Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.083715

ARTICLE

Federated Learning with Consistency Optimization Algorithms under Non-IID Data

Rui Wu[1], **Yehong Li**[2], **Hongjie Guo**[3,*], **Gangqiang Hu**[3], **Changjun Zhou**[3,*] **and Qile Zou**[4]

[1]School of Software Engineering, Jiangxi University of Science and Technology, Nanchang, China
[2]Jiangxi Provincial Key Laboratory of Multidimensional Intelligent Perception and Control, Jiangxi University of Science and Technology, Ganzhou, China
[3]School of Computer Science and Technology, Zhejiang Normal University, Jinhua, China
[4]Faculty of Science and Engineering, University of Nottingham Ningbo China, Ningbo, China

*Corresponding Authors: Hongjie Guo. Email: guohongjie@zjnu.edu.cn; Changjun Zhou. Email: zhouchangjun@zjnu.edu.cn

Received: 09 April 2026; Accepted: 03 July 2026; Published: 13 August 2026

ABSTRACT: Federated learning (FL) enables collaborative training of deep neural architectures while preserving data privacy, yet its performance often deteriorates in non-IID scenarios, which stems from client-side distribution drift and divergent local updates induced by pervasive data heterogeneity. This challenge is particularly critical for maintaining the structural consistency and generalization of neural models across diverse, distributed sources with significant distribution shifts. In this paper, we investigate how to effectively mitigate label distribution shift and feature distribution skew to enhance the global representation stability of neural architectures. We propose Federated Learning with Consistency Optimization Algorithms (FedCO), a novel optimization framework that incorporates a label-skew-aware correction loss and neural feature distribution regularization during local training. Specifically, our method aligns the internal statistics of architectural components between local and global models to suppress feature-space drift. Combined with an adaptive global aggregation mechanism guided by label entropy, this approach ensures that model updates from heterogeneous clients are consistently integrated into the global architectural parameters. Experimental results on multiple benchmarks demonstrate that FedCO significantly improves accuracy and convergence under diverse non-IID settings. For instance, on CIFAR-10 with extreme heterogeneity ($\alpha = 0.05$), FedCO achieves 73.64% accuracy, outperforming FedAvg by 7.32%; it also attains the highest accuracies on CIFAR-100 (62.63%) and TinyImageNet (37.82%) under the same setting. Our findings provide a robust strategy for integrating distribution-aware local training with adaptive structural aggregation, offering new insights into enhancing the reliability of distributed neural systems in real-world deployments. The code related to this algorithm is available at https://github.com/Donglin0730/FedCO.

KEYWORDS: Data heterogeneity; federated learning; federated optimization

1 Introduction

With the rapid development of machine learning, data privacy and security have become increasingly important concerns. Federated Learning, a novel distributed framework introduced by Konečnyý [1], is a decentralized training paradigm which enables collaborative model training across multiple clients without the need to centralise data on a single server, thus effectively safeguarding data privacy. While standard FL methods like FedAvg can ideally achieve global convergence, in practice, differences in local data distributions across clients—known as data heterogeneity—can lead to inconsistencies in local solutions [2].

This inconsistency may introduce non-zero Skew, resulting in local overfitting and causing the global model to degrade into a simple average of the clients' local models. This implies that data heterogeneity significantly impacts the convergence of the global model, as reflected by $\mathbf{x}^* \neq \frac{\sum_{i\in[m]} \mathbf{x}_i^*}{m}$ [3].

Data heterogeneity is primarily reflected in three aspects: feature Skew, label Skew, and quantity Skew. Feature Skew refers to inconsistencies in data feature distributions across clients, while label Skew indicates differences in label distributions. Quantity Skew refers to the imbalance in the number of data samples across clients [1]. This paper focuses on addressing the issues of feature Skew and label Skew. Severe label Skew can amplify client drift, further reducing the global model's convergence speed and generalization performance. As noted in earlier studies, heterogeneous data negatively impacts the effectiveness of FL [4]. Guo et al. [5] provided a theoretical analysis showing that the performance of the traditional FedAvg [1] is affected by the product of the length of local updates and the count of partial participants, along with an upper limit on the heterogeneity of gradient dispersion, which significantly influences the convergence rate. By increasing the local update interval and reducing participation rates, this disparity is significantly magnified. Next, we will explore how existing methods address data heterogeneity issues and propose a novel approach that combines local Skew correction with adaptive consistency-based aggregation to mitigate the impact of feature and label Skew on the global model. This method not only improves model convergence but also maintains superior global model performance in heterogeneous data environments.

Previous attempts to address this issue have generally focused on introducing proximal regularization terms in local client training [6,7] to control and reduce parameter differences between clients, thereby better handling data heterogeneity. Alternatively, some methods apply client model weighting during global aggregation by re-normalizing gradients. These methods neither integrate model parameter optimization with representation space nor effectively combine local training with global aggregation, making them less adaptable to addressing label and feature skew. For instance, approaches like FedProx [6], Scaffold [8], FedBN [9], FedOpt [10], and FedNova [11] are primarily aimed at mitigating accuracy degradation in non-IID settings. However, these methods have distinct scopes and limitations. **FedBN** only aligns batch normalization (BN) statistics across clients but does not correct label skew or feature drift beyond BN layers. **FedProx** adds a proximal term to penalize large deviations from the global model, yet it lacks explicit control over feature representation drift and cannot handle missing labels. **SCAFFOLD** uses control variates to correct gradient updates, addressing client drift at the gradient level, but it does not calibrate logits or align feature distributions directly. **FedLogitCal** performs logit calibration on the client side to balance classifier outputs, but it ignores feature-level inconsistencies and BN mismatch. To better understand why existing methods struggle under extreme heterogeneity, we analyze their failure scenarios: when data heterogeneity is very high (e.g., Dirichlet distribution parameter $\alpha = 0.05$), each client sees only a few classes (severe label skew) and the feature distribution under the same label varies drastically across clients (feature drift). Under these conditions, FedAvg suffers from severe client drift because local models overfit to their own label sets and feature distributions; simple averaging produces a global model that performs poorly on almost all clients. FedProx reduces drift by penalizing parameter differences, but the proximal term cannot prevent classifiers from being skewed toward locally present classes, resulting in low accuracy on classes unseen during local training (label missing problem). SCAFFOLD corrects gradient direction using control variates, which reduces the variance of updates, but it does not align feature representations; the global BN statistics become inconsistent, and the model loses discriminative power for rare classes. FedBN only exchanges and aggregates BN layers; while it mitigates feature drift to some extent, it cannot address label missing or label shift: if a client has never seen class "A", its classifier logit for "A" remains arbitrary and uncalibrated. FedLogitCal calibrates logits based on the observed label distribution, but it assumes that feature extractors across clients are already aligned, which is false under high feature drift; logit calibration

alone cannot correct misaligned features, leading to limited improvement. Consequently, these methods converge slowly (often requiring 2–3× more communication rounds to reach moderate accuracy) and yield low final accuracy (e.g., below 60% on CIFAR-10 with $\alpha = 0.05$). In contrast, our proposed FedCO jointly handles label missing, label shift, and feature drift through three synergistic components: a label-skew-aware loss with unbiased probabilities that explicitly regularizes predictions for missing classes, preventing the model from producing overconfident wrong predictions; a feature-statistics regularization that aligns local BN layers with the global model, directly counteracting feature drift without requiring additional data exchange; and an adaptive consistency aggregation method weighted by label entropy that reweights client contributions, reducing the influence of clients with highly skewed label distributions which are likely to provide inconsistent updates. These three components together address three distinct but intertwined problems—label missing, label shift, and feature drift—rather than a single issue, enabling fast and stable convergence even under $\alpha = 0.05$ heterogeneity.

From the perspective of neural architecture dynamics, data heterogeneity inherently leads to the collapse of shared representational spaces across distributed nodes. Even when a consistent structural backbone is utilized, the stochastic nature of local data distributions forces the neural layers—particularly the normalization and alignment components—to converge toward divergent manifolds. This phenomenon, often termed as structural representation drift, necessitates a more granular optimization strategy that goes beyond simple parameter averaging, focusing instead on the architectural consistency and the alignment of latent feature distributions. Beyond the above approaches, the FL community has also explored various orthogonal directions, including semi-supervised learning [12,13], personalization [14], causal structure learning [15], asynchronous bias mitigation [16], out-of-distribution generalization [17], incentive mechanism design [18], dynamical system modeling [19], and privacy preservation [20,21]. However, none of these methods simultaneously address the intertwined challenges of label missing, label shift, and feature drift under extreme data heterogeneity.

