Medical Image Analysis 102 (2025) 103524

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

FedBM: Stealing knowledge from pre-trained language models for heterogeneous federated learning

Meilu Zhu[a], Qiushi Yang[b], Zhifan Gao[c], Yixuan Yuan[d,*], Jun Liu[e,a,**]

[a] Department of Mechanical Engineering, City University of Hong Kong, Hong Kong Special Administrative Region, China
[b] Department of Electrical Engineering, City University of Hong Kong, Hong Kong Special Administrative Region, China
[c] School of Biomedical Engineering, Sun Yat-sen University, China
[d] Department of Electronic Engineering, Chinese University of Hong Kong, Hong Kong Special Administrative Region, China
[e] Department of Data and Systems Engineering, The University of Hong Kong, Hong Kong Special Administrative Region, China

ARTICLE INFO

Keywords:
Federated learning
Medical Image Classification
Pre-trained language model

ABSTRACT

Federated learning (FL) has shown great potential in medical image computing since it provides a decentralized learning paradigm that allows multiple clients to train a model collaboratively without privacy leakage. However, current studies have shown that data heterogeneity incurs local learning bias in classifiers and feature extractors of client models during local training, leading to the performance degradation of a federation system. To address these issues, we propose a novel framework called Federated Bias eliMinating (FedBM) to get rid of local learning bias in heterogeneous federated learning (FL), which mainly consists of two modules, *i.e.*, Linguistic Knowledge-based Classifier Construction (LKCC) and Concept-guided Global Distribution Estimation (CGDE). Specifically, LKCC exploits class concepts, prompts and pre-trained language models (PLMs) to obtain concept embeddings. These embeddings are used to estimate the latent concept distribution of each class in the linguistic space. Based on the theoretical derivation, we can rely on these distributions to pre-construct a high-quality classifier for clients to achieve classification optimization, which is frozen to avoid classifier bias during local training. CGDE samples probabilistic concept embeddings from the latent concept distributions to learn a conditional generator to capture the input space of the global model. Three regularization terms are introduced to improve the quality and utility of the generator. The generator is shared by all clients and produces pseudo data to calibrate updates of local feature extractors. Extensive comparison experiments and ablation studies on public datasets demonstrate the superior performance of FedBM over state-of-the-arts and confirm the effectiveness of each module, respectively. The code is available at https://github.com/CUHK-AIM-Group/FedBM.

1. Introduction

With the explosive growth of data, training deep models has become a promising path to achieve high-precision computer-aided diagnosis (CAD) (Razzak et al., 2018; Yan et al., 2018; Zhu et al., 2021a, 2024b). However, centralizing data from different hospitals or institutions to construct large-scale medical training datasets is unrealistic, thanks to growing privacy concerns or legal restrictions (Zhu et al., 2024a). To conquer this problem, a new training paradigm, federated learning (FL) (McMahan et al., 2017; Li et al., 2020b; Zhu et al., 2023), is proposed to learn deep models across different clients (hospitals) under the coordination of a cloud server. In each round of FL training, clients independently train local models on their private data and upload them to the server, where the models are aggregated. The aggregated model is then sent back to the clients, serving as the initialization for the next training round. Importantly, clients are not required to share their raw data during this process, thereby preserving their privacy. Unfortunately, the heterogeneity among client datasets significantly contributes to *local learning bias* at the client side, leading to performance degradation in federation systems (Li et al., 2021b, 2020b; Jeong et al., 2018; Guo et al., 2023b). The learning bias primarily manifests in two aspects from the perspective of representation learning, as shown in Fig. 1.

Firstly, local learning bias appears in classifiers of local models during training (Luo et al., 2021; Xu et al., 2023). When data from clients are heterogeneous, the local classifiers are dominated by their local class distributions, leading to the shifted decision boundaries across

* Corresponding author.
** Corresponding author at: Department of Data and Systems Engineering, The University of Hong Kong, Hong Kong Special Administrative Region, China.
E-mail addresses: yxyuan@ee.cuhk.edu.hk (Y. Yuan), dr.jun.liu@hku.hk (J. Liu).

https://doi.org/10.1016/j.media.2025.103524
Received 25 October 2024; Received in revised form 21 January 2025; Accepted 22 February 2025
Available online 7 March 2025

