One-Shot Federated Learning with Classifier-Free Diffusion Models

Obaidullah Zaland[1*], Shutong Jin[2*], Florian T. Pokorny[2] and Monowar Bhuyan[1]
[1]Department of Computing Science, Umeå Unviersity, Umeå, SE-90187, Sweden
emails: {ozaland, monowar}@cs.umu.se
[2]KTH Royal Institute of Technology, Stockholm, SE-10044, Sweden
emails: {shutong, fpokorny}@kth.se

***Abstract*—Federated learning (FL) enables collaborative learning without data centralization but introduces significant communication costs due to multiple communication rounds between clients and the server. One-shot federated learning (OSFL) addresses this by forming a global model with a single communication round, often relying on the server's model distillation or auxiliary dataset generation - mostly through pre-trained diffusion models (DMs). Existing DM-assisted OSFL methods, however, typically employ classifier-guided DMs, which require training auxiliary classifier models at each client, introducing additional computation overhead. This work introduces OSCAR (One-Shot Federated Learning with Classifier-Free Diffusion Models), a novel OSFL approach that eliminates the need for auxiliary models. OSCAR uses foundation models to devise category-specific data representations at each client which are integrated into a classifier-free diffusion model pipeline for server-side data generation. In our experiments, OSCAR outperforms the state-of-the-art on four benchmark datasets while reducing the communication load by at least 99%[1].**

***Index Terms*—Federated Learning, One-Shot Federated Learning, Diffusion Model, Foundation Model**

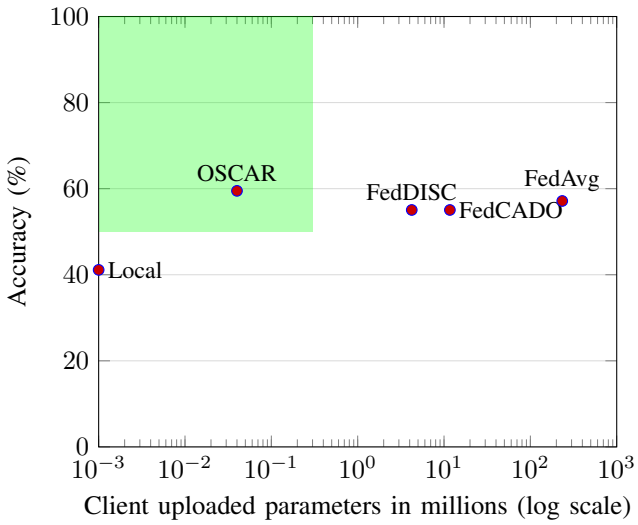

Fig. 1. Uploaded parameters by each client and accuracy for various algorithms on the OpenImage dataset and ResNet-18.

I. Introduction

Federated Learning (FL) [1] is a decentralized machine learning (ML) training methodology that enables multiple clients to collaboratively train a global model without moving the data to a central location, thus addressing concerns about data privacy and ownership in the age of growing data privacy regulations. This has led to the application of FL to various domains, including autonomous vehicles [2], the Internet of Things (IoT) [3], and healthcare [4]. However, since the participating clients usually own data that is not independent and identically distributed (non-IID) data, the task of training an *optimal* global model typically requires multiple communication rounds between the clients and the server, causing high communication overhead [5]. Several strategies, including client selection [6], update compression [7], and update dropping [8], have been proposed to reduce the communication load in each communication round. However, these strategies still require client synchronization and suffer under non-IID data distribution across clients, as local client models can drift from the global model at each communication round [9].

One-shot federated learning (OSFL) [10] offers an alternative, where the global model is learned through a single communication round between the clients and the server. OSFL can also reduce the impact of heterogeneous data distribution, as the global model is not directly formed from the local models. The single communication round can further benefit in scenarios where FL suffers from client dropout or stragglers (i.e., clients that communicate slowly) [11]. Existing OSFL approaches [12]–[14] rely on auxiliary dataset generation or knowledge distillation to form the global model. Knowledge distillation approaches usually require an auxiliary public dataset as the knowledge transfer medium [13]. On the other hand, dataset generation methods employ pre-trained generative models (e.g., diffusion models) to generate *new* data for training the global model. DMs are pre-trained with vast amounts

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation via the WASP NEST project "Intelligent Cloud Robotics for Real-Time Manipulation at Scale." The computations and data handling essential to our research were enabled by the supercomputing resource Berzelius provided by the National Supercomputer Centre at Linköping University and the gracious support of the Knut and Alice Wallenberg Foundation.