To further optimize FL in addressing client drift caused by data heterogeneity and ensure a consistent and stable global state, this paper simultaneously considers label and feature skew at the local client level, aiming to improve global optimization and generalization. We propose a novel algorithm, FedCO which notably integrates three key components to achieve state-of-the-art (SOTA) performance. (i) First, we employ a Label Skew-Aware loss (LASA) with un Skewed probabilities for unobserved structures during local training [22]. This allows the global model to adjust class predictions during local updates, thereby enhancing global knowledge. (ii) Second, to further mitigate feature drift, we introduce a regularization correction that aligns the local model's average and variance of samples with the parameters of the batch normalization layers in global model [22]. This reduces feature space discrepancies caused by data heterogeneity during training, improving the performance of the global model. (iii) Lastly, to further reduce the effects of heterogeneity, we present an adaptive aggregation method that calculates consistency through an un Skewed estimation of parameters. The consistency is then adjusted using label entropy, and the resulting optimized weights are sent from local clients to the server for global aggregation. We conducted a theoretical analysis of the proposed method and extensively validated it using three public datasets under three non-IID scenarios. The results demonstrate its significant potential in handling tasks involving heterogeneous data. In summary, our contributions are as follows:

- This paper introduces a novel federated algorithm, FedCO, designed for non-IID settings, with the dual objective of optimizing global consistency and generalization while maintaining fast convergence and high adaptability.
- This paper introduces a loss function based on un Skewed probability and label-skew awareness, along with a regularization term based on feature statistics to address label and feature skew issues.

These components integrate local and global knowledge to extract robust feature distributions during local training.
- This paper presents a consistency-based adaptive aggregation method, weighted by label entropy, to further optimize the server aggregation process, enhancing the global model's ability to handle heterogeneity and improving its generalization performance.
- Comprehensive experiments conducted on three public datasets showcase the enhanced performance of the proposed approach, outperforming several classic baselines, especially under high heterogeneity, confirming its applicability across various scenarios.

The structure of the remaining sections are organized as follows: Section 2 presents technical methods in FL aimed at tackling the issues posed by non-IID data. We also discuss methods for handling label distribution skew and review recent advancements in overcoming the problem of model forgetting. In Section 3, we provide a detailed description of the proposed FL framework and its novel techniques. This includes an introduction to the standard FL architecture and the data heterogeneity issues it faces, followed by our new method based on un Skewed-aware loss, feature-skew-aware regularization, and consistency-optimized aggregation. Section 4 covers the experimental setup and overall evaluation to verify the effectiveness and performance of the proposed method, along with factors that may influence the experimental outcomes. Section 5 shows, through a short theoretical analysis, that the proposed method ensures stable convergence under non-IID conditions. Lastly, Section 6 concludes with a summary of the paper's contributions and offers perspectives on future research avenues.

2 Method

2.1 Problem Settings and Overall Framework

The standard FL architecture employs a distributed decentralized training paradigm. A central server periodically samples from a set of S comprising a total of N clients, followed by communication interactions. Each client possesses a private local dataset $|D_n|$. The global objective of FL is to minimize the following finite and non-convex problem:

$$F(\Theta) = \frac{1}{n}\sum_{i=1}^{N} F_i(\Theta) \tag{1}$$

$$f_i(\Theta) \triangleq \mathbb{E}[\mathcal{L}_{(x,y)\sim\mathcal{D}_i}(\Theta;(x,y))] \tag{2}$$

Here, f_i represents the global objective function, which is associated with the population risk related to the randomly sampled data from the local distribution D_i and the global model Θ. Due to heterogeneity, this may vary across local clients. Our approach performs local optimization for label skew by introducing a novel loss function and feature skew correction regularization, combined with a global aggregation optimization strategy. FedCO is a consistency optimization FL framework for label skew and feature drift, which integrates local correction and global adaptive aggregation to better alleviate data heterogeneity in FL.

Fig. 1 and Algorithm 1 illustrate the overall framework and process of our proposed FL method. To reduce the effects of heterogeneity in the federated environment, we employ unbiased-aware loss and feature skew-aware regularization to address label drift and partial labeling issues. Additionally, we introduce a global aggregation method that dynamically adjusts consistency based on label entropy. This comprehensive framework not only considers the training of local clients but also emphasizes the consistency of global aggregation. In the following sections, we will elaborate on these novel techniques incorporated into our FL architecture.

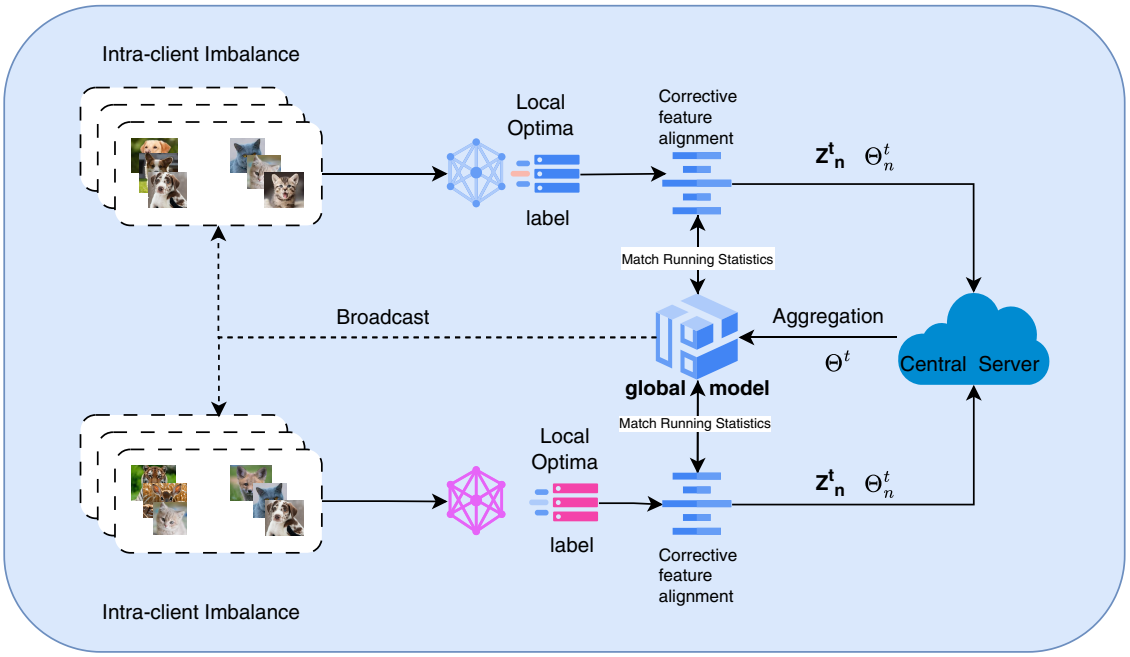


Figure 1: A basic framework for federated learning for consistency optimization.

Algorithm 1: FedCO

Require: Initial server model Θ, local learning rate η_l, regularization weight λ

for each round $t = 0$ to $T-1$ **do**
  Sample subset $S \subseteq [N]$ of clients
  Communicate Θ^t to all clients $n \in S$
  for each client n in parallel **do**
    Initialize local model $\Theta^t_n = \Theta^t$
    for $k = 0$ to $K-1$ **do**
      Sample minibatch $\xi_{n,k}$ from the local dataset
      Compute unbiased loss gradient: $g_{n,k} = \nabla L(\Theta_{n,k}; \xi_{n,k})$
      Compute feature correction regularization: $\text{FeSA}_n = \nabla \text{FeSA}(f_n, f_g)$
      Update local model:
      $\Theta^{k+1}_n = \Theta^k_n - \eta_l \left(g_{n,k} + \lambda \cdot \text{FeSA}_n\right)$
    end for
    Calculate Z^t_n
    Send locally updated model Θ^{k+1}_n and Z^t_n back to the server
  end for
  Calculate aggregation weight w^t_n
  Update global model:
  $\Theta^{t+1} = \sum_{n=1}^{N} w^t_n \cdot \Theta^t_n$
end for

2.2 Label Skew-Awared Loss

In the traditional cross-entropy loss function, it is assumed that all classes are uniformly distributed in the training data. However, in real-world FL scenarios, the data distribution across different clients often

varies significantly, leading to label skew. To address this issue, the l_{LASA} modifies the loss function to better adapt to these imbalanced distributions, thereby improving the global model's ability to generalize. In scenarios with partial labels, where certain labels are completely absent from the training data, l_{LASA} effectively handles label absence by assigning the sum of the unseen labels to specific positions (e.g., the first or second position in the outputs). This ensures that even when certain labels are missing, the model still receives meaningful training signals. In cases of label skew, l_{LASA} mitigates the impact of label imbalance on model training by calculating the log-sum of unseen labels and adjusting the probabilities of observed labels. By modifying the output probabilities, this method effectively reduces the skew introduced by label skew, ultimately improving the generalization capability of the global model.

Let z_i represent the model output for the i-th sample of a client, and y_i represent the true label of the i-th sample. The traditional cross-entropy loss function is defined as:

$$\mathcal{L}_{\mathrm{CE}} = -\frac{1}{N}\sum_{i=1}^{N}\log p(y_i|\mathbf{z}_i) \tag{3}$$

where $p(y_i|z_i)$ is the softmax probability derived from the model output.

In the context of label skew, we introduce the LASA to retain knowledge of the unseen classes. The LASA is defined as follows: Given a dataset with C total classes, suppose the dataset on client i contains samples from only a subset of these classes. We define an unseen label set U and a seen label set V. For scenarios with label skew or partial labels, the loss function incorporates the effect of the unseen labels by computing the softmax probabilities for these labels, summing them, and then taking the logarithm. This allows the loss to account for the impact of unseen labels during training.

