to any feedback!

[1] Liu, Zechun, et al. "MobileLLM: Optimizing Sub-billion Parameter Language Models for On-Device Use Cases." Forty-first International Conference on Machine Learning.

[2] Xu, Can, et al. "WizardLM: Empowering large pre-trained language models to follow complex instructions." The Twelfth International Conference on Learning Representations. 2024.

[3] Gou, Jianping, et al. "Knowledge distillation: A survey." International Journal of Computer Vision 129.6 (2021): 1789-1819.

↪ *Replying to Thanks for the Rebuttal*

Official Comment by Authors

Official Comment ✎ Authors (👁 Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 📅 14 Aug 2024, 15:30

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer MURY

Comment:
Dear Reviewer MURY:

The rebuttal deadline is approaching in less than 5 hours, and we have carefully addressed your recent concerns with detailed responses. We kindly ask that you review our replies at your earliest convenience. If there are any additional questions or issues, please do not hesitate to reach out. If our responses have satisfactorily resolved your concerns, we would greatly appreciate a higher score.

Thank you for your attention and consideration.

Official Review of Submission6501 by Reviewer PCj3

Official Review ✎ Reviewer PCj3 📅 12 Jul 2024, 16:27 (modified: 25 Sept 2024, 23:53) 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer PCj3

Revisions

Summary:
The authors propose a new direction for tackling data heterogeneity in federated learning by introducing generative content. They propose FedGC, where clients train local models on both private real and generative data. The authors present a comprehensive empirical study across datasets, heterogeneity types, modalities, and baselines. Experiments verify that FedGC not only improves task performance but also preserves privacy better.

Soundness: 3: good
Presentation: 3: good
Contribution: 3: good
Strengths:

- The paper is well-written, and the motivation is clear. It is a good attempt to explore the interplay between FL and generative content.
- The proposed FedGC is a flexible framework that enables diverse designs while keeping the framework simple to deploy in practice.
- Sufficient experiments are provided. FedGC can improve the performance of many existing baselines in multiple scenarios. It can also mitigate the risk of membership inference attacks.

Weaknesses:

- The authors compare their FedGC with many baselines, however, there is no detailed description of the baselines.
- It is unclear how the sequential training strategy is implemented. Is it round-level or epoch-level? More explanation is expected.

Questions:
Please refer to weakness.

Limitations:
The authors adequately addressed the limitations.

Flag For Ethics Review: No ethics review needed.
Rating: 7: Accept: Technically solid paper, with high impact on at least one sub-area, or moderate-to-high impact on more than one areas, with good-to-excellent evaluation, resources, reproducibility, and no unaddressed ethical considerations.
Confidence: 5: You are absolutely certain about your assessment. You are very familiar with the related work and checked the math/other details carefully.
Code Of Conduct: Yes

Rebuttal by Authors

Rebuttal ✎ Authors (👁 Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 📅 07 Aug 2024, 06:47 (modified: 07 Aug 2024, 20:58)

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors Revisions

Rebuttal:
Thanks for your recognition, time and suggestions. Here are our detailed replies to your questions.

W1: The authors compare their FedGC with many baselines, however, there is no detailed description of the baselines.

Response: We apologize for the missing details. Here are the details and we will include these in the revision.

- FedAvg is the most basic federated learning method
- FedAvgM introduces a momentum term when updating the aggregated global model on the server side.
- FedProx applies an additional L2 regularization term between local model and global model during local model training on the client side
- SCAFFOLD introduces a control variate for correcting gradient during local model training
- MOON uses a contrastive loss term to maximize the agreement between the features of current local model and global model, while minimizing the agreement between the features of current local model and previous local model.
- FedDecorr applies a regularization term during local training that encourages different dimensions of representations to be uncorrelated
- FedDyn proposes dynamic regularizer for each device at each round, so that in the limit the global and local solutions are aligned
- FedSAM leverages Sharpness Aware Minimization (SAM) local optimizer for local learning generality
- FedDisco proposes to aggregate local models based on dataset size and discrepency between local and global distributions

---

W2: It is unclear how the sequential training strategy is implemented. Is it round-level or epoch-level? More explanation is expected.