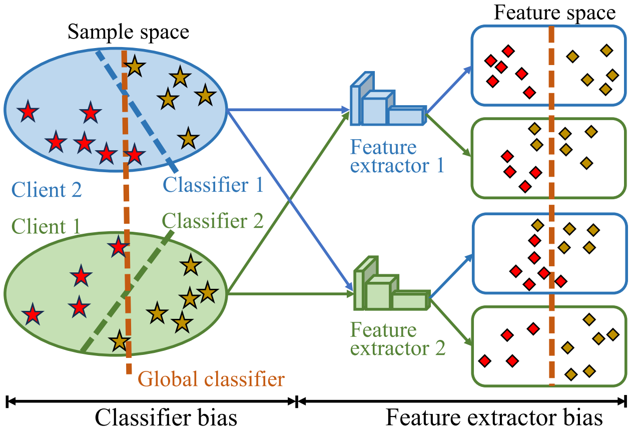


Fig. 1. Data heterogeneity causes local learning bias, including classifier bias and feature extractor bias. (Best viewed in color).

clients in Fig. 1. Recent studies (Huang et al., 2023; Dai et al., 2023; Tan et al., 2022; Qi et al., 2023; Long et al., 2023) have attempted to exploit class prototypes as classifier to avoid this problem. However, these approaches obtain limited performance since the quality of class prototypes for one client is affected by its biased local feature extractor. In experiments, we surprisingly found that a simple strategy of using a fixed randomly-initialized classifier for all clients outperforms the baseline method, FedAvg (McMahan et al., 2017), as shown in Table 8. The results indicate that sharing a fixed classifier across clients is a feasible path to alleviate the classifier bias problem. Intuitively, random initialization is not the optimal strategy to build the fixed classifier since it does not consider intra-class semantic information and inter-class distance relations. This inspires us to explore a better solution to pre-construct a high-quality classifier for clients and freeze it during federated training.

Secondly, the heterogeneous data of clients would produce inconsistent local feature extractors. The features from the feature extractor of one client may differ significantly from those extracted by the feature extractor of another client, even for the same input data in Fig. 1. Consequently, the global feature extractor, obtained by aggregating these inconsistent local feature extractors, will fail to extract generalizable features for adapting to all clients (Guo et al., 2023a). Previous works (Li et al., 2020b, 2021a; Acar et al., 2021; Karimireddy et al., 2020) reduce the inconsistency by regularizing the distance between local models of the current round and the global model from the last round. However, it is hard to balance the trade-offs between optimization and regularization to perform well (Wang et al., 2020). Different from these methods, we try to directly narrow the gap between the data distribution of clients by exploiting textual prior to estimate global distribution to supplement the local distribution.

To tackle the aforementioned issues, we propose a novel framework called Federated Bias eliMinating (FedBM) to get rid of local learning bias in heterogeneous federated learning (FL). FedBM mainly consists of two modules, *i.e.*, Linguistic Knowledge-based Classifier Construction (LKCC) and Concept-guided Global Distribution Estimation (CGDE). Specifically, LKCC exploits class concepts, prompts, and pre-trained language models (PLMs) to obtain concept embeddings. These embeddings are used to estimate the latent concept distribution of each class in the linguistic space. Based on the theoretical derivation, we can rely on these distributions to construct a high-quality global classifier for alignment between visual and linguistic spaces, avoiding classifier bias. CGDE samples probabilistic concept embeddings from the latent concept distributions to learn a conditional generator to capture the input space of the global model. Three regularization terms are introduced to improve the quality and utility of the generator. The generator is shared by all clients and produces pseudo data to calibrate updates of local feature extractors. The contributions of this work are summarized as follows:

▶ We present a novel Federated Bias eliMinating (FedBM) framework, which represents the first effort to use linguistic knowledge to address heterogeneous FL.

▶ We propose Linguistic Knowledge-based Classifier Construction (LKCC) that exploits linguistic knowledge from pre-trained language models (PLMs) to pre-define a high-quality global classifier to avoid classifier bias.

▶ We design Concept-guided Global Distribution Estimation (CGDE) that utilizes probabilistic concept embeddings to learn a conditional generator to produce pseudo data to calibrate updates of local feature extractors.

▶ We conduct extensive experiments on public datasets to evaluate the proposed framework. The results show the superior performance of FedBM against state-of-the-arts and the effectiveness of different modules.