First, we calculate the total probability of the unseen labels as $p(O)$. For each sample, the model output is processed through a softmax function to generate the class probabilities. The total probability for the unseen labels is computed as:

$$p(O) = \log\sum_{j\in U}\exp\left(z_{ij}\right) \tag{4}$$

where z_{ij} represents the model output for the j-th class of the i-th sample. Next, for each seen label $k \in V$, we calculate the adjusted softmax probability as follows:

$$\hat{p}(y_i = k|\mathbf{z}_i) = \frac{\exp\left(z_{ik}\right)}{\exp\left(z_{ik}\right) + \sum_{j\in U}\exp\left(z_{ij}\right)} \tag{5}$$

here, $\exp(z_{ik})$ denotes the exponential value of the model output for the label k corresponding to the i-th sample. The adjusted probability $\hat{p}(y_i = k|z_i)$ estimates the actual probability of label k by comparing the output value of label k to the total probability of the unseen labels.

We then compute the loss value L_{LASA} as the average negative log probability across all samples:

$$\mathcal{L}_{\mathrm{LASA}} = -\frac{1}{N}\sum_{i=1}^{N}[\log\hat{p}(y_i|\mathbf{z}_i)] \tag{6}$$

The LASA effectively addresses label skew and partial label issues by maximizing the predicted probabilities of the most likely classes during local training. In essence, it enables local models to retain the global model's predictions on unseen labels, thereby reinforcing global knowledge derived from fully labeled data. Notably, this loss relies solely on the global model's predictions, which are already communicated

during standard FL rounds. Consequently, it introduces no additional label distribution information, fully preserving privacy. Moreover, the proposed loss is not limited to label skew in FL; it can also be readily extended to tackle general label imbalance problems in other domains.

2.3 Adaptive Consistency Aggregation Methods

After addressing the issues of label skew and feature discrepancies in local client training, our next task is to solve another fundamental problem in heterogeneous FL: how to appropriately weight and aggregate the local models sent by clients to update the global model. In basic FL, the most common aggregation approach is FedAvg, which uses a weighted average. However, under heterogeneous conditions, this simple weighted averaging is unfair and fails to fully leverage the optimization capabilities of each client. In fact, clients with highly skewed local data often perform well in the early stages of FL. Assigning larger weights to these clients during aggregation may impede the global model's convergence and negatively impact its performance. To tackle this issue, we introduce an innovative adaptive dynamic aggregation strategy on the server side. By measuring the degree of optimization of different local models under privacy-preserving constraints, we allocate appropriate weights to each model, taking into account the optimization degree, dataset distribution, and dataset size simultaneously. In the design, we select label entropy as the core metric to measure the difference in local data distribution, combined with model parameter consistency to measure the degree of optimization, which exhibits significant advantages over other client weighting schemes (such as gradient norm-based or loss exponential weighting). First, label entropy can more intuitively quantify the distribution difference of local data labels, which is crucial for weight allocation in heterogeneous FL. Second, compared with gradient norms that may be distorted by noise interference or loss exponential weighting that may excessively amplify weights due to local optima, label entropy is more robust and less sensitive to noise, more accurately reflecting data distribution characteristics. Furthermore, label entropy avoids excessive skew toward client models with high data heterogeneity, effectively enhancing the aggregation fairness and convergence performance of the global model.

We generally assume that the degree of optimization of a local client correlates with the amount of local training iterations or the size of the local dataset, and inversely proportional to the data heterogeneity in a heterogeneous environment. For each client, we assess the local data distribution by calculating the client's label entropy, and further calculate the consistency between the local model parameters and the global model parameters using an unbiased estimate, reflecting the model's optimization level during every communication iteration.

Specifically, when the server aggregates the global model, the target client transmits its local model and corresponding optimization coefficient to the server. The implementation is as follows: First, by calculating the label entropy $H(p_n)$ for each client, we obtain the uniformity of its label distribution. Then, the label entropy is used to adjust the weighting factor:

$$H(p_n) = -\sum_{y} p_n(y)\log(p_n(y)) \tag{7}$$

$$H_{max} = \log(C) \tag{8}$$

$$\alpha_n = 1 - \frac{H(p_n)}{H_{max}} \tag{9}$$

here, H_{max} is the maximum possible entropy used to normalize the entropy values. $\hat{p}_n$ is the unbiased probability of client n, and p_g is the predicted probability of the global model. After obtaining the label entropy, we further compute the optimization coefficient by evaluating the consistency of the model parameters:

$$Z_n^t = \frac{\alpha_n}{|D_n|} \cdot \frac{1}{HW} \sum_{i=1}^{|D_n|} \sum_{h,w} 1\{ \arg\max(\hat{p}_n(x_{i,n})) = \arg\max(p_g(x_{i,n}))\} \tag{10}$$

where $|D_n|$ is the size of the dataset for client n, HW is the spatial dimension of each sample (e.g., height and width for images), and $x_{i,n}$ is the i-th sample. The transmission of the optimization coefficient to the server does not involve sharing any private information from the client, thus ensuring effective protection of the local client's data privacy. Importantly, the optimization coefficient Z_n^t is a scalar post-processing function of the already-shared model parameters; thus, by the post-processing property of differential privacy, it does not incur any extra privacy leakage beyond that of standard FL (e.g., FedAvg). Consequently, our method remains compatible with existing privacy-preserving techniques such as DP-SGD and secure aggregation.

We then account for both the size of the local dataset and the optimization coefficient to compute the weight for each target client:

$$w_n^t = \frac{Z_n^t \cdot |D_n|}{\sum_{j=1}^{N} Z_j^i \cdot |D_j|} \tag{11}$$

where $|D_n|$ is the dataset size of client n and N is the total number of clients. Finally, we perform weighted aggregation of the model parameters sent by the local clients to obtain the updated global model:

$$\Theta^{t+1} = \sum_{n=1}^{K} w_n^t \cdot \Theta_n^t \tag{12}$$

3 Experiment

In the experimental section, we conducted extensive qualitative and quantitative experimentsto verify the feasibility and effectiveness of our proposed approach. Our method was compared against several advanced baselines across three different data heterogeneity scenarios. Additionally, we performed ablation studies on the number of clients, data heterogeneity, and our proposed method to further demonstrate its robustness and effectiveness under various conditions. The detailed hyperparameter settings for the federated basic configurations and all baseline methods are summarized in Table 1.

Table 1: Hyperparameters for federated basic settings and baseline methods.

Parameter	Value
Federated Basic Settings	
Number of registered clients	10
Client selection strategy	Uniform random sampling per round
Communication rounds (CIFAR-10/100)	100
Communication rounds (TinyImageNet)	50
Local training epochs per client	10
Local learning rate	0.01
Optimizer	SGD with momentum 0.9
FedProx	Proximal term $\mu = 10^{-3}$
MOON	Contrastive regularization $\mu = 1.0$

(Continued)

Table 1 (continued)

Parameter	Value
FedLogitCal	Calibration temperature = 0.1
FedRS	Restricted strength = 0.5
FedAvg/FedNova/SCAFFOLD	Default original settings (no extra hyperparameters)

3.1 Experimental Setup

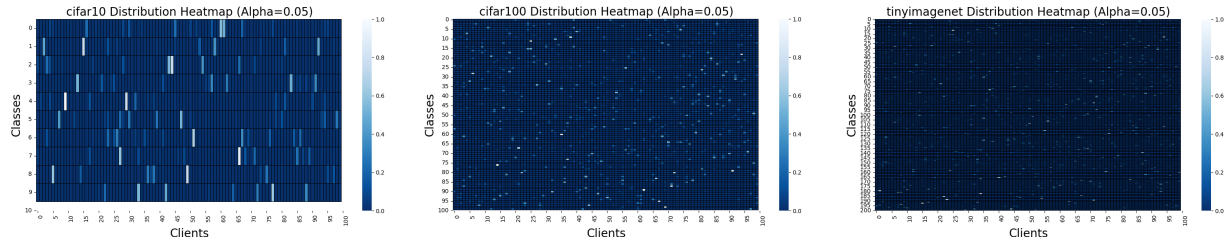
To ensure reproducible federated learning workflows, we first define a unified global architecture and default hyperparameters. This work adopts a classic centralized server-client architecture with one global server and multiple edge clients. For all comparison experiments, the default number of registered clients is set to 10. In each communication round, the server uniformly selects a random subset of clients for local updating. The default global round is 100, and each activated client performs 10 local training epochs per round. The aggregation workflow follows a two-stage pipeline: activated clients optimize locally on private data and upload encrypted local model deltas to the server; the server then executes the proposed label-entropy-based adaptive client aggregation (ACA) to fuse updates into a new global model, which is broadcast to all clients for the next round.

