FAFI: Bridging Model Inconsistency in One-shot Federated Learning

Anonymous Authors[1]

Abstract

Turning the multi-round vanilla Federated Learning (FL) into one-shot FL (OFL) significantly reduces the communication burden and makes a big leap toward practical deployment. However, we note that existing OFL methods all focus on designing better merging with inconsistent local models, which naturally leads to poor performance. We empirically and theoretically demonstrate the occurrence of model inconsistencies on both intra-model and inter-model levels. In this paper, we present a novel OFL framework named FAFI, which uses Federated Self-Alignment Local Training and Informative Feature Fused Inference to overcome the above challenges. Specifically, the self-alignment local training which adopts self-supervised learning and learnable prototypes to ensure intra-model consistency, and the informative feature fused inference which fuses the knowledge of the local model while mitigating the negative impact of inconsistency. Extensive experiments on various datasets and models demonstrate that FAFI achieves state-of-the-art performance.

1. Introduction

As a distributed machine learning paradigm featured with privacy-preserving, Federated Learning (FL) enables multiple clients to collaboratively integrate their knowledge without exposing their local data (McMahan et al., 2017; Zeng et al., 2021; 2025; Karimireddy et al., 2020; Yurochkin et al., 2019). Basically, FL allows clients to train local models independently, collects the locally trained models for aggregation, and broadcasts the aggregated global model for iterative local training. However, such a multi-round client-server interaction process would incur a heavy communication burden (e.g., more than 250GB for a simple VGG19 model cumulatively (Zeng et al., 2024; Wu et al., 2020)) and high communication time (e.g., more than 194 hours for one-round transmission of GPT-3 (Tang et al., 2024)), criticized for being prohibitive in real-world applications (Zhang et al., 2022a; Dai et al., 2023; Chen et al., 2023; Tang et al., 2024; Liu et al., 2024).

To reduce the communication costs, one-shot Federated Learning (OFL) has emerged recently by compressing the multi-round communications of vanilla FL into just *one round* (Guha et al., 2019). In OFL, clients perform long-term local training individually based on their private data and upload these well-trained local models to the server for aggregation. In this manner, OFL is believed to be well-suited for the prevalent model market scenarios (Zhang et al., 2022a; Zeng et al., 2024), where users are willing to trade their models for next-stage knowledge fusion instead of joining into a redundant training process. In current OFL, clients are supposed to train locally for only one time, gaining one-shot local models, while the server are expected to provide a deliberate aggregation by reforming a global model with one-shot local ones.

Existing OFL aggregation designs focus on the server side and fall into three categories. Wherein, the optimization-based methods focus on designing dedicated averaging (McMahan et al., 2017; Zeng et al., 2021; Jhunjhunwala et al., 2024; Liu et al., 2024), clustering (Dennis et al., 2021), or solving Pareto optimum (Su et al., 2023) to tune the parameters; The distillation-based methods (Li et al., 2021b; Zhou et al., 2020; Heinbaugh et al., 2022; Yang et al., 2023) use the assumed-to-be-available auxiliary dataset to distill all local models into one; Selective ensemble methods (Zhang et al., 2022a; Dai et al., 2023; Zeng et al., 2024) assign proportion for each local model and use them for inference based on ensemble learning. Unfortunately, there is no free lunch for the communication cost reduction. The reported performance of OFL methods has a significant gap compared with multi-round FL, especially on heterogeneous data (Tang et al., 2024; Zeng et al., 2021; Gao et al., 2022).

This work identifies, for the first time, that existing OFL falls in a garbage (inferior one-shot local models) in and garbage (degraded global model) out pitfall. Vanilla FL overcomes such a pitfall by implicitly exchanging local knowledge via multi-round iterations while existing OFL

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

lacks such a measure by reactively gathering the garbage inputs. We further unravel the root cause of such 'garbage input' as two aspects of inconsistency in the face of data heterogeneity § 3.

(1) Intra-model inconsistency. A one-shot local model would have divergent predictions for samples of the same semantics. We owe this to

To unravel the underlying reasons for the ineffectiveness of existing OFL, we first observe that these methods manifests model inconsistencies at both intra-model level and inter-model level due to data heterogeneity (details in § 3). (1) For intra-model inconsistency, we observe that the locally trained model exhibits inconsistent behaviors (including inconsistent features and predictions) when presented with samples of the same semantics. (2) For inter-model inconsistency, we notice that local models from different clients manifests both semantical and statistical inconsistencies (including inconsistent features and parameters). Unfortunately, *you can't make a silk purse out of a sow's ear*, we note that existing OFL methods focus mostly on designing better model merging methods with the pre-trained inconsistent local models, which naturally leads to poor performance.

The OFL paradigm would be more effective if the one-shot is attained with consistent local models. For this purpose, this work presents the novel design of One-shot Federated Learning with Self Alignment and Informative Feature Fused Inference to address the model inconsistencies at both intra-model and inter-model levels. (1) To address intra-model inconsistency, our proposed strategy aims to ensure the local model behaves consistently by leveraging semantics as an implicit anchor. Specifically, we utilize self-supervised learning to learn invariant features, ensuring the extracted features are aligned with semantics. Then, we design learnable contrastive prototypes for semantically consistent classification boundaries, thereby guaranteeing the predictions are aligned with semantics. This Self-Alignment Local Training strategy is able to bridge the intra-model inconsistencies and improve the performance on client-specific data. (2) In response to the inter-model inconsistency issue, the crux lies in how to fuse the knowledge contained in inconsistent local models. Instead of directly fusing knowledge irrelevant model parameters, we propose to fuse the knowledge of local models by informatively fusing semantically consistent features during inference, thus alleviating the pitfalls of model-wise fusion. Meanwhile, we evaluate the information contained in the extracted features by assessing their similarity to noise features. Features with less information tend to have lower weights. This differentiated feature weighting reduces the interference of noise features in knowledge fusion, thereby enhancing the effectiveness and robustness of the knowledge fusion process. This Informative Feature Fused Inference strategy can alleviate the inter-model inconsistency while mitigating noise interruption. We believe these two components together make FAFI a competitive method for OFL. Our main contribution are summarized as follows:

- We focus on one-shot federated learning, empirically and theoretically demonstrating the occurrence of model inconsistencies on both intra-model and inter-model levels due to data heterogeneity, which naturally leads to poor performance for existing model merging designs.
- we propose a novel one-shot federated learning framework, obtaining more effective and consistent local models by leveraging client-side Self-Alignment and server-side Informative Feature Fused Inference.
- We conduct extensive evaluations on real datasets with various level of data heterogeneity. Accompanied by a set of ablative studies, promising results validate the efficacy of FAFI and indispensability of each module.

2. Related Work

2.1. One-shot Federated Learning

One-shot Federated Learning (OFL) is a variant of FL that requires only one round of interaction between the clients and server to reduce the heavy communication cost. Existing OFL methods focus on designing aggregation mechanisms on the server side, which can be categorized into parameter optimization-based and knowledge distillation-based methods.

Parameter Optimization-based methods focus on reconstructing a better global model with the parameters of local models. Traditional aggregation methods such as FedAvg (McMahan et al., 2017), Median (Yin et al., 2018), Krum (Blanchard et al., 2017), and FedCav (Zeng et al., 2021; 2025) can be directly applied in OFL, but achieve low performance. MA-Echo (Su et al., 2023) tries to get the Pareto optimum of the local clients via exploring common harmonized optima. FedFisher (Jhunjhunwala et al., 2024) and FedLPA (Liu et al., 2024) require additional Fisher information matrices for reaggregation. However, the nonlinear structure of DNNs makes it difficult to obtain a comparable global model through parameter optimization (Tang et al., 2024). Besides, directly analyzing the model parameters requires all local models to have the same architecture, whose setting is impractical in real-world heterogeneous scenarios.

Knowledge Distillation-based methods try to introduce the knowledge distillation (KD) to transfer the massive local knowledge into one global model. The local models (Guha et al., 2019; Li et al., 2021b; Diao et al., 2022) or locally distilled data (Zhou et al., 2020) are viewed as teachers and the newly constructed global model is the student. (Guha et al., 2019) firstly uses the ensemble prediction of local

models as the teachers' output. FedKT (Li et al., 2021b) designs a two-tier PATE structure relying on public data to improve the ensemble of local models. To alleviate the label skews, FedOV (Diao et al., 2022) adopts open-set voting in OFL to enhance the generalization. k-FED (Dennis et al., 2021) runs a variant of Lloyd's method for k-means clustering and obtains an aggregated model through one round iteration of exchanging local cluster means. Some dataset distillation-based one-shot methods, such as FedD3 (Song et al., 2023) and FedMD (Li & Wang, 2019), transmit the locally distilled dataset rather than models to the server in a one-shot manner. However, these methods require auxiliary public data or pre-trained models which may be impractical in privacy-sensitive scenarios, such as biomedical domains. To solve the scarcity problem of public data or models, some methods propose using existing powerful generative models to generate auxiliary samples for next-stage global model training. DENSE (Zhang et al., 2022a) firstly proposes to use GAN for data generation to assist the KD process. Co-Boosting (Dai et al., 2023) designs a mutually enhanced process to optimize high-quality samples and ensemble weights. FedCAVE (Heinbaugh et al., 2022) modifies the local learning task into training a conditional variation auto-encoder (CAVE) and uses KD to compress the ensemble into a powerful decoder. The decoder can be used to generate training samples for the global model. FedCADO (Yang et al., 2023) adopts the popular diffusion models to get the synthetic data. IntactOFL (Zeng et al., 2024) trains an MoE network by generative samples for better prediction. FedDEO (Yang et al., 2024b) trains local descriptions which serve as the medium for conditional generation with diffusion models. However, these generative data samples may leak the privacy of the local clients.