[1]https://github.com/obaidullahzaland/oscar

[*]Equal contributions.

of data, and *with proper guidance*, these DMs can generate realistic images that resemble a desired distribution. DMs can have an immense impact on FL, as new data resembling the clients' distribution can be generated without access to the raw dataset of the participating clients.

Current studies incorporating diffusion models in OSFL [11], [14]–[16] utilize classifier-guided DMs. Employing classifier-guided DMs requires auxiliary classifier training at each client, introducing computational and communication burdens. Furthermore, in some cases, the diffusion model needs to be downloaded to the clients [14]. Classifier-free DMs [17] solve these challenges by integrating the conditioning directly into the model, which is also adopted by most of the prevalent image generative models [18], [19]. Foundation models (FMs) [20], [21] can be employed for the encoding generation that can be used as conditioning without training or fine-tuning. Replacing the classifier models with encodings significantly reduces the client upload size compared to classifier-guided DM-based OSFL approaches, as shown in *Fig* 1. The seamless integration of pre-trained FMs and DMs in OSFL simplifies the overall framework, reduces communication load, and enhances scalability and efficiency across heterogeneous client datasets.

In this work, we present **OSCAR**, **O**ne-**S**hot federated learning with **C**l**A**ssifier-F**R**ee diffusion models. OSCAR leverages the strengths of FMs and a classifier-free diffusion model to train a global FL model in a single communication round between the clients and the server. OSCAR relies on each participating party's category-specific encodings to generate data through classifier-free DMs. The generated data is then used to train the global model on the server. By removing the need for classifier training at each client, OSCAR reduces the client upload size by 99% compared to current state-of-the-art (SOTA) DM-assisted OSFL approaches. In addition to reducing the communication overhead at each client, OSCAR outperforms existing SOTA on four different benchmarking datasets.

II. Prior Work

A. One-Shot Federated Learning

Existing OSFL approaches can be divided into two categories based on their methodology. The first category utilizes knowledge distillation to learn a global model through either data distillation [13] or model distillation [22]. In distilled one-shot federated learning (DOSFL [13], the clients share distilled synthetic data with the server, which is utilized for global model training. FedKT [22] utilizes a public auxiliary dataset and student-teacher models trained by clients to learn a global student model. The second category of methods uses auxiliary data generation at the server based on intermediary information shared by the clients. DENSE [12] trains a generator model on local classifiers, later used to generate auxiliary data for global model training. In FedCVAE [23], the server aggregates the decoder part of the conditional variational encoders (CVAE) trained at each client and generates auxiliary data for the global model. FedDiff [15] aggregates locally trained diffusion models to form a global diffusion model for data generation. FedCADO [11] utilizes classifiers trained at each client to generate data for global model training via classifier-guided pre-trained diffusion models (DMs). FedDISC [24] utilizes data features for data generation via pre-trained DMs.

B. Federated Learning with Foundation Models

The emergence of foundation models (FMs), both large language models (LLMs) [21] and vision language models (VLMs) [20], has impacted the landscape of machine learning. The application of these FMs, however, has not yet been fully explored in FL. Yu et al., [25], and Charles et al., [26] explore training FMs in FL setups. PromptFL [27] investigates prompt learning for FMs under data scarcity in FL settings. FedDAT [28] proposes a federated fine-tuning approach for multi-modal FMs. FedPCL [29] integrates FMs into the traditional FL process to act as class-wise prototype extractors. While FMs have the potential to mitigate data heterogeneity and communication load in FL, their full potential has not been utilized in FL settings.

III. Preliminaries

A. Diffusion Models

Denoising Diffusion Probabilistic Models (DDPMs) [30] employ a U-Net architecture [31], denoted as ϵ_θ, to model data distribution $x \sim q(x_0)$. For any given timestamp $t \in \{0, \ldots, T\}$, during the *forward process*, Gaussian noise $\mathbf{I}$ is progressively added according to:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1},\ \beta_t\mathbf{I}), \tag{1}$$

with β_t as a learned variance scheduler. In the *reverse process*, x_0 is sampled from:

$$p_\theta(x_{t-1}|x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t),\ \Sigma_\theta(x_t,\ t)), \tag{2}$$

where $\mu_\theta(x_t, t)$ is derived from $\epsilon_\theta(x_t, t)$, and $\Sigma_\theta(x_t, t)$ is a time-dependent constant.