(a) Dataset: We conducted experiments on three image public datasets: CIFAR-10 [23], CIFAR-100 [24], and TinyImageNet [23]. CIFAR-10 has 10 categories and a total of 60,000 images, with 50,000 for training and 10,000 for testing. On the other hand, CIFAR-100 has 100 categories with the same total number of images of 60,000. TinyImageNet has 200 categories and a total of 110,000 images. For quantitative construction, control and evaluation of cross-client data heterogeneity, we adopt the canonical Dirichlet-based non-IID partitioning strategy DirPartition [11] to split training samples, which is the most widely recognized label-skew partitioning scheme in federated vision tasks. Specifically, label distribution across clients obeys the Dirichlet distribution $\boldsymbol{p}_k \sim \mathrm{Dir}(\alpha)$, where $\boldsymbol{p}_k$ denotes the class proportion vector assigned to the k-th client. The concentration parameter α serves as the quantitative control coefficient for data heterogeneity: a smaller α yields more imbalanced label proportion distribution, namely higher client-side heterogeneity. We set three gradient heterogeneity levels for comprehensive evaluation: $\alpha \in \{0.05, 0.1, 0.5\}$, corresponding to extreme heterogeneity, high heterogeneity and mild heterogeneity, respectively. To quantitatively evaluate heterogeneity degree, we adopt global label distribution divergence as the evaluation metric, which calculates the KL divergence between per-client label distribution and the original global label distribution. Higher KL divergence values represent more severe non-IID degree. To intuitively visualize quantitative heterogeneity differences, we generated label distribution heatmaps to reflect per-client class sample occupancy. The heatmaps in Fig. 2 show the class allocation on 100 statistical clients for intuitive observation, where the color of each rectangle represents the proportion of samples belonging to a specific class for each client—dark blue indicates a low proportion, while light blue indicates a high proportion. As shown in our extreme heterogeneous setting with $\alpha = 0.05$, less than 10% of clients have fully IID datasets, represented by the white rectangles, which verifies the effectiveness of our Dirichlet partitioning for constructing extreme client distribution drift.

(b) Evaluation Metrics: In our study, we conducted a comprehensive evaluation by using accuracy as the primary metric, following the settings outlined in previous works. We tested the global model's accuracy on the joint test set, which allows us to assess the ability of the learned global model to generalise. Additionally, the differences in test accuracy revealed the generalization abilities of various methods. To ensure fairness, we trained all algorithms for the same number of rounds.

(c) Baseline Settings: Our method primarily addresses data heterogeneity induced client distribution drift and divergent local updates in the context of non-IID data, so the selected advanced FL baselines are all optimization methods aimed at tackling client-side data heterogeneity. We compared our approach with several methods, including FedAvg [1], FedNova [11], FedProx [6], Scaffold [8], FedLogitCal [25], MOON [26], FedRS [27], and FedProc [28]. Specifically, the regularization coefficient for FedProx was set to $\mu = 10^{-3}$; the calibration temperature for FedLogitCal was set to 0.1; the regularization coefficient for MOON was chosen as $\mu = 1.0$; and the restricted strength for FedRS was set to 0.5. Consistent with our unified FL configuration, all baseline methods adopt 10 total clients, per-round random client selection, 10 local epochs and 100 global communication rounds for fair comparison under gradient non-IID Dirichlet partitioning.

(d) Implementation Details: The project is based on the work of Shi et al. [29]. For the comprehensive experiments, we set the local training rounds to 10, with each client's data volume partitioned according to the previously described non-IID method. We configured the communication rounds to 100 for CIFAR-10/100 and 50 for TinyImageNet, utilizing the SGD optimizer with a momentum of 0.9. In our FL setup, the local learning rate was consistently set to 0.01. All experiments employed MobileNetV2 as the backbone network, incorporating data augmentation techniques as proposed by Cubuk et al. [30]. The experimental environment was configured with an Intel(R) Xeon(R) Platinum 8352V CPU @ 2.10 GHz and an RTX 4090 (24 GB).



(**a**) CIFAR-10 (**b**) CIFAR-100 (**c**) TinyImageNet

Figure 2: The heatmaps of the heterogeneity weights from the Dirichlet distribution at $\alpha = 0.05$ for different datasets.

For the proposed FedCO, the regularization coefficient λ controlling the feature-statistics alignment term (FeSA) was set to 0.005 by default. This choice is based on a sensitivity analysis on CIFAR-10 under the highest heterogeneity ($\alpha = 0.05$), as illustrated in Table 2: $\lambda = 0.005$ achieved the best accuracy (62.58%) while larger values (e.g., $\lambda = 1$) caused slight degradation due to over-regularization, and smaller values yielded comparable performance. The LASA employs a temperature parameter of 0.5, and the adaptive client aggregation (ACA) uses label entropy weighting without additional hyperparameters.

Table 2: Sensitivity of accuracy (%) to the regularization coefficient λ on CIFAR-10 under $\alpha = 0.05$ (non-IID).

λ	0.005	0.01	0.05	0.1	0.5	1
Accuracy (%)	62.58	62.14	62.10	62.14	62.14	61.83

3.1.1 Ablation Studies Analysis

We conducted a quantitative ablation study to evaluate the contribution of each key component in our proposed FedCO framework Label Skew-Aware loss (LASA), Adaptive Client Aggregation (ACA), and Feature Skew-Aware Regularization (FeSA)—under the non-IID setting with Dirichlet parameter $\alpha = 0.05$

on three datasets: CIFAR-10, CIFAR-100, and TinyImageNet. FedAvg serves as the baseline. The results are summarized in Table 3.

Table 3: Quantitative ablation study on three datasets under $\alpha = 0.05$ non-IID setting.

LASA	ACA	FeSA	ACC (%) at $\alpha = 0.05$		
			CIFAR-10	CIFAR-100	TinyImageNet
			66.32	59.45	35.16
		✓	70.31	61.13	36.23
	✓		68.14	61.42	36.15
✓			69.83	62.08	36.53
	✓	✓	70.75	62.57	36.78
✓		✓	71.65	62.42	36.54
✓	✓		72.58	62.34	36.87
✓	✓	✓	**73.64**	**62.78**	**37.45**

Note: Bold numbers indicate the highest accuracy in each column.

As shown in the table, on CIFAR-10, the baseline FedAvg achieves 66.32% accuracy. Incorporating ACA alone raises the accuracy to 70.31%, demonstrating the benefit of dynamic weight adjustment in high heterogeneity. Adding only FeSA or LASA yields 68.14% and 69.83%, respectively, indicating that feature alignment and label-skew correction each provide moderate gains. Pairs of components—ACA+FeSA (70.75%), LASA+FeSA (71.65%), and ACA+LASA (72.58%)—outperform any single component, confirming their complementarity. The full model with all three components achieves the best accuracy of 73.64%, underscoring the synergy among adaptive aggregation, feature regularization, and label-skew-aware loss. A similar trend is observed on CIFAR-100, a more challenging dataset with 100 classes. The baseline accuracy is 59.45%. Adding ACA, FeSA, or LASA individually improves accuracy to 61.13%, 61.42%, and 62.08%, respectively. Two-component combinations reach 62.34%–62.57%, and the full FedCO model achieves 62.78%. These results validate the generalizability of FedCO beyond CIFAR-10 and highlight its effectiveness even under high class cardinality. On TinyImageNet, a large-scale dataset with 200 classes, the baseline FedAvg achieves 35.16% accuracy. Incorporating ACA, FeSA, or LASA individually yields 36.23%, 36.15%, and 36.53%, respectively. Two-component combinations improve accuracy to 36.54%–36.87%, and the full FedCO model achieves the best accuracy of 37.45%. Although the absolute gains are smaller due to the increased task difficulty, the consistent improvement trend across all datasets confirms that FedCO's three components synergistically address label skew, feature drift, and client heterogeneity under extreme non-IID conditions.

To evaluate the impact of different aggregation strategies, we compared our proposed ACA with standard FedAvg and FedExp [31]. As shown in Fig. 3, FedExp yields performance nearly identical to FedAvg, indicating that naive aggregation rules are insufficient to handle severe label distribution skew. In contrast, our ACA consistently achieves superior accuracy across all datasets and heterogeneity levels. This clear margin justifies our choice of ACA as the final aggregation mechanism, as it adaptively mitigates the negative effects of diverse local updates via label-entropy guided weighting.

Overall, this ablation study demonstrates that each of the three proposed components contributes positively to handling label skew, feature drift, and client heterogeneity, and their combination yields the most robust performance across different datasets under extreme non-IID conditions.

