FedSM: Semantic-Guided Feature Mixup for Bias Reduction in Federated Learning with Long-Tail Data

Jingrui Zhang, Yimeng Xu, Shujie Li, Feng Liang* *Member, IEEE*, Haihan Duan *Member, IEEE*, Yanjie Dong *Member, IEEE*, Victor C.M. Leung *Life Fellow, IEEE*, and Xiping Hu* *Senior Member, IEEE*

***Abstract*—Federated Learning (FL) has emerged as a promising paradigm for decentralized machine learning, where a central server coordinates distributed clients to collaboratively train a global model without direct access to raw data. Despite its advantages, heterogeneous and long-tail data distributions across clients remain a major bottleneck, particularly in IoT scenarios with diverse devices and sensing modalities. To address these challenges, we propose FedSM, a novel framework that integrates multimodal semantic knowledge with balanced pseudo features to enhance global model optimization. Unlike conventional approaches that rely on single-modal information, FedSM leverages CLIP's cross-modal representations and open-vocabulary priors to guide semantic-aware data augmentation. A probabilistic selection mechanism further refines local features by mixing them with global prototypes, ensuring pseudo features are semantically reliable and reducing bias caused by skewed client distributions. Almost all computations are performed locally at the client side, thereby alleviating server overhead and improving scalability in resource-constrained IoT environments. Extensive experiments on long-tail benchmarks including CIFAR-10-LT, CIFAR-100-LT, and ImageNet-LT demonstrate the superiority of FedSM over state-of-the-art baselines, highlighting its potential for robust communication-efficient FL in IoT networks.**

***Index Terms*—Federated Learning, Internet of Things, Long-tail Distribution, Semantic-Guided Data Augmentation.**

I. Introduction

REAL-world data distributions universally exhibit long-tail characteristics [1]. Oriented towards the physical world, data in Internet of Things (IoT) systems are inherently highly imbalanced [2]. Furthermore, due to significant discrepancies in deployment environments and functional capabilities of edge devices, different clients often capture only partial and limited class distributions, or even completely lack specific tail categories. For instance, in industrial anomaly detection scenarios, sensors predominantly collect data representing normal operational states (head classes), while critical fault conditions (tail classes) are extremely rare. This leads to severe non-IID data imbalance. The compounded effects of long-tail data distributions and non-IID characteristics pose a fundamental challenge for federated learning (FL) in IoT.

Under this interplay, standard FL algorithms (e.g., FedAvg [3]) are prone to specific failure modes: **1) Local overfitting**: Local models tend to be dominated by head classes within their respective clients [4], [5], leading to biased decision boundaries and insufficient discriminative capability for tail classes. **2) Aggregation bias**: The dominance of head classes exerts a cumulative and propagating effect during federated aggregation [6]. Head classes, being prevalent across multiple clients, dominate the gradients and continuously accumulate advantages. Conversely, the key representations of tail samples are ignored, forcing the decision boundary to skew severely towards head classes. These factors ultimately result in severe client drift, making it difficult for the global model to converge to a stable solution that balances both majority and minority classes.

While existing approaches attempt to address these challenges, they exhibit distinct limitations when applied to IoT scenarios. First, re-weighting methods [4] are often ineffective when specific categories are entirely absent from local clients, as they rely on existing samples to adjust weights. Second, client selection [7] strategies rely on prior knowledge of local label distributions, which constitutes a potential privacy leakage risk. Third, server-side synthesis methods (e.g., CReFF [8], CLIP2FL [9]) typically depend on uploading gradients. This imposes a heavy computational and storage burden on the server, which can impede the system's ability to scale to a large number of clients.

Consequently, there is a pressing need to address long-tail distributions in FL with a solution that can *correct long-tail and absent-category biases*, *preserve privacy*, and *perform locally in clients to increase scalability*. Feature augmentation strategies, such as Mixup [10], offer a potential path by interpolating samples to smooth decision boundaries efficiently. However, standard Mixup may fail in the fragmented view of FL. As shown in Fig. 1, blindly applying Mixup in a non-IID setting can be detrimental. For example, images of squirrels often share background elements (like pine trees or mountains) with train images (Fig. 1(a)). Mixing semantically unrelated samples (e.g., squirrel and train) generates misleading noise, limiting the classifier's ability to refine decision boundaries.

*Feng Liang and Xiping Hu are corresponding authors.

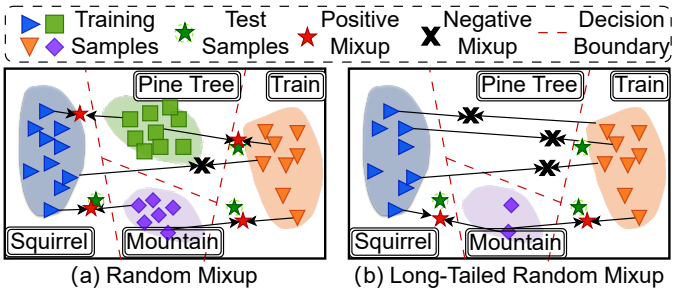
Jingrui Zhang and Yimeng Xu are with the School of Computer Science & Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: jingrui1119@bit.edu.cn; xuym@bit.edu.cn;).

Shujie Li is with the Department of Electrical and Electronic Engineering (HKU EEE), Hong Kong University, Pokfulam, Hong Kong (e-mail: shujie.li@connect.hku.hk.

Feng Liang, Haihan Duan, Yanjie Dong, and Xiping Hu are with the Artificial Intelligence Research Institute, Shenzhen MSU-BIT University, Shenzhen 518172, China (e-mail: fliang, duanhaihan, yanjiedong, huxp@smbu.edu.cn).

Victor C.M. Leung is with the Artificial Intelligence Research Institute, Shenzhen MSU-BIT University, Shenzhen 518172, China, and also with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC V6T 1Z4, Canada.

The source code and training scripts are available for research purposes at https://github.com/DistriAI/FedSM.



(a) Random Mixup (b) Long-Tailed Random Mixup

Fig. 1: Illustration of Mixup problems. (a) Random Mixup ignores semantic relevance between categories and may blend unrelated samples, such as squirrel and train, across boundaries, producing synthetic data that misguides boundary refinement. (b) When the mountain category has significantly fewer samples or the pine tree category is absent, random Mixup has a higher chance of generating unrepresentative or even misleading synthetic samples.

This is exacerbated in IoT settings (Fig. 1(b)) where clients may lack the semantic context of absent classes, leading to distorted feature representations [11].

To address these challenges, we propose *FedSM*, a novel *S*emantic relevance-guided *M*ixup framework tailored for FL in IoT environments. As illustrated in Fig. 2, the simplified pipeline of FedSM depicts major stages in clients, as well as the information exchange between clients and the server. Unlike prior works that rely on heavy server-side interventions, FedSM empowers clients to locally rectify classifier bias using lightweight operations. Specifically, we leverage a pre-trained image-text-aligned model (e.g., CLIP [12]) to introduce external semantic knowledge. By computing the semantic relevance between categories, FedSM guides the generation of pseudo-features in the feature space. These pseudo-features are synthesized by mixing local features with global prototypes, which are generated from class prototypes from clients. Crucially, this enables clients to synthesize representative features for locally absent classes by leveraging global prototypes. Each client re-trains its classifier using the synthesized features to mitigate bias propagation during model aggregation with the server.