However, all these methods focus on improving performance through server-side design, ignoring that the root cause of low performance in OFL is the low-quality local models. In this work, we focus on client-side design by improving the quality of local models.

2.2. Prototype Learning

Prototype refers to the representative feature vector of the instances belonging to a specific class. It is a popular and effective method widely used in various tasks, such as supervised classification tasks and unsupervised learning. Since the prototypes can provide abstract knowledge while preserving data privacy, few works introduce prototypes into federated learning. FedProto (Tan et al., 2022) and FedProc (Mu et al., 2023) aim to achieve a feature-wise alignment with global prototypes. CCVR (Luo et al., 2021) generates virtual features based on approximated Gaussian Mixture Model (GMM). However, existing federated prototype learning methods rely on multi-round interactions between the clients and server, which is not practical in one-shot federated learning.

2.3. Contrastive Learning

Contrastive learning (CL) is a promising direction in self-supervised learning. Contrastive learning constructs positive and negative pairs for each training instance and designs various loss functions to contrast positiveness against negativeness, such as InfoNCE (Oord et al., 2018), and SupCon (Khosla et al., 2020). One branch of CL focuses on selecting the informative positive pairs and negative pairs (Chongjian et al., 2023). Another branch investigates the semantic structure and involves clustering methods to construct more representative prototypes (Caron et al., 2020; Li et al., 2021a). In this work, contrastive learning is used to obtain more representative features and prototypes, thus improving the quality of local models.

3. Motivation

Existing methods all focus on designing better aggregation methods to improve the OFL performance, ignoring the crucial role played by the local models. To this end, in this section, we empirically and theoretically demonstrate the occurrence of model inconsistencies on both intra-model levels and inter-model levels, which motivates us to consider a new form of aggregation framework in OFL.

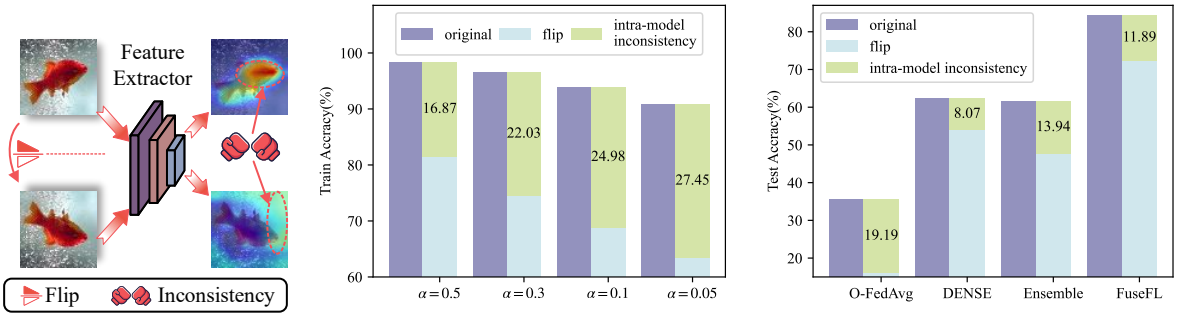
3.1. Intra-model Inconsistency

In this part, we first investigate the intra-model inconsistency in existing OFL local training strategies. As shown in Figure 1a, the local models exhibit significant inconsistencies in their prediction for identical samples when subjected to simple augmented transformations, such as flipping, suggesting that the local models fail to learn semantically consistent features.

We quantify this intra-model inconsistency by measuring the averaged performance discrepancy of the model on identical samples. As illustrated in Figure 1b, we notice that there exists a significant performance drop when the local model is evaluated on the flipped samples. And with the increased non-IID degree (lower α), the intra-model inconsistency becomes larger.

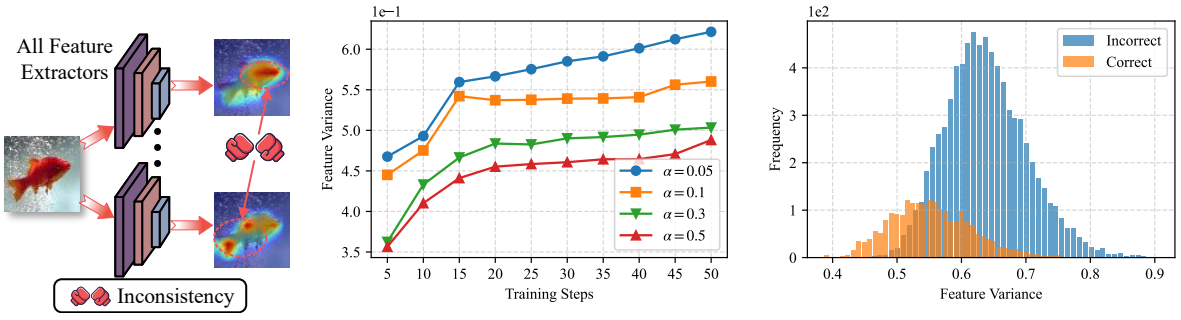
In addition, we further investigate the impact of intra-model inconsistency on existing methods. We test the global models aggregated by different methods on the original samples and flipped samples in Figure 1c. We observe that all these global model aggregated by existing methods still exhibit high intra-model inconsistency, indicating that the current OFL methods fail to alleviate the intra-model inconsistency.

In summary, we highlight the presence of intra-model inconsistency in existing OFL methods. We note that the current



(a) (b) (c)

Figure 1. (a) **Model exhibits divergent attention under identical semantic sample.** TODO...



(a) (b) (c)

Figure 2. X

training strategies' inability to learn consistent semantic features, leading to poor performance in aggregation.

3.2. Inter-model Inconsistency

Notably, inconsistency not only exists within individual models but also manifests significantly across models trained on different clients due to the heterogeneous nature of data distributions.

The existing federated learning community has long recognized the inconsistency brought by data heterogeneity, such as the concept of 'client drift' (Gao et al., 2022; Karimireddy et al., 2020). However, most of these studies focus on the impact of heterogeneous data on model parameters, overlooking the fundamental challenge of extracting consistent features. To investigate the inter-model inconsistency in OFL, we provide the analysis from the view of features.

As illustrated in Figure 2a, we visualize the features extracted by different local models for the same sample. We observe that the features extracted by different local models are significantly different, namely, the inter-model inconsistency.

Besides, we quantify the inter-model inconsistency by measuring the feature variance of local models, which is $\sqrt{\frac{1}{M}\sum_{i=1}^{M}(\mathcal{F}_i - \hat{\mathcal{F}})^2}$. The M is the number of clients, $\mathcal{F}_i$ is the feature extracted by the i-th client, and $\hat{\mathcal{F}}$ is the average feature. As shown in Figure 2b, the feature variance increases with the local training steps, indicating that the existing closed training paradigm cannot alleviate the inter-model inconsistency.

Furthermore, we dissect the inference process of existing methods to investigate the impact of feature inconsistency on performance. As presented in Figure 2c, we compute the feature variance for all correctly classified samples and incorrectly classified samples. We observe that correctly classified samples statistically exhibit smaller feature variance compared to incorrectly classified samples, suggesting that more consistent features may contribute to the correct classification by the model.

In this part, we present the inter-model inconsistency in existing OFL methods. We note that existing OFL local training paradigms lead to significant inter-model inconsistency, making it challenging for server-side aggregation algorithms to achieve superior performance. Besides, we observe that consistent features are more likely to contribute

to correct classification, motivating that mitigating inter-model inconsistency could be a key factor for performance improvement.

3.3. Theoretical Analysis

We follow (Zhou et al., 2024) to represent the m^{th} client local model $w_m = \theta_m \cdot \Phi_m$, where Φ_m is the classifier and θ_m is the feature extractor . For simplicity, we set all the bias vectors of the feature extractor and classifier to zero. Existing methods primarily train local models in a supervised manner, using the cross-entropy loss as the supervised loss. Considering a classification task with C categories, the supervised loss can be represented as:

$$\begin{aligned}\mathcal{L}_i^{sup}(w_m) &= \mathbb{E}_{(x,y)\in\mathcal{D}_m}\left[\ell_m^{sup}(\Phi_m,\theta_m;(x,y))\right] \\ &= -\frac{1}{n_m}\sum_{j=1,y_j=c}^{n_m}\log\frac{\exp(\Phi_{m,c}^T\theta_{m,c}^T x_j)}{\sum_{k=1}^{C}\exp(\Phi_{m,k}^T\theta_{m,k}^T x_j)},\end{aligned} \tag{1}$$

where n_m is the quantity of samples in the m-th client. We will demonstrate the intra-model inconsistency and inter-model inconsistency in the supervised loss as follows.

