International Journal of Innovative
Computing, Information and Control
Volume **21**, Number **6**, December **2025**

ICIC International ©2025 ISSN 1349-4198
pp. **1569–1582**

ADAPTIVE FEDERATED LEARNING FRAMEWORK WITH DISCREPANCY WEIGHT AGGREGATION

Jiaqi Wang[1], Xu Li[2], Chunlong Yao[1,*] and Yang Li[3]

[1]School of Information Science and Engineering
[2]Innovation and Entrepreneurship College
Dalian Polytechnic University
No. 1, Qinggong Yuan, Ganjingzi District, Dalian 116034, P. R. China
{ wangjiaqi; lixu }@dlpu.edu.cn; *Corresponding author: yaochunlong@dlpu.edu.cn

[3]Dalian Cloud Force Technologies Co., Ltd.
Building B, Unit 1, 6th Floor, No. 23, Honggang Road, Dalian 116023, P. R. China
Leon@cloudforce.cn

Received April 2025; revised July 2025

Abstract. *Although federated learning inherently preserves data privacy, statistical heterogeneity across clients often undermines the global model's generalization performance. To address this limitation, we propose FedADW, an adaptive federated learning framework featuring discrepancy weighted aggregation. At the client side, an adaptive local aggregation module dynamically fuses the received global model with local updates according to client specific objectives, thereby optimizing model initialization. Concurrently, during server side aggregation, dual weights derived from categories distribution discrepancy and data volume are applied to precisely capturing each client's data characteristics and effectively mitigating distributional gaps. Comprehensive experiments on four standard benchmark datasets demonstrate that FedADW consistently outperforms existing baselines in accuracy, validating its enhanced effectiveness and robustness.*

Keywords: Federated learning, Adaptive local aggregation, Discrepancy weight aggregation, Category distribution discrepancy, Data volume

1. **Introduction.** Federated learning (FL), as a privacy preserving distributed machine learning paradigm [1], demonstrates significant potential in scenarios such as mobile device intelligent input and cross institutional medical analysis through its collaborative mechanism of client side local training and server side model aggregation. Its core principle lies in clients sharing only model parameters rather than raw data, which not only complies with privacy regulations but also overcomes data silo limitations [2].

In FL, the not independent and identically distributed (Non-IID) nature of client data manifests in multiple dimensions [3]. Among these, heterogeneity in category distribution is particularly critical, as such discrepancies lead to divergent local model optimization directions, making it challenging for traditional data volume weighted averaging aggregation strategies to converge to a robust global model. Theoretical analysis demonstrates that when client class distributions significantly deviate from the globally assumed distribution, aggregation relying solely on data volume weighting causes substantial model performance degradation.

Existing solutions primarily follow two approaches: local model adjustment at the client side and global model optimization at the server side. Local model adjustment refers to the process by which each client independently trains a model using its local data. Specifically,

DOI: 10.24507/ijicic.21.06.1569

each client starts with the global model and performs several rounds of training using its local data to update its own model. During training, clients typically employ gradient descent methods to optimize the model by minimizing the loss function on the local data. Upon completion of the local training, the clients send the updated models or gradients to the server for global model optimization. Global model optimization involves aggregating the model updates received from all clients. A common aggregation method is federated averaging, in which the server computes a weighted average of the clients' model updates based on the volume of their data, thereby generating a new global model. This new model is then redistributed to the clients as the starting point for the next training round. In this way, federated learning enables the collaborative training of a global model across multiple distributed devices while ensuring data privacy and local data security [4, 5].

This paper proposes an adaptive federated learning framework with discrepancy weighted aggregation to improve efficiency and adaptability. The framework introduces two key innovations: fine grained parameter fusion on the client side and an aggregation strategy based on category distribution discrepancy. First, an adaptive local aggregation module fuses global and local model parameters using an element wise weighted matrix, with learnable parameter selection in higher level networks [6, 7]. This preserves generalizable features while reducing interference. Second, a dual-weighting strategy leverages KL divergence to quantify distribution shifts, assigning higher weights to well-balanced clients with sufficient data. These modular enhancements strengthen the framework's effectiveness.

This paper makes three key contributions. First, it establishes a theoretical link between category distribution and aggregation weights, providing an interpretable basis for weight allocation. Second, it introduces a dual level optimization mechanism with category-aware aggregation, balancing model personalization and global convergence. Third, extensive experiments demonstrate the effectiveness of the proposed approach in diverse heterogeneous settings.

2. **Related Work.** Despite its advantages, FL faces two key challenges in practice: statistical heterogeneity and category imbalance [8]. Non-IID data distributions and class imbalances across clients hinder both generalization and personalization in traditional FL methods [9].

To address these challenges, researchers have recently proposed various personalized federated learning (PFL) approaches [10]. The central innovation of these methods lies in their client-centric designs that enhance local model adaptability through specialized model generation strategies and personalized aggregation mechanisms. However, existing PFL methods exhibit a fundamental limitation: they primarily focus on personalized optimization of local models while overlooking the necessity of personalized design in the global model aggregation phase. Specifically, during global model updates, these approaches predominantly adopt simplistic category-agnostic weighted averaging strategies based solely on client data volume, failing to incorporate category distribution characteristics that better reflect intrinsic data patterns.

Since the introductory work of McMahan et al. [1] on FedAvg in 2017, federated learning has been applied in numerous domains. However, prior empirical studies have revealed limitations of standard aggregation schemes when faced with statistical heterogeneity. To this end, several enhancements have been proposed. Yuan and Li [11] introduced FedProx, which incorporates a proximal regularizer to stabilize local updates; Wang et al. [12] developed FedMA, employing layer-wise matching to align and merge client models into a global network. While these methods yield improvements under particular conditions, they do not fully mitigate the adverse effects of Non-IID data.