When a conditioning signal y, such as text, is added, the network is trained to minimize:

$$\begin{aligned} \mathcal{L}_t(\theta) = \mathbb{E}_{z_0 \sim q(\mathcal{E}(x_0)),\ \epsilon \sim \mathcal{N}(0,\ \mathbf{I}),\ t}\left[\|\epsilon - \hat{\epsilon}_t\|^2\right], \\ \hat{\epsilon}_t = \epsilon_\theta(x_t,\ t,\ y), \end{aligned} \tag{3}$$

where $\hat{\epsilon}_t$ provides an estimate of the score function used for data generation during the reverse process, and $\mathcal{E}(x_0)$ denotes the encoded representation of the original input x_0.

Classifier-Guided Models [32] generate conditional samples by combining the diffusion model's score estimate with the input gradient of a classifier's log probability:

$$\hat{\epsilon}_t = \epsilon_\theta(x_t, t, y) - s\sigma_t \nabla_{x_t} \log p(y|x_t), \tag{4}$$

where $p(y \mid x_t)$ is the probability of class y with respect to the input x_t, obtained from the classifier. σ_t denotes the noise scale at timestep t, while s controls the influence of the classifier's guidance on the classifier's output.

Classifier-Free Models [17], in contrast, combine the score estimate from a conditional diffusion model with that of a

jointly trained unconditional model, guiding the generation by their difference:

$$\hat{\epsilon}_t = (1+s)\epsilon_\theta(x_t, t, y) - s\epsilon_\theta(x_t, t, \emptyset), \tag{5}$$

where $\epsilon_\theta(x_t, t, \emptyset)$ represents prediction without conditioning.

In conclusion, classifier-guided models rely on a trained classifier to guide the generative process by predicting the likelihood that the generated output matches a given text description. Classifier-free Models integrate text conditioning directly into the generative process, eliminating the need for an external classifier. This approach has driven recent trends in training text-driven generative models, such as DALL·E [33] and Stable Diffusion [19].

IV. OSCAR: One Shot Federated Learning with Classifier-Free Diffusion Models

The traditional FL setup consists of a central server and a set of clients $\mathcal{R}$, where each client $r \in \mathcal{R}$ trains a local model $\mathbf{w}_r$ on its local dataset $\mathcal{D}_r$ in each iteration and communicates it to the server. The server is responsible for forming a global model $\mathbf{w}$ from the local client models through an aggregation function like federated averaging (FedAvg). Considering the intrinsic data heterogeneity in FL, traditional FL algorithms require multiple communication rounds between the clients and the server to form a global model, leading to significant communication overhead.

To reduce the communication load in the FL setup, we propose OSCAR, a novel one-shot federate learning (OSFL) approach. OSCAR integrates foundation and pre-trained classifier-free generative models, specifically Stable Diffusion [19], and learns a global model from the clients' category-specific representations. OSCAR facilitates global model learning within a single communication round under a non-IID data distribution among the clients. As illustrated in *Fig.* 2, the OSCAR pipeline can be divided into four steps: **(1)** generating descriptions of the client's client-specific data, **(2)** encoding features from the generated descriptions, and transmitting them to the server **(3)** transmitting the category-specific representations to the server, and **(4)** generating data on the server to facilitate final model training.

a) Description Generation and Text Encoding: Unlike existing DM-assisted OSFL approaches, OSCAR eliminates classifier training and utilizes the clients' category-specific data features as conditioning for the DM. Each client follows a two-step approach to generate category-specific encodings. First, the client uses a VLM, specifically BLIP [34], to generate textual descriptions for all their images. Then, the client uses a text encoder, specifically CLIP [35], denoted as $\text{CLIP}_{\text{Text}}$, to generate category-specific text encodings from the textual descriptions, as shown in *Eq.* 6. A classifier-free diffusion model can utilize the CLIP encodings directly as text conditioning to generate new data.