Our main contributions are summarized as follows:

- We propose FedSM, a client-centric framework designed for the resource-constrained IoT context. It mitigates classifier bias locally without requiring raw data sharing or heavy server-side generative modeling.
- We introduce a semantics-guided mixup strategy that utilizes cross-modal priors (e.g., CLIP) and probabilistic pairing. This ensures that augmented features preserve semantic relevance, even when local data is severely skewed or missing categories.
- Extensive experiments on CIFAR-10-LT, CIFAR-100-LT, and ImageNet-LT demonstrate that FedSM consistently outperforms state-of-the-art methods in accuracy while maintaining superior communication efficiency suitable for IoT networks.

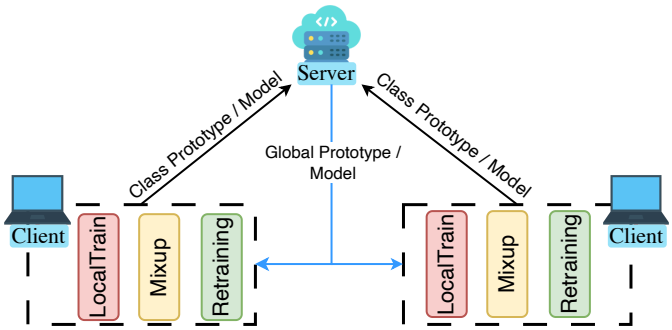


Fig. 2: Illustration of major stages and the information flow between clients and the server in FedSM, where only class-wise prototypes and model parameters are exchanged during federated training.

II. Related Work

Long-Tail Learning: Two primary strategies dominate long-tail learning: *re-weighting* and *decoupled retraining*. Re-weighting assigns varying weights to samples based on category frequency, increasing the emphasis on tail classes to counterbalance head-class dominance. For example, Cui *et al.* [4] proposed an exponential weighting method to redistribute importance across categories. Similarly, AREA [5] recalibrates classifier updates by estimating the effective area in the feature space. Decoupled retraining, in contrast, separates feature learning and classifier learning. Kang *et al.* [13] introduced a two-stage training pipeline, feature learning and classifier learning, to learn balanced classifiers. BBN [14] further evolved this into a dual-branch architecture with shared parameters, one branch for standard training and the other for classifier refinement. While effective, these approaches are designed for centralized settings and do not directly translate to FL, where decentralized data introduces additional challenges.

Federated Learning with Heterogeneous Data: Most existing works address client-level heterogeneity in FL but often assume class distributions are uniform, overlooking global class imbalance. Solutions typically fall into two categories: server-side methods that mitigate the impact of heterogeneity [15], and methods that preserve consistency between local and global models [16]–[18]. For example, CCVR [19] retrains classifiers using virtual features sampled from a Gaussian Mixture Model to address heterogeneity, though its performance deteriorates under long-tail distributions. Other methods focus on client selection for data complementarity [20], [21], often requiring revealing local data distribution, undermining FL's privacy guarantees. Our method is applicable to global class heterogeneity and requires retraining the aggregated classifier with locally augmented data.

Federated Learning with Long-Tail Data: When long-tail data is distributed across clients, local models often develop severely biased representations due to data heterogeneity. RUCR [22] employs a Mixup-inspired strategy [10] to generate pseudo-features. CReFF [8] and CLIP2FL [9] necessitate uploading averaged gradients of local data to the server, which then synthesizes balanced pseudo-features for classifier retraining. However, relying on gradient transmission rises

the risk of privacy leakage. Specifically, regarding semantic guidance, CLIP2FL optimizes pseudo-features on the server side using a contrastive loss aligned with text features. This iterative optimization process on the server imposes a significant computational burden, thereby impeding scalability for many clients. Unlike these methods, FedSM adopts a prototype-based approach. It avoids server-side gradient sharing and heavy computations by leveraging semantic guidance locally, ensuring robust, scalable, and privacy-preserving in long-tail FL scenarios.

III. Method

A. Problem Setting

We assume a standard FL setup with K clients holding non-IID, long-tail data. The goal is to train a shared feature extractor and classifier that generalizes well despite client drift and label imbalance. Let $\mathcal{D}^k$ denote the local dataset on client k, with size $n^k = |\mathcal{D}^k|$. The global dataset is defined as $\mathcal{D} = \bigcup_{k=1}^{K} \mathcal{D}^k$ and consists of C classes. For class c, let $\mathcal{D}_c^k = \{(x, y) \in \mathcal{D}^k \mid y = c\}$ be the subset of samples distributed to client k, and $n_c^k = |\mathcal{D}_c^k|$ its size. The total number of samples in class c across all clients is $N_c = \sum_{k=1}^{K} n_c^k$. The global data follows a long-tail distribution, i.e., when sorted by class frequency such that $N_1 \geq N_2 \geq \cdots \geq N_C$, we have $N_1 \gg N_C$.

The standard FL process involves: 1) The server broadcasts the global model to clients; 2) Clients update local models using private data; 3) Locally updated models are sent back to the server for aggregation; This cycle repeats until convergence. Our objective is to learn a high-performance global model for image classification under the constraint of long-tail distributed data in the FL setting.

B. Overview of FedSM

FedSM follows the standard federated learning process: 1) The server distributes the global model to each client; 2) Clients update their local models using private data; 3) Clients send the updated models back, and the server aggregates them. These steps repeat until convergence. This study primarily focuses on the client-side training with three phases: a) local training, b) label semantics relevance-guided feature mixup, and c) classifier retraining, as shown in Fig. 3.

First, in the local training phase, we employ knowledge distillation between the CLIP image encoder and our local feature extractor. This step aims at regularizing the local feature space learned by different clients reside within a shared and comparable semantic embedding space. By transferring the teacher's generalized visual representations, we prevent the local model from overfitting to the sparse tail classes. This ensures that the local features serve as the viable inputs for the subsequent mixup process.