In response, personalized federated learning techniques have emerged [13, 14], which can be grouped into three main categories. First, fine-tuning approaches adapt a global model locally (e.g., Per-FedAvg [15], and FedREP [16]). Second, client-specific model methods train additional personal networks alongside the global model (e.g., pFedMe [17], and Ditto [18]). Third, personalized aggregation strategies generate customized local models by weighting or selecting client updates (e.g., FedAMP [19], FedPHP [20], FedFomo [21], and APPLE [22]).

Concurrently, model adaptation on both client and server sides has been investigated to further alleviate distributional divergence. On the client side, dynamic regularization (FedDyn [23]), gradient-correction mechanisms (FedDC [24]), and contrastive feature alignment (MOON [25]) have been proposed. Although these methods refine objective functions, they continue to employ aggregation weights based primarily on data volume and do not adequately account for client-level distribution characteristics. On the server side, reweighting schemes such as FedNova [5], post-aggregation calibration (CCVR [26]), momentum-enhanced updates (FedAvgM [27]), and knowledge-distillation-based refinement (FedDF [28]) have been introduced. Nevertheless, these approaches often incur substantial computational overhead on the central server.

However, existing approaches demonstrate significant shortcomings in their weighting methodologies, being constrained either by reliance on simplistic data volume measures or by inadequate consideration of the fundamental distributional disparities between client data and the global model. Our objective is to assign greater participation weights to those clients whose category distributions deviate most substantially from the global distribution and thus contribute the highest information gain to the global model. The KL divergence perfectly aligns with this requirement. It both quantifies distributional divergence and inherently amplifies the influence of minority categories, thereby guiding server side aggregation decisions with precision within the dual-weighting mechanism. In contrast, alternative metrics – such as Euclidean distance, cosine similarity, or Jensen-Shannon divergence – neglect distributional sparsity and information content, exhibit weaker discriminative power for sparse distributions, incur higher computational complexity, and provide less pronounced amplification of rare categories, making them less suitable than KL divergence for this purpose.

Our proposed method adopts a dual level optimization strategy. At the client level, customized local models are generated through personalized aggregation mechanisms; At the server level, a hybrid weighting mechanism combines dataset size with KL divergence. This mechanism continuously evaluates each client's potential contribution to the global model and preferentially schedules, in each round, those clients that best fill the current global distributional gaps. Its hybrid weighting scheme ensures that large sample clients are fully leveraged while minority category samples remain impactful, with weights updated dynamically throughout training. Unlike personalized FL algorithms that focus exclusively on server side global initialization, aggregate weights solely by sample size, or impose regularization only at the client side. FedADW employs a dual level, adaptive, distribution aware update strategy to significantly enhance global model performance.

3. **Method.**

3.1. **Problem statement.**

In FL framework, multiple clients perform training in parallel. Each client iteratively updates its local model parameters through several training iterations on its local data. These local parameter updates are then aggregated at the server via a specific aggregation strategy [1]. Conventionally, most federated learning algorithms employ a data volume weighted averaging mechanism to update the global model

parameters. The standard aggregation strategy can be formalized as follows:

$$\theta_t = \sum_{k=1}^{K} \frac{n_k}{N} \theta_k \tag{1}$$

θ_k denote the model parameters of the k client, n_k the local sample size of client k, N the total sample size across all clients, and θ_t the global model at the server.

The conventional strategy weights clients proportionally to their sample sizes, granting clients with larger datasets greater influence in global model updates. While effective under IID data conditions where balanced client distributions enable efficient model fusion and rapid convergence to global optima.

However, it exhibits critical limitations when handling Non-IID data: aggregation challenges in Non-IID environments, data distribution discrepancies, and client resource heterogeneity.

3.2. **Overview of the FedDWA framework.** To address the prevalent statistical heterogeneity issue in federated learning, where the Non-IID characteristics of client data distributions severely constrain the generalization performance of global models on clients, we propose a category distribution driven adaptive federated learning mechanism.

As illustrated in Figure 1, the FL iterative process commences with the server distributing current global model parameters to participating clients. Subsequently, each client performs local training based on its private data to generate model parameter updates while simultaneously computing a category discrepancy metric d_k that reflects data distribution characteristics both of which are transmitted back to the server. During this process, as depicted in Figure 2, each client generates a weighting coefficient matrix W_i through local training, which adaptively integrates local features with global knowledge by operating on the global model parameters through Hadamard product. Ultimately, the server employs a discrepancy weighting aggregation strategy based on d_k to perform weighted aggregation of the integrated local models, thereby generating an updated global model to initiate subsequent iterations.

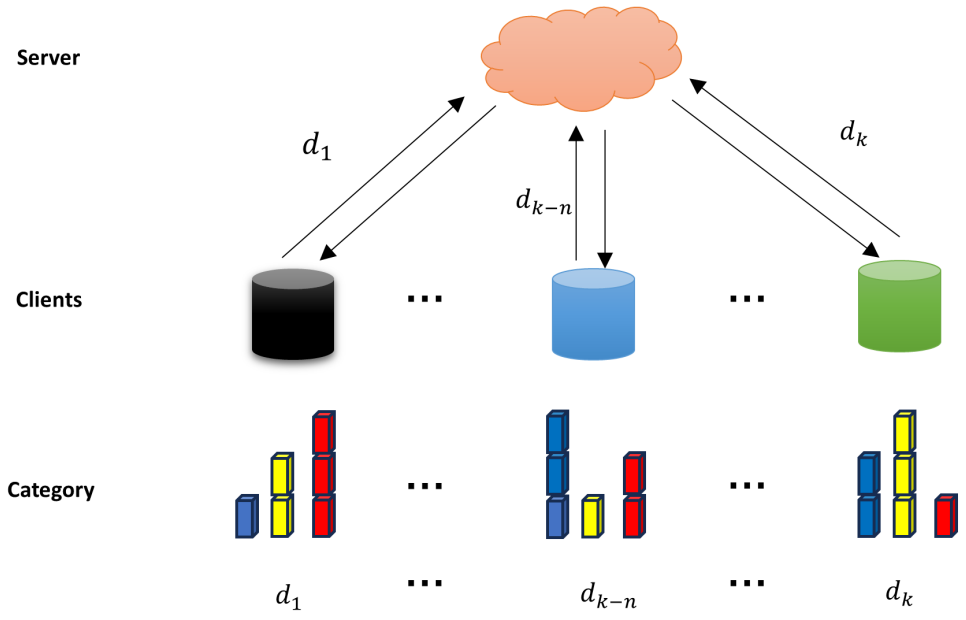


FIGURE 1. Workflow of the adaptive federated learning framework with discrepancy weighted aggregation