$$\begin{aligned} y_{cn} &= \text{CLIP}_{\text{Text}}\left(\text{BLIP}(x_{cn})\right), \\ &\text{for } c = 1, \ldots, C \text{ and } n = 1, \ldots, N \end{aligned} \tag{6}$$

where C represents the number of categories for the client, N is the number of category-specific images at the party, and y_{cn} denotes the encoded text corresponding to the input x_{cn}.

b) Client Representation and Server Data Synthesis: Each client averages the category-specific encodings to form a unified representation for the specific category. Despite its simplicity, averaging the category-specific text encodings aligns well with the classifier-free approach:

$$\bar{y}_c = \frac{1}{N}\sum_{n=1}^{N} y_{cn} \tag{7}$$

The averaged feature $\bar{y}_c$ for each category c is then sent directly to the server to initiate classifier-free sampling:

$$\hat{\epsilon}_T(\bar{y}_c) = (1+s)\epsilon_\theta(x_T, T, \bar{y}_c) - s\epsilon_\theta(x_T, T, \emptyset) \tag{8}$$

where the guidance scale s is fixed at 7.5. In this process, x_T (with T set to 50) is randomly sampled from the noise space, and the subsequent sampling is carried out according to:

$$x_{T-1} = \frac{1}{\sqrt{\alpha_T}}\left(\sqrt{\alpha_T}x_T - \hat{\epsilon}_T(\bar{y}_c)\right) + \sigma_T\mathcal{N}(0, \mathbf{I}) \tag{9}$$

Starting from random noise x_T, the model iteratively refines the image through the sequence $x_T \to x_{T-1} \to \cdots \to x_0$, guided by the category-specific encoding $\bar{y}_c$, to produce the final output image x_0 within the client's distribution. In our comparisons to SOTA models, replacing the classifier model with the clients' category-specific representations has the advantage of reducing each client's upload size by at least 99The server generates **ten** images for each category-specific client representation to form a global synthesized dataset $\mathcal{D}_{syn}$, with $10\times|\mathcal{R}|\times C$ new images, where $|\mathcal{R}|$ is the number of clients and C is the number of categories. As the global dataset $\mathcal{D}_{syn}$ is constructed based on the category-specific representations from individual clients, it effectively captures the heterogeneous data distribution present at each client.

c) Model Training: After generating the global synthesized dataset $\mathcal{D}_{syn}$, the server trains a centralized model $\mathbf{w}$, specifically a ResNet-18 classifier, on the synthesized dataset. This approach not only reduces the dependency on client synchronization and availability at each training and communication round but also ensures that the model can generalize well to non-IID data across clients. The server communicates the global model $\mathbf{w}$ to all clients after training, to be later used for inference locally.

V. Experimental Setup

a) Datasets:

- **NICO++** [36]: NICO++ contains images of size 224×224 from 60 different categories, across six domains. The dataset has two settings. In *Common NICO++*, all the categories share the same six domains: [autumn, dim, grass, outdoor, rock, and water]. On the other hand, the *Unique NICO++* contains different domains for each category.