Next, in the semantic relevance-guided mixup phase, we aim to generate balanced pseudo-features to compensate for distributional gaps within each client. We use a image-text-aligned model (e.g., the text tower in CLIP) to compute label text semantic relevance, based on which we probabilistically select sample pairs for mixup with global prototypes. This

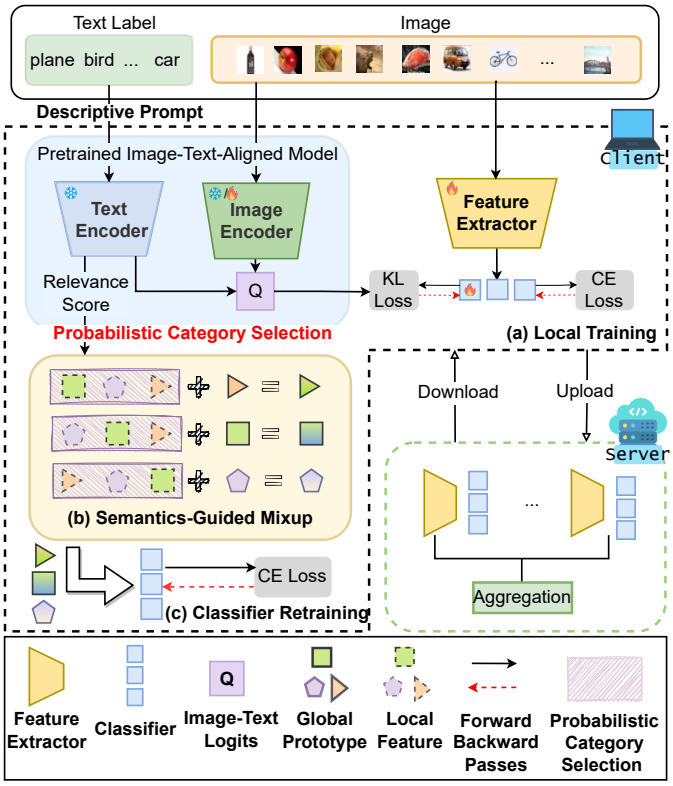


Fig. 3: Overview of the FedSM framework. The client side consists of three key phases: a) local training, b) label relevance-guided feature mixup, and c) classifier retraining.

probabilistic selection, rather than a deterministic top-1 selection, helps alleviate relevance estimate bias and increase robustness. This relevance-guided mixup ensures that generated pseudo-features remain semantically consistent and do not overlap with unrelated classes.

Finally, the classifier is retrained using the synthesized balanced pseudo-features to correct the decision boundaries, mitigating the head-class bias before aggregation. Motivated by prior work [8] showing that classification bias mainly stems from the classifier rather than the feature extractor, we retrain only the classifier after several local training rounds to correct these biases. The server side remains unchanged during retraining, executing standard FL procedures without any additional modifications.

The causal chain for solving the long-tail problem in FL by these phases is summarized as follows. Knowledge distillation provides the foundation for a semantically consistent feature space. The probabilistic semantic relevance estimation enhances cross-domain robustness, while retraining with mixed-up features based on global prototypes directly addresses classifier bias.

C. Local Training

In the local training phase, our goal is to enhance the model's representation capability of aligning image features with text semantics by transferring knowledge from a pre-trained image-text-aligned model. To this end, we adopt a knowledge distillation strategy within a teacher–student

framework, where the image-text-aligned model serves as the teacher, guiding the local model (student) during learning. This image-text-aligned model is required to have strong semantic understanding of both visual and textual modalities with two encoders: an image encoder Enc_I and a text encoder Enc_T. Given an input image x and its corresponding text label l, we compute the visual feature as $h_v = Linear(Enc_I(x)) \in \mathbb{R}^d$ and the text feature as $h_t = Linear'(Enc_T(l)) \in \mathbb{R}^d$, where d is the feature dimension. The image-text-aligned model output logits are calculated as:

$$q = [\langle h_v, h_{t_1} \rangle, \langle h_v, h_{t_2} \rangle, \ldots, \langle h_v, h_{t_C} \rangle],$$

where $\langle \cdot, \cdot \rangle$ denotes cosine similarity between visual and textual features across all C categories. In client k, The local model prediction $p = w^k(x)$ is obtained by forwarding x through the local model w^k. The total training loss combines supervised and distillation objectives:

$$L = L_{\text{CE}}(y, p) + L_{\text{KL}}(q \parallel p) \tag{1}$$

where y is the category label of p, L_{CE} is the cross-entropy loss, and L_{KL} denotes the Kullback–Leibler divergence [23].

After local updates in the t-th round, clients in a randomly selected subset U^t upload their models w^k to the server. Following standard FL aggregation, the server computes the updated global model as a weighted average of client models, given by:

$$w = \sum_{k \in U^t} \frac{|\mathcal{D}^k|}{\sum_{k \in U^t} |\mathcal{D}^k|} w^k. \tag{2}$$

D. Image Feature Mixup Guided by Label Relevance

Feature mixup. Sample-level augmentation techniques such as MixUp [10] and CutMix [24] are simple yet effective for mitigating long-tail distributions. However, these methods originally operate at the pixel level and do not exploit higher-level feature-space mixing, limiting their applicability in FL, where decentralized data and privacy constraints make raw image sharing impractical. To overcome this limitation, FedSM performs mixup in the feature space, leveraging both global category prototypes and local features. This approach maintains a global perspective to reduce bias while preserving client-specific characteristics and adhering to FL privacy principles.

The global prototype for category c is the aggregation of local category prototypes from all clients, which is defined as:

$$z_c^{\text{global}} = \frac{1}{N_c} \sum_{k=1}^{K} f_c^k \cdot n_c^k, \quad f_c^k = \frac{1}{n_c^k} \sum_{i=1}^{n_c^k} g(x_{c,i}^k), \tag{3}$$

where $g(\cdot)$ is the local feature extractor, $x_{c,i}^k$ is the i-th sample of category c on client k, and f_c^k is the client-level category prototype uploaded to the server.

A pseudo feature r_c^k for category c on client k is generated by mixing the global prototype of category c with a local feature from the most semantically relevant category v:

$$r_c^k = (1 - \lambda) \cdot z_v^k + \lambda \cdot z_c^{\text{global}}, \tag{4}$$

where z_v^k is a local feature of category v, and λ is a mixup coefficient that balances the importance of generalization (global prototype) and personalization (local feature).

From a broader perspective, λ serves as a trade-off coefficient for different long-tail scenarios. When λ is small, the pseudo-features generated via mixup are dominated by potentially biased local features, hindering their ability to effectively correct the decision boundary. Whereas, a larger λ ensures that global prototypes occupy a dominant position in the pseudo-feature generation process, thereby providing more stable guidance for the model's discriminative direction. In scenarios characterized by highly long-tail distributions, appropriately increasing λ can strengthen the reliance on global prototypes, enabling them to serve as stable feature anchors. In contrast, in scenarios when local data is relatively sufficient and long-tail distributions are moderate, a lower λ can preserve more local feature information, thereby avoiding the issue of feature homogeneity caused by excessive reliance on global prototypes.

Category relevance estimation. This step selects the most semantically relevant category v for a target category c. Unlike prior methods that rely on co-occurrence or feature similarity, FedSM leverages label semantics via a pretrained image-text-aligned model.

Specifically, FedSM uses the model's text encoder to estimate the similarity between categories based solely on their textual labels. Each label l_i is converted into a descriptive phrase_i (e.g., "a photo of {label}"). The semantic relevance score $\alpha_{i,j}$ between categories i and j is computed as:

$$\alpha_{i,j} = Nonlinear(\langle Enc_T(\text{phrase}_i), Enc_T(\text{phrase}_j) \rangle), \tag{5}$$

where $\langle \cdot, \cdot \rangle$ denotes similarity between encoded text features, default to cosine, and $Nonlinear$ refers to an optional transformation (e.g., softmax).