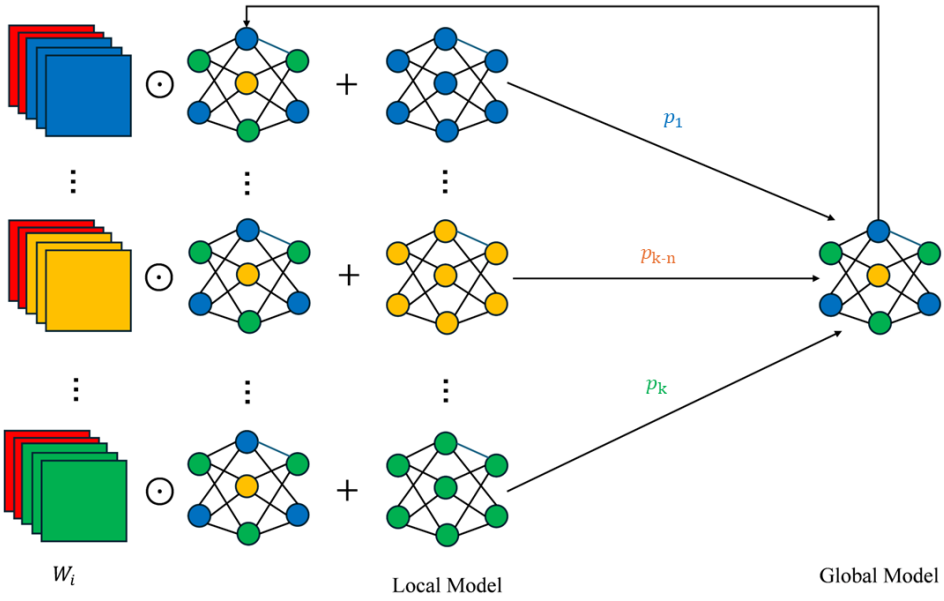


FIGURE 2. Model parameter aggregation process under adaptive fusion and discrepancy weighted aggregation mechanisms

3.3. **Theoretical analysis.** Motivated by the degradation of FL performance under Non-IID heterogeneity, we introduce an adaptive aggregation framework that weights each client by both its relative sample size and its distributional discrepancy. By down weighting clients whose category distribution deviates substantially from the global distribution, our method attenuates the skew induced bias present in conventional FL.

Under the standard FL assumptions including L-smoothness of the objective function, bounded gradients, unbiased stochastic gradients, and bounded dissimilarity, we extend the existing theory by introducing a bounded distributional discrepancy hypothesis. Here, n_k denotes the fraction of samples at client k, p_k represents its aggregation weight, and d_k measures the divergence between client k label distribution and the global distribution. In the multi-round local update analysis, the following major error components can be identified.

Local update drift (S_0): When each client performs τ local SGD steps with learning rate η, the parameters drift cumulatively from the global optimum. Algebraically, this drift aggregates to $T = 2\tau(\tau - 1)\eta^2 L^2$ which behaves like an "error budget" reducing the ideal descent factor of unity to

$$S_0 = 1 - 3T - P_D(1 - T) \tag{2}$$

Larger τ or η amplifies drift, shrinking the headroom S_0 for true descent. If clients each run 10 steps instead of 1, then T increases roughly by factor τ^2, so the global model may "overshoot" the optimal direction.

Global descent budget (S_1): Given an initial optimality gap $\Delta_0 = F\left(w^{(0,0)}\right) - F^{\infty}$, the effective descent is scaled by $(1 - T)$. Thus,

$$S_1 = \frac{2(1 - T)\Delta_0}{\tau\eta T} \tag{3}$$

This term quantifies how much of the original "budget" Δ_0 remains after drift T. With small T, $S_1 \approx 2\Delta_0/(\tau\eta T)$, so more local steps (large τ) actually slow convergence.

Distribution induced bias (S_2): A mismatch between sample weights n_k and aggregation weights p_k injects bias. Defining $P_D = 2\sum_{k=1}^{K}(n_k - p_k)^2$, and combining with the discrepancy bound B, we obtain

$$S_2 = (1-T)P_D B \sum_{k=1}^{K} d_k \tag{4}$$

When p_k deviates from the true data proportion n_k skewed clients pull the global model off course in proportion to their distributional gap d_k. If one client holds 50% of data but is weighted only 10%, then its strong influence on certain labels is under represented, raising S_2.

Stochastic noise amplification (S_3): Local SGD at each client introduces gradient noise variance σ^2, which upon aggregation yields

$$S_3 = 2(1-T)L\eta\sigma^2 \sum_{k=1}^{K} p_k^2 \tag{5}$$

higher learning rates η and uneven p_k magnify sampling noise in the global update. If one client's weight p_k doubles, its noise contribution quadruples, slowing convergence.