Intra-model Inconsistency. We first analyze the intra-model inconsistency in the supervised loss. We consider the performance discrepancy of the local model on the original samples (x, y) and augmented samples $(A(x), y)$, where A is the data augmentation matrix. We assume that the augmented samples have significant differences from the original samples without losing semantics (Cao et al., 2024), that is, $\|A(x) - x\| > 0$. Without losing generality, we characterize the prediction discrepancy between the original samples and augmented samples as follows.

Lemma 3.1. *(Prediction discrepancy. See proof in Appendix A). The prediction discrepancy of the local model on the original samples (x, y) and augmented samples $(A(x), y)$ can be represented as:*

$$\Delta_{intra} \geq (p \cdot \nabla g_a \cdot \nabla A)^T (x - A(x)), \tag{2}$$

where $p = \sum_{c=1}^{C}(z_c - y_c)$, z is the prediction of w_i activated by softmax function with the augmented samples $A(x)$, ∇g_a is the gradient of the local model w_i, and ∇A is the gradient of the data augmentation function A.

Lemma 3.1 indicates that prediction discrepancy is mainly caused by three factors. (1) The performance on augmented samples, which is represented by $p\nabla g_a$. (2) The transformations property of the data augmentation function, which is represented by ∇A. (3) The discrepancy between the original samples and augmented samples, which is represented by $(x - A(x))$. With Lemma 3.1, we demonstrate that the intra-model inconsistency is inevitable if adopting the supervised loss function, such as the cross-entropy function.

Theorem 3.2. *(Intra-model inconsistency. See proof in Appendix B). For local models trained with the cross-entropy loss and activated by the softmax function. When $\nabla g_a \neq 0$, $\|A(x) - x\| > 0$, and $\nabla A \neq 0$, then $\|\Delta_{intra}\|^2 > 0$.*

Theorem 3.2 reveals that the intra-model inconsistency is inevitable in the existing supervised training paradigm. Specifically, models trained with current local training methods exhibit inconsistent behavior when predicting samples with the same semantics but different augmentations. This inconsistency arises due to the inability of the model to learn semantically consistent features, leading to poor performance in aggregation.

Inter-model Inconsistency. We then analyze the inter-model inconsistency between different local models under heterogeneous data distributions. We first provide a lemma to demonstrate that data heterogeneity leads to classifier deviation among different local models. With the lemma, we characterize the inter-model inconsistency in heterogeneous scenarios. Let $z_c^j = \frac{exp(\Phi_c^T \mathcal{F}_c)}{\sum_{k=1}^{C} exp(\Phi_k^T \mathcal{F}_k)}$, where $\mathcal{F}_c$ is the feature of the c-th class. We denote $\overline{F}$ and $\overline{z}$ as the average feature and prediction of all local models, respectively.

Lemma 3.3. *(Classifier deviation. See proof in Appendix C). For client u and v with the same quantity of samples $n_u = n_v$, the classifier deviation between the two clients $\Delta_{\Phi_c} = \nabla\Phi_{u,c} - \nabla\Phi_{v,c}$ can be represented as:*

$$\begin{aligned}\Delta_{\Phi_c} = \frac{\eta}{n_u}[&(n_{u,c}(1-\overline{z}_{u,c})\overline{\mathcal{F}}_{u,c} - n_{v,c}(1-\overline{z}_{v,c})\overline{\mathcal{F}}_{v,c}) \\ &- (\sum_{c'\in[C_u]\setminus c} n_{u,c'}\overline{z}_{u,c'}\overline{\mathcal{F}}_{u,c'} - \sum_{c'\in[C_v]\setminus c} n_{v,c'}\overline{z}_{v,c'}\overline{\mathcal{F}}_{v,c'})],\end{aligned} \tag{3}$$

where η is the learning rate, $n_{u,c}$ and $n_{v,c}$ is the sample quantity of c-th class, c' is the negative classes except c. Thus, we can induce that $\|\Delta_{\Phi_c}\|^2 > 0$.

Lemma 3.3 formulates the deviation of the classifier between any two clients. All classifiers under heterogeneous data are different, inducing $\Phi_{u,c} \neq \Phi_{v,c}$.

Theorem 3.4. *(Inter-model inconsistency. See proof in Appendix D). For any two clients u and v with the same quantity of one class. the deviation of mean class features $\Delta_{\mathcal{F}_c} = \nabla\overline{\mathcal{F}}_{u,c} - \nabla\overline{\mathcal{F}}_{v,c}$ can be formulated as:*

$$\begin{aligned}\Delta_{\mathcal{F}_c} = \eta[&((1-\overline{z}_{u,c})\Phi_{u,c} - (1-\overline{z}_{v,c})\Phi_{v,c}) \\ &- (\sum_{c'\in[C]\setminus c}\overline{z}_{u,c'}\Phi_{u,c'} - \sum_{c'\in[C]\setminus c}\overline{z}_{v,c'}\Phi_{v,c'})],\end{aligned} \tag{4}$$

Base on $\Delta_{\mathcal{F}_c}$, we can induce $\|\Delta_{\mathcal{F}_c}\|^2 > 0$.

Theorem 3.4 unravels the negative effect of data heterogeneity, namely, inter-model inconsistency.

In this section, we empirically and theoretically demonstrate the occurrence of model inconsistencies on both intra-model

levels and inter-model levels. We show that the current OFL local training strategies lead to significant inconsistencies in the local models, making it challenging for server-side aggregation algorithms to achieve superior performance. We highlight the necessity of considering a new form of aggregation framework in OFL to alleviate the model inconsistencies.

4. Method

4.1. Overview

Motivated by the analysis in § 3, we propose a novel OFL framework, namely FAFI. FAFI consists of two components: self-alignment local training and informative feature fused inference. (1) On the client side, we train the feature extractor to enable the model to learn invariant features that can generalize to diverse augmented samples. We also design learnable contrastive prototypes to replace the original classifier, thereby mitigating the negative impact on the prediction. (2) On the server side, we aggregate the prototypes from all clients into a global prototype. During the inference stage, we informatively fuse the features extracted by the local models to alleviate the inter-model inconsistency. The overview of FAFI is illustrated in Figure 3.

4.2. Self-Alignment Local Learning

Motivation. According to Lemma 3.1 and Theorem 3.2, we note that the key factor leading to intra-model inconsistency is the model's inability to handle augmented samples with the same semantics, i.e., when $\Delta g_a \gg 0$. Unfortunately, the current supervised learning paradigm can only learn fixed semantics based on labels and original input, lacking generalization to diverse augmented samples. If the model can perform well on any augmented samples, i.e., $\Delta g_a = 0$, then the intra-model inconsistency will be alleviated. Intuitively, it seems that directly adopting data augmentation could resolve this problem. However, constrained by the supervised training paradigm, as long as training with labels, $p \neq 0$, the intra-model inconsistency will inevitably occur. Besides, existing works have shown that solely adopting data augmentation can lead to the problem of *complete collapse* (Grill et al., 2020). To this end, we introduce self-alignment learning to learn invariant features and an unbiased classifier that can generalize to diverse augmented samples.

Feature Contrastive Alignment. To enable the model to learn more generalized features and reduce the impact of biased data, we focus on learning invariant features related to themselves rather than features aligned with labels. We introduce a contrastive learning approach. Specifically, consider the m^{th} clients with its local dataset $\mathcal{D}_m = \{x_i, y_i\}_{i=1}^{n_m}$ and the augmented dataset $\mathcal{D}_m^a = \{A(x_i), y_i\}_{i=1}^{n_m}$, where A denotes the augmentation operator. Through self-supervised learning, we aim to learn invariant features $\mathcal{F}_{x_i} = f(\theta_m; x_i)$ and $\mathcal{F}_{A(x_i)} = f(\theta_m; A(x_i))$ that can generalize to diverse augmented samples. With similarity function sim, the self-supervised learning loss, whose objective is to minimize the discrepancy with the same semantics and different the representation to all other semantics, can be formulated as follows:

$$\mathcal{L}_{\text{SSL}} = -\frac{1}{n_m}\sum_{i=1}^{n_m}\log\frac{s(\mathcal{F}_{x_i}, \mathcal{F}_{x_i^+})}{\sum_{j\in N_g(y_i)} s(\mathcal{F}_{x_i}, \mathcal{F}_{x_j})}, \quad (5)$$

where $s() = exp(cos(\mathcal{F}_{x_i}, \mathcal{F}_{x_i^+})/\tau)$, τ denotes the temperature parameter, x_i^+ represents the set of samples that have the same label with x_i, $N_g(y_i)$ denotes the set of sample indexes that are different from y_i, and the cos function is cosine similarity.