The resulting relevance score $\alpha_{i,j}$ is interpreted as the probability of selecting a local feature z_j^k from category j in Eq. 4, ensuring semantic consistency in pair selection. In FedSM, each client ranks its available categories based on relevance scores and assigns higher selection probabilities to more relevant categories, promoting semantically meaningful mixup and generating balanced pseudo data.

This probabilistic strategy offers two key advantages: 1) It mitigates domain shift between the pretrained model and the downstream FL task by introducing controlled randomness, reducing over-reliance on potentially misaligned semantic priors. 2) It enhances mixup diversity and robustness by allowing feature synthesis from a broader pool of relevant categories, especially beneficial when the top-matching categories are absent from a client's local dataset.

Moreover, applying a nonlinear transformation to the similarity scores allows fine-grained control over the distribution sharpness, amplifying confidence in top choices or smoothing across multiple candidates, further improving flexibility and stability in pseudo feature generation.

Each client generates S semantics-guided pseudo-features per category. Let $r_{c,i}^k$ denote the i-th pseudo feature for

Algorithm 1 FedSM Training at Communication Round t

Input: Global model $w^t = \{f^t, g^t\}$
Output: Updated global model $w^{t+1} = \{f^{t+1}, g^{t+1}\}$
Server-side:
Randomly sample a set of online clients U^t
Send global model w^t to all $k \in U^t$
Client-side (for each $k \in U^t$):
Update local model using Eq. 1
/ generate pseudo-features for classifier retraining */*
if $t \geq$ `total_rounds` $-$ `retraining_rounds` **then**
  Compute category relevance via Eq. 5
  Obtain global prototypes via Eq. 3
  Generate pseudo feature set via Eq. 4
  Retrain classifier using pseudo-features
  Set local model $w_k^{t+1} = \{f_k^{t+1}, g_k^{t+1}\}$
end if
Send updated local model w_k^{t+1} to server
Server-side:
Aggregate received models via Eq. 2

category c on client k, then the complete pseudo feature set on client k is defined as:

$$\mathcal{R}^k = \{r_{c,i}^k \mid c \in C,\, i = 1, \ldots, S\}. \tag{6}$$

Classifier Retraining. After local training, client k refines its classifier g^k using the generated pseudo feature set $\mathcal{R}^k$. This retraining step aims to mitigate classification bias and further enhance robustness to domain shift by leveraging semantically enriched, balanced synthetic data. The loss function is the cross entropy loss:

$$L_{\text{CE}}(g^k; \mathcal{R}^k) = \frac{1}{|\mathcal{R}^k|} \sum_{(r,y) \in \mathcal{R}^k} -y \log\big(\sigma(g^k(r))\big), \tag{7}$$

where σ denotes the softmax function.

This lightweight retraining phase is performed only on the classifier g^k, making it computationally efficient while improving model generalization. The final model $w^k = \{f^k, g^k\}$, consisting of the locally updated feature extractor and the calibrated classifier, is then uploaded to the server for aggregation. Unlike prior methods [8], [9] that retrain local models at every FL communication round, FedSM performs classifier retraining only in the final few rounds, significantly reducing computational overhead.

Overall, the global model is iteratively updated through client-side local training and classifier retraining with semantics-guided mixed-up features, as outlined in Algorithm 1.

E. Discussion

We discuss FedSM's characteristics in the following aspects: privacy risk, computational cost, and the scenario gap in the long-tail and feature drift problems.

Privacy Risk. FedSM is practically privacy-preserving. FedSM's prototypes contain only aggregated abstract representations, carrying significantly lower privacy leakage risks compared to gradient-based methods. Although there is a theoretical possibility of decoding or reconstructing feature prototypes, such attacks typically require the attacker to possess the complete feature encoder architecture and simultaneously train a decoder model, which poses a high barrier in practical settings.

Computational Cost. FedSM exhibits low server-side computational cost, which is advantageous in large-scale scenarios. In addition to the necessary model aggregation in common FL frameworks, FedSM only performs the aggregation and broadcast of class-wise prototypes on the server. This aggregation operation is only on simple vectors of class prototypes, rendering its computational load negligible. Furthermore, classifier retraining is performed at the client side, avoiding additional computational burdens on the server. As a comparison, CLIP2FL requires iterative gradient optimization of pseudo-features on the server, involving contrastive learning losses related to text semantics, which incurs a high centralized computational cost, especially for a large scale of clients. The client-centric design of FedSM can effectively distribute this pseudo-feature generation workload across clients while preserving greater privacy.

Long-tail and Feature Drift. Long-tail distribution (label skew) and feature drift represent two distinct yet interconnected dimensions of data heterogeneity. Long-tail distribution emphasizes that different clients possess varying class proportions. For example, in IoT environments, data representing normal operations constitutes the vast majority, while faults or accidents (tail classes) are minorities. Whereas, feature drift highlights that the representation of the same category varies across different clients, such as variations in sensor types, camera viewpoints, or lighting conditions. While both frequently occur in IoT scenarios, they present distinct challenges. This paper primarily focuses on the long-tail distribution issue. However, because FedSM leverages pre-trained models to guide high-level semantic encoding, the encoded features tend to be more robust across diverse devices and conditions. By encouraging local models to align with a shared semantic space, FedSM may mitigate the effects of device-related feature perturbations in feature drift scenarios.

IV. Evaluation

A. Experimental Setup and Implementation Details

Datasets. We evaluate FedSM on three long-tail benchmarks: CIFAR-10-LT [8], CIFAR-100-LT [8], and ImageNet-LT [29]. CIFAR-10-LT and CIFAR-100-LT are derived from CIFAR-10 and CIFAR-100 [30], respectively, by sampling with varying imbalance factors (IF): 100, 50, and 10. An imbalance factor of 100 means the most frequent class has 100 times more samples than the least frequent one. ImageNet-LT is a long-tail subset of ImageNet [31], containing 115.8K images across 1000 categories. It has a predefined distribution with up to 1280 images in head classes and as few as five in tail classes. To simulate non-IID data across clients, we adopt the Dirichlet distribution, which allows us to simulate varying degrees of non-IID scenarios by controlling a concentration parameter α. This setup enables a more comprehensive

TABLE I: Top-1 accuracy(%) of different FL algorithms on the CIFAR-10-LT and CIFAR-100-LT datasets. "I", "II", and "III" represent types of heterogeneity-oriented, imbalance-oriented, and heterogeneity and imbalance-oriented, respectively.