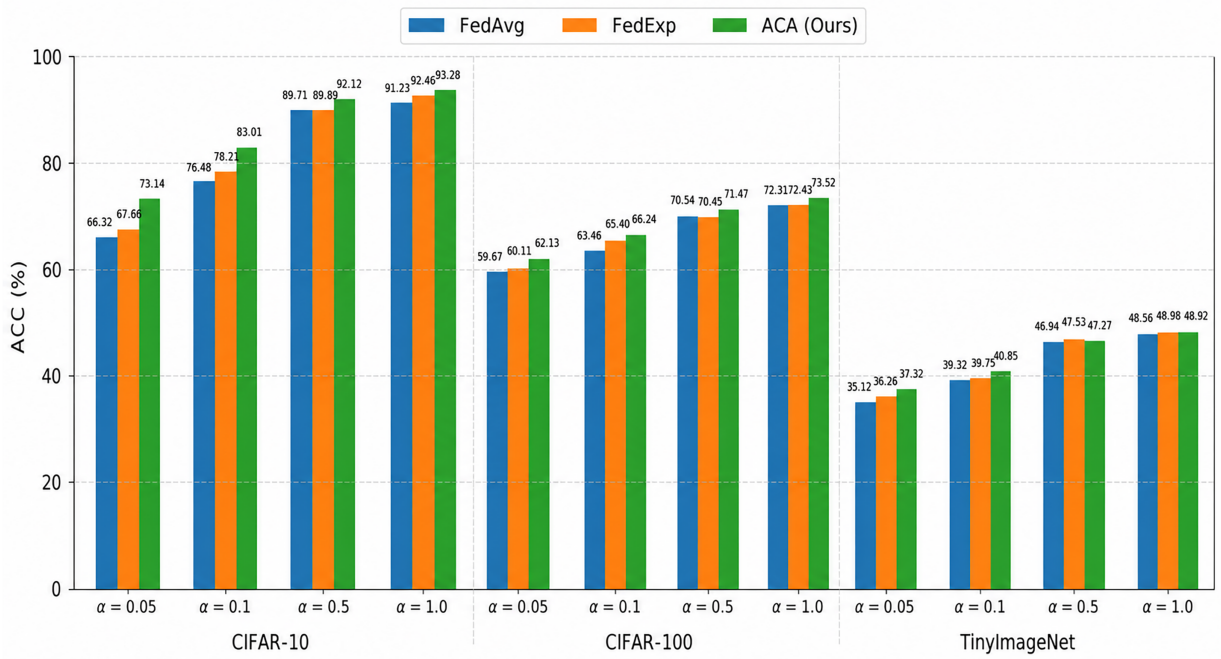


Figure 3: Ablation study of aggregation strategies.

3.2 Overall Performance Comparison

In the overall performance comparison, FedCO demonstrates remarkable adaptability and robustness, especially in highly heterogeneous environments. Its core advantage lies in its ability to adjust global generalization through label correction and dynamic weighting, enabling it to maintain superior model convergence and final performance even in the face of extremely imbalanced client data distributions.

In Tables 4 and 5, we compared various FL optimization methods on the CIFAR-10 and CIFAR-100 datasets, all of which were run for 100 rounds using different non-IID data partitions ($\alpha \in \{0.05, 0.1, 0.5, 1\}$). It is evident that our proposed FedCO method excels in high heterogeneity environments, showcasing strong robustness, particularly under the more challenging $\alpha = 0.05$ setting, where it still achieves high accuracy. This firmly demonstrates that FedCO can reliably perform well when addressing data distribution imbalances. Specifically, on the CIFAR-10 dataset, FedCO achieves an accuracy of 73.64% under $\alpha = 0.05$, outperforming FedAvg by 7.32% and surpassing strong baselines such as Fedlogitcal (72.72%) and FedRS (71.82%). Under the near-IID setting ($\alpha = 1$), FedCO reaches 93.78%, again leading all compared methods, including FedProc (93.07%). Even under $\alpha = 0.5$, FedCO retains its competitive edge, demonstrating strong adaptability.

On the CIFAR-100 dataset, FedCO also performs exceptionally well, achieving an accuracy of 62.63% under $\alpha = 0.05$, which is 2.96% higher than FedAvg and notably exceeds Fedlogitcal (56.77%) and FedRS (59.10%). Under $\alpha = 1$, FedCO attains 74.02%, still the highest among all methods, with FedProc reaching 72.61%. Although the performance improvements diminish somewhat at $\alpha = 0.1$ and $\alpha = 0.5$, FedCO consistently leads the pack. This indicates that FedCO's optimization is particularly effective in scenarios of high heterogeneity, aligning with its design intent.

Table 4: Test accuracy under different heterogeneities on CIFAR-10.

Method	CIFAR-10			
	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1$
FedAvg	66.32 ± 0.18	76.48 ± 0.17	89.71 ± 0.15	91.23 ± 0.14
Scaffold	53.24 ± 0.15	75.46 ± 0.19	87.05 ± 0.16	89.87 ± 0.15
MOON	68.79 ± 0.17	78.70 ± 0.16	90.08 ± 0.15	91.94 ± 0.13
FedNova	64.57 ± 0.19	79.96 ± 0.16	90.23 ± 0.15	91.76 ± 0.14
FedProx	64.11 ± 0.10	76.13 ± 0.18	89.57 ± 0.16	90.95 ± 0.14
Fedlogitcal	72.72 ± 0.16	80.53 ± 0.15	92.18 ± 0.14	93.05 ± 0.13
FedRS	71.82 ± 0.17	76.45 ± 0.18	90.33 ± 0.15	91.68 ± 0.14
FedProc	67.50 ± 0.18	69.54 ± 0.15	92.27 ± 0.15	93.07 ± 0.21
Our Method	**73.64 ± 0.15**	**83.51 ± 0.17**	**92.62 ± 0.14**	**93.78 ± 0.12**

Note: Bold numbers denote the highest accuracy in each column.

Table 5: Test accuracy under different heterogeneities on CIFAR-100.

Method	CIFAR-100			
	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1$
FedAvg	59.67 ± 0.12	63.46 ± 0.11	70.54 ± 0.18	72.31 ± 0.15
Scaffold	54.77 ± 0.14	61.54 ± 0.11	68.83 ± 0.19	70.45 ± 0.16
MOON	56.79 ± 0.13	65.48 ± 0.20	71.81 ± 0.17	73.45 ± 0.14
FedNova	60.50 ± 0.11	66.43 ± 0.19	71.60 ± 0.17	73.60 ± 0.15
FedProx	60.12 ± 0.12	65.42 ± 0.20	70.68 ± 0.18	72.11 ± 0.16
Fedlogitcal	56.77 ± 0.15	66.23 ± 0.19	70.83 ± 0.17	72.44 ± 0.15
FedRS	59.10 ± 0.12	66.35 ± 0.17	70.03 ± 0.18	71.89 ± 0.16
FedProc	56.72 ± 0.14	64.51 ± 0.26	68.80 ± 0.11	72.61 ± 0.13
Our Method	**62.63 ± 0.11**	**66.74 ± 0.18**	**71.97 ± 0.16**	**74.02 ± 0.14**

Note: Bold numbers denote the highest accuracy in each column.

Additionally, in Table 6, we tested the more challenging Tiny-ImageNet dataset across the same four heterogeneity levels. FedCO achieves the highest test accuracy for all α values, reaching 37.82% at $\alpha = 0.05$, 41.35% at $\alpha = 0.1$, 47.77% at $\alpha = 0.5$, and 49.42% at $\alpha = 1$. Notably, at $\alpha = 0.05$, FedCO surpasses FedProc (37.96%) and Fedlogitcal (36.86%), and at $\alpha = 1$, it outperforms FedProc (49.34%) and MOON (48.91%). Given that Tiny-ImageNet is more complex than the CIFAR datasets, FedCO's ability to maintain its leading position under all heterogeneity conditions further proves its superiority in handling more complex datasets. This has significant implications for addressing the complex and unpredictable data distributions encountered in real-world scenarios.

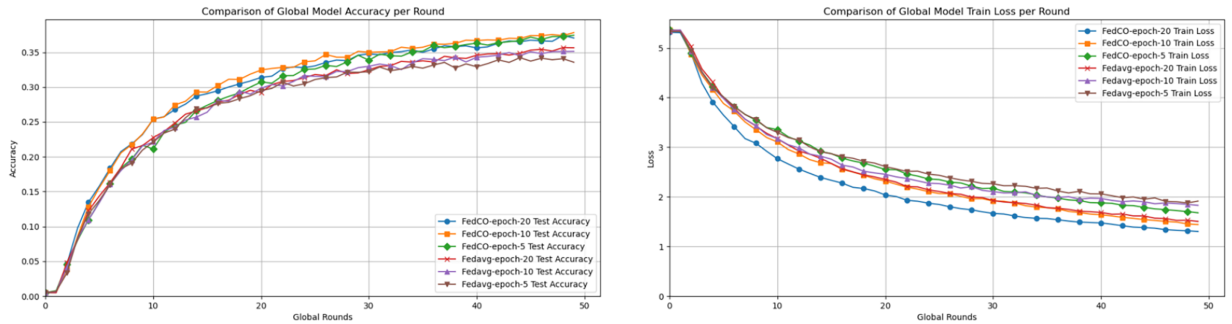
Table 6: Test accuracy under different heterogeneities on TinyImageNet.

Method	TinyImageNet			
	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1$
FedAvg	35.12 ± 0.21	39.32 ± 0.18	46.94 ± 0.15	48.56 ± 0.14
Scaffold	34.71 ± 0.24	37.88 ± 0.20	44.54 ± 0.17	46.23 ± 0.16
MOON	36.13 ± 0.19	40.61 ± 0.16	47.39 ± 0.13	48.91 ± 0.12
FedNova	35.75 ± 0.22	39.73 ± 0.19	47.35 ± 0.14	48.72 ± 0.13
FedProx	35.16 ± 0.23	39.65 ± 0.21	46.16 ± 0.16	47.85 ± 0.15
Fedlogitcal	36.86 ± 0.20	41.12 ± 0.17	46.82 ± 0.14	48.23 ± 0.13
FedRS	35.66 ± 0.22	40.52 ± 0.18	47.07 ± 0.15	48.44 ± 0.14
FedProc	37.96 ± 0.21	40.45 ± 0.10	46.80 ± 0.23	49.34 ± 0.14
Our Method	**37.82** ± **0.20**	**41.35** ± **0.12**	**47.77** ± **0.14**	**49.42** ± **0.12**

Note: Bold numbers denote the highest accuracy in each column.