This work builds upon our conference paper (Zhu et al., 2024c) and extends it in the following aspects: (1) It comprehensively discusses the local learning bias problem from the perspectives of the classifier and feature extractor of a local model. In addition to debiasing local classifiers as in the conference version, it proposes a novel CGDE module to eliminate the learning bias of local feature extractors, thereby achieving robust local training; (2) It provides an exhaustive review of existing methods focusing on the local learning bias problem; (3) It introduces three new datasets to further verify the effectiveness and generalization of the proposed method across various medical tasks; (4) The experimental results show that this extended version achieves better performance than the conference version, with significant improvements; (5) More comprehensive ablation experiments are conducted to verify the effectiveness of different modules of the proposed method and its scalability to different numbers of clients.

Roadmap. The rest of the paper is organized as follows. In Section 2, we review previous methods focusing on classifier debiasing, model alignment and data augmentation in FL. In Section 3, the proposed FedBM is introduced in detail. We describe the implementation details and verify of the effectiveness of the proposed FedBM in Section 4. Finally, the paper is closed with the conclusion in Section 5.

2. Related work

We introduce existing methods of classifier debiasing, model alignment and data augmentation in federated learning.

2.1. Classifier debiasing in federated learning

Federated learning (FL) provides a new solution to handle privacy concerns in distributed training. As the pioneering method, FedAvg (McMahan et al., 2017), trains a global model by aggregating local models from multiple clients without accessing their raw data. However, it undergoes considerable performance degradation when the data of clients are heterogeneous due to various imaging protocols, disease incidences, or population demographics. One of the main reasons is that data heterogeneity results in divergent local classifiers. Current approaches to this problem can be divided into three categories.

The first type of approaches (Luo et al., 2021; Huang et al., 2024; Zhou et al., 2024) aims to generate a balanced feature set to train local classifiers. For example, CCVR (Luo et al., 2021) exploits feature representations of all clients to build an approximated Gaussian mixture model, which is sent to each client for sampling more virtual representations. RUCR (Huang et al., 2024) broadcasts global prototypes to clients and arbitrarily fuses them and local features to synthesize virtual features. The second category (Huang et al., 2023; Dai et al., 2023; Tan et al., 2022; Qi et al., 2023; Long et al., 2023) tends to replace local classifiers with class prototypes. For instance, FedProto (Tan et al., 2022) directly aggregates prototypes of each class as local classifiers of clients. FedNH (Dai et al., 2023) produces uniformly-distributed class prototypes as initial local classifiers, and then smoothly infuses the class semantics into class prototypes. FPL (Huang et al., 2023) uploads feature representations of all clients to the server and clusters them to get different prototype centers for each class. These prototype centers are further averaged as local classifiers.

The third branch of works (Li et al., 2023; Zhang et al., 2024a) pre-constructs a fixed classifier before federated training. According to the theory of neural collapse (Papyan et al., 2020) that classifier vectors converge to an optimal simplex equiangular tight frame (ETF) when the dataset is balanced and sufficient, FedETF (Li et al., 2023) and FedKTL (Zhang et al., 2024a) introduce a synthetic simplex ETF as a fixed classifier for all clients. However, the orthogonal relation between classifier vectors is too strict and lacks of semantic interpretability. In this work, we propose to borrow linguistic knowledge from pre-trained language models to construct local classifiers.

2.2. Model alignment in federated learning

Data heterogeneity also leads to misalignment between client models, i.e., client-level variance, resulting in unstable and slow convergence during federated learning (Li et al., 2020a). FedProx (Li et al., 2020b) is the first work to solve this problem by introducing a proximal term into the objective during local training to restrict the distance between the current global model and the local model. SCAFFOLD (Karimireddy et al., 2020) introduces control variates to correct the drift in local updates. Nevertheless, the direct constraint in the parameter space may negatively affect model learning.

Apart from the above solutions, another way is to introduce the constraint in the feature space to solve this problem. For example, MOON (Li et al., 2021a) presents model-level contrastive learning to maximize the similarity between the features of local models in the current round and the global model and minimize the similarity between the features of local models of the current round and the previous round. FedFA (Zhou et al., 2024), FedFM (Ye et al., 2023), and FedPAC (Xu et al., 2023) collect local class prototypes to generate global prototypes. These global prototypes are sent to clients as the alignment objective of feature representations during local training. RUCR (Huang et al., 2024) pulls features within the same class towards corresponding global prototypes and pushes features of the other classes away. Although these prototype-based methods can improve representation learning, their performance highly relies on the quality of global prototypes. Unlike the existing methods, we directly narrow the distribution gap between client data.