Accumulated local noise (S_4): Within each local trajectory of τ SGD steps, noise accumulates approximately as

$$S_4 = 2\left(\tau - 1\right)\sigma^2 L^2 \eta^2 \tag{6}$$

More local iterations permit greater stochastic fluctuation before synchronization. Doubling τ roughly doubles S_4, adding bias to the global solution.

Category imbalance scaling (S_5): Even with balanced sample sizes, label distribution heterogeneity distorts gradient estimation. We derive

$$S_5 = 2TB \sum_{k=1}^{K} p_k d_k \tag{7}$$

Clients whose labels are over or under represented (large d_k) should be down weighted (p_k smaller) to reduce this term. If a client's class distribution is orthogonal to the global one (d_k maximal), setting $p_k = 0$ eliminates its adverse influence.

Putting these together, the expected squared norm of the global gradient obeys

$$\min \mathbb{E}\left\|\nabla F\left(w^{(t,0)}\right)\right\|^2 \leq \frac{S_1 + S_2 + S_3 + S_4 + S_5}{S_0} \tag{8}$$

This bound makes explicit how local update drift (T), sampling noise (σ^2), and distributional discrepancy (d_k) interact through the aggregation weights p_k. Increasing d_k without adjusting p_k simultaneously lowers S_0 and raises S_2 and S_5, worsening the bound. Conversely, choosing p_k negatively correlated with d_k (down weighting skewed clients) directly reduces the dominant error terms S_2 and S_5.

3.4. **Adaptive local aggregation and discrepancy weight design.** Client: When the central server distributes the global model to clients, rather than simply overwriting the previous local model with the global model as in conventional federated learning, it performs element-wise aggregation between the global and local models. This client-specific adaptive aggregation is achieved through a weighted Hadamard product, formulated as

$$\hat{Q}_i^t = Q_i^{t-1} \odot W_{i,1} + Q^{t-1} \odot W_{i,2} \tag{9}$$

In FL, handling such constrained weighting coefficient matrices is challenging. Since gradient descent is employed during training, constrained optimization problems typically encounter non-smoothness issues and constraint enforcement difficulties. The constraints may render the objective function non-smooth as they enforce specific relationships among parameters, potentially causing gradient updates to become unstable or discontinuous at these boundaries. Directly updating the weighting coefficients may prevent effective weight adjustment under constraints. Therefore, more sophisticated mechanisms must be designed to ensure weight updates comply with constraints. Consequently, the aggregation process is reformulated as

$$\hat{Q}_i^t = Q_i^{t-1} + \left(Q^{t-1} - Q_i^{t-1}\right) \odot W_i \tag{10}$$

All elements of the weight matrix W_i^p for client i are first initialized to one. At each communication round t, the client randomly samples $s\%$ of its local dataset D_i, denoted $D_i^{s,t}$. Holding both the global model parameters Q^{t-1} and the current local model $\hat{Q}_i^t$ fixed, it computes the gradient of the loss with respect to W_i^p: $\nabla_{W_i^p}\mathcal{L}\left(\hat{Q}_i^t, D_i^{s,t}; Q^{t-1}\right)$. The weight matrix is then updated by a gradient descent step with learning rate η: $W_i^p \leftarrow W_i^p - \eta\nabla_{W_i^p}\mathcal{L}\left(\hat{Q}_i^t, D_i^{s,t}; Q^{t-1}\right)$. Subsequently, each entry of W_i^p is clipped to lie in $[0, 1]$, and a projection operator is applied to enforcing the element-wise constraint $w_1^q + w_2^q = 1$. Throughout this procedure, all other trainable parameters – namely, the global model and the local model parameters – remain frozen. Finally, client i performs τ steps of local SGD on its full dataset D_i, to update $\hat{Q}_i^t$.

Furthermore, each client needs to perform a discrepancy calculation by evaluating the difference between its local category distribution and the assumed global category distribution. To promote fairness across categories and enhance the generalization ability of the global model, we assume that the global category distribution follows a uniform distribution. Under this assumption, clients can independently compute the discrepancy without sharing additional data, effectively preventing the leakage of category distribution information. Let the local category distribution be C_k and the global category distribution be I. Each client can determine its discrepancy d_k based on the difference between these distributions. We use KL divergence as the measurement function, and the formula for computing category distribution discrepancy is

$$d_k = \sum_{s=1}^{S} C_{k,s} \log \frac{I}{C_{k,s}} \tag{11}$$

Weight Update: The aggregation weights for clients are determined by their d_k and n_k. In the discrepancy weighting mechanism, each client achieves more effective weight allocation for local model aggregation based on both its local data size and d_k. The formulation is derived as follows:

$$p_k = \frac{n_k}{\sum_{j \in S} n_j} - x \cdot d_k + y \tag{12}$$

n_k is the data size of client k, S is the set of clients participating in training during the current communication round, and $\sum_{j \in S} n_j$ is the total data size of all participating clients. x and y are hyperparameters used to adjust the aggregation weights.

Central Server: The central server aggregates the local model parameters uploaded by clients, assigning specific weights to each client during aggregation. Using a category distribution weighting mechanism, the global model aggregation is formulated as follows:

$$Q^{(t+1)} = \sum_{k=1}^{K} p_k Q_k^t \tag{13}$$

Algorithm 1 FedADW

Input: N clients, ρ (client joining ratio), L (loss function), Q^0 (initial global model), η (learning rate), $s\%$ (the percent of local data), x (balancing dataset size and discrepancy across clients), y (adjusting the aggregation weight)