Response:

Sorry for the missing details. The sequential training strategy is epoch-level. To be more specific, suppose there are two sets (A and B) of data and the number of local epochs for each round is set to be 2x. Then, at each round, we first train x epochs on set A and then train on set B for the following x epochs.

Overall, we hope that our responses can fully address your concerns and will be grateful for any feedback.

↪ *Replying to Official Comment*

oops. Reviewer K6BJ gave a response at an improper section.

Official Comment ✎ Authors (👁 Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 📅 08 Aug 2024, 15:09

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

Comment:
Dear Reviewer **K6BJ**:

Thanks for your timely feedback and we are glad that our responses addressed your concerns.

It seems that you are responding at the section of Reviewer **PCj3**. Would you mind to delete this one and put it to the right position to avoid any confusion? Thanks.

↪ *Replying to Rebuttal by Authors*

Official Comment by Reviewer PCj3

Official Comment ✎ Reviewer PCj3 📅 13 Aug 2024, 21:48 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

Comment:
Thanks for the author's response. After browsing the discussions between the authors and Reviewer MURY and GY7j, I agree with the authors that it is still necessary to train small models even if we have available large models and that leveraging large generative models to facilitate federated learning on small models is an interesting and promising direction. Thus, I increased my score.

Official Review of Submission6501 by Reviewer GY7j

Official Review ✎ Reviewer GY7j 📅 11 Jul 2024, 23:43 (modified: 25 Sept 2024, 23:53) 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer GY7j
Revisions

Summary:
The paper introduces FedGC, a data-centric framework designed to address the issue of data heterogeneity in federated learning. By enriching client data with diverse generative content, FedGC aims to mitigate overfitting and improve the generalization of local models. The framework explores four critical dimensions: budget allocation, prompt design, generation guidance, and training strategy. Extensive empirical studies on multiple datasets and baselines demonstrate that FedGC consistently enhances task performance and privacy preservation when combining it with the other federated learning approaches.

Soundness: 3: good
Presentation: 3: good
Contribution: 2: fair
Strengths:

1. The studied problem is emerging. It is important to study federated learning in the field of generative models.
2. The method is simple and easy to understand.
3. It is interesting to see that FedGC also improve the privacy of local data.

Weaknesses:

1. My main concern is about the setting of the study. FedGC utilizes generative models to generate data and test on the popular tasks. However, in the case where generative data is rich, the public data is also rich to train the generative model. When we have enough public data to train generative models, we can also use the data to train models for corresponding tasks. From my view, the paper should study the case where the generative model is not able to generate high-quality task-specific data.
2. The paper misses important baselines. The paper should compare FedGC with 1) the approach where no local data is used and only generated data is used, 2) the approach where centralized learning is applied to the generative data. It is may be the case that federated learning is not needed and generative data is enough for training the model.

Questions:

1. How does FedGC compare with centralized learning on generative data and federated learning on generative data without local data?
2. Can you try FedGC on the settings where the generative model is not able to generate high-quality task-specific data?

Limitations:
Please see the weaknesses.

Flag For Ethics Review: No ethics review needed.
Rating: 4: Borderline reject: Technically solid paper where reasons to reject, e.g., limited evaluation, outweigh reasons to accept, e.g., good evaluation. Please use sparingly.
Confidence: 4: You are confident in your assessment, but not absolutely certain. It is unlikely, but not impossible, that you did not understand some parts of the submission or that you are unfamiliar with some pieces of related work.
Code Of Conduct: Yes

Rebuttal by Authors

Rebuttal ✎ Authors (👁 Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 📅 07 Aug 2024, 06:58 (modified: 07 Aug 2024, 20:58)
👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors Revisions

Rebuttal:
Thanks for your time and suggestions. Here are our detailed replies to your questions.

W1: My main concern is about the setting of the study. FedGC utilizes generative models to generate data and test on the popular tasks. However, in the case where generative data is rich, the public data is also rich to train the generative model. When we have enough public data to train generative models, we can also use the data to train models for corresponding tasks. From my view, the paper should study the case where the generative model is not able to generate high-quality task-specific data.