2.3. Data augmentation in federated learning

Data Augmentation is a commonly-used way to relieve data heterogeneity issues in federated learning. (de Luca et al., 2022) have verified that some common data augmentation techniques can significantly improve out-of-domain generalization in federated settings, such as random cropping, horizontal flipping and color transformations. Besides, Mixup (Zhang, 2018) also obtains the widespread attention (Guo et al., 2023b; Yoon et al., 2021; Shin et al., 2020). For example, FedMix (Yoon et al., 2021) averages local batches to produce synthetic data. The server gathers these data and then sends them to clients. Clients combine these synthetic data with their local data to perform Mixup in local training.

In addition, FedOV (Diao et al., 2023) and FedOSS (Zhu et al., 2024a) are inspired by adversarial training and use fast gradient sign method (FGSM) to generate unknown samples. Moreover, FedRDN (Yan and Zhu, 2023) and CCVR (Luo et al., 2021) hypothesize that the images of each client are sampled from a multivariate Gaussian distribution. By sharing Gaussian distributions across clients, each client can sample augmented data to enhance local data, thereby reducing the domain gap. SDA-FL (Li et al., 2022) pre-trains a generative adversarial network (GAN) (Goodfellow et al., 2014) to generate synthetic data in each client. These synthetic data are then collected by the server to construct a global synthetic dataset to optimize global model. FedDiff (Mendieta et al., 2024) trains a class-conditioned diffusion model (Ho et al., 2020) on local data at the clients. These local diffusion models are sent to the server to generate data for training global model.

Data-free knowledge distillation is also a popular way to synthesize samples in the federated setting (Zhang et al., 2022b,a; Luo et al., 2024). For instance, DENSE (Zhang et al., 2022a) and FedFTG (Zhang et al., 2022b) utilize the ensemble client models to train a generator and then generate synthetic data to train global model at the server. However, these methods learn a generator to capture the mapping between random noises (from Gaussian distribution) and samples in the image space. The random noises lack semantically meaningful information and do not form well-organized class clusters, enabling the generator to produce low-quality images. The proposed FedBM framework introduces the text information of classes to remedy this disadvantage.

3. Method

This section first presents the workflow of FedBM and then introduces its submodules as well as optimization process.

3.1. Overview of FedBM

We present a Federated Bias eliMinating (FedBM) framework to remove local learning bias in heterogeneous federated learning. FedBM follows a standard FL training paradigm and consists of C distributed clients and a trustworthy server. Each client possesses a local dataset $\mathcal{D}^c = \{(\mathbf{x}_i^c, \mathbf{y}_i^c)\}_{i=1}^{N_c}$ with K classes, where N_c is the sample number of $\mathcal{D}^c$, and $\mathbf{x}_i^c$ is a training instance with the label $\mathbf{y}_i^c$. The goal of FedBM is to coordinate these clients to train a global model $\mathcal{F}(\mathrm{W}_{fe}, \mathrm{W}_{fc})$, where $\mathcal{F}$ contains a feature extractor $\mathcal{F}(\mathrm{W}_{fe})$ and a feature classifier $\mathcal{F}(\mathrm{W}_{fc})$. The overall training process proceeds through communication between clients and the server for multiple rounds. Concretely, we first pre-construct a high-quality global classifier via Linguistic Knowledge-based Classifier Construction (LKCC) before distributed training. Next, the cth client downloads the global feature extractor and classifier from the server to initialize the weights of its local model. During local training, all clients freeze local classifiers to avoid the classifier bias problem and only train their feature extractors. After local training, the local feature extractors $\mathcal{F}_c(\mathrm{W}_{fe}^c)$ of clients are uploaded to the server to update the global feature extractor via model aggregation: $\mathcal{F} = \frac{1}{C}\sum_{c\in[C]} \mathcal{F}_c(\mathrm{W}_{fe}^c)$. Concept-guided Global Distribution Estimation (CGDE) exploits the aggregated global model and concept prior to train a conditional generator that can capture the input space of the global model. The global feature extractor is sent to each client as the initialization of the next round. The generator is also distributed to per client to produce domain-invariant samples to regularize local updates in a consistent direction. The overall framework is shown in Fig. 2.

3.2. Linguistic knowledge-based classifier construction

Although pre-constructing a global classifier is a flexible way to mitigate local learning bias in classifiers of client models, it is very challenging since we do not have any knowledge about a high-quality classifier. Motivated by recent language-to-vision models (Radford et al., 2021; Zhang et al., 2024b), natural language descriptions (such as diagnosis reports) carry rich semantic information and can represent diverse images or clinical scans of different categories. Based on this insight, we propose Linguistic Knowledge-based Classifier Construction (LKCC), which exploits linguistic knowledge from pre-trained language models to construct a high-quality classifier for all clients.