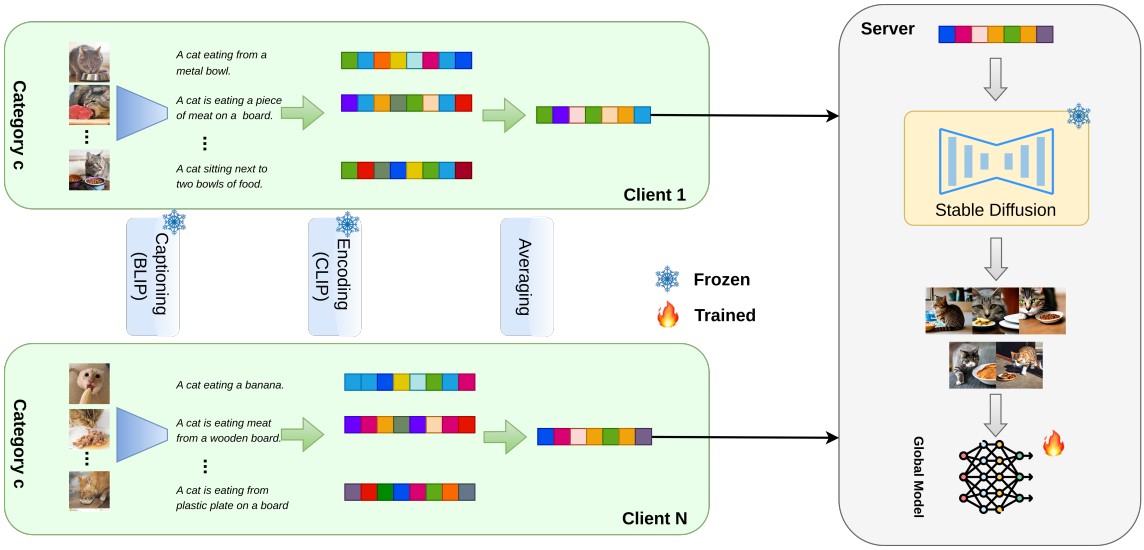

Fig. 2. An illustration of the proposed OSCAR pipeline, where BLIP [34], CLIP [35] Text Encoder, and Stable Diffusion [19] are all used with frozen weights and in a zero-shot manner.

- **DomainNet** [37]: The DomainNet dataset contains images of 345 categories over six domains. This work uses a subset of the DomainNet dataset with 90 categories.
- **OpenImage** [38]: OpenImage is a multi-task dataset with over 1.7 million images across 600 categories. This work uses a subset of 120 categories from the dataset following the pre-processing in [11].

b) Data Division: The data has been divided among all parties in a non-IID manner. Specifically, the data is **feature distribution skewed**, as each client owns data about a single domain from each category in the NICO++ and DomainNet datasets. For Openimage, the classes have been divided into six similar subgroups, where each client owns a single category from each subgroup. The number of clients is fixed to **six**, aligning with the number of domains in all the datasets.

c) Baselines: Local learning, federated learning, and state-of-the-art DM-assisted OSFL approaches have been considered as baselines for comparison against OSCAR. Each client trains a local standalone model on its data in local training. FedAvg [1], FedProx [39], and FedDyn [40] consider traditional FL setups with minor variations in the local objective and aggregation functions. FedDISC [24] and FedCADO [11] are DM-assisted OSFL approaches and train auxiliary models for image generation at the server.

VI. Results and Analysis

a) Main Results: In this section, experimental results are provided to compare OSCAR against baselines. The main comparisons are carried out on the four benchmarking datasets. In local and traditional FL settings, original images are used for training client models, while in FedCADO, FedDISC, and OSCAR, the global models are trained with synthetic data. The test set images, however, are the actual dataset test images in all the experiments. We have compared OSCAR against two SOTA DM-assisted approaches, FedCADO [11] and FedDISC [24], alongside traditional FL algorithms and local training. The experiments consider accuracy as the performance measure, calculated as the number of images classified correctly by the trained model divided by the number of images in the test set. Specifically, we only consider *top-1* accuracy. OSCAR performs better than the baselines on all the considered datasets, as shown in *Table*. I. Aside from superior *average* accuracy on the overall test set, OSCAR also performs better than the SOTA on domain-specific test sets. As we have assigned each domain to a single client, we consider the domain-specific test sets as client-specific test sets. All the experiments were carried out with the ResNet-18 classifier network, and the number of images per category for each client was set to 30.

Like other DM-assisted OSFL approaches, OSCAR performs better on datasets that consist of real images (i.e., NICO++ and OpenImage). The difference is evident in the two domains (sketch and clipart) corresponding to client2 and client3 in the DomaiNnet dataset. While OSCAR performs lower than average in these domains, FL algorithms struggle similarly.

b) Classifier Networks: To facilitate direct comparison with existing baseline approaches, the classifier network in the main results reported is a ResNet-18. However, the synthesized data appears to have potential that more advanced backbones can utilize. The results for NICO++ Unique and NICO++ Common datasets with different classifier networks are reported in *Table* II. The results indicate that the generated data can potentially improve the global model's optimality with an improved model architecture and may even improve more as the number of images per category increases. In the ResNet family, ResNet-101 performs the best, while the base version of the vision transformer (ViT B-16) has the best overall performance.

c) Number of Generated Images: This section examines the impact of the generated dataset size on the global model performance in OSCAR. Traditionally, the increase in dataset

TABLE I
ACCURACY (IN %) ON THE TEST SET FOR THE BASELINES AND OSCAR ON FOUR BENCHMARKING DATASETS. THE BEST RESULTS ARE IN BOLD.