Output: Reasonable local models $\hat{Q}_1, \ldots, \hat{Q}_N$

Server sends Q^0 to all clients to initialize local models.
Clients initialize $W_i^p \leftarrow W_i^p - \eta \nabla_{W_i^p} \mathcal{L}\left(\hat{Q}_i^t, D_i^{s,t}; Q^{t-1}\right)$, $\forall i \in [N]$, to ones.
for $t = 1, \ldots, T$ **do**
  Server samples a subset L^t of clients according to ρ.
  Server sends Q^{t-1} to each client in L^t.
  for each client $i \in L^t$ **in parallel do**
    Client i samples $s\%$ of its local data.
    if $t = 2$ **then**
      while W_i^P does not converge **do**
        Client i trains W_i^P.
      end while
    else
      Client i trains W_i^P.
    end if
    Client i obtains $\hat{Q}_i^t$.
    $Q_i^t \leftarrow \hat{Q}_i^t - \alpha \nabla_{\hat{Q}_i} L\left(\hat{Q}_i^t, D_i; Q^{t-1}\right)$.
    Client i calculates the local discrepancy d_k using Equation (11).
    Client i sends Q_i^t and d_k to the server.
  end for
  Server calculates the aggregation weights using Equation (12).
  Server obtains Q^t using Equation (13).
end for
return $\hat{Q}_1, \ldots, \hat{Q}_N$.

4. Experiments.

4.1. Experimental setup.

In our experiments, we selected four datasets: CIFAR-10, Fashion-MNIST, TINY, and AG News, which were distributed across $N = 20$ clients. All methods were implemented in PyTorch 1.8 and executed on an NVIDIA GTX 1050 GPU. Each experiment was run for 1500 iterations to ensure that all approaches empirically reached convergence.

We designed two heterogeneous data distribution scenarios for dataset configuration: pathological label skew setting and practical label skew setting. The pathological label skew setting simulates the most severe Non-IID scenario with maximum distribution divergence among clients. The practical label skew setting better reflects real world data distribution characteristics, where clients exhibit moderate but non-extreme distribution differences. Specifically, the practical label skew setting employs Dirichlet distribution ($\beta = 0.1$) for data partitioning [28].

As shown in Table 1, different model architectures were chosen for different datasets: a 4-layer CNN was used for FMNIST, CIFAR-10, and TINY [1], while FastText was adopted for AG News [29, 30, 31, 32]. To further validate the method's effectiveness, we additionally introduced ResNet18 for comparative experiments on TINY* [21]. The personalization layer depth was set to $p = 2$, allowing the model to start personalization from relatively deep (but not the deepest) layers, thereby maintaining global information while enabling local adaptation at clients. We randomly selected 80% of the data ($s = 80$) for training, with a batch size of 10 and local training epochs set to 1.

Table 1. Parameter configuration for experiments

N	β	Model	p	s	batch_size	local_steps	x	y	local_learning_rate
20	0.1	CNN	2	80	10	1	0.5	0.1	0.005
20	0.1	ResNet18	2	80	10	1	0.5	0.1	0.1
20	0.1	FastText	2	80	10	1	0.5	0.1	0.1

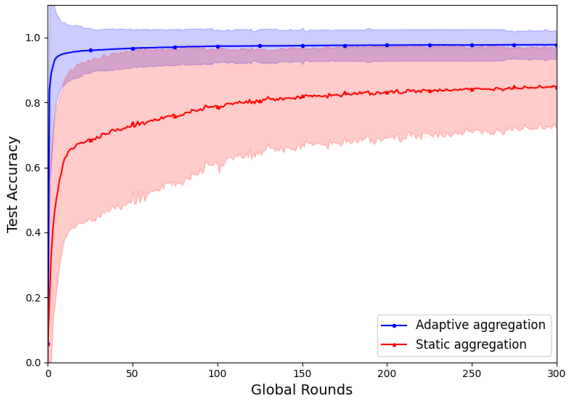
4.2. **Main results.** Through optimization of the global aggregation phase, we have mitigated the challenges posed by Non-IID data distribution and data imbalance. In the local training phase, an adaptive strategy is employed, enabling each client to make effective adjustments according to its data characteristics, thereby enhancing the model's fitting capability on local data. The models generated by individual clients not only better align with their respective data distributions but also demonstrate improved quality when facing significant data distribution biases, providing high-quality updates for global aggregation. Simultaneously, we introduce category distribution discrepancy in the global aggregation phase, calculating aggregation weights for each client to balance both data quantity and distribution consistency. This combined approach makes the global aggregation process preferentially incorporate client models that possess both sufficient data volume and reasonable data distribution. Consequently, the overall model achieves significant improvements in both accuracy and generalization capability, as demonstrated in Table 2.

Table 2. The test accuracy of the adaptive federated learning framework with discrepancy weighted aggregation under two heterogeneous settings