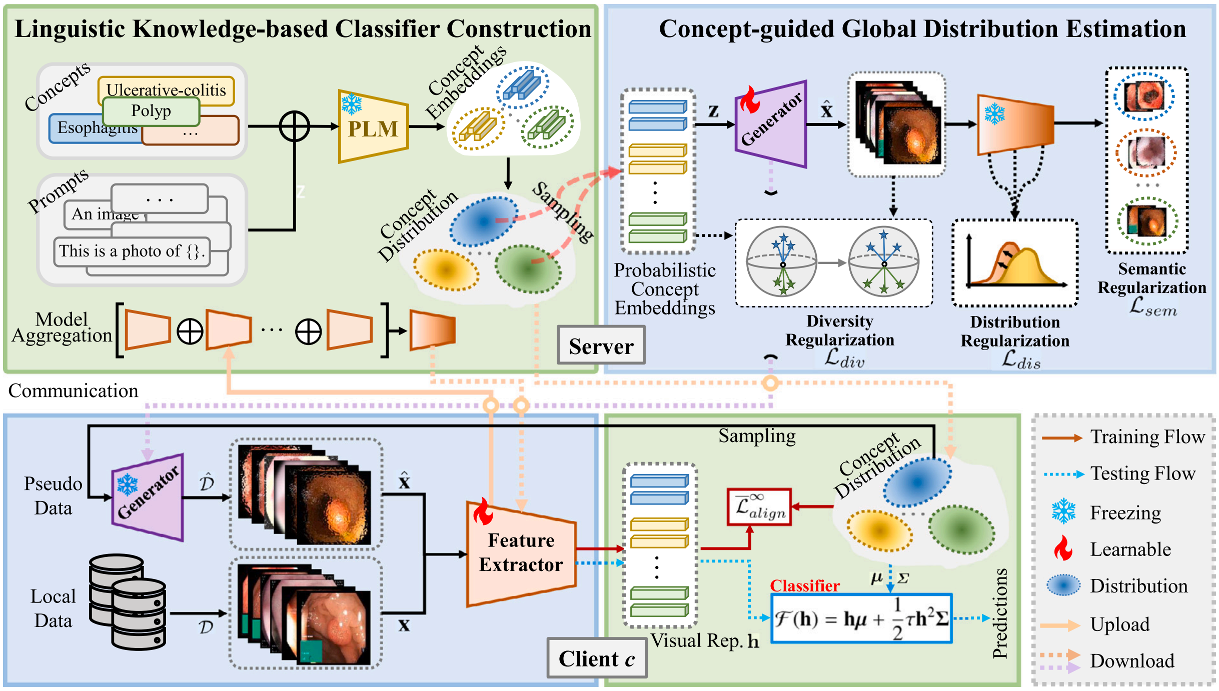


Fig. 2. The overview of the proposed FedBM framework. FedBM contains Linguistic Knowledge-based Classifier Construction (LKCC) and Concept-guided Global Distribution Estimation (CGDE). LKCC uses class concepts, prompts and PLMs to build latent concept distributions, which are sent to clients as local classifiers. CGDE samples probabilistic concept embeddings from the distributions to train a conditional generator. The generator is shared by all clients and produces pseudo data to calibrate updates of local feature extractors. (Best viewed in color).

As shown in Fig. 2, the server first collects a concept set $\{P_k\}_{k=1}^K$ from clients, where P_k is the class name of the kth class and K is the total class number. A set of M predetermined prompts (such as "This is an image of {concept}" and "The image shows {concept}". and so on) is used to contextualize the concepts. We input the contextualized concepts into a pre-trained language model (PLM) (such as the text encoder of BiomedCLIP Zhang et al., 2024b) to obtain a set of concept embeddings $\boldsymbol{E}$, where $\boldsymbol{E} = \cup_{k=1}^K\{e_1^{(k)}, e_2^{(k)}, \ldots, e_M^{(k)}\}$. PLM is trained on large-scale datasets based on contrastive learning and demonstrates strong feature transferability. Therefore, the obtained concept embeddings in $\boldsymbol{E}$ contain rich semantics, which has two favorable properties: (i) the distance relationship between concepts can be reflected through their similarities, (ii) concept embeddings in the linguistic space are domain-agnostic. Next, we employ these concept embeddings to build a high-quality classifier for clients.