Model	Client Test Set Accuracy							Model	Client Test Set Accuracy						
	client1	client2	client3	client4	client5	client6	avg		client1	client2	client3	client4	client5	client6	avg
	DomainNet								**OpenImage**						
Local	22.22	8.54	7.67	28.95	19.16	16.10	17.64	Local	37.72	39.95	49.01	47.41	49.20	41.13	43.97
FedAvg	35.27	11.99	5.68	36.99	22.97	22.33	21.88	FedAvg	51.84	**52.63**	62.70	58.53	63.08	54.86	57.14
FedProx	42.10	11.73	6.29	42.61	27.53	25.60	25.33	FedProx	54.08	51.30	**63.96**	60.53	63.11	51.19	57.20
FedDyn	37.62	13.92	6.71	40.21	26.09	23.87	23.24	FedDyn	51.60	49.08	62.75	56.07	59.55	53.06	55.22
FedCADO	57.31	17.51	9.43	44.25	**38.74**	38.44	34.28	FedCADO	51.66	48.99	62.41	55.59	58.86	52.80	55.05
FedDISC	56.19	14.84	8.35	43.89	38.38	36.82	33.07	FedDISC	49.65	47.42	54.73	53.41	60.74	52.81	53.12
OSCAR	**66.95**	**23.25**	**10.02**	**44.54**	34.14	**38.97**	**37.60**	OSCAR	**55.42**	51.14	63.42	**61.12**	**68.55**	**58.11**	**59.49**
	NICO++ Common								**NICO++ Unique**						
Local	54.10	53.95	42.49	56.68	53.86	46.14	51.29	Local	49.19	54.77	56.48	50.62	56.06	56.13	53.89
FedAvg	58.57	55.36	44.60	58.63	55.90	50.27	54.17	FedAvg	69.16	71.34	74.22	67.58	**79.59**	77.14	73.15
FedProx	58.63	52.12	44.96	58.12	54.68	50.43	53.66	FedProx	69.48	71.75	74.43	67.68	78.73	76.61	73.09
FedDyn	62.13	56.62	48.08	61.76	57.61	51.60	56.67	FedDyn	66.60	72.46	74.84	66.84	77.66	78.62	72.83
FedCADO	49.21	58.13	54.63	54.75	54.64	47.03	53.06	FedCADO	75.13	70.31	73.60	68.88	73.30	72.51	72.28
FedDISC	51.43	**59.45**	**56.17**	56.82	52.32	45.64	53.64	FedDISC	74.32	71.25	**75.28**	66.79	73.47	70.06	71.86
OSCAR	**59.11**	59.32	52.96	**64.04**	**62.18**	**51.70**	**58.19**	OSCAR	**75.95**	**71.32**	75.13	**70.14**	75.00	**73.93**	**73.62**

TABLE II
ACCURACY (IN %) ON THE TEST SET FOR OSCAR WITH DIFFERENT BACKBONE NETWORKS AT THE SERVER.

Model	Client Test Set Accuracy						
	client1	client2	client3	client4	client5	client6	avg
	NICO++ Unique						
ResNet-18	65.42	71.14	74.02	68.52	71.00	70.69	70.15
VGG-16	75.53	69.14	71.96	67.13	74.71	73.51	72.06
ResNet-50	80.72	73.50	76.40	77.34	79.61	78.52	77.73
ResNet-101	80.61	75.03	79.05	76.48	80.49	81.86	78.97
DenseNet-121	80.51	77.10	76.93	75.94	79.61	78.73	78.17
VIT B-16	84.53	77.86	83.49	79.91	81.08	82.48	81.58
	NICO++ Common						
ResNet-18	57.46	58.55	60.12	61.83	60.05	50.85	56.43
VGG-16	60.31	62.45	51.68	63.11	62.04	54.39	58.95
ResNet-50	62.19	65.70	54.44	67.07	67.78	55.19	61.76
ResNet-101	64.04	69.82	58.01	69.01	69.32	57.46	64.16
DenseNet-121	60.28	65.86	55.00	64.93	66.61	55.53	60.93
VIT B-16	65.19	70.97	58.17	71.86	70.85	58.60	65.60

TABLE III
IMPACT OF SAMPLE COUNT PER CATEGORY ON OSCAR.