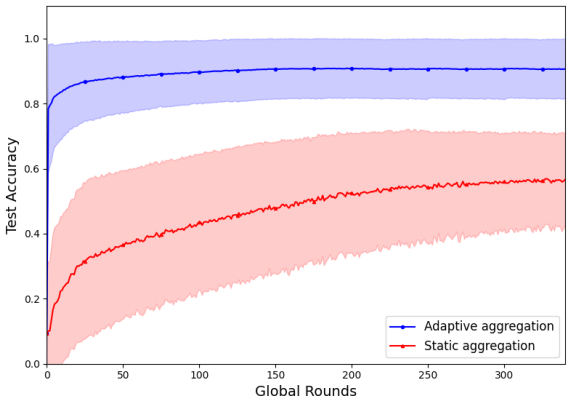
Method	Pathological label skew setting			Practical label skew setting				
	FMNIST	CIFAR-10	TINY	FMNIST	CIFAR-10	TINY	TINY*	AG News
FedAvg	80.41 ± 0.08	55.09 ± 0.83	14.20 ± 0.47	85.85 ± 0.19	59.16 ± 0.47	19.46 ± 0.20	19.45 ± 0.13	79.57 ± 0.17
FedProx	78.08 ± 0.15	55.06 ± 0.75	13.85 ± 0.25	85.63 ± 0.57	59.21 ± 0.40	19.37 ± 0.22	19.27 ± 0.23	79.35 ± 0.23
FedTGP	90.18 ± 0.06	90.02 ± 0.10	34.56 ± 0.42	87.78 ± 0.10	88.15 ± 0.05	27.37 ± 0.19	28.92 ± 0.32	88.45 ± 0.25
FedDual	83.21 ± 0.07	50.13 ± 0.11	30.09 ± 0.44	81.99 ± 0.17	48.70 ± 0.05	28.46 ± 0.21	29.15 ± 0.25	81.78 ± 0.22
FedFomo	99.46 ± 0.01	91.85 ± 0.02	36.55 ± 0.50	97.21 ± 0.02	88.06 ± 0.02	26.33 ± 0.22	26.84 ± 0.11	95.84 ± 0.15
pFedMe	99.35 ± 0.14	90.11 ± 0.10	27.71 ± 0.40	97.25 ± 0.17	88.09 ± 0.32	26.93 ± 0.19	33.44 ± 0.43	91.41 ± 0.22
APPLE	99.30 ± 0.01	90.97 ± 0.05	36.22 ± 0.40	97.06 ± 0.07	89.37 ± 0.11	35.04 ± 0.47	39.93 ± 0.52	95.63 ± 0.21
FedAMP	99.42 ± 0.03	90.79 ± 0.16	36.12 ± 0.30	97.20 ± 0.06	88.70 ± 0.18	27.99 ± 0.11	29.11 ± 0.15	94.18 ± 0.09
FedADW	$\mathbf{99.57 \pm 0.01}$	$\mathbf{91.32 \pm 0.02}$	$\mathbf{41.02 \pm 0.30}$	$\mathbf{97.74 \pm 0.02}$	$\mathbf{90.83 \pm 0.03}$	$\mathbf{44.09 \pm 0.02}$	$\mathbf{45.49 \pm 0.05}$	$\mathbf{97.13 \pm 0.08}$

4.3. **The impact of aggregation mechanisms and adaptability on the framework.**

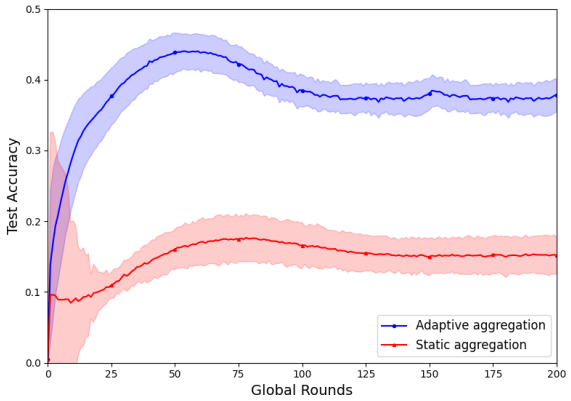
4.3.1. *The impact of adaptability on the framework.* This study examines the impact of adaptive mechanisms on federated learning performance by comparing adaptive aggregation, which dynamically adjusts the aggregation strategy based on real-time model performance, and static aggregation, which follows a fixed preset rule. Experiments on



(a) FMNIST



(b) CIFAR-10



(c) TINY

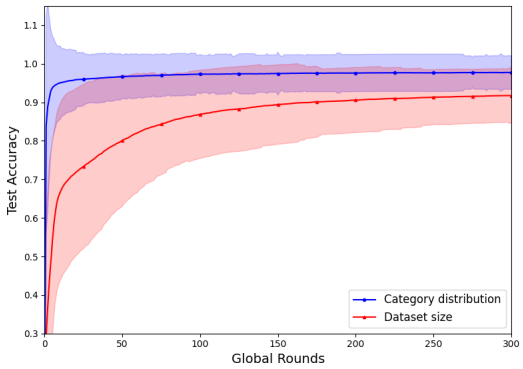
FIGURE 3. Accuracy comparison of adaptive and static aggregation strategies

FMNIST, CIFAR-10, and TINY maintained identical network architectures and hyperparameters, with only the adaptability of the aggregation strategy varied. As shown in Figure 3, results indicate that adaptive aggregation achieves higher accuracy than the static approach.

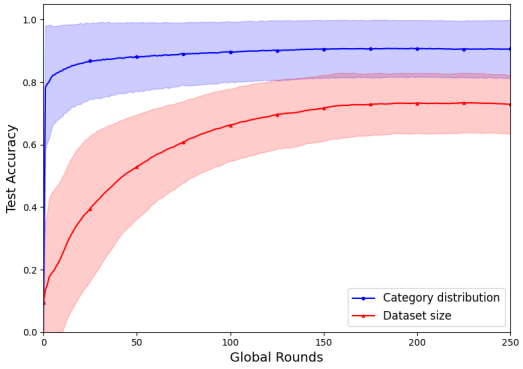
4.3.2. *The impact of aggregation mechanisms on the framework.* This study compares category distribution based aggregation, which adjusts weights based on client category distribution differences, and data size based aggregation, which assigns weights by data volume. Experiments on FMNIST, CIFAR-10, and TINY controlled all conditions except the aggregation strategy. As shown in Figure 4, the category distribution based method consistently outperforms conventional approaches.

4.4. **Effect of hyperparameters.** The parameter x controls the penalty on clients with deviating data distributions. When x is too low, weak penalties allow highly divergent clients to retain high weights, reducing accuracy. When x is too high, excessive penalties diminish their contributions, also lowering accuracy. The optimal balance occurs at $x = 0.5$, where penalties effectively account for deviations without overly suppressing local models.