3.3 Performance with Different Local Epochs

We validated the effectiveness of our method by adjusting the number of local epochs for each client, either increasing or decreasing them. We utilized the Tiny-ImageNet dataset and set the local epochs to 5, 10, and 20. The results are displayed in Fig. 4. Under various epoch settings, our method consistently outperformed other baselines. When the number of local updates is too low, the model fails to train effectively, resulting in poor performance. Our approach aims to mitigate label skew during local updates and optimize global consistency, which explains why FedAvg performs poorly at lower epoch counts. In contrast, our method remains relatively stable. While baseline performance improves with a larger number of epochs, our method consistently maintains superior performance compared to the baselines.



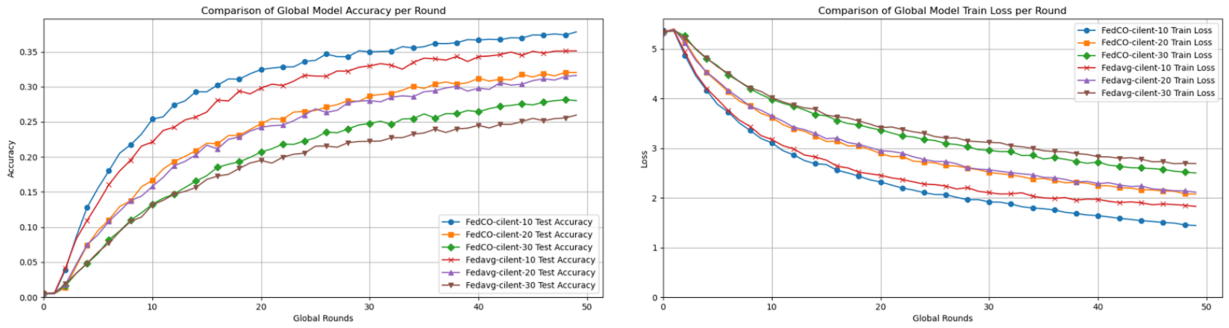
(**a**) Global Model Accuracy per Round (**b**) Global Model Train Loss per Round

Figure 4: Visualization of accuracy and loss under ablation of local client rounds.

3.4 Ablation with Varying Client Numbers

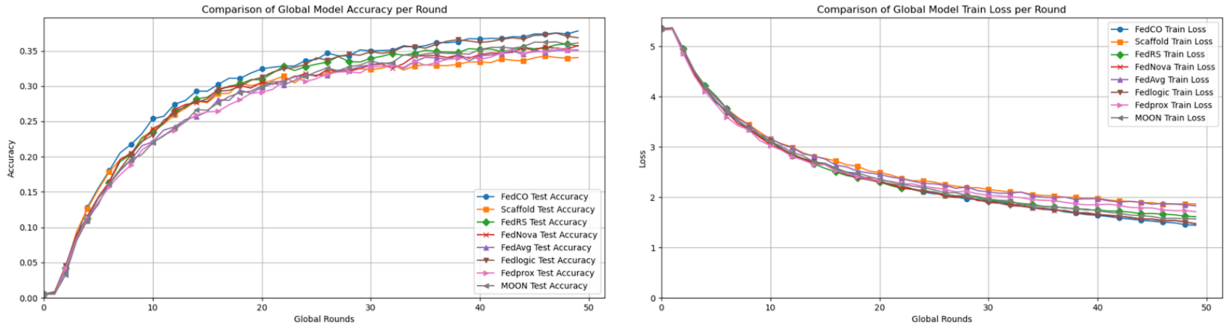
We further investigated whether the optimization effect of the proposed FedCO method remains significant when the number of clients increases. In this experiment, we divided the Tiny-ImageNet dataset into 10, 20, and 30 clients for comparative experiments to verify the robustness of the method. The experimental results are shown in Figs. 5 and 6, which respectively demonstrate the changes in global model accuracy and loss with the number of communication rounds. The results show that as the number of global communication rounds increases, the performance of the FedCO method remains stable under different

numbers of clients and significantly outperforms the traditional FedAvg method. Specifically, FedCO exhibits a faster and more significant improvement in accuracy, and with the increase in the number of clients, it can still bring a 2%~3% performance improvement. In addition, in terms of training loss, the decline rate of FedCO is also significantly better than that of the comparative method.



(**a**) Global Model Accuracy per Round (**b**) Global Model Train Loss per Round

Figure 5: Visualization of accuracy and loss under ablation of the number of clients.



(**a**) Global Model Accuracy per Round (**b**) Global Model Train Loss per Round

Figure 6: Visualization of accuracy and loss compared to multiple baselines.

This advantage is mainly attributed to the innovative design of FedCO: by combining local optimization with global aggregation, FedCO effectively alleviates the model bias problem caused by data heterogeneity among clients. In particular, when the number of clients increases, the feature skew correction regularization and optimization strategy designed by FedCO can better balance the adaptability of the global model across different clients, thereby improving overall performance. This result further verifies the robustness and wide applicability of FedCO.

3.5 Computation Cost & Communication Cost

To ensure a fair comparison, all methods were evaluated on the same machine (fixed Intel Xeon CPU and a single RTX 4090 GPU), with identical batch size, local learning rate, SGD optimizer, and 10 local epochs per round. The normalized training time reported in Table 7 is defined as the net GPU runtime consumed to complete exactly one global communication round under a unified non-IID Dirichlet partition with $\alpha = 0.05$, eliminating interference from system jitter, idle memory, and background processes. This serves as a fair indicator of actual computational overhead. FedAvg demonstrates the lowest training times across all datasets, recording 54.4 s on CIFAR-10, 4 min 10 s on CIFAR-100, and 10 min 8 s on Tiny-ImageNet,

consistent with its design that avoids additional loss terms or control variables. In contrast, FedProx and MOON introduce extra loss terms on top of FedAvg, while SCAFFOLD incorporates additional control variables for both server and clients. Among these, MOON exhibits the highest computational overhead, with training times of 1 min on CIFAR-10, 5 min 45 s on CIFAR-100, and 18 min 16 s on Tiny-ImageNet, attributable to the need for forward propagation through three models (current round, previous round, and global models) during local training. FedProx, with training times of 1 min 2 s on CIFAR-10, 3 min 40 s on CIFAR-100, and 12 min 20 s on Tiny-ImageNet, performs comparably to most methods (e.g., SCAFFOLD's 52.3 s and 6 min 47 s) on CIFAR-10 and CIFAR-100, yet outperforms others (except FedAvg) on Tiny-ImageNet, such as SCAFFOLD's 16 min 12 s and MOON's 18 min 16 s, indicating a growing advantage in computational efficiency as dataset size and local network scale increase. Our proposed method, FedCO, achieves a balanced performance with training times of 52.1 s on CIFAR-10, 3 min 18 s on CIFAR-100, and 11 min 42 s on Tiny-ImageNet, aligning closely with other methods such as FedNova (57.7 s, 4 min 20 s, 15 min 37 s), Fedlogitcal (54.3 s, 3 min 12 s, 11 min 15 s), and FedRS (50.6 s, 2 min 41 s, 10 min 44 s), while demonstrating competitiveness in large-scale scenarios supported by its innovative optimization strategies.

Table 7: The normalized training time for different approaches.

Method	CIFAR-10	CIFAR-100	Tiny-ImageNet
FedAvg	54.4 s	4 min 10 s	**10 min 8 s**
SCAFFOLD	52.3 s	6 min 47 s	16 min 12 s
MOON	1 min	5 min 45 s	18 min 16 s
FedNova	57.7 s	4 min 20 s	15 min 37 s
FedProx	1 min 2 s	3 min 40 s	12 min 20 s
Fedlogitcal	54.3 s	3 min 12 s	11 min 15 s
FedRS	**50.6 s**	**2 min 41 s**	10 min 44 s
Our Method	52.1 s	3 min 18 s	11 min 42 s

Note: Bold values represent the fastest training time in each column.

Regarding communication complexity, FedCO introduces statistical information such as label entropy and consistency estimates to dynamically adjust the aggregation weights of the global model. Represented as scalars, these statistics impose negligible storage overhead. In our experiments using MobileNetV2 as the backbone model (with a parameter size of approximately 9.6 MB), the additional transmission of label entropy and consistency statistics contributes only about 0.01% to the communication overhead, which can be considered negligible. This represents a significant advantage over traditional FedAvg in terms of communication cost, while simultaneously enhancing the performance of the global model, highlighting FedCO's efficiency and robustness in optimizing heterogeneous FL.

4 Convergence Analysis

To demonstrate the convergence of FedCO, we need to establish some common assumptions regarding the objective function and gradient:

Assumption 1: *Considering a general non-convex minimization problem, the local objective functions f and the global objective function F satisfy the following conditions:*

(1) The gradient of the loss function $F_n(w)$ for each client is Lipschitz continuous, meaning there exists a constant $L > 0$ such that:

$$\| \nabla F_n(w) - \nabla F_n(w') \| \leq L \| w - w' \| \tag{13}$$

This ensures that the gradient changes are not excessively large.