Response: Thanks for your valuable output. We would like to kindly inform the reviewer that we have considered the related issues in our paper. We would like to further clarify from the following two aspects:

[Results on medical and satellite images, for which the generative model is not able to generate high-quality task-specific data] Table 1 includes results on EuroSAT (satellite image) and Table 5 includes results on HAM10000 (medical image). Here, we report the results again for convinience. For both domains, generative models are not able to generate high-quality task-specific data, especially for the medical domain! However, we still see that our FedGC brings significant performance gain. Additionally, we visualized the generated satellite samples in Figure 11 but did not show medical examples because the generated medical images could cause discomfort.

[**Table R1.** Experiments on uncommon domains]

Method	Satellite	Medical
FedAvg	53.82	48.57
FedGC (ours)	**74.83**	**56.67**

[Generated data alone is not sufficient for training performant models] Here, we report our results in Table 6 to here for convenience. We can see that training entirely on generated data achieves significantly worse performance than training entirely on private real data, indicating that there is a big gap between generated and real data. Therefore, despite that generative models are good at generating high-quality data, they could fail to generate task-specific or domain-specific data. Our FedGC that leverages both private and generated data achieves the siginifncalty best performance.

[**Table R2.** Only generative data is not sufficient to train good FL models]

Method	FedAvg (private data)	FedAvg (generative data)	FedGC (private and generative data, ours)
Acc	60.77	41.85	**73.99**

W2: The paper misses important baselines. The paper should compare FedGC with 1) the approach where no local data is used and only generated data is used, 2) the approach where centralized learning is applied to the generative data. It is may be the case that federated learning is not needed and generative data is enough for training the model.

Response: Thanks for the advice. In the following, we show the results where centralized learning is applied to generative data. From the table, we see that centralized learning on generative data achieves very poor performance; while our FedGC achieves the best performance. This verifies that **federated learning is needed and generative data is not enough for training the model.**

We believe that this is a convincing result to address the reviewer's concern. Note that since centralized learning on generative data cannot achieve good performance, we did not experiment on federated learning on generative data anymore, which would performs worse than centralized learning.

[**Table R3.** Comparison with centralized learning on generative data]

Method	Centralized learning (generative data)	FedAvg (private data)	FedGC (private and generative data, ours)
CIFAR-10	44.91	61.25	**74.50**
EuroSAT	19.70	53.82	**74.83**

Overall, we are sorry for causing the potential confusion and we believe that our responses can fully address your concerns. We will be grateful for any feedback.

Official Comment by Authors

Official Comment Authors (Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 11 Aug 2024, 17:41

Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer GY7j

Comment:
Dear Reviewer:

Thanks again for the comments. We have now provided more clarifications, explanations, and experiments to address your concerns. Specifically, we:

- show that our method still brings **significant perfomance gain (up to 20% improvement)** in case where the generative model is not able to generate high-quality task-specific data.
- compare with centralized learning on generative data (as recommended by the reviewer) and verify that our method **performs significantly better (up to 55% improvement).**

Please kindly let us know if anything is unclear. We truly appreciate this opportunity to improve our work and shall be most grateful for any feedback you could give to us.

Replying to Official Comment by Authors

Official Comment by Reviewer GY7j

Official Comment Reviewer GY7j 12 Aug 2024, 16:37 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer GY7j

Comment:
Thank you for your response. I appreciate the effort and have updated my score to 4 as some of my concerns were addressed. However, I still have a few remaining issues:

1. The core idea of the paper is to generative models to generate data that can improve federated learning. The idea is somewhat straightforward, and the results seem expected.
2. You demonstrate that even when the quality of generative data is low, it can still benefit federated learning. However, for the generative model, the training of the model may already include the test dataset or other medical data. This raises the question of whether it is necessary to employ a generative model to create data, rather than simply utilizing publicly available related datasets directly.

Replying to Official Comment by Reviewer GY7j

Official Comment by Authors

Official Comment Authors (Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 12 Aug 2024, 18:05

Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer GY7j

Comment:
Thank you for increasing the score and we would like to further address your remaining concerns.