Samples	Client Test Set Accuracy						
	client1	client2	client3	client4	client5	client6	avg
	NICO++ Unique						
10	65.42	71.14	74.02	68.52	71.00	70.69	70.15
20	67.62	71.14	74.95	68.94	74.11	70.90	71.19
30	75.95	71.32	75.13	70.14	75.00	73.93	73.62
40	69.26	72.56	75.26	70.72	73.47	72.70	72.34
50	68.52	71.44	74.95	70.72	73.36	70.69	71.62
	NICO++ Common						
10	57.46	58.55	60.12	61.83	60.05	50.85	56.43
20	58.03	57.78	49.52	62.42	62.13	52.17	57.06
30	59.11	59.32	52.96	64.04	62.18	51.70	58.19
40	58.57	59.10	61.68	61.90	61.14	52.17	57.36
50	59.41	57.07	52.08	62.26	61.42	53.38	57.93

size impacts performance positively. In this case, while the initial increase in synthesized dataset size boosts the model performance, the performance remains constant, or in some cases decreases, after a certain threshold. This may indicate that data synthesized by diffusion models may best be utilized as auxiliary data for training OSFL approaches rather than as a replacement. *Table* III shows the result for OSCAR with varying number of samples synthesized per category of each client.

TABLE IV
TOTAL NUMBER OF PARAMETERS (IN MILLIONS) UPLOADED BY EACH CLIENT.

Model	Local	FedAvg	FedCADO	FedDISC	OSCAR
Parameters	-	234	11.69	4.23	0.03

d) Communication Analysis: As OSCAR only uploads the data encodings, it uploads the least parameters from each client. In our experiments, OSCAR uploads less than 1% of the number of parameters compared to the evaluated SOTA models. OSCAR achieves this by eliminating the classifier training, and hence classifier uploading, and each client only communicates 512 parameters for each category. *Table* IV shows the number of parameters each client uploads. FedCADO trains a classifier model; hence, each client uploads a model with 11.69 million parameters. While FedDISC reduces the communication size by more than 60% compared to FedCADO at each client, the upload size from each client is still more than 100 times higher than OSCAR.

VII. CONCLUSION

In this work, we propose OSCAR, a novel one-shot federated learning approach that utilizes pre-trained vision language models and classifier-free diffusion models (DMs) to train a global model in a single communication round in FL settings. OSCAR eliminates the need for training a classifier model at each client by replacing the classifier-guided DM with a classifier-free DM in the image synthesis phase. OSCAR generates category-specific data representations for each client through BLIP and CLIP foundation models, which are communicated to the server. The server generates *new* data samples and trains a global model on the generated *data*. In our experiments, OSCAR reduces the communication load by reducing the client upload size more than 100X compared to state-of-the-art DM-assisted

OSFL approaches while exhibiting superior performance on four benchmarking datasets.

In future work, we want to extend OSCAR to fuse the knowledge of auxiliary generated datasets with existing learned knowledge at each client.

REFERENCES

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[2] J. Xue, D. Yuan, Y. Sun, T. Zhang, W. Xu, H. Zhou *et al.*, "Spatial-temporal attention model for traffic state estimation with sparse internet of vehicles," *arXiv preprint arXiv:2407.08047*, 2024.

[3] N. Wang, W. Yang, X. Wang, L. Wu, Z. Guan, X. Du, and M. Guizani, "A blockchain based privacy-preserving federated learning scheme for internet of vehicles," *Digital Communications and Networks*, 2022.

[4] S. Banerjee, R. Misra, M. Prasad, E. Elmroth, and M. H. Bhuyan, "Multi-diseases classification from chest-x-ray: A federated deep learning approach," in *AI 2020: Advances in Artificial Intelligence: 33rd Australasian Joint Conference, AI 2020, Canberra, ACT, Australia, November 29–30, 2020, Proceedings 33*. Springer, 2020, pp. 3–15.

[5] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[6] P. Singhal, S. R. Pandey, and P. Popovski, "Greedy shapley client selection for communication-efficient federated learning," *IEEE Networking Letters*, 2024.

[7] G. Lan, H. Wang, J. Anderson, C. Brinton, and V. Aggarwal, "Improved communication efficiency in federated natural policy gradient via admm-based gradient updates," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023, pp. 59 873–59 885.

[8] H. Zhou, M. Li, P. Sun, B. Guo, and Z. Yu, "Accelerating federated learning via parameter selection and pre-synchronization in mobile edge-cloud networks," *IEEE Transactions on Mobile Computing*, 2024.