Learnable Prototype Alignment. While contrastive learning can achieve invariant features, the classifier often displays biased behaviors and is sensitive to data heterogeneity, resulting in inconsistent predictions. Inspired by the success of prototype learning in various heterogeneous data scenarios, the learnable contrastive prototypes to replace the original classifier, thereby mitigating the negative impact of data heterogeneity on the classification (**?**). Our goal is to obtain a set of representative and highly discriminative prototypes. (1) Closely align with the features to retain semantic information. (2) maintain class distinctions between each prototype. Specifically, we define each client maintains a set of prototypes $\mathcal{P}_m = \{p_{m,1}, p_{m,2}, \ldots, p_{m,C}\}$, where C is the number of classes. During local training, the prototypes are updated to minimize the contrastive loss between the features extracted by the local model and the prototypes. The contrastive loss for the m^{th} client can be formulated as:

$$\mathcal{L}_{\text{proto}} = -\frac{1}{n_m}\sum_{i=1}^{n_m}\log\frac{\exp(\mathcal{F}_{x_i}^T p_{m,y_i}/\tau)}{\sum_{j\in[C\setminus y_i]}\exp(\mathcal{F}_{x_i}^T p_{m,j}/\tau)}, \quad (6)$$

where $\mathcal{F}_{x_i}$ is the feature extracted by the local model for sample x_i, p_{u,y_i} is the prototype corresponding to the ground truth class y_i, and τ is the temperature parameter.

Overall Objective. Thus, the overall local training objective can be:

$$\mathcal{L}_{\text{local}} = \mathcal{L}_{\text{ssl}} + \mathcal{L}_{\text{proto}}. \quad (7)$$

Notably, the objective of self-supervised local learning loss is consistent with alleviating inter-model inconsistency. And the two losses in it are complementary to each other. For learnable contrastive prototypes $\mathcal{P}_u$, the prediction of the original sample and augmented sample can be formulated as $\mathcal{F}_{x_i}^T\mathcal{P}_u$ and $\mathcal{F}_{A(x_i)}^T\mathcal{P}_u$, respectively. Using the cosine similarity as the distance metrics, the prediction discrepancy,

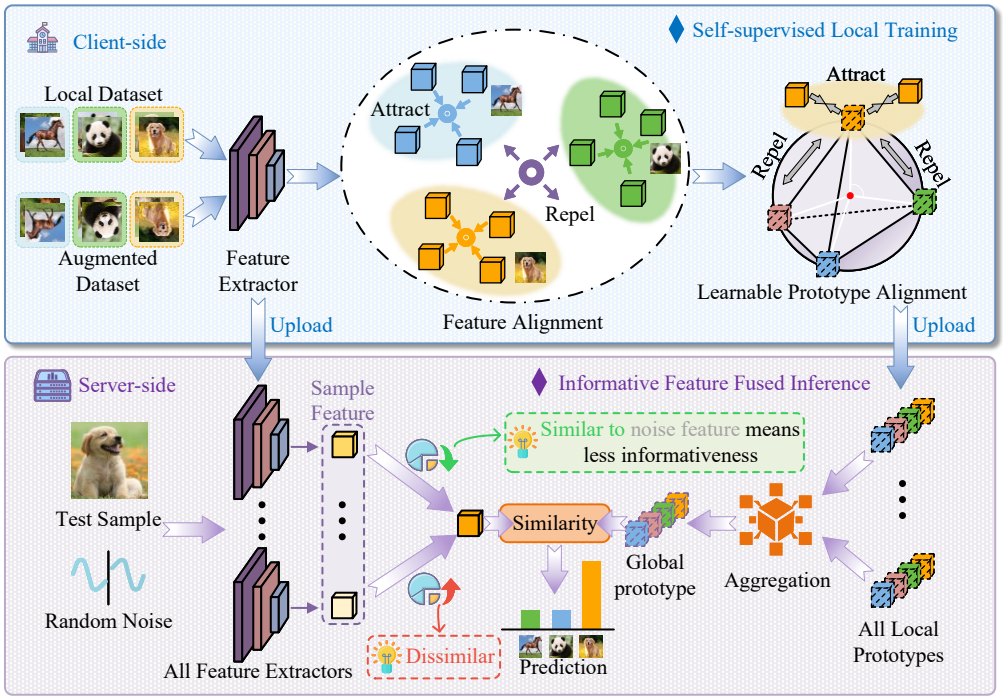


Figure 3. The overview of FAFI. All clients perform local training and upload the trained models to the server once. (1) Through self-alignment local training, clients could upload generalizable feature extractors and class-distinct prototypes. (2) The server aggregates all local prototypes into a global prototype, while in the inference stage, the server informative fuses the features extracted from the local models. It is best viewed in color. Zoom in for details.

i.e., inter-model inconsistency, can be formulated as:

$$\begin{aligned}\Delta_{intra} &= -cos(\underbrace{\mathcal{F}_{x_i}^T\mathcal{P}_m}_{\mathcal{L}_{\text{proto}}\propto}, \underbrace{\mathcal{F}_{A(x_i)}^T\mathcal{P}_m}_{\mathcal{L}_{\text{proto}}\propto}) \\ &= -\underbrace{cos(\mathcal{F}_{x_i}, \mathcal{F}_{A(x_i)})}_{\mathcal{L}_{\text{ssl}}\propto}\mathcal{P}_m,\end{aligned} \tag{8}$$

we can observe that the optimization objective of $\mathcal{L}_{\text{proto}}$ in Eq.(6) is to learn a discriminative representation, ensuring that the samples with the same semantics have similar predictions, which aligns with the goal of reducing intra-model inconsistency (first line in Eq.(8)). Simultaneously, the objective of $\mathcal{L}_{\text{ssl}}$ in Eq.(5) is to enable the local model to learn informative and consistent features (second line in Eq.(8)), which also help to reduce intra-model inconsistency.

4.3. Informative Feature Fused Inference

Motivation. The aforementioned theoretical analysis Theorem 3.4 and recent studies indicate that inter-model inconsistency caused by data heterogeneity is inevitable. The root cause of the low performance due to inter-model inconsistency lies in the inability to reconstruct a well-performing global model through parameter-level aggregation from significantly different local models. Inspired by the Mixture of Experts (MoE) (Zeng et al., 2024; Zhu et al., 2024b), an architecture which enhances the inference capability by leveraging the outputs of expert models. Instead of aggregating model parameters, we fuse the features extracted by inconsistent local models to integrate semantics, thereby mitigating the negative impact of data heterogeneity and enhancing the inference capability. Additionally, we also note that the features extracted by different models exhibit discrepancies during the fusion process. To address this, we design a feature rescale fusion mechanism.

Informative Feature Fusion. Specifically, let $\mathcal{F}_m$ be the feature extracted by the u-th client's local model. To fuse the features from different clients, we design a mechanism to informatively aggregating the features. The features with less information which is similar with noise should be down-weighted, while the features with more information should be up-weighted. Thus, we define the rescaling factor α_u as:

$$\alpha_m = 1 - cos(\mathcal{F}_m, \mathcal{F}_m^{\mathcal{N}(\mu,\sigma)}), \tag{9}$$

where $sim(\cdot,\cdot)$ is the cosine similarity, and $\mathcal{F}_u^{\mathcal{N}(\mu,\sigma)}$ is a feature extracted by a Gaussian distribution with mean μ and standard deviation σ.

Next, we aggregate the features from different clients using

the weighted average:

$$\mathcal{F}_{\text{fused}} = \sum_{m}^{M} \frac{\alpha_m}{\sum_{v}^{M} \alpha_v} \mathcal{F}_m, \tag{10}$$

where M is the number of clients.

Inference with Global Prototype. After obtaining the fused feature $\mathcal{F}_{\text{fused}}$, we use the similarity between the fused feature and the discriminative global prototype for prediction. Specifically, let $\mathcal{P}_g = \frac{1}{M}\sum_{m=1}^{M} \mathcal{P}_m$ be the global prototype aggregated from the clients' learnable prototypes. The prediction $\hat{y}$ can be formulated as:

$$\hat{y} = \arg\max_{c \in [C]} cos(\mathcal{F}_{\text{fused}}, \mathcal{P}_{g,c}), \tag{11}$$

where $cos(\cdot, \cdot)$ is a the cosine similarity. This approach leverages the fused features and the discriminative power of the global prototype to make accurate predictions.

4.4. Disccusion

Privacy Security. Transmitting prototypes from clients to the server instead of classifier is a common practice in FL (Tan et al., 2022; Mu et al., 2023; Huang et al., 2023; Wan et al., 2024) that does not compromise privacy security, as prototypes are statistical-level information of class that does not contain the privacy of individual samples.

Comparison with Analogous Methods. FedProto (Tan et al., 2022) and FedTGP (Zhang et al., 2024) both use prototype learning to alleviate the negative impact caused by data heterogeneity. However, both of these methods are based on a supervised learning paradigm and rely on iterative interactions between clients and the server, which limits their applicability in one-shot federated learning scenarios. In contrast, our proposed method leverages self-supervised learning and learnable prototypes to achieve consistent feature extraction and effective aggregation in a one-shot manner, without the need for multiple communication rounds. Besides, some studies consider that the aggregation process in OFL is akin to model merging and try to improve the performance by either using weighted-based model merging (Tao et al., 2024; Yang et al., 2024a), subspace-based merging (Zhu et al., 2024a; Yadav et al., 2024), or routing-based merging (Zeng et al., 2024). However, all these methods focuses on parameter-level merging and require auxiliary knowledge for post-calibration after merging, which is not suitable for one-shot OFL. In contrast, our proposed method focuses on feature-level merging and does not require any additional calibration.