Type	Method	CIFAR-10-LT			CIFAR-100-LT		
		IF=100	IF=50	IF=10	IF=100	IF=50	IF=10
I	FedAvg [3]	57.3 ± 1.7	61.0 ± 3.6	72.0 ± 3.6	31.6 ± 0.7	35.9 ± 0.3	47.6 ± 0.8
	FedAvgM [25]	56.7 ± 1.6	61.2 ± 4.0	71.9 ± 4.0	31.7 ± 0.7	36.3 ± 0.5	47.3 ± 0.9
	FedProx [26]	54.4 ± 2.2	60.4 ± 2.5	69.8 ± 2.9	30.4 ± 0.4	34.3 ± 0.4	43.9 ± 0.4
	FedNova [27]	56.5 ± 1.6	61.0 ± 4.4	72.6 ± 5.1	31.6 ± 1.0	36.1 ± 0.3	47.5 ± 0.6
	CCVR [19]	60.4 ± 2.2	68.2 ± 2.0	74.4 ± 2.3	25.1 ± 0.9	27.1 ± 2.0	36.0 ± 1.0
	MOON [28]	57.5 ± 1.1	61.6 ± 3.6	73.0 ± 3.2	31.9 ± 0.9	36.1 ± 0.3	47.5 ± 0.8
II	Fed-Focal [6]	52.9 ± 1.9	58.1 ± 2.6	74.9 ± 5.5	30.3 ± 0.7	34.6 ± 0.9	41.4 ± 0.8
	RatioLoss [7]	56.0 ± 2.2	65.0 ± 2.7	72.8 ± 5.4	31.7 ± 0.9	34.7 ± 0.9	42.6 ± 1.1
III	CReFF [8]	69.9 ± 1.2	72.6 ± 1.1	79.6 ± 1.5	26.9 ± 0.7	30.3 ± 0.6	37.8 ± 1.0
	RUCR [22]	61.3 ± 0.8	65.1 ± 3.4	79.3 ± 1.2	33.7 ± 0.1	37.4 ± 0.0	48.8 ± 0.2
	CLIP2FL [9]	71.2 ± 0.8	72.6 ± 1.8	80.7 ± 1.7	36.0 ± 0.7	39.6 ± 0.6	47.2 ± 0.5
	FedSM+MetaCLIP (Ours)	70.4 ± 0.7	71.6 ± 0.9	80.9 ± 1.1	35.6 ± 0.7	39.5 ± 0.5	50.2 ± 0.8
	FedSM+CLIP (Ours)	**72.2 ± 0.9**	**74.4 ± 1.0**	**82.2 ± 0.5**	**37.8 ± 0.5**	**41.2 ± 0.4**	**50.7 ± 0.7**

evaluation of our method's robustness for long-tail datasets under different non-IID scenarios and is consistent with related SOTA studies (e.g., CReFF [8], CLIP2FL [9]), ensuring a fair and rigorous comparison. For datasets CIFAR-10-LT and CIFAR-100-LT, we use $\alpha = 0.5$, following CReFF [8]. For ImageNet-LT, we use $\alpha = 0.1$ to introduce higher data heterogeneity among clients.

Implementation and Setup. For CIFAR-10-LT and CIFAR-100-LT, we use ResNet-8 [32] as the feature extractor, and for the larger ImageNet-LT dataset, we adopt ResNet-50 [32]. We use CLIP [12] or MetaCLIP [33] as the image-text-aligned model. These models have been pretrained on rich image and text data from diverse domains and can be used to verify FedSM's performance under domain shifts. To align with the image-text-aligned model, a projection layer is added atop the base model to match the feature dimension. Both its text and image encoders are frozen during training. For the CLIP image encoder, we use the ViT-B/32 variant, consistent with the setup in CLIP2FL [9]. CLIP is the default choice for other experiments if the image-text-aligned model is not specifically mentioned. FedSM and other baseline methods are implemented within the FLGO framework [34], [35] relying on PyTorch. Each experiment is repeated five times with different random seeds for CIFAR-10-LT and CIFAR-100-LT, and three times for ImageNet-LT. All experiments are run on a single node equipped with four NVIDIA A800 GPUs.

Training. By default, we simulate 20 clients, with 40% randomly selected for participation in each communication round. The classifier is retrained using 100 pseudo-features per class, following the common practice in recent works [8], [9]. We use the standard cross-entropy loss and run totally 200 communication rounds with 10 epochs per round. The baseline methods [8], [9] retrain local models at every round, while FedSM performs classifier retraining with pseudo-features only in the final 50 rounds. All experiments use Stochastic Gradient Descent (SGD) with a learning rate of 0.1 for local training and 0.01 for classifier retraining. The mixup coefficient λ in Eq. 3 is chosen randomly from range 0.65 to 0.90 and batch size is 32 across all datasets. Rather than fixing λ to a deterministic value, this stochastic perturbation from a value range not only enhances the diversity of synthetic features but also prevents the model from overfitting to a single interpolation ratio, thereby contributing to improved generalization capabilities.

B. Results

We compare FedSM against a range of FL algorithms that address data heterogeneity at varying levels. General approaches [3], [19], [25]–[28] target standard heterogeneous settings, while others [6], [7] specifically focus on class imbalance. The most relevant to our work are recent state-of-the-art (SOTA) methods [8], [9], [22] designed for FL with long-tail data.

a) Results on CIFAR-10/100-LT.: Table I reports the classification accuracy of various FL algorithms on CIFAR-10-LT and CIFAR-100-LT. FedSM with CLIP consistently outperforms all baselines across different IFs, with performance improvement ranging from 1.0 to 1.9 percentage points compared to second best results. Performance gains on CIFAR-100-LT are generally slightly higher than on CIFAR-10-LT. A possible reason is that CIFAR-100-LT has finer-grained labels, which enhances the effect of semantic guidance for feature mixup in FedSM. When CLIP is replaced by MetaCLIP, representing a specific domain shift, the results remain close to those obtained with CLIP and are competitive with other baseline results. This demonstrates FedSM's robustness to domain shift between the pretrained model and training data.

b) Results on ImageNet-LT.: For fine-grained analysis, we divide categories of the full ImageNet-LT dataset into three groups based on samples amounts: *many* (>100 samples), *medium* (20–100 samples), and *few* (<20 samples). Table II shows the results of the *overall* dataset along with divided groups. Despite the substantial imbalance in ImageNet-LT, FedSM with CLIP and MetaCLIP achieves the overall accuracy of 30.9% and 29.3%, an improvement of 3.4 and 1.8 percentage points compared to the previous SOTA (27.5%). Even with fewer retraining rounds, our method matches or surpasses others, particularly on tail classes (*Few*) with the

TABLE II: Top-1 accuracy(%) of different federated learning algorithms on the ImageNet-LT.