Given the concept embeddings of each class, a natural idea is to regard them as a multi-way classifier to train the local feature extractor by performing alignment between image representations and these embeddings, which can be formulated as minimizing the following contrastive loss:

$$\mathcal{L}_{align} = \frac{1}{N_c}\sum_{i=1}^{N_c}\mathbb{E}_{e^{(\mathbf{y}_i)}\in\boldsymbol{\Omega}^{(\mathbf{y}_i)}}\left(-\log\frac{e^{\tau\mathbf{h}_i^{\mathrm{T}}e^{(\mathbf{y}_i)}}}{e^{\tau\mathbf{h}_i^{\mathrm{T}}e^{(\mathbf{y}_i)}}+\sum_{k\neq\mathbf{y}_i}^{K}\mathbb{E}_{e^{(k)}\in\boldsymbol{\Theta}^{(\mathbf{y}_i)}}e^{\tau\mathbf{h}_i^{\mathrm{T}}e^{(k)}}}\right), \quad (1)$$

where $\mathbf{h}_i = \frac{\mathcal{F}(\mathbf{x}_i)}{\|\mathcal{F}(\mathbf{x}_i)\|_2}$ is the normalized representation of a sample $\mathbf{x}_i$ of the client c. We add a fully-connected layer on top of the feature extractor to align the dimension of $\mathbf{h}_i$ and $e_m^{(\mathbf{y}_i)}$. $\boldsymbol{\Omega}^{(\mathbf{y}_i)}$ is the positive embedding set of the class $\mathbf{y}_i$ and contains the embeddings $\{e_1^{(\mathbf{y}_i)}, e_2^{(\mathbf{y}_i)}, \ldots, e_M^{(\mathbf{y}_i)}\}$. $\boldsymbol{\Theta}^{(\mathbf{y}_i)}$ is the negative embedding set and contains the concept embeddings of the other categories. τ is the temperature coefficient. Generally, more diverse prompts can obtain richer concept embeddings (corresponding to language descriptions) to comprehensively describe one class. Aligning image representations to these concept embeddings can force the model to learn exhaustive visual details. Hence, the performance of the model $\mathcal{F}$ highly depends on the number M of prompts under the supervision of Eq. (1). However, it is difficult to obtain all prompts for a specific task via prompt engineering. Besides, different prompts should not be treated equally due to different importance. Considering these issues, we propose to further generalize Eq. (1) to the infinite space, namely, aligning the image representations and the concept embedding distribution of each class.

Assuming that the concept embeddings $\{e_1^{(k)}, e_2^{(k)}, \ldots, e_M^{(k)}\}$ of the kth class are sampled from a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, we compute the mean $\boldsymbol{\mu}_k$ and variance $\boldsymbol{\Sigma}_k$ as follows:

$$\boldsymbol{\mu}_k = \frac{1}{M}\sum_{m=1}^{M}e_m^{(k)}, \qquad \boldsymbol{\Sigma}_k = \frac{1}{M-1}\sum_{m=1}^{M}(e_m^{(k)}-\boldsymbol{\mu}_k)(e_m^{(k)}-\boldsymbol{\mu}_k). \quad (2)$$

After estimating the distributions $\{\mathcal{N}^{(k)}\}_{k=1}^K$ of all classes, we can sample infinite concept embeddings, which correspond to instances with different characteristics in the image space. In the context, Eq. (1) can be reformulated as

$$\mathcal{L}_{align}^{\infty} = \frac{1}{N_c}\sum_{i=1}^{N_c}\mathbb{E}_{e^{(\mathbf{y}_i)}\sim\mathcal{N}^{(\mathbf{y}_i)}}\left(-\log\frac{e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}}}{e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}}+\sum_{k\neq\mathbf{y}_i}^{K}\mathbb{E}_{e^{(k)}\sim\mathcal{N}^{(k)}}e^{\tau\mathbf{h}_i e^{(k)}}}\right). \quad (3)$$