Limitations. Our approach uses the semantics consistent prototypes for training and Inference, which is model agnostic and auxiliary information free. However, we also note limitations on the requirements of task consistency and data corruption. For multi-task settings, prototypes may not only have distinct size but also contain different meanings. For data corruption, it is difficult to guarantee a consistent prototype can be learned from semantics inconsistent data. These limitations are shared by related methods (Tan et al., 2022; Huang et al., 2023; Zhang et al., 2024; Mu et al., 2023).

5. Experiments

5.1. Experimental Setup

Datasets. Adhere to the previous work (Tan et al., 2022; Huang et al., 2023; Zeng et al., 2024; Zhang et al., 2022a; Tang et al., 2024), we evaluate the efficacy on three widely used benchmarks:

- **CIFAR-10** contsins 50k, 10k images for training and testing. Images are in size 32×32 with 10 classes.
- **CIFAR-100** have the same format and size as CIFAR-10, but with 100 classes.
- **Tiny-Imagenet** contains 100k, 10k images for training and testing. Images are in size 64×64 with 200 classes.

Data Heterogeneity. Considering the heterogeneous environment, we partitioned the dataset through a widely-used non-IID partition method, namely Dirichlet Sampling, in which the coefficient α refers to the non-IID degree. A small α represents a biased distribution. Following the setting in (Zeng et al., 2024; Zhang et al., 2022a), we set $\alpha \in \{0.05, 0.1, 0.3, 0.5\}$, respectively.

Models. Following the advanced OFL works (Zeng et al., 2024), we train ResNet-18 on all datasets. We use the SGD optimizer with a momentum coefficient of 0.9, set batch size to 256, and make local train $E = 200$ steps. We report the best results by varying the learning rate in $0.05, 0.01, 0.001$.

Baselines. We compare with several OFL methods, categorized into three types:

- **Parameter optimization-based**: one-shot FedAvg (O-FedAvg) (McMahan et al., 2017; Guha et al., 2019) [arXiv'19], MA-Echo (Su et al., 2023) [NN'23], and FedFisher (Jhunjhunwala et al., 2024) [AISTATS'24].
- **Distillation-based**: DENSE (Zhang et al., 2022a) [NeurIPS'22], FedDF (Lin et al., 2020) [NeurIPS'20], F-ADI (Yin et al., 2020) [CVPR'20], and F-DAFL (Chen et al., 2019) [ICCV'19].
- **Selective ensemble learning-based**: directly Ensemble, Co-Boosting (Dai et al., 2023) [ICLR'23], IntactOFL (Zeng et al., 2024) [MM'24].

Besides, we also consider some iterative methods, i.e., FuseFL (Tang et al., 2024) [NeurIPS'24], which tries to enhance the performance through iteratively sharing intermediate features. To ensure fair comparisons, we neglect some methods that require additional public data or models, such as FedKT (Li et al., 2021b), FedOV (Diao et al., 2022), and FedGen (Zhu et al., 2021).

Table 1. **Comparison with the state-of-the-art OFL methods**: in CIFAR-10, CIFAR-100, and Tiny-ImageNe scenarios with skew ratio $\alpha \in \{0.05, 0.1, 0.3, 0.5\}$. Underline/**bold** fonts highlight the best baseline/the proposed FAFI. Δ represents the performance improvement compared with the best baseline. We report the mean and variance (behind $\pm$) of performance over 5 trials.

Methods	CIFAR-10				CIFAR-100				Tiny-ImageNet			
	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$
MA-Echo	$36.77_{\pm 0.91}$	$51.23_{\pm 0.28}$	$60.14_{\pm 0.21}$	$64.21_{\pm 0.23}$	$19.54_{\pm 0.45}$	$29.11_{\pm 0.26}$	$37.77_{\pm 0.24}$	$41.94_{\pm 0.21}$	$15.46_{\pm 0.66}$	$22.23_{\pm 0.56}$	$23.46_{\pm 0.19}$	$28.21_{\pm 0.42}$
O-FedAvg	$12.13_{\pm 2.11}$	$17.43_{\pm 0.51}$	$28.07_{\pm 0.89}$	$35.42_{\pm 0.67}$	$4.77_{\pm 0.21}$	$6.45_{\pm 0.71}$	$10.67_{\pm 0.31}$	$12.13_{\pm 0.05}$	$5.67_{\pm 0.45}$	$8.31_{\pm 0.21}$	$13.61_{\pm 0.10}$	$13.71_{\pm 0.16}$
FedFisher	$40.03_{\pm 1.11}$	$47.01_{\pm 1.81}$	$49.33_{\pm 1.52}$	$50.34_{\pm 1.32}$	$16.56_{\pm 2.67}$	$18.98_{\pm 2.09}$	$27.24_{\pm 1.92}$	$31.44_{\pm 1.87}$	$15.65_{\pm 1.54}$	$17.89_{\pm 1.46}$	$19.54_{\pm 1.31}$	$20.77_{\pm 1.15}$
FedDF	$35.53_{\pm 0.67}$	$41.58_{\pm 0.80}$	$44.78_{\pm 0.60}$	$54.58_{\pm 0.73}$	$15.07_{\pm 0.74}$	$27.17_{\pm 0.55}$	$31.23_{\pm 0.79}$	$35.39_{\pm 0.47}$	$11.45_{\pm 0.40}$	$16.32_{\pm 0.33}$	$17.79_{\pm 0.57}$	$27.55_{\pm 0.66}$
F-ADI	$35.93_{\pm 1.56}$	$48.35_{\pm 1.23}$	$52.66_{\pm 1.44}$	$58.78_{\pm 1.67}$	$14.65_{\pm 0.98}$	$28.13_{\pm 1.24}$	$33.18_{\pm 0.67}$	$39.44_{\pm 1.11}$	$13.92_{\pm 1.99}$	$19.00_{\pm 1.78}$	$26.01_{\pm 1.44}$	$29.98_{\pm 1.34}$
F-DAFL	$38.32_{\pm 1.40}$	$46.34_{\pm 1.12}$	$54.03_{\pm 1.71}$	$59.09_{\pm 2.23}$	$16.31_{\pm 0.33}$	$26.80_{\pm 1.33}$	$34.89_{\pm 1.45}$	$37.88_{\pm 1.34}$	$15.12_{\pm 1.34}$	$19.01_{\pm 1.11}$	$23.78_{\pm 1.23}$	$27.98_{\pm 1.10}$
DENSE	$38.37_{\pm 1.08}$	$50.26_{\pm 0.24}$	$59.76_{\pm 0.45}$	$62.19_{\pm 0.12}$	$18.37_{\pm 2.43}$	$32.03_{\pm 0.44}$	$37.33_{\pm 0.48}$	$38.84_{\pm 0.39}$	$18.77_{\pm 0.67}$	$22.25_{\pm 0.33}$	$28.14_{\pm 0.34}$	$32.34_{\pm 0.32}$
Ensemble	$41.36_{\pm 0.67}$	$45.43_{\pm 0.32}$	$62.18_{\pm 0.34}$	$61.61_{\pm 0.23}$	$20.46_{\pm 0.62}$	$26.23_{\pm 0.55}$	$38.01_{\pm 0.67}$	$41.61_{\pm 0.77}$	$13.28_{\pm 0.67}$	$15.38_{\pm 0.23}$	$17.53_{\pm 0.31}$	$28.50_{\pm 0.46}$
Co-Boosting	$39.20_{\pm 0.81}$	$58.49_{\pm 1.24}$	$67.21_{\pm 1.76}$	$70.24_{\pm 2.34}$	$20.19_{\pm 1.44}$	$27.59_{\pm 1.35}$	$39.30_{\pm 1.30}$	$42.67_{\pm 1.40}$	$19.00_{\pm 1.45}$	$21.90_{\pm 1.20}$	$29.24_{\pm 1.32}$	$30.78_{\pm 2.01}$
FuseFL	$54.42_{\pm 0.41}$	$73.79_{\pm 0.34}$	$84.58_{\pm 0.91}$	$84.34_{\pm 0.88}$	$29.12_{\pm 0.23}$	$36.86_{\pm 0.38}$	$45.12_{\pm 0.51}$	$49.30_{\pm 0.32}$	$22.15_{\pm 2.11}$	$29.28_{\pm 2.04}$	$33.04_{\pm 1.79}$	$34.34_{\pm 1.81}$
IntactOFL	$48.22_{\pm 0.43}$	$61.13_{\pm 0.63}$	$70.21_{\pm 0.60}$	$79.93_{\pm 0.23}$	$27.99_{\pm 0.67}$	$39.15_{\pm 0.46}$	$41.86_{\pm 0.60}$	$46.78_{\pm 0.78}$	$20.45_{\pm 0.34}$	$28.43_{\pm 0.17}$	$30.15_{\pm 0.12}$	$35.09_{\pm 0.14}$
Ours	**71.84**$_{\pm 1.53}$	**77.83**$_{\pm 1.32}$	**84.76**$_{\pm 0.46}$	**88.74**$_{\pm 0.11}$	**31.02**$_{\pm 1.17}$	**45.48**$_{\pm 1.01}$	**56.65**$_{\pm 0.91}$	**61.07**$_{\pm 0.55}$	**36.96**$_{\pm 0.92}$	**43.62**$_{\pm 0.77}$	**53.32**$_{\pm 0.50}$	**56.48**$_{\pm 0.32}$
Δ	$\uparrow$ 17.42	$\uparrow$ 6.04	$\uparrow$ 0.18	$\uparrow$ 4.40	$\uparrow$ 1.90	$\uparrow$ 6.33	$\uparrow$ 11.53	$\uparrow$ 11.77	$\uparrow$ 14.81	$\uparrow$ 14.34	$\uparrow$ 20.28	$\uparrow$ 21.39