Concern 1: The idea is somewhat straightforward, and the results seem expected.

Response: We would like to address your concerns from two aspects.

First, we would like to kindly remind the reviewer that our results are not as easily expected as the reviewer thought. Please refer to the following table, where we show that **FedAvg on only generative data performs much worse than FedAvg on only private data** (41.85 v.s. 60.77). This indicates that the quality of generative data is not as high as the real private data. Therefore, it is **not straightforward** to expect that introducing generative data to real private data could bring such benefits! We reveal such finding in our paper, which points out a new direction for tackling data heterogeneity.

[**Table R2.** Only generative data is not sufficient to train good FL models]

Method	FedAvg (private data)	FedAvg (generative data)	FedGC (private and generative data)
Acc	60.77	41.85	**73.99**

Second, we would like to defend for ourselves that the conciseness of our solution should not be regarded as a weakness. Rather, such conciseness makes our solution **easy to deploy** in real-world applications since we do not need to modify too much on the well-constructed FedAvg framework, making it compatible with a series of mature techniques such as secure aggregation and differential privacy. Also, we would like to direct the reviewer's attention towards two published papers that also seem to be 'straightforward'. [1,2] are both published by ICLR2023, which introduces pre-trained models as the initialization of global model in federated learning. And, that's all. We argue that the community should appreciate such methods that are **simple yet effective**.

[1] Nguyen, John, et al. "Where to Begin? On the Impact of Pre-Training and Initialization in Federated Learning." The Eleventh International Conference on Learning Representations. 2023.

[2] Chen, Hong-You, et al. "On the importance and applicability of pre-training for federated learning." The Eleventh International Conference on Learning Representations. 2023.

Concern 2: Whether it is necessary to employ a generative model to create data, rather than simply utilizing publicly available related datasets directly.

Response: We would like to address this concern from two aspects.

First, using a generative model offers a strong advantage over manually searching for appropriate public dataset: automation. Specifically, for any task, the data can be generated by simply inputting the label space (e.g., 10 words for CIFAR-10). In contrast, if we are searching for public data, we need to search for a set of images for each category, which is time-consuming especially when the label space is large (e.g., 1000 for ImageNet).

Second, we propose an effective solution for generating data for rare tasks. Specifically, we propose a real-data-guided generation method (Figure 8 and Figure 9), which promotes the fidelity of generated data. Please refer to Table 5, where we show that using text-guided and real-data-guided generation together yields the best performance.

We hope that our responses can fully address your concerns and look forward to your feedback!

Replying to Official Comment by Reviewer GY7j

Official Comment by Authors

Official Comment Authors (Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) 14 Aug 2024, 15:31

Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer GY7j

Comment:
Dear Reviewer GY7j:

The rebuttal deadline is approaching in less than 5 hours, and we have carefully addressed your recent concerns with detailed responses. We kindly ask that you review our replies at your earliest convenience. If there are any additional questions or issues, please do not hesitate to reach out. If our responses have satisfactorily resolved your concerns, we would greatly appreciate a higher score.

Thank you for your attention and consideration.

Official Review of Submission6501 by Reviewer K6BJ

Official Review · Reviewer K6BJ · 11 Jul 2024, 02:42 (modified: 25 Sept 2024, 23:53) · Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer K6BJ

Revisions

Summary:
FL facilitates collaborative model training using dispersed private data while maintaining privacy. However, data heterogeneity, a prevalent concern, significantly hampers current FL methods' effectiveness. This paper introduces a new approach, data-centric intervention, which directly reduces data heterogeneity by augmenting clients' local datasets with generative content. This leads to the proposal of FedGC, a streamlined yet potent framework where clients combine advanced generative data and their private data, guided by a four-pronged analysis. Experimental comparisons against nine baselines and examination on seven datasets validate that FedGC reliably and significantly improves both task performance and privacy preservation.