(2) Each client's gradient estimation has bounded variance, which means there exists a constant σ^2 *such that:*

$$\mathbb{E}_{\xi\sim D_n}[\| \nabla f(w;\xi) - \nabla F_n(w) \|^2] \leq \sigma^2 \tag{14}$$

This limits the level of noise introduced by local data.

(3) Each client's gradient estimation is unbiased, meaning for any client n:

$$\mathbb{E}_{\xi\sim D_n}[\nabla f(w;\xi)] = \nabla F_n(w) \tag{15}$$

This guarantees that local updates from each client do not introduce skew.

Assumption 2: *In the presence of client heterogeneity, the following condition holds: For all w, there exist constants* $G \geq 0$ *and* $B \geq 1$ *such that:*

$$\frac{1}{N}\sum_n \mathbb{E} \| \nabla f_n(w) \|^2 \leq G^2 + B^2 \mathbb{E} \| \nabla F(w) \|^2 \tag{16}$$

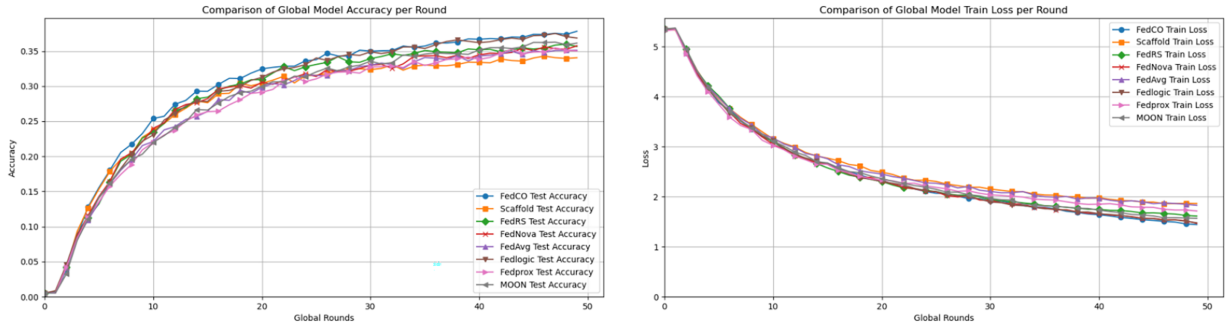
While Assumption 1 addresses general properties for non-convex minimization problems, Assumption 2 is widely referenced in previous works [8,11,32] regarding heterogeneity. Under the conditions of bounded gradients and Lipschitz continuity for each client's loss function $F_n(w)$*, the FL algorithm, incorporating adaptive label entropy weights and feature skew-aware regularization, maintains stable convergence of the global model when the learning rate* $\eta_l \leq \frac{S}{24NKL(B)^{1/2}}$ *and regularization coefficient* λ *are applied.*

Specifically, after T iterations, the global loss function $F(w)$ converges to an optimal solution w^* at the following rate:

$$\mathbb{E}[F(w_T) - F(w^*)] \leq \frac{C}{T} + \mathcal{O}(\lambda), \tag{17}$$

where C is a constant. When λ is suitably small, this algorithm can converge at a rate of $O(1/T)$. Through theoretical analysis, we conclude that the introduction of adaptive label entropy weights effectively controls client contributions, preventing excessive skew due to client heterogeneity, thereby ensuring stable convergence of the global model. Additionally, feature skew-aware regularization penalizes discrepancies in client distributions without significantly interfering with gradient updates, enabling the algorithm to converge at the rate of $O(1/T)$ under appropriate regularization strength.

We also conducted a convergence analysis based on the visualization of the data in Tiny-Imagenet with $\alpha = 0.05$, as shown in Fig. 7. It is evident that FedCO consistently enhances the global model during long-term training compared to the strongest baselines, potentially demonstrating better generalization ability than other algorithms with the same number of training rounds. From the loss curves, we observe that the training loss of FedCO gradually decreases with each round and begins to converge around the 30th round. This indicates that the optimization process of the FedCO algorithm effectively reduces training errors and converges quickly.



(**a**) Global Model Accuracy per Round (**b**) Global Model Train Loss per Round

Figure 7: Visualization of accuracy and loss compared to multiple baselines.

5 Discussion & Future Work

Despite the significant progress achieved by FedCO in heterogeneous federated learning, several limitations remain that warrant further investigation.

While the extra transmission of label entropy and consistency statistics introduces only about 0.01% overhead per round under our experimental setting, this overhead may become non-negligible in ultra-large-scale deployments involving thousands of clients or low-bandwidth environments. The server must collect scalar statistics from all participating clients each round, leading to linear accumulation. Therefore, future work should develop lightweight sketching techniques (e.g., Count-Min sketches or probabilistic counting) that compress entropy statistics in a sub-linear manner relative to the number of label classes, preserving the benefits of adaptive aggregation while scaling to massive client populations.

Our current experiments are limited to 10–50 clients. The aggregation strategy—particularly the server-side computation of label entropy consistency—has not been validated on thousands of clients or high-dimensional data. To address this, we plan to explore hierarchical federated learning architectures and sparse update strategies that reduce per-round communication and computation, enabling FedCO to efficiently leverage distributed resources. Trade-off between efficiency and performance on edge devices. The dynamic adjustment of aggregation weights requires the server to compute label entropy consistency, which is lightweight on a powerful server but could become a bottleneck on resource-constrained edge devices if the server were decentralized. For future work, adaptive resource allocation mechanisms (e.g., early exit or client-side sub-sampling of statistical information) can be investigated to balance latency and accuracy on heterogeneous edge hardware. Adversarial robustness against label-entropy manipulation. FedCO relies on clients honestly reporting their label distributions. A malicious client could fabricate label entropy values to artificially increase its aggregation weight, thereby biasing the global model. This specific vulnerability to Byzantine attacks has not been tested. To address this, we propose integrating robust aggregation rules tailored for entropy-based statistics, such as trimmed mean, median, or Krum. Additionally, cryptographic verification (e.g., zero-knowledge proofs or commitment schemes) can allow the server to verify the correctness of reported label entropy without accessing raw client data. Defending against data poisoning and model inversion attacks—where an adversary injects poisoned samples or reconstructs private information—remains an open direction that requires further study. Communication security. Although FedCO does not currently implement encryption for transmitted parameters or label entropy, this limitation is not unique to our method. In privacy-sensitive applications (e.g., healthcare, finance), any FL framework must secure communication against man-in-the-middle attacks. Future work can adopt standard encryption

techniques (e.g., TLS) as a baseline, and for stronger guarantees, homomorphic encryption or secure multi-party computation can be integrated. Importantly, these security mechanisms are orthogonal to FedCO's heterogeneity handling and can be added without altering the core algorithm.

In summary, the above future directions are directly motivated by the limitations identified in our current study. By addressing them, FedCO can become more scalable, robust, and secure, thereby advancing its applicability in real-world heterogeneous federated learning systems.

6 Conclusions

In this work, Our proposed FedCO has demonstrated effectiveness in handling label skew and feature heterogeneity under extreme non-IID conditions. However, several limitations of the current study need to be acknowledged, which also point to concrete directions for future research.

The performance of FedCO relies on manually tuned balancing weights between the label-shift correction loss and the feature regularization term. All experiments in this work adopt fixed weights for individual datasets, yet static parameter settings tend to be suboptimal in practical federated learning systems where data distributions evolve over time. Future research can explore adaptive hyperparameter control mechanisms. Meta-learning schemes or online adjustment strategies driven by real-time heterogeneity estimation are promising solutions. For instance, monitoring the variance of label entropy across clients enables automatic tuning of feature correction strength, making FedCO a self-adaptive solution better suited for dynamic operating environments. Although the extra communication overhead introduced by label entropy and consistency statistics is marginal at approximately 0.01% per communication round, the server still needs to collect such scalar information from all participating clients. Such small overhead may become noticeable when accumulated in large-scale cross-device federated learning deployments with thousands of clients or limited network bandwidth. Replacing direct transmission of raw statistics with lightweight sketching techniques presents a viable improvement. Methods such as Count-Min sketches and probabilistic counting can compress entropy statistics in a sub-linear manner relative to the number of label classes. This approach retains the merits of adaptive aggregation while supporting deployment on massive client groups.

The current framework operates under the assumption that all clients behave honestly and report authentic label distributions. Malicious participants can manipulate label entropy values to bias global aggregation results, such as fabricating entropy data to gain higher aggregation weights. The vulnerability to Byzantine attacks remains unaddressed in this work. Follow-up studies may develop robust aggregation rules tailored for entropy-based statistics, including trimmed mean, median and Krum algorithms. Additionally, cryptographic techniques such as zero-knowledge proofs and commitment schemes can be adopted. These tools allow the server to verify the validity of reported label entropy without accessing raw client data, which delivers provable security against adversarial clients while preserving the core capability of mitigating label skew.