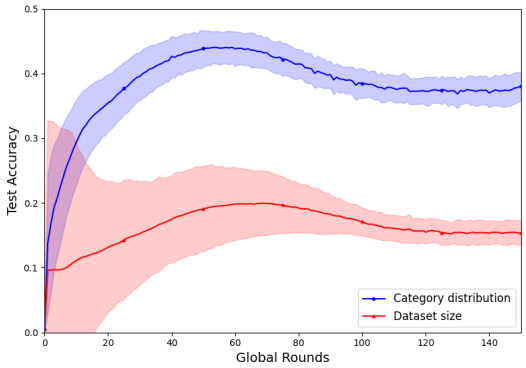
The offset y ensures all clients retain a minimum weight. A small y lets penalties dominate, potentially excluding some clients, while a large y counteracts penalties too much, allowing misaligned clients excessive influence. The best balance is achieved at $y = 0.1$, maintaining meaningful client contributions without weakening the penalty effect.



(a) FMNIST



(b) CIFAR-10



(c) TINY

FIGURE 4. Comparison of aggregation strategies showing the superiority of category distribution based aggregation

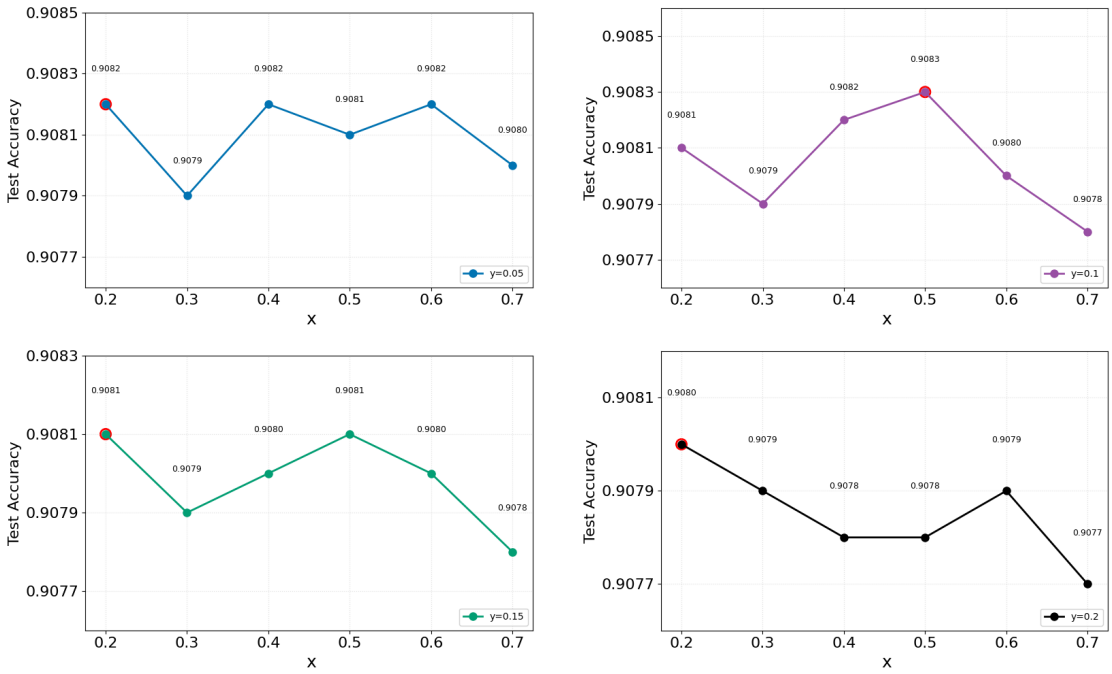


FIGURE 5. Analysis of parameters x and y effects in weight adjustment

Thus, $x = 0.5$ and $y = 0.1$ provide the best trade-off between penalizing distribution deviations and preserving valuable local model contributions, as demonstrated in Figure 5.

5. **Conclusions.** This paper proposes a novel FL framework that tackles data heterogeneity through adaptive local aggregation and discrepancy weight aggregation. By considering both data volume and category distribution, it achieves a more balanced global model. Experiments show superior performance in heterogeneous settings, surpassing existing methods. Future work will explore its application to other heterogeneity challenges to further advance federated learning.

Acknowledgment. This work was supported by the Scientific Research Fund for the Higher Education Institutions of Liaoning Province of China under Grant LJ2124101520 70.

REFERENCES

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

[2] J. Zhang, Y. Liu, Y. Hua and J. Cao, FedTGP: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning, *Proceedings of the AAAI Conference on Artificial Intelligence*, vol.38, no.15, pp.16768-16776, 2024.

[3] P. Sahoo, A. Tripathi, S. Saha and S. Mondal, FedDUAL: A dual-strategy with adaptive loss and dynamic aggregation for mitigating data heterogeneity in federated learning, *arXiv Preprint*, arXiv: 2412.04416, 2024.

[4] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie and R. Pedarsani, FedPAQ: A communication-efficient federated learning method with periodic averaging and quantization, *International Conference on Artificial Intelligence and Statistics*, pp.2021-2031, 2020.

[5] 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.

[6] J. Yosinski, J. Clune, Y. Bengio and H. Lipson, How transferable are features in deep neural networks?, *Advances in Neural Information Processing Systems*, vol.27, 2014.

[7] Y. LeCun, Y. Bengio and G. Hinton, Deep learning, *Nature*, vol.521, no.7553, pp.436-444, 2015.

[8] Y. Ding, C. Niu, F. Wu, S. Tang, C. Lyu, G. Chen et al., Federated submodel optimization for hot and cold data features, *Advances in Neural Information Processing Systems*, vol.35, pp.1-13, 2022.

[9] J. Wang, Z. Charles, Z. Xu, G. Joshi, H. B. McMahan, M. Al-Shedivat, G. Andrew, S. Avestimehr, K. Daly, D. Data et al., A field guide to federated optimization, *arXiv Preprint*, arXiv: 2107.06917, 2021.

[10] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings et al., *Advances and Open Problems in Federated Learning*, Now Foundations and Trends, 2021.