Soundness: 3: good
Presentation: 3: good
Contribution: 3: good
Strengths:

- This work offers FedGC, a FL with generative learning, which aims to solve the data heterogeneity problem and maintain good privacy performance.
- Comprehensive and thorough evaluation.
- Compared with previous work, this paper provides a new path to solve the data heterogeneity and privacy problems in FL, providing new ideas for the research community.

Weaknesses:

- Communication efficiency needs to be further improved.
- Evaluation on more various tasks is needed.

Questions:

- Do the authors consider further improving communication efficiency? As far as the results in Table 10 are concerned, FedGC still has higher communication overhead than traditional schemes, which is intolerable for resource-constrained FL.
- Do the authors consider further evaluation on other types of data such as text, table, and graph? Extensive and comprehensive evaluation will help demonstrate the superior performance of FedGC.

Limitations:
Please kindly refer to the above comments.

Flag For Ethics Review: No ethics review needed.
Rating: 7: Accept: Technically solid paper, with high impact on at least one sub-area, or moderate-to-high impact on more than one areas, with good-to-excellent evaluation, resources, reproducibility, and no unaddressed ethical considerations.
Confidence: 4: You are confident in your assessment, but not absolutely certain. It is unlikely, but not impossible, that you did not understand some parts of the submission or that you are unfamiliar with some pieces of related work.
Code Of Conduct: Yes

Rebuttal by Authors

Rebuttal · Authors (Jingyi Chai, Siheng Chen, Rui Ye, Lingjuan Lyu, +3 more) · 07 Aug 2024, 07:08 (modified: 07 Aug 2024, 20:58)
Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors · Revisions

Rebuttal:
Thanks for your recognition, time and suggestions. Here are our detailed replies to your questions.

W1: Communication efficiency needs to be further improved.

Q1: Do the authors consider further improving communication efficiency? As far as the results in Table 10 are concerned, FedGC still has higher communication overhead than traditional schemes, which is intolerable for resource-constrained FL.

Response:

Thanks for the comments. Actually, our FedGC can achieve better performance while requiring less communication cost compared to baselines since FedGC can speed up the convergence. On one hand, in Table 10, we are fixing the number of communication rounds between client and server, showing that our FedGC can brings 5.07% performnace improvement at the negaligible cost (0.007%). On the other hand, we show in Table 11 that applied on SCAFFOLD, FedGC can contribute to better performance with less communication cost.

To further verify the efficiency of FedGC, we further show the following table, where we run baselines for 100 rounds and our FedGC for 98 rounds. From the table, we see that our FedGC achieves the best performance with the lowest cost.

[**Table R1.** Comparisons on communication cost and accuracy]

Method	Cost	Accuracy
FedAvg	215,777,600	61.25
SCAFFOLD	431,555,200	63.98
FedGC	214,534,048	74.26

Besides, we also compare the required number of rounds to achieve a target accuracy (60% here) among baselines. From the table, we see that our proposed FedGC requires the least communication rounds to achieve the target accuracy. Specifically, compared to FedAvg, FedGC can save communication cost up to 63%.

[**Table R2.** Comparisons on the required number of rounds to achieve target accuracy]

Method	Round
FedAvg	73
FedProx	62
SCAFFOLD	53
FedGC (ours)	**27**

W2: Evaluation on more various tasks is needed.

Q2: Do the authors consider further evaluation on other types of data such as text, table, and graph? Extensive and comprehensive evaluation will help demonstrate the superior performance of FedGC.

Response: Thanks for the advice. We report our results on text modality in the following table, where we consider two datasets: Sentiment140 and Yahoo! Answers. From the table, we see that for text modality, our proposed FedGC still achieves significantly better performance compared to the baseline.

[**Table R3.** Experiments on text data]

Method	Sentiment140	Yahoo! Answers
FedAvg	66.76	49.79
FedGC	72.45	53.74

Overall, we hope that our responses can fully address your concerns and will be grateful for any feedback.

Replying to Rebuttal by Authors

Thanks for Authors' Rebuttal

Official Comment · Reviewer K6BJ · 08 Aug 2024, 19:30 · Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

Comment:
Thanks to the authors for their detailed responses! The above responses have addressed most of my concerns.