Type	Method	Overall	Divided Categories		
			Many	Medium	Few
Heterogeneity-oriented	FedAvg [3]	23.0 ± 2.0	34.9 ± 1.2	19.1 ± 1.0	7.0 ± 1.3
	FedAvgM [25]	22.5 ± 2.2	33.9 ± 1.4	18.7 ± 1.4	6.0 ± 1.2
	FedProx [26]	22.9 ± 1.6	35.0 ± 1.8	17.1 ± 1.2	7.0 ± 0.9
	FedNova [27]	24.7 ± 2.0	35.4 ± 0.8	20.6 ± 1.6	11.6 ± 0.5
	CCVR [19]	25.7 ± 1.5	36.8 ± 1.5	20.6 ± 1.6	10.0 ± 0.9
	MOON [28]	24.1 ± 1.1	34.7 ± 0.5	20.4 ± 0.9	9.9 ± 1.2
Imbalance-oriented	Fed-Focal [6]	21.5± 1.8	31.0 ± 1.6	15.8 ± 1.6	6.8 ± 1.3
	RatioLoss [7]	25.0 ± 3.0	35.9 ± 2.3	18.9 ± 1.9	7.4 ± 1.4
Heterogeneity and Imbalanced	CReFF [8]	19.7 ± 1.5	34.8 ± 2.1	18.7 ± 1.8	8.3 ± 0.7
	CLIP2FL [9]	27.5 ± 1.0	35.7 ± 2.1	26.9 ± 1.8	**23.4 ± 1.4**
	FedSM+MetaCLIP (Ours)	29.3 ± 0.4	37.0 ± 0.6	**28.4 ± 1.5**	22.1 ± 1.3
	FedSM+CLIP (Ours)	**30.9 ± 0.2**	**38.0 ± 0.3**	27.4 ± 0.1	23.0 ± 0.2

TABLE III: Accuracy (%) of mixup strategies guided by probabilistic ($\mathcal{P}$, ours) semantic relevance, deterministic ($\mathcal{D}$) semantic relevance, and random category without using semantic relevance.

FedSM	CIFAR-10			CIFAR-100		
	$\mathcal{P}$	$\mathcal{D}$	w/o SR	$\mathcal{P}$	$\mathcal{D}$	w/o SR
λ =0.20	**71.4**	68.1	68.8	**36.0**	34.6	35.3
λ =0.35	**71.6**	68.3	68.9	**36.0**	34.8	35.3
λ =0.50	**71.7**	69.0	69.1	**36.7**	34.9	35.5
λ =0.65	**72.1**	70.0	70.3	**37.4**	35.3	35.9
λ =0.80	**72.2**	70.8	71.2	**38.0**	35.5	36.1

accuracy of 23.0%. Note that FedSM achieves this performance efficiently with classifier retraining only in the final 50 communication rounds (50 epochs each), while CLIP2FL requires gradient matching for 300 epochs in every round throughput the training (totally 200 rounds). These results highlight FedSM's computational efficiency and robustness under severely skewed data.

C. Ablation Study

Effect of Probabilistic Semantic Relevance. We explore the effectiveness of probabilistic semantic relevance-guided mixup. The synthesized pseudo-feature is a result of using the probabilistic selection of a relevant category j guided by $\alpha_{i,j}$ and the balance coefficient λ. We conduct experiments to explore the compound effect of varying λ across probabilistic and deterministic approaches. Table III presents a performance comparison under various λ settings across three strategies: mixup guided by probabilistic semantic relevance, deterministic semantic relevance, and randomly without semantic relevance guidance.

Results show that the probabilistic approach consistently outperforms the deterministic mechanism under various λ settings. It is noteworthy that the deterministic modeling approach performs even worse than the randomized approach in all cases. This can be attributed to the fact that the deterministic mechanism mixes features exclusively from a single selected class, resulting in rigid feature combinations. This lack of diversity limits its capacity to calibrate decision boundaries across classes. In contrast, the probabilistic approach generates diverse pseudo-features, introducing richer semantic perturbations. This facilitates the learning of more robust decision boundaries, ultimately yielding superior overall performance. Additionally, the performance consistently improves as λ increases across all approaches, validating the effectiveness of global prototypes in rectifying classifier bias.

TABLE IV: Accuracy (%) on CIFAR-100-LT with and without fine-tuning.

FedSM	IF=100	IF=50	IF=10
w/o fine-tuning	37.8	41.2	50.7
w/ fine-tuning	**38.4**	**42.4**	**52.0**

Effect of Fine-tuning. To investigate FedSM potential ability to mitigate the domain shift problem, we explore the effect of fine-tuning during local training. Instead of freezing the image encoder of CLIP, we optimize it by the loss of Margin Metric Softmax [36], which adds an adaptive margin for each negative feature pair between the image and text encoders. Since fine-tuning the teacher model during knowledge distillation based on logits between the teacher and student models can lead to unstable training, we further replace the Kullback–Leibler divergence between logits in Eq. 1 with the mean square error between features to optimize the feature extractor, following the practice suggested by FitNets [37]. Table IV shows the results on CIFAR-100-LT, a dataset with finer-grained categories and is more likely subject to problems caused by domain shift. After fine-tuning, FedSM delivered notably improved accuracy across different imbalance factors, e.g., an extra 1.3 percentage point improvement when IF=10. This fine-tuning helps quickly refine pretrained image feature spaces to align with out-of-domain training data, enhancing the results of data augmentation based on semantic relevance.

Effect of Distance Functions on Relevance Calculation. We evaluate the impact of different similarity distance functions used in relevance score calculation, as defined in Eq. 5. Experiments on CIFAR-10-LT are conducted under various IF settings, comparing four common distance functions: cosine similarity, L2 distance, L1 distance, and dot product. As shown

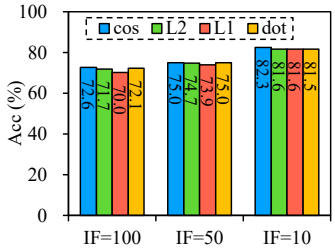
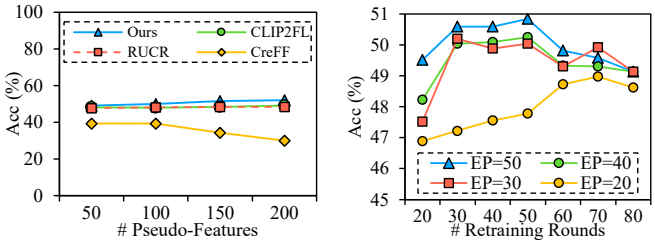


Fig. 4: Results on CIFAR-10-LT under different similarity functions for relevance score.



(a) Varying number of pseudo-features.

(b) Varying number of retraining epochs (EP). The y axis ranges between 45% and 51%.

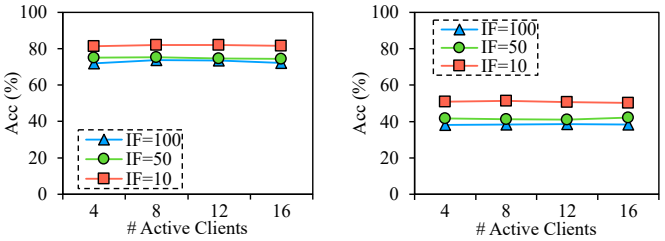
Fig. 5: Results in various classifier retraining settings on CIFAR-100-LT with IF=10.

in Fig. 4, cosine similarity yields the best performance across all IF levels, consistent with its widespread use in semantic similarity tasks. L1 and L2 distances yield lower accuracy, especially under high imbalance (IF=100), suggesting that they are less robust in capturing meaningful semantic relationships in sparse or skewed feature distributions. These results highlight the importance of selecting an appropriate similarity function to guide relevance-aware mixup in long-tail FL scenarios.