5.2. Effectiveness

Table 1 illustrates the effectiveness of our proposed FAFI compared with popular OFL methods in non-IID settings. (1) It clearly depicts that our method achieves a significant performance improvement over the baselines on all datasets and all settings (10.86% averaged). (2) Notably, in some extreme cases such as $\alpha = \{0.05, 0.1\}$ on Tiny-Imagenet, our method exhibits a more significant performance advantage (17.71% averaged) over the baseline algorithms. (3) Besides, the FuseFL which iteratively sharing intermediate features between clients can achieve the second-best performance in some settings. We attribute this to the fact that the FuseFL mitigates the feature inconsistencies by sharing intermediate features, leaving intra-model inconsistency unsolved. **In summary, the FAFI is effective in various data heterogeneous scenarios and achieves competitive performance over all baselines.**

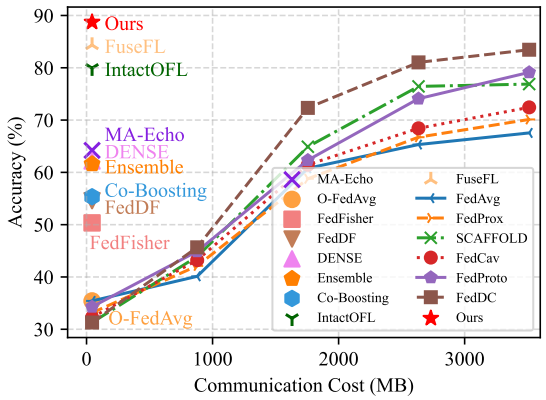


Figure 4. **Test Accuracy v.s. Communication Cost** on CIFAR-10. FAFI is more efficient than all other baselines, while achieving higher performance without large communication overhead.

Table 2. **Scalability under different number of clients**, $m = \{5, 10, 25, 50, 100\}$ on CIFAR-10 with $\alpha = 0.5$.

Methods	Client scales m				
	5	10	25	50	100
MA-Echo	64.21	52.64	48.36	45.35	38.54
O-FedAvg	35.42	32.09	28.03	28.24	27.14
FedFisher	50.34	45.67	34.66	29.09	28.89
FedDF	54.58	48.88	35.44	29.91	25.66
F-ADI	59.34	46.33	31.83	27.66	24.89
F-DAFL	58.59	45.45	32.88	29.98	28.91
DENSE	62.19	54.67	49.32	48.67	43.34
Ensemble	61.61	60.44	58.44	52.51	45.72
Co-Boosting	55.34	51.11	49.32	44.56	42.45
FuseFL	84.34	78.28	62.12	42.18	37.11
IntactOFL	79.93	69.11	64.32	59.45	53.21
Ours	**88.74**	**86.96**	**85.25**	**81.32**	**75.37**

5.3. Scalability

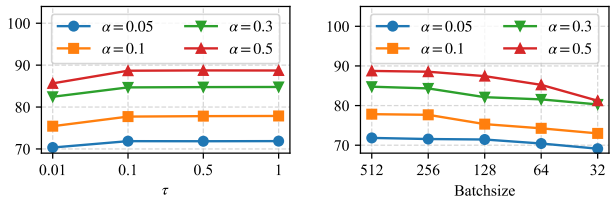
We assess the scalability of FAFI by varying the number of clients m. As presented in Table 2, FAFI consistently achieves the best performance across different client scales. As suggested in (Lian et al., 2017), the server can become a major bottleneck while the number of clients increases. We also reach a similar conclusion, with the number of clients m increasing, the performance decreases, which is consistent with (Zhang et al., 2022a; Dai et al., 2023; Lian et al., 2017; Zeng et al., 2024). **In summary, the FAFI is scalable across diverse distributed networks of varying sizes.**

5.4. Efficiency

We compare the accuracy and communication cost of FAFI to existing OFL methods and multi-round FL baselines. We

Table 3. **Ablation Study on Key Components** for FAFI on CIFAR-10.

SALT	IFFI	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$
		12.13	17.43	28.07	35.42
✓		53.12	55.76	58.95	61.67
	✓	55.23	63.75	68.49	70.44
✓	✓	71.84	77.83	84.76	88.74



(a) Contrastive parameter τ (b) Contrastive batch size

Figure 5. **Analysis on hyper-parameter**. Performance with hyper-parameter τ and batchsize on four data heterogeneity $\alpha \in \{0.05, 0.1, 0.3, 0.5\}$.

select six representative multi-round FL methods, including FedAvg (McMahan et al., 2017), FedProx (Li et al., 2020) SCAFFOLD (Karimireddy et al., 2020), FedCav (Zeng et al., 2021; 2025), FedProto (Tan et al., 2022), FedDC (Gao et al., 2022). For a fair comparison, we use the same hyperparameters for all methods, including the number of clients, heterogeneity, and learning rate. As illustrated in Fig. 4, FAFI achieves higher performance than all other baselines, while incurring a lower communication overhead. Notably, with more communication budget, multi-round FL methods can achieve better performance (Fig. 4 only presents the first 80 rounds results of multi-round FL methods). However, as these methods get convergent, the performance improvement diminishes, and communication costs become prohibitive. In summary, **FAFI achieves a better efficiency compared with existing FL methods (both OFL and multi-round FL), making it more suitable for practical deployment**.

5.5. Ablation Study

Overall Design. For thoroughly analyzing the efficacy of each module, we perform an ablation study to investigate the effectiveness of Self-Alignment Local Training (SALT) and Informative Feature Fused Inference (IFFI). The results in Table 3 show that the two modules contribute significantly to the performance improvement of FAFI. The combination of the two modules achieves the best performance, underscoring the effectiveness of our proposed method.

Hyper-parameter. We first investigate the τ and batch size in self-alignment local learning. For τ, we note that its impact on performance when $\tau \in [0.1, 1]$. When $\tau = 0.01$, it would have a certain degree of degradation. For batch size, we note that a larger batch size would benefit the performance while requiring more resources. With the minimum batch size, the performance is still competitive.

6. Conclusion

In this paper, we propose a one-shot Federated Learning with self-alignment Local Training and Informative Feature Fused Inference with objective of bridging the gap between local models. We empirically and theoretically observe the model inconsistencies in existing OFL methods. We propose self-alignment Local Training for consistent features implicitly aligned by feature semantics, and use Informative Feature Fused Inference to fuse the local models with informative features. We conduct extensive experiments on different datasets and show that our method outperforms the state-of-the-art methods.

References

Blanchard, P., El Mhamdi, E. M., Guerraoui, R., and Stainer, J. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, pp. 118–128, 2017.

Cao, C., Zhou, F., Dai, Y., Wang, J., and Zhang, K. A survey of mix-based data augmentation: Taxonomy, methods, applications, and explainability. *ACM Computing Surveys*, 57(2):1–38, 2024.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 9912–9924, 2020.

Chen, H., Wang, Y., Xu, C., Yang, Z., Liu, C., Shi, B., Xu, C., Xu, C., and Tian, Q. Data-free learning of student networks. In *Proc. of ICCV*, pp. 3514–3522, 2019.

Chen, J., Zhu, J., and Zheng, Q. Towards fast and stable federated learning: Confronting heterogeneity via knowledge anchor. In *Proc. of ACM International Conference on Multimedia (ACM MM)*, pp. 8697–8706, 2023.

Chongjian, G., Wang, J., Tong, Z., Chen, S., Song, Y., and Luo, P. Soft neighbors are positive supporters in contrastive visual representation learning. In *Proc. of the International Conference on Learning Representations (ICLR)*, pp. 1–16, 2023.

Dai, R., Zhang, Y., Li, A., Liu, T., Yang, X., and Han, B. Enhancing one-shot federated learning through data and

ensemble co-boosting. In *Proc. of International Conference on Learning Representations (ICLR)*, pp. 1–21, 2023.

Dennis, D. K., Li, T., and Smith, V. Heterogeneity for the win: One-shot federated clustering. In *Proc. of International Conference on Machine Learning (ICML)*, pp. 2611–2620. PMLR, 2021.

Diao, Y., Li, Q., and He, B. Towards addressing label skews in one-shot federated learning. In *Proc. of International Conference on Learning Representations (ICLR)*, 2022.