In summary, FedCO serves as a practical solution for heterogeneous federated learning. Its real-world deployment can be further enhanced via adaptive hyperparameter tuning, a further reduction of communication overhead, and strengthened resilience against malicious clients. The research directions discussed above are derived directly from the existing limitations and possess solid technical feasibility. We hope this discussion can inspire more studies to advance the robustness and scalability of heterogeneous federated learning systems.

Acknowledgement: Thank you very much for Dr. Donglin Zhu's technical support.

Funding Statement: This work is supported by the National Natural Science Foundation of China (Nos. 62272418, 62102058), Basic public welfare research program of Zhejiang Province (No. LGG18E050011), the Major Open Project of Key Laboratory for Advanced Design and Intelligent Computing of the Ministry of Education under grant ADIC2023ZD001.

Author Contributions: Study conception, Design, Conceptualization, Data curation, Writing—original draft, Methodology: Rui Wu; Supervision, Methodology, Writing—original draft, Writing—review & editing: Yehong Li; Resources, Software, Validation, Visualization: Hongjie Guo; Investigation, Formal analysis, Writing—original draft: Gangqiang Hu; Supervision, Conceptualization, Writing—review & editing, Project administration: Changjun Zhou; Data curation, Draft manuscript preparation: Qile Zou. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Data that support the findings of this study are available from the corresponding author upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Konečnỳ J. Federated learning: strategies for improving communication efficiency. arXiv:1610.05492. 2016.
2. Ouyang C, Mao J, Liu Z, Zhu D, Zhou C, Hu G, et al. Federated learning with dynamics-aware loss for label noise. Expert Syst Appl. 2026;312(11):131523. doi:10.1016/j.eswa.2026.131523.
3. Sun Y, Shen L, Huang T, Ding L, Tao D. Fedspeed: larger local interval, less communication round, and higher generalization accuracy. arXiv:2302.10429. 2023.
4. Li L, Fan Y, Tse M, Lin KY. A review of applications in federated learning. Comput Ind Eng. 2020;149(5):106854. doi:10.1016/j.cie.2020.106854.
5. Guo Y, Guo K, Cao X, Wu T, Chang Y. Out-of-distribution generalization of federated learning via implicit invariant relationships. In: Proceedings of the 40th International Conference on Machine Learning; 2023 Jul 23–29; Honolulu, HI, USA. p. 11905–33.
6. Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V. Federated optimization in heterogeneous networks. Proc Mach Learn Syst. 2020;2:429–50. doi: 10.48550/arxiv.1812.06127.
7. Kim G, Kim J, Han B. Communication-efficient federated learning with accelerated client gradient. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2024 Jun 17–21; Seattle, WA, USA. p. 12385–94.
8. Karimireddy SP, Kale S, Mohri M, Reddi S, Stich S, Suresh AT. Scaffold: stochastic controlled averaging for federated learning. In: Proceedings of the 37th International Conference on Machine Learning; 2020 Jul 13–18; Vienna, Austria. p. 5132–43.
9. Li X, Jiang M, Zhang X, Kamp M, Dou Q. FedBN: federated learning on non-IID features via local batch normalization. arXiv:2102.07623. 2021.
10. Reddi S, Charles Z, Zaheer M, Garrett Z, Rush K, Konečnỳ J, et al. Adaptive federated optimization. arXiv:2003.00295. 2020.
11. Wang J, Liu Q, Liang H, Joshi G, Poor HV. Tackling the objective inconsistency problem in heterogeneous federated optimization. Adv Neural Inf Process Syst. 2020;33:7611–23. doi: 10.48550/arxiv.2007.07481.
12. Wang Y, Chen H, Heng Q, Hou W, Fan Y, Wu Z, et al. Freematch: self-adaptive thresholding for semi-supervised learning. arXiv:2205.07246. 2022.
13. Chen H, Tao R, Fan Y, Wang Y, Wang J, Schiele B, et al. Softmatch: addressing the quantity-quality trade-off in semi-supervised learning. arXiv:2301.10921. 2023.

14. Cao Q, Zhu Z, Lian Z, Zhang R, Li B, Xiong Y, et al. PPFL: a parameter behavior-driven plug-in personalization engine for federated learning. In: Koenig S, Jenkins C, Taylor ME, editors. Proceedings of the Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026; 2026 Jan 20–27; Singapore. Menlo Park, CA, USA: AAAI Press; 2026. p. 19898–906.
15. Chen W, Gu W, Peng L, Yan T, Cai R, Hao Z, et al. Horizontal and vertical federated causal structure learning via higher-order cumulants. In: Koenig S, Jenkins C, Taylor ME, editors. Proceedings of the Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026; 2026 Jan 20–27; Singapore. Menlo Park, CA, USA: AAAI Press; 2026. p. 20280–8.
16. Chen Y, Lu J, Cao S, Wang W, Li G, Wen G. FedCure: mitigating participation bias in semi-asynchronous federated learning with non-IID data. In: Koenig S, Jenkins C, Taylor ME, editors. Proceedings of the Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026; 2026 Jan 20–27; Singapore. Menlo Park, CA, USA: AAAI Press; 2026. p. 20418–26.
17. Zhang J, Liu X, Niu J, Tang S, Yang H, Wu X. Causality inspired federated learning for OOD generalization. In: Singh A, Fazel M, Hsu D, Lacoste-Julien S, Berkenkamp F, Maharaj T, et al., editors. Proceedings of the Forty-Second International Conference on Machine Learning, ICML 2025; 2025 Jul 13–19; Vancouver, BC, Canada. p. 75615–37.
18. Tang X, Yu H, Li Z, Li X. Multi-session budget optimization for forward auction-based federated learning. In: Singh A, Fazel M, Hsu D, Lacoste-Julien S, Berkenkamp F, Maharaj T, et al., editors. Proceedings of the Forty-Second International Conference on Machine Learning, ICML 2025; 2025 Jul 13–19; Vancouver, BC, Canada. p. 59005–16.
19. Agarwal A, Joshi G, Pileggi LT. FedECADO: a dynamical system model of federated learning. In: Singh A, Fazel M, Hsu D, Lacoste-Julien S, Berkenkamp F, Maharaj T, et al., editors. Proceedings of the Forty-Second International Conference on Machine Learning, ICML 2025; 2025 Jul 13–19; Vancouver, BC, Canada. p. 531–49.
20. Bienstock A, Kumar U, Polychroniadou A. DMM: distributed matrix mechanism for differentially-private federated learning based on constant-overhead linear secret resharing. In: Singh A, Fazel M, Hsu D, Lacoste-Julien S, Berkenkamp F, Maharaj T, et al., editors. Proceedings of the Forty-Second International Conference on Machine Learning, ICML 2025; 2025 Jul 13–19; Vancouver, BC, Canada. p. 4360–83.
21. Hou C, Wang M, Zhu Y, Lazar D, Fanti G. Private federated learning using preference-optimized synthetic data. In: Singh A, Fazel M, Hsu D, Lacoste-Julien S, Berkenkamp F, Maharaj T, et al., editors. Proceedings of the Forty-Second International Conference on Machine Learning, ICML 2025; 2025 Jul 13–19; Vancouver, BC, Canada. p. 24025–44.
22. Gao Z, Wu F, Gao W, Zhuang X. A new framework of swarm learning consolidating knowledge from multi-center non-IID data for medical image segmentation. IEEE Trans Med Imag. 2022;42(7):2118–29. doi:10.1109/tmi.2022.3220750.
23. Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images. Handb Syst Autoimmune Dis. 2009;1(4):32–3. doi:10.1016/j.tics.2007.09.004.
24. Ma Z, Cao A, Yang F, Wei X. Curriculum dataset distillation. arXiv:2405.09150. 2024.
25. Zhang J, Li Z, Li B. Federated learning with label distribution skew via logits calibration. arXiv:2209.00189. 2022.
26. Li Q, He B, Song D. Model-contrastive federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021 Jun 19–25; Nashville, TN, USA. p. 10713–22.
27. Li XC, Zhan DC. FedRS: federated learning with restricted softmax for label distribution non-IID data. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining; 2021 Aug 14–18; Virtual. p. 995–1005.
28. Mu X, Shen Y, Cheng K, Geng X, Fu J, Zhang T, et al. FedProc: prototypical contrastive federated learning on non-IID data. Future Gener Comput Syst. 2023;143:93–104.

29. Shi Y, Liang J, Zhang W, Tan VYF, Bai S. Towards understanding and mitigating dimensional collapse in heterogeneous federated learning. In: Proceedings of the 11th International Conference on Learning Representations; 2023 May 1–5; Kigali, Rwanda.
30. Cubuk ED, Zoph B, Mane D, Vasudevan V, Le QV. Autoaugment: learning augmentation policies from data. arXiv:1805.09501. 2018.
31. Jhunjhunwala D, Wang S, Joshi G. FedExP: speeding up federated averaging via extrapolation. In: Proceedings of the Eleventh International Conference on Learning Representations, ICLR 2023; 2023 May 1–5; Kigali, Rwanda.
32. Dai R, Yang X, Sun Y, Shen L, Tian X, Wang M, et al. Fedgamma: federated learning with global sharpness-aware minimization. IEEE Trans Neural Netw Learn Syst. 2023;35(12):17479–92.