[9] Y. Shi, Y. Zhang, Y. Xiao, and L. Niu, "Optimization strategies for client drift in federated learning: A review," *Procedia Computer Science*, vol. 214, pp. 1168–1173, 2022.

[10] N. Guha, A. Talwalkar, and V. Smith, "One-shot federated learning," *arXiv preprint arXiv:1902.11175*, 2019.

[11] M. Yang, S. Su, B. Li, and X. Xue, "One-shot federated learning with classifier-guided diffusion models," *arXiv preprint arXiv:2311.08870*, 2023.

[12] J. Zhang, C. Chen, B. Li, L. Lyu, S. Wu, S. Ding, C. Shen, and C. Wu, "Dense: Data-free one-shot federated learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 21 414–21 428, 2022.

[13] Y. Zhou, G. Pu, X. Ma, X. Li, and D. Wu, "Distilled one-shot federated learning," *arXiv preprint arXiv:2009.07999*, 2020.

[14] M. Yang, S. Su, B. Li, and X. Xue, "Feddeo: Description-enhanced one-shot federated learning with diffusion models," in *Proceedings of the 32nd ACM International Conference on Multimedia*, ser. MM '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 6666–6675.

[15] M. Mendieta, G. Sun, and C. Chen, "Navigating heterogeneity and privacy in one-shot federated learning with diffusion models," *arXiv preprint arXiv:2405.01494*, 2024.

[16] J. Zhang, X. Qi, and B. Zhao, "Federated generative learning with foundation models," *arXiv preprint arXiv:2306.16064*, 2023.

[17] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[18] J. Betker, G. Goh, L. Jing, T. Brooks, J. Wang, L. Li, L. Ouyang, J. Zhuang, J. Lee, Y. Guo *et al.*, "Improving image generation with better captions," 2023.

[19] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022.

[20] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.

[21] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[22] Q. Li, B. He, and D. Song, "Practical one-shot federated learning for cross-silo setting," *arXiv preprint arXiv:2010.01017*, 2020.

[23] C. E. Heinbaugh, E. Luz-Ricca, and H. Shao, "Data-free one-shot federated learning under very high statistical heterogeneity," in *The Eleventh International Conference on Learning Representations*, 2023.

[24] M. Yang, S. Su, B. Li, and X. Xue, "Exploring one-shot semi-supervised federated learning with pre-trained diffusion models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, 2024, pp. 16 325–16 333.

[25] S. Yu, J. P. Munoz, and A. Jannesari, "Federated foundation models: Privacy-preserving and collaborative learning for large models," in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024, pp. 7174–7184.

[26] Z. Charles, N. Mitchell, K. Pillutla, M. Reneer, and Z. Garrett, "Towards federated foundation models: Scalable dataset pipelines for group-structured learning," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[27] T. Guo, S. Guo, J. Wang, X. Tang, and W. Xu, "Promptfl: Let federated participants cooperatively learn prompts instead of models-federated learning in age of foundation model," *IEEE Transactions on Mobile Computing*, 2023.

[28] H. Chen, Y. Zhang, D. Krompass, J. Gu, and V. Tresp, "Feddat: An approach for foundation model finetuning in multi-modal heterogeneous federated learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 10, pp. 11 285–11 293, Mar. 2024.

[29] Y. Tan, G. Long, J. Ma, L. Liu, T. Zhou, and J. Jiang, "Federated learning from pre-trained models: A contrastive learning approach," *Advances in neural information processing systems*, vol. 35, pp. 19 332–19 344, 2022.

[30] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, 2020.

[31] O. Ronneberger, P. Fischer, and T. 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*, 2015.

[32] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[33] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[34] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *International conference on machine learning*. PMLR, 2022, pp. 12 888–12 900.

[35] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[36] X. Zhang, Y. He, R. Xu, H. Yu, Z. Shen, and P. Cui, "Nico++: Towards better benchmarking for domain generalization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 16 036–16 047.

[37] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1406–1415.

[38] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov *et al.*, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," *International journal of computer vision*, vol. 128, no. 7, pp. 1956–1981, 2020.

[39] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proceedings of Machine Learning and Systems*, vol. 2, 2020, pp. 429–450.

[40] D. A. E. Acar, Y. Zhao, R. Matas, M. Mattina, P. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," in *International Conference on Learning Representations*, 2021.