Gao, L., Fu, H., Li, L., Chen, Y., Xu, M., and Xu, C.-Z. Feddc: Federated learning with non-iid data via local drift decoupling and correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10112–10121, 2022.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. Bootstrap your own latent-a new approach to self-supervised learning. In *Proc. of Advances in Neural information Processing Systems (NeurIPS)*, volume 33, pp. 21271–21284, 2020.

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

Heinbaugh, C. E., Luz-Ricca, E., and Shao, H. Data-free one-shot federated learning under very high statistical heterogeneity. In *Proc. of International Conference on Learning Representations (ICLR)*, pp. 1–17, 2022.

Huang, W., Ye, M., Shi, Z., Li, H., and Du, B. Rethinking federated learning with domain shift: A prototype view. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16312–16322. IEEE, 2023.

Jhunjhunwala, D., Wang, S., and Joshi, G. Fedfisher: Leveraging fisher information for one-shot federated learning. In *Proc. of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 1612–1620. PMLR, 2024.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., and Suresh, A. T. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proc. of International Conference on Machine Learning (ICML)*, pp. 5132–5143. PMLR, 2020.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, pp. 18661–18673, 2020.

Li, D. and Wang, J. Fedmd: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*, 2019.

Li, J., Zhou, P., Xiong, C., and Hoi, S. C. Prototypical contrastive learning of unsupervised representations (2020). In *Proc. of the International Conference on Learning Representations (ICLR)*, pp. 4–8, 2021a.

Li, Q., He, B., and Song, D. Practical one-shot federated learning for cross-silo setting. In *Proc. of IJCAI*, 2021b.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. Federated optimization in heterogeneous networks. *Proc. of Machine learning and systems (MLSys)*, 2:429–450, 2020.

Lian, X., Zhang, C., Zhang, H., Hsieh, C.-J., Zhang, W., and Liu, J. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Lin, T., Kong, L., Stich, S. U., and Jaggi, M. Ensemble distillation for robust model fusion in federated learning. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 2351–2363, 2020.

Liu, X., Liu, L., Ye, F., Shen, Y., Li, X., Jiang, L., and Li, J. Fedlpa: One-shot federated learning with layer-wise posterior aggregation. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1–39, 2024.

Luo, M., Chen, F., Hu, D., Zhang, Y., Liang, J., and Feng, J. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 5972–5984, 2021.

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

Mu, X., Shen, Y., Cheng, K., Geng, X., Fu, J., Zhang, T., and Zhang, Z. Fedproc: Prototypical contrastive federated learning on non-iid data. *Future Generation Computer Systems*, 143:93–104, 2023.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Song, R., Liu, D., Chen, D. Z., Festag, A., Trinitis, C., Schulz, M., and Knoll, A. Federated learning via decentralized dataset distillation in resource-constrained edge

environments. In *Proc. of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10. IEEE, 2023.

Su, S., Li, B., and Xue, X. One-shot federated learning without server-side training. *Neural Networks*, 164:203–215, 2023.

Tan, Y., Long, G., Liu, L., Zhou, T., Lu, Q., Jiang, J., and Zhang, C. Fedproto: Federated prototype learning across heterogeneous clients. In *Proc. of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 36, pp. 8432–8440, 2022.

Tang, Z., Zhang, Y., Dong, P., ming Cheung, Y., Zhou, A. C., Han, B., and Chu, X. Fusefl: One-shot federated learning through the lens of causality with progressive model fusion. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1–37, 2024.

Tao, Z., Mason, I., Kulkarni, S., and Boix, X. Task arithmetic through the lens of one-shot federated learning. *arXiv preprint arXiv:2411.18607*, 2024.

Wan, G., Huang, W., and Ye, M. Federated graph learning under domain shift with generalizable prototypes. In *Proc. of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pp. 15429–15437, 2024.

Wu, X., Yao, X., and Wang, C.-L. Fedscr: Structure-based communication reduction for federated learning. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 32(7):1565–1577, 2020.

Yadav, P., Tam, D., Choshen, L., Raffel, C. A., and Bansal, M. Ties-merging: Resolving interference when merging models. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2024.

Yang, E., Shen, L., Guo, G., Wang, X., Cao, X., Zhang, J., and Tao, D. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666*, 2024a.

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

Yang, M., Su, S., Li, B., and Xue, X. Feddeo: Description-enhanced one-shot federated learning with diffusion models. In *Proc. of the ACM International Conference on Multimedia (ACM MM)*, pp. 6666–6675, 2024b.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. Byzantine-robust distributed learning: Towards optimal statistical rates. In *Proc. of International Conference on Machine Learning (ICML)*, pp. 5650–5659. PMLR, 2018.

Yin, H., Molchanov, P., Alvarez, J. M., Li, Z., Mallya, A., Hoiem, D., Jha, N. K., and Kautz, J. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pp. 8715–8724, 2020.

Yurochkin, M., Agarwal, M., Ghosh, S., Greenewald, K., Hoang, N., and Khazaeni, Y. Bayesian nonparametric federated learning of neural networks. In *Proc. of International Conference on Machine Learning (ICML)*, pp. 7252–7261, 2019.

Zeng, H., Zhou, T., Guo, Y., Cai, Z., and Liu, F. Fedcav: contribution-aware model aggregation on distributed heterogeneous data in federated learning. In *Proc. of International Conference on Parallel Processing (ICPP)*, pp. 1–10, 2021.

Zeng, H., Xu, M., Zhou, T., Wu, X., Kang, J., Cai, Z., and Niyato, D. One-shot-but-not-degraded federated learning. In *Proc. of the ACM International Conference on Multimedia (ACM MM)*, pp. 11070–11079, 2024.

Zeng, H., Zhou, T., Guo, Y., Cai, Z., and Liu, F. Towards value-sensitive and poisoning-proof model aggregation for federated learning on heterogeneous data. *Journal of Parallel and Distributed Computing (JPDC)*, 196:104994, 2025.

Zhang, J., Chen, C., Li, B., Lyu, L., Wu, S., Ding, S., Shen, C., and Wu, C. Dense: Data-free one-shot federated learning. In *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 21414–21428, 2022a.

Zhang, J., Li, Z., Li, B., Xu, J., Wu, S., Ding, S., and Wu, C. Federated learning with label distribution skew via logits calibration. In *Proc. of the International Conference on Machine Learning (ICML)*, volume 162, pp. 26311–26329, 17–23 Jul 2022b.

Zhang, J., Liu, Y., Hua, Y., and Cao, J. Fedtgp: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In *Proc. of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pp. 16768–16776, 2024.

Zhou, T., Zhang, J., and Tsang, D. H. K. Fedfa: Federated learning with feature anchors to align features and classifiers for heterogeneous data. *IEEE Transactions on Mobile Computing (TMC)*, 23(6):6731–6742, 2024.

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

Zhu, D., Sun, Z., Li, Z., Shen, T., Yan, K., Ding, S., Kuang, K., and Wu, C. Model tailor: Mitigating catastrophic forgetting in multi-modal large language models. 2024a.

Zhu, T., Qu, X., Dong, D., Ruan, J., Tong, J., He, C., and Cheng, Y. Llama-moe: Building mixture-of-experts from llama with continual pre-training. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 15913–15923, 2024b.

Zhu, Z., Hong, J., and Zhou, J. Data-free knowledge distillation for heterogeneous federated learning. In *Proc. of International Conference on Machine Learning (ICML)*, pp. 12878–12889. PMLR, 2021.

A. Proof of Lemma 3.1

Proof. We derive the performance discrepancy of the local model on the original samples (x, y) and augmented samples $(A(x), y)$ as follows.

$$\Delta_{intra} = \mathcal{L}_i^{sup}(w_i; (x, y)) - \mathcal{L}_i^{sup}(w_i; (A(x), y)). \tag{12}$$

We use the Talyer expansion to approximate the loss function on the original samples (x, y) as follows:

$$\mathcal{L}_i^{sup}(w_i; (x, y)) \approx \mathcal{L}_i^{sup}(w_i; (A(x), y)) + \nabla \mathcal{L}_i^{sup}(w_i; (A(x), y))^T (x - A(x)). \tag{13}$$

We can further simplify Equ. 12 as follows:

$$\Delta_{intra} \geq \nabla \mathcal{L}_i^{sup}(w_i; (A(x), y))^T (x - A(x)). \tag{14}$$

If we adopt cross-entropy loss as the supervised loss $\mathcal{L}(z, y) = -\sum_{c=1}^{C} y_k \log z_k$, where z is the softmax prediction, y is the ground truth label, we have:

$$\begin{aligned} \Delta_{intra} &\geq \left(\frac{\partial \mathcal{L}}{\partial z} \frac{\partial z}{\partial A} \frac{\partial A}{\partial x} \right)^T (x - A(x)) \\ &\geq \left(\sum_{c=1}^{C} (z_c - y_c) \frac{\partial z}{\partial A} \nabla A \right)^T (x - A(x)). \end{aligned} \tag{15}$$