$\mathcal{L}_{align}^{\infty}$ is difficult to compute its exact form when the sampled concept embeddings are infinite. Here, we can derive its upper bound based on the existing method (Wang et al., 2019) and find a surrogate loss $\overline{\mathcal{L}}_{align}^{\infty}$:

$$\mathcal{L}_{align}^{\infty} \leq \overline{\mathcal{L}}_{align}^{\infty} = \frac{1}{N^c}\sum_{i=1}^{N_c}\left(-\log\frac{e^{\mathcal{F}(\mathbf{h}_i,\mathbf{y}_i)}}{\sum_{k=1}^{K}e^{\mathcal{F}(\mathbf{h}_i,k)}}+\frac{\tau^2}{2}\mathbf{h}_i^2\boldsymbol{\Sigma}_{(\mathbf{y}_i)}\right), \quad (4)$$

where $\mathcal{F}(\mathbf{h}_i, k) = \mathbf{h}_i\boldsymbol{\mu}_{(k)} + \frac{1}{2}\tau\mathbf{h}_i^2\boldsymbol{\Sigma}_{(k)}$. The detailed derivation is shown in Appendix A. By minimizing the loss $\overline{\mathcal{L}}_{align}^{\infty}$, we can implement the alignment between the image representations and the concept embedding distributions. It can be observed that $\overline{\mathcal{L}}_{align}^{\infty}$ is a softmax-based cross-entropy loss over $\mathcal{F}(\mathbf{h}_i, k)$, with a constraint on variance of features. Therefore, we redefine the local classifier as

$$\mathcal{F}(\mathbf{h}) = \mathbf{h}\boldsymbol{\mu} + \frac{1}{2}\tau\mathbf{h}^2\boldsymbol{\Sigma}, \quad (5)$$

where $\mathbf{h} \in \mathbb{R}^{B\times D}$, $\boldsymbol{\mu} \in \mathbb{R}^{D\times K}$ and $\boldsymbol{\Sigma} \in \mathbb{R}^{D\times K}$. B, D and K are the batch size, the feature dimension and the class number. $\mathcal{F}(\mathbf{h}) \in \mathbb{R}^{B\times K}$ denotes the logit outputs of the batch images. During the inference phase, the calculation of $\mathcal{F}(\mathbf{h}) \in \mathbb{R}^{B\times K}$ do not rely on the class labels of $\mathbf{h}$. Noticeably, the construction process is only conducted once and does not incur a high computation cost. Meanwhile, the constructed classifier only needs one round of transmission, which reduces communication overhead.

Essentially, Eq. (1) averages concept embeddings as the local classifier and treats all prompts equally. By comparison, the classifier $\mathcal{F}(\mathbf{h})$ in Eq. (5) considers the variance of the concept embeddings and thus are more robust to match the semantic diversification of image representations, thereby achieving more accurate classification. Additionally, concept embedding distributions are derived from embeddings generated by a pre-trained language model (PLM). Since PLM is trained on large-scale datasets, these embeddings carry rich semantic information and can represent specific image samples. A concept embedding distribution can represent a specific cluster of samples, enabling the classification model to easily capture the relationship between concept embedding distributions and visual images. In contrast to FedETF (Li et al., 2023) that utilizes orthogonal initialization to construct the classifier, our method has stronger semantic interpretability and the elastic constraints on the inter-class angular margin.

3.3. Concept-guided global distribution estimation

Data heterogeneity causes local client models to gradually forget the global knowledge learned in previous rounds during local training because they can only receive local data information and thus always optimize towards their own local distributions, resulting in inconsistency in local feature extractors $\mathcal{F}_c(\mathrm{W}^c_{fe})$. It has been confirmed that the inconsistency will incur sharper loss landscape and performance degradation of the global model (Shi et al., 2023; Chen et al., 2021). To break this dilemma, we propose Concept-guided Global Distribution Estimation (CGDE) to train a conditional generator based on the aggregated model and concept prior at the server, which can generate data that have a similar distribution to the input space of the global model. These generated data are combined with local data to train local feature extractors and restrict local updates, reducing local learning bias, as illustrated in Fig. 2.

Specifically, we consider a conditional generator $\mathcal{G}(\cdot)$ with the parameters $\mathrm{W}_{\mathcal{G}}$, which takes a condition pair $(\mathbf{z}_i, \hat{\mathbf{y}}_i)$ as input to generate the synthetic sample $\hat{\mathbf{x}}_i = \mathcal{G}(\mathbf{z}_i, \mathrm{W}_{\mathcal{G}})$. The label $\hat{\mathbf{y}}_i$ is randomly sampled from a uniform distribution. Generally, synthetic data generated by a well-trained conditional generator should satisfy several key characteristics: **semantic similarity**, **data diversity**, **distribution consistency**, and **interpretability of conditions**. Towards these goals, we introduce some regularization terms into the optimization objective to restrict the training of the generator $\mathcal{G}(\cdot)$ from these aspects, ensuring its quality and utility.