Effect of the Number of Pseudo-features. We evaluate FedSM's performance on CIFAR-100-LT with IF=10 when generating varying numbers of pseudo-features for classifier retraining, as shown in Fig. 5a. FedSM consistently benefits from increasing the number of pseudo-features, with each additional 50 samples yielding an approximate 1 percentage point improvement in accuracy. This gain is not solely due to quantity, but also to more mixup operations that encourage a more uniform and balanced feature distribution, helping to reduce classifier bias and refine decision boundaries.

Interestingly, CLIP2FL and RUCR also exhibit slight performance gains with more pseudo-features, albeit at a lower speed than FedSM. In contrast, CReFF shows declining accuracy as the number increases. A possible explanation is that CReFF relies on average gradient matching to optimize pseudo-features, which may yield lower-quality samples. Additionally, increasing the pseudo feature count in CReFF likely exacerbates the optimization burden, hindering effective classifier retraining.

Effect of the Number of Classifier Retraining Epochs. We examine the impact of varying classifier retraining epochs



(a) CIFAR-10-LT (b) CIFAR-100-LT

Fig. 6: Impact of varying the number of active clients.

TABLE V: Accuracy (%) under varying hyperparameter λ.

	CIFAR-10-LT			CIFAR-100-LT		
	IF=100	IF=50	IF=10	IF=100	IF=50	IF=10
λ=0.20	71.4	73.6	81.2	36.0	39.9	50.0
λ=0.35	71.6	74.2	81.5	36.0	39.9	**50.2**
λ=0.50	71.7	74.6	81.5	36.0	39.9	50.0
λ=0.65	72.1	75.0	81.8	37.4	41.1	49.1
λ=0.80	**72.2**	**75.1**	**81.9**	**38.0**	**41.5**	49.4

on CIFAR-100 with IF=10. As shown in Fig. 5b, FedSM achieves comparable performance using only 50 retraining rounds with 50 epochs each, limited to the final phase of training. In contrast, prior methods [8], [9] perform retraining in every communication round with 300 epochs, leading to significantly higher computational costs. This highlights the efficiency of our approach in reducing training overhead without sacrificing accuracy.

Effect of the Number of Active Clients. We evaluate FedSM's performance with different numbers of active clients, a key factor in FL. As shown in Fig. 6, FedSM demonstrates strong robustness to the number of active clients. Performance on CIFAR-10 exhibits slightly more fluctuation than on CIFAR-100, possibly due to less distinct label semantics in CIFAR-10. Across all settings, lower imbalance (i.e., smaller IF values) consistently yields higher accuracy, which aligns with trends observed in the main results.

Hyperparameter for Pseudo Feature Mixup. We study the effect of the mixup coefficient λ in Eq. 4, which controls the interpolation between the global prototype and local feature. As shown in Table V, FedSM performs robustly across a range of λ values from 0.20 to 0.80 with a gap of 0.15. This is because, in long-tail non-IID scenarios, local features z_v^k from tail classes are often sparse and noisy. Relying more on the global prototype z_c^{global} (i.e., a larger λ) ensures that the generated pseudo-features are closer to the true class centers, thereby providing more stable guidance for classifier boundary refinement.

V. Conclusion

In this paper, we propose FedSM, a semantics-guided mixup framework designed to alleviate classification bias in federated learning with long-tail and heterogeneous data distributions. By leveraging a pretrained image–text-aligned model, FedSM performs feature-level mixup between local features and global

prototypes, generating balanced pseudo-features that facilitate few-round classifier retraining. This semantic-aware design enables effective mitigation of domain shift while preserving privacy, as all procedures are executed locally on IoT devices without exposing raw data. Moreover, the lightweight client-side operations reduce server burden, making the framework highly suitable for large-scale IoT networks with constrained communication and computational resources. Extensive experiments validate that FedSM achieves superior accuracy, robustness against domain shifts, and efficiency compared to prior methods, highlighting its potential for enabling reliable and communication-efficient federated learning in IoT environments.

REFERENCES

[1] Z. Liu *et al.*, "Large-scale long-tailed recognition in an open world," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[2] H. Kopetz and W. Steiner, "Internet of things," in *Real-time systems: design principles for distributed embedded applications*. Springer, 2022, pp. 325–341.

[3] 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.

[4] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[5] X. Chen, Y. Zhou, D. Wu, C. Yang, B. Li, Q. Hu, and W. Wang, "Area: adaptive reweighting via effective area for long-tailed classification," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 277–19 287.

[6] D. Sarkar, A. Narang, and S. Rai, "Fed-focal loss for imbalanced data classification in federated learning," *arXiv preprint arXiv:2011.06283*, 2020.

[7] L. Wang, S. Xu, X. Wang, and Q. Zhu, "Addressing class imbalance in federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, 2021, pp. 10 165–10 173.

[8] X. Shang, Y. Lu, G. Huang, and H. Wang, "Federated learning on heterogeneous and long-tailed data via classifier re-training with federated features," *arXiv preprint arXiv:2204.13399*, 2022.

[9] J. Shi, S. Zheng, X. Yin, Y. Lu, Y. Xie, and Y. Qu, "Clip-guided federated learning on heterogeneity and long-tailed data," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 13, 2024, pp. 14 955–14 963.

[10] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018.

[11] D. Teney, J. Wang, and E. Abbasnejad, "Selective mixup helps with distribution shifts, but not (only) because of mixup," in *International Conference on Machine Learning*, 2024, pp. 47 948–47 964.

[12] 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.

[13] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis, "Decoupling representation and classifier for long-tailed recognition," *arXiv preprint arXiv:1910.09217*, 2019.

[14] B. Zhou, Q. Cui, X.-S. Wei, and Z.-M. Chen, "Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9719–9728.

[15] H.-Y. Chen and W.-L. Chao, "Fedbe: Making bayesian model ensemble applicable to federated learning," in *ICLR*, 2021. [Online]. Available: https://openreview.net/forum?id=dgtpE6gKjHn

[16] H. Huang, F. Shang, Y. Liu, and H. Liu, "Behavior mimics distribution: Combining individual and group behaviors for federated learning," in *IJCAI*, 2021, pp. 2556–2552.

[17] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *MLSys*, 2020, pp. 429–450. [Online]. Available: https://proceedings.mlsys.org/paper/2020/file/38af86134b65d0f10fe33d30dd76442e-Paper.pdf

[18] Y. Zhang, F. Liang, G. Yuan, M. Yang, C. Li, and X. Hu, "Fedpall: Prototype-based adversarial and collaborative learning for federated learning with feature drift," in *International Conference on Computer Vision*, 2025, pp. 1 – 10.

[19] M. Luo, F. Chen, D. Hu, Y. Zhang, J. Liang, and J. Feng, "No fear of heterogeneity: Classifier calibration for federated learning with non-iid data," *Advances in Neural Information Processing Systems*, vol. 34, pp. 5972–5984, 2021.

[20] M. Yang, X. Wang, H. Zhu, H. Wang, and H. Qian, "Federated learning with class imbalance reduction," in *2021 29th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 2174–2178.