[11] X. Yuan and P. Li, On convergence of FedProx: Local dissimilarity invariant bounds, non-smoothness and beyond, *Advances in Neural Information Processing Systems*, vol.35, pp.10752-10765, 2022.

[12] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos and Y. Khazaeni, Federated learning with matched averaging, *arXiv Preprint*, arXiv: 2002.06440, 2020.

[13] L. Albshaier, S. Almarri and A. Albuali, Federated learning for cloud and edge security: A systematic review of challenges and AI opportunities, *Electronics*, vol.14, no.5, DOI: 10.3390/electronics 14051019, 2025.

[14] D. Thakur, A. Guzzo, G. Fortino and F. Piccialli, Green federated learning: A new era of green aware AI, *ACM Computing Surveys*, 2025.

[15] A. Fallah, A. Mokhtari and A. Ozdaglar, Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach, *Advances in Neural Information Processing Systems*, vol.33, pp.3557-3568, 2020.

[16] M. A. Husnoo, A. Anwar, N. Hosseinzadeh, S. N. Islam, A. N. Mahmood and R. Doss, FedREP: Towards horizontal federated load forecasting for retail energy providers, *2022 IEEE PES 14th Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pp.1-6, 2022.

[17] C. T. Dinh, N. Tran and J. Nguyen, Personalized federated learning with Moreau envelopes, *Advances in Neural Information Processing Systems*, vol.33, pp.21394-21405, 2020.
[18] C. He, S. Li, J. So, X. Zeng, M. Zhang, H. Wang, X. Wang, P. Vepakomma, A. Singh, H. Qiu et al., FedML: A research library and benchmark for federated machine learning, *arXiv Preprint*, arXiv: 2007.13518, 2020.
[19] Y. Huang, L. Chu, Z. Zhou, L. Wang, J. Liu, J. Pei and Y. Zhang, Personalized cross-silo federated learning on Non-IID data, *Proceedings of the AAAI Conference on Artificial Intelligence*, vol.35, no.9, pp.7865-7873, 2021.
[20] X.-C. Li, D.-C. Zhan, Y. Shao, B. Li and S. Song, FedPHP: Federated personalization with inherited private models, *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp.587-602, 2021.
[21] K. He, X. Zhang, S. Ren and J. Sun, Deep residual learning for image recognition, *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770-778, 2016.
[22] Y. Wu, Y. Kang, J. Luo, Y. He and Q. Yang, FedCG: Leverage conditional GAN for protecting privacy and maintaining competitive performance in federated learning, *arXiv Preprint*, arXiv: 2111.08211, 2021.
[23] C. Jin, X. Chen, Y. Gu and Q. Li, FedDyn: A dynamic and efficient federated distillation approach on recommender system, *2022 IEEE 28th International Conference on Parallel and Distributed Systems (ICPADS)*, pp.786-793, 2023.
[24] L. Gao, H. Fu, L. Li, Y. Chen, M. Xu and C.-Z. Xu, FedDC: Federated learning with Non-IID data via local drift decoupling and correction, *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.10112-10121, 2022.
[25] Q. Li, B. He and D. Song, Model-contrastive federated learning, *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.10713-10722, 2021.
[26] 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.
[27] 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.
[28] T. Lin, L. Kong, S. U. Stich and M. Jaggi, Ensemble distillation for robust model fusion in federated learning, *Advances in Neural Information Processing Systems*, vol.33, pp.2351-2363, 2020.
[29] X. Zhang, J. Zhao and Y. LeCun, Character-level convolutional networks for text classification, *Advances in Neural Information Processing Systems*, vol.28, 2015.
[30] Y. Liao, J. Geng, L. Guo, B. Geng, K. Cui and R. Li, Transfer learning rolling bearing fault diagnosis method based on deep domain adaptive network, *Information and Control*, vol.21, no.1, pp.209-225, 2025.
[31] S. Li and X. Liu, A study on detection algorithm of safety apparatus wearing by workers at heights based on deep learning, *International Journal of Innovative Computing, Information and Control*, vol.19, no.5, pp.1593-1603, 2023.
[32] W. Wang, J. Yu, Y. Ma, Z. Pan and T. Chen, Bearing fault diagnosis based on deep learning and array stochastic resonance under strong noise background, *International Journal of Innovative Computing, Information and Control*, vol.21, no.2, pp.549-563, 2025.

Author Biography

Jiaqi Wang received with a Bachelor's degree in Network Engineering from University of Jinan Quancheng College, China, in 2022. He is studying for a Master's degree in Dalian Polytechnic University, China. His main research interests include federated learning and artificial intelligence.

Xu Li received B.S. degree in Computer Science from University of Science and Technology Anshan, China, in 2003. She received M.E. and Ph.D. degrees in Computer Application Technology from Yanshan University, China, in 2006 and 2010, respectively. She is currently an associate professor in the Innovation and Entrepreneurship College, Dalian Polytechnic University, China. Her current research interests include natural language processing and deep learning.

Chunlong Yao received B.S. and M.S. degrees in Computer Science from Northeast Heavy Machinery Institute, China in 1994 and 1997, respectively. He received Ph.D. degree in Computer Software and Theory from Harbin Institute of Technology, China, in 2005. He is currently a professor in the School of Information Science and Engineering, Dalian Polytechnic University, China. His current research interests include data mining and intelligent information system.

Yang Li received his B.S. degree in Mathematics from University College London, UK, in 2010, followed by both M.S. and Ph.D. degrees in Computer Science from Imperial College London, UK, which he completed in 2016. His academic research primarily focuses on machine learning and its applications. In 2013, during his doctoral studies, Dr. Li founded Dalian Cloud Force Technologies Co., Ltd., China, where he leads initiatives in cloud computing and AI-driven enterprise solutions, aiming to bridge cutting-edge machine learning technologies with real-world industrial applications.