Note that $\frac{\partial z}{\partial A}$ is the gradient ∇g of local model w_i with respect to the augmented data $A(x)$, and $\nabla A = \frac{\partial A}{\partial x}$ is the transformations property of the data augmentation function A. Since $z_c = \frac{exp(\Phi_{i,c}^T \theta_{i,c}^T A(x_j))}{\sum_{k=1}^{C} \exp(\Phi_{i,k}^T \theta_{i,k}^T A(x_j))}$ is the output of a softmax function, $z_c \in (0, 1)$, y is the one-shot encoded vector, $y_c \in \{0, 1\}$, thus, $|z_c - y_c| > 0$. Finally, we finish the proof:

$$\Delta_{intra} \geq (p \cdot \nabla g_a \cdot \nabla A)^T (x - A(x)), \tag{16}$$

where $p = \sum_{c=1}^{C} (z_c - y_c)$, ∇g_a is the gradient of the local model w_i, and ∇A is the gradient of the data augmentation function A. □

B. Proof of Theorem 3.2

Proof. Based on Lemma 3.1, we derive the intra-model inconsistency for any local models trained by existing OFL methods. When $\nabla g_a \neq 0$ and $\|A(x) - x\| > 0$, $\nabla A \neq 0$, we have:

$$\begin{aligned} \|\Delta_{intra}\|^2 &= \|(p \cdot \nabla g_a \cdot \nabla A)^T (x - A(x))\|^2 \\ &= p^2 \cdot \|\nabla g_a\|^2 \cdot \|\nabla A\|^2 \cdot \|(x - A(x))\|^2 \end{aligned} \tag{17}$$

Since $|p| > 0$, $\|\nabla g_a\|^2 > 0$, $\|\nabla A\|^2 > 0$, and $Vert(x - A(x))\|^2 > 0$, we can induce that $\|\Delta_{intra}\|^2 > 0$.

We finish the proof. □

Note that in Theorem 3.2, the $\nabla g_a = 0$ represents the augmented data $A(x)$ is not sensitive to the local model w_i, which is a rare case in practice. If the $\nabla A = 0$, it means the data augmentation function A have fully changed the semantics of the original data x. In conclusion, we can infer that the intra-model inconsistency is inevitable in existing OFL methods.

C. Proof of Lemma 3.3

Proof. We derive the gradient of cross-entropy loss as :

$$\begin{aligned} \Delta_{\Phi_c} &= \nabla \Phi_{u,c} - \nabla \Phi_{v,c} \\ &\approx \frac{\eta n_{u,c}}{n_u} (1 - \overline{z}_{u,c}) \overline{\mathcal{F}}_{u,c} - \frac{\eta n_{v,c}}{n_v} (1 - \overline{z}_{v,c}) \overline{\mathcal{F}}_{v,c} \\ &- (\frac{\eta}{n_u} \sum_{c' \in [C_u] \setminus c} n_{u,c'} \overline{z}_{u,c'} \overline{\mathcal{F}}_{u,c'} - \frac{\eta}{n_v} \sum_{c' \in [C_v] \setminus c} n_{v,c'} \overline{z}_{v,c'} \overline{\mathcal{F}}_{v,c'}) \end{aligned} \tag{18}$$

where the term of $\approx$ holds from the Property 1 in (Zhang et al., 2022b). If $n_u = n_v$, we have

$$\begin{aligned}\Delta_{\Phi_c} = \frac{\eta}{n_u}[&\underbrace{(n_{u,c}(1-\overline{z}_{u,c})\overline{\mathcal{F}}_{u,c} - n_{v,c}(1-\overline{z}_{v,c})\overline{\mathcal{F}}_{v,c})}_{\Delta^+_{\Phi_c}} \\ &\underbrace{-(\sum_{c'\in[C_u]\backslash c} n_{u,c'}\overline{z}_{u,c'}\overline{\mathcal{F}}_{u,c'} - \sum_{c'\in[C_v]\backslash c} n_{v,c'}\overline{z}_{v,c'}\overline{\mathcal{F}}_{v,c'})}_{\Delta^-_{\Phi_c}}].\end{aligned} \tag{19}$$

We consider the $\|\Delta_{\Phi_c}\|^2$ under heterogeneous data distribution.

When $\overline{\mathcal{F}}_{u,c} \neq \overline{\mathcal{F}}_{v,c}$, we have

$$\|\Delta_{\Phi_c}\|^2 \geq \frac{\eta^2}{n^2}\left|\|\Delta^+_{\Phi_c}\|^2 - \|\Delta^-_{\Phi_c}\|^2\right| > 0. \tag{20}$$

When $\overline{\mathcal{F}}_{u,c} = \overline{\mathcal{F}}_{v,c}$, $c \in [C_u] \cap [C_v]$, $\overline{z}_{u,c} = \overline{z}_{v,c}$, that is, two clients have same prediction and feature on selected positive class c, then $c' \in [C_u]\backslash\{[C_u] \cap [C_v]\}$, we have

$$\begin{aligned}\|\Delta_{\Phi_c}\|^2 &= \frac{\eta^2}{n^2}\|\Delta^-_{\Phi_c}\|^2 \\ &= \frac{\eta^2}{n^2}\|\sum_{c'\in[C_u]\backslash\{[C_u]\cap[C_v]\}} n_{u,c'}\overline{z}_{u,c'}\overline{\mathcal{F}}_{u,c'} - \sum_{c'\in[C_u]\backslash\{[C_u]\cap[C_v]\}} n_{v,c'}\overline{z}_{v,c'}\overline{\mathcal{F}}_{v,c'}\|^2\end{aligned} \tag{21}$$

If we assume the equation above equals to zero, we have $\overline{z}_{u,c'}\overline{\mathcal{F}}_{u,c'} = \overline{z}_{v,c'}\overline{\mathcal{F}}_{v,c'}$, which requires a perfect feature extractor for all clients. It is not a practical condition in FL according to (Zhou et al., 2024).

When $\overline{\mathcal{F}}_{u,c} = \overline{\mathcal{F}}_{v,c}$, $c \in [C]\backslash\{[C_u] \cup [C_v]\}$, $n_{u,c} = n_{v,c} = 0$, we have the same formulation with Equ. 21, we can obtain $\|\Delta_{\Phi_c}\|^2 > 0$.

When $\overline{\mathcal{F}}_{u,c} = \overline{\mathcal{F}}_{v,c}$, $c \in [C_u]\backslash\{[C_u] \cap [C_v]\}$, $n_{v,c} = 0$, we have

$$\begin{aligned}\|\Delta_{\Phi_c}\|^2 &= \frac{\eta^2}{n^2}\|n_{u,c}(1-\overline{z}_{u,c})\overline{\mathcal{F}}_{u,c} - (\sum_{c'\neq c} n_{u,c'}\overline{z}_{u,c'}\overline{\mathcal{F}}_{u,c'} - \sum_{c'\neq c} n_{v,c'}\overline{z}_{v,c'}\overline{\mathcal{F}}_{v,c'})\|^2 \\ &\approx \frac{\eta^2}{n^2}\|n_{u,c}\overline{\mathcal{F}}_{u,c} - \overline{z}_{u,c}\overline{\mathcal{F}}_{u,c}\|\end{aligned} \tag{22}$$

where the equality holds if and only if $n_{u,c}(1-\overline{z}_{u,c})\overline{\mathcal{F}}_{u,c} = \Delta^-_{\Phi_c}$. In the training phase, it is challenging to maintain this condition between any two clients, so we use the term of $\approx$ since $\overline{z}_{u,c'} \ll \overline{z}_{u,c} < 1$. We say that the $\|\Delta_{\Phi_c}\|^2$ is more likely to be positive. The other cases, $c \in [C_v]\backslash\{[C_u] \cap [C_v]\}$ can get the same conclusion.

In summary, we conclude that the $\|\Delta_{\Phi_c}\|^2 > 0$ under different heterogeneous scenarios. We finish the proof. □

D. Proof of Theorem 3.4

Proof. Similar to Lemma 3.3, we derive the gradient of the cross-entropy loss function, we have:

$$\begin{aligned}\Delta_{\mathcal{F}_c} &= \nabla\overline{\mathcal{F}}_{u,c} - \nabla\overline{\mathcal{F}}_{v,c} \\ &= \eta[\underbrace{((1-\overline{z}_{u,c})\Phi_{u,c} - (1-\overline{z}_{v,c})\Phi_{v,c})}_{\Delta^+_{\mathcal{F}_c}} \\ &\quad \underbrace{-(\sum_{c'\in[C]\backslash c}\overline{z}_{u,c'}\Phi_{u,c'} - \sum_{c'\in[C]\backslash c}\overline{z}_{v,c'}\Phi_{v,c'})]}_{\Delta^-_{\mathcal{F}_c}}\end{aligned} \tag{23}$$

With Lemma 3.3, all classifiers under heterogeneous data exhibits discrepancies, that is, $\Phi_{u,c} \neq \Phi_{v,c}$, inducing $\Delta^+_{\mathcal{F}_c} \neq \Delta^-_{\mathcal{F}_c}$. Therefore, we have $\|\Delta_{\mathcal{F}_c}\|^2 > 0$. □