[21] M. Duan, D. Liu, X. Chen, R. Liu, Y. Tan, and L. Liang, "Self-balancing federated learning with global imbalanced data in mobile systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 1, pp. 59–71, 2020.

[22] W. Huang, Y. Liu, M. Ye, J. Chen, and B. Du, "Federated learning with long-tailed data via representation unification and classifier rectification," *IEEE Transactions on Information Forensics and Security*, 2024.

[23] J. M. Joyce, "Kullback-leibler divergence," in *International encyclopedia of statistical science*. Springer, 2011, pp. 720–722.

[24] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6023–6032.

[25] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, 2019.

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

[27] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in neural information processing systems*, vol. 33, pp. 7611–7623, 2020.

[28] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 10 713–10 722.

[29] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Large-scale long-tailed recognition in an open world," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2537–2546.

[30] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[33] H. Xu, S. Xie, X. Tan, P.-Y. Huang, R. Howes, V. Sharma, S.-W. Li, G. Ghosh, L. Zettlemoyer, and C. Feichtenhofer, "Demystifying clip data," in *The Twelfth International Conference on Learning Representations*.

[34] Z. Wang, X. Fan, J. Qi, C. Wen, C. Wang, and R. Yu, "Federated learning with fair averaging," 2021.

[35] Z. Wang, X. Fan, Z. Peng, X. Li, Z. Yang, M. Feng, Z. Yang, X. Liu, and C. Wang, "Flgo: A fully customizable federated learning platform," 2023.

[36] Y. Shu, X. Guo, J. Wu, X. Wang, J. Wang, and M. Long, "CLIPood: Generalizing CLIP to out-of-distributions," in *Proceedings of the 40th International Conference on Machine Learning*, vol. 202. PMLR, 2023, pp. 31 716–31 731.

[37] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," 2015.

Jingrui Zhang received the B.S. degree in Information and Computational Science from Shandong University of Science and Technology, Shandong, China, in 2023. He is currently working toward the M.S. degree in Computer Technology with the School of Computer Science & Technology, Beijing Institute of Technology, Beijing, China. His research interests include federated learning, multi-agent collaboration and MLLM(Multimodal Large Language Model).

Yimeng Xu received the B.S. degree in Software Engineering from Hangzhou Dianzi University, Zhejiang, China, in 2024. He is currently working toward the M.S. degree in Computer Technology with the School of Computer Science & Technology, Beijing Institute of Technology, Beijing, China. His research interests include distributed training, multi-agent collaboration and automatic optimization.

Shujie Li , a first-year Ph.D. student at The University of Hong Kong (HKU). I received my M.Sc. degree from the University of Science and Technology of China (USTC). My research has been published in top-tier conferences and journals, including NeurIPS, TOIS, TACL, SIGIR, COLING, and CIKM. My current research interests focus on AI agents and large language models (LLMs) for AI for Science.

Feng Liang (Member, IEEE), is an associate professor in the Artificial Intelligence Research Institute of Shenzhen MSU-BIT University. He received his Ph.D. degree in computer science from the University of Hong Kong. His research interests are in broad areas of distributed intelligence, including distributed systems, distributed machine learning, and large-scale intelligence. He has published papers in prestigious venues including IEEE TPDS, ICCV, AAAI, HPDC, IPDPS, Information Sciences, EDBT, and ACSAC.

Haihan Duan (Member, IEEE) received his B.Eng. degree in Computer Science and Technology from East China Normal University, Shanghai, China, in 2017, and his M.Eng. degree in Software Engineering from Sichuan University, Chengdu, China, in 2020, and his Ph.D. degree in Computer and Information Engineering from The Chinese University of Hong Kong, Shenzhen, China, in 2023. He is currently an associate professor with Artificial Intelligence Research Institute, Shenzhen MSU-BIT University (SMBU), and also with Guangdong-Hong Kong-Macao Joint Laboratory for Emotion Intelligence and Pervasive Computing, Shenzhen, China. Before joining SMBU, he worked as a postdoctoral research fellow at University of Ottawa and Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), located in Abu Dhabi, United Arab Emirates. His research interests include multimedia, blockchain and Web3, metaverse, human-centered computing, and medical image analysis.

Yanjie Dong (Member, IEEE) received the MASc and PhDs degrees from the University of British Columbia, Canada, in 2020 and 2016, respectively. He is an associate professor and the assistant dean of Artificial Intelligence Research Institute, Shenzhen MSU-BIT University. His research interests include focus on the design and analysis of machine learning algorithms, machine learning based resource allocation algorithms, and quantum computing technologies.

Victor C.M. Leung (Life Fellow, IEEE) is currently with Artificial Intelligence Research Institute, Shenzhen MSU-BIT University, Shenzhen, China. He is also an Emeritus Professor of electrical and computer engineering and the Director of the Laboratory for Wireless Networks and Mobile Systems, The University of British Columbia (UBC), Canada. His research interests include wireless networks and mobile systems. He has published widely in these areas. He is a Fellow of the Royal Society of Canada, the Canadian Academy of Engineering, and the Engineering Institute of Canada. He received the 1977 APEBC Gold Medal, the 1977–1981 NSERC Postgraduate Scholarships, the IEEE Vancouver Section Centennial Award, the 2011 UBC Killam Research Prize, the 2017 Canadian Award for Telecommunications Research, the 2018 IEEE TCGCC Distinguished Technical Achievement Recognition Award, and the 2018 ACM MSWiM Reginald Fessenden Award. He has coauthored papers that won the 2017 IEEE ComSoc Fred W. Ellersick Prize, the 2017 IEEE Systems Journal Best Paper Award, the 2018 IEEE CSIM Best Journal Paper Award, and the 2019 IEEE TCGCC Best Journal Paper Award. He has been serving on the editorial boards of the IEEE Transactions on Green Communications and Networking, IEEE Transactions on Cloud Computing, IEEE Access, IEEE Network, and several other journals. He is named in the current Clarivate Analytics list of "Highly Cited Researchers." He is a fellow of the Royal Society of Canada (Academy of Science), Canadian Academy of Engineering, and Engineering Institute of Canada, and a life fellow of IEEE.

Xiping Hu (Member, IEEE) received the Ph.D. degree from the University of British Columbia, Vancouver, BC, Canada. He is currently a professor with Beijing Institute of Technology, and with Shenzhen MSU-BIT University, China. He has more than 150 papers published and presented in prestigious conferences and journals, such as IEEE TPAMI/TMC/TPDS/TIP/JSAC, IEEE COMST, ACM MobiCom/MM/SIGIR/WWW, AAAI, and IJCAI. He has been serving as associate editor of IEEE TCSS, and the lead guest editors of IEEE IoT Journal and IEEE TASE etc. He has been granted several key national research projects as principal investigator. He was the Co-Founder and CTO of Bravolol Ltd., Hong Kong, a leading language learning mobile application company with over 100 million users, and listed as the top 2 language education platform globally. His research areas consist of mobile cyber-physical systems, crowd sensing and affective computing.