Semantic Regularization The generator is expected to produce synthetic samples with semantic similarity to instances in the input space of the global model. To put it differently, we hope that a generated sample $\hat{\mathbf{x}}_i$ can be classified into the class $\hat{\mathbf{y}}_i$ with a high probability by the global model. Therefore, we treat the global model $\mathcal{F}$ as the teacher model to optimize the generator $\mathcal{G}(\cdot)$ via the following loss:

$$\mathcal{L}_{sem} = CE(\mathcal{F}(\mathcal{G}(\mathbf{z}_i, \mathrm{W}_{\mathcal{G}}), \mathrm{W}_{fe}, \mathrm{W}_{fc}), \hat{\mathbf{y}}_i), \tag{6}$$

where $\mathcal{F}(\mathcal{G}(\mathbf{z}_i, \mathrm{W}_{\mathcal{G}}), \mathrm{W}_{fe}, \mathrm{W}_{fc})$ is the logit values of the generated sample $\hat{\mathbf{x}}_i$. $CE(\cdot)$ denotes the cross-entropy function. The parameters $\mathrm{W}_{fe}, \mathrm{W}_{fc}$ of the global model are frozen. By minimizing $\mathcal{L}_{sem}$, the generator is forced to generate pseudo data to capture the input space (data distribution) of the global model.

Diversity Regularization If we only employ the loss $\mathcal{L}_{sem}$, the generator probably undergoes the mode collapse problem and fails to achieve a good performance (Bang and Shim, 2021). This problem is caused by shortcut learning of deep neural networks (Geirhos et al., 2020). For example, given two condition codes $\mathbf{z}_i$ and $\mathbf{z}_j$, with the same class label $\hat{\mathbf{y}}$, the synthetic samples $\hat{\mathbf{x}}_i = \mathcal{G}(\mathbf{z}_i, \mathrm{W}_{\mathcal{G}})$ and $\hat{\mathbf{x}}_j = \mathcal{G}(\mathbf{z}_j, \mathrm{W}_{\mathcal{G}})$ may be collapsed into a point (namely, repeating samples), leading to low data diversity. To tackle this issue, we propose a diversity regularization loss to dynamically penalize the distance between $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$:

$$\mathcal{L}_{div} = \frac{|\mathbf{z}_i - \mathbf{z}_j|}{|\mathcal{G}(\mathbf{z}_i, \mathrm{W}_{\mathcal{G}}) - \mathcal{G}(\mathbf{z}_j, \mathrm{W}_{\mathcal{G}})|} \tag{7}$$

where $|\cdot|$ denotes the L_1 norm distance. We minimize the loss $\mathcal{L}_{div}$ to provide a greater punishment on the distance between $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ when $\mathbf{z}_i$ and $\mathbf{z}_j$ are closer, thus encouraging the generator to create diverse images.

Distribution Regularization To improve the training stability of the generator, we follow the previous methods (Yin et al., 2020; Liu et al., 2021) to use a distribution regularization loss to align feature map statistics of the synthetic data at the Batch Normalization (BN) layer with their running counterparts:

$$\mathcal{L}_{dis} = \sum_l \left(\|\mu_l(\hat{\mathbf{x}}) - \mu_l\| + \|\sigma_l^2(\hat{\mathbf{x}}) - \sigma_l^2\| \right), \tag{8}$$

where $\mu_l(\hat{\mathbf{x}})$ and $\sigma_l^2(\hat{\mathbf{x}})$ are the batch-wise mean and variance estimates of feature maps corresponding to the lth BN layer of the generator $\mathcal{G}(\cdot)$. μ_l and σ_l^2 are the mean and variance of the lth BN layer of the global model. $\|\cdot\|$ denotes the L_2 norm distance.

Explainable Conditions A common type of the condition $\mathbf{z}_i$ is random noise sampled from standard normal distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ in previous methods (Zhu et al., 2021b; Yu et al., 2023). The generator is expected to learn the mapping between the noise space and the input space of the global model by minimizing $\mathcal{L}_{sem}$. However, there are two drawbacks to this strategy: (1) randomly-sampled noises do not have any semantic information and interpretability; (2) the correspondence between random noises $\mathbf{z}$ and $\hat{\mathbf{y}}$ are unclear and enables the generator to produce a lot of low-quality or repeating samples. Due to the drawbacks, it is difficult to learn a well-behaved generator to capture the global distribution of the global model. To solve this problem, we propose to build the condition pair $(\mathbf{z}_i, \hat{\mathbf{y}}_i)$ based on the concept embedding space. In particular, we first randomly sample the pseudo label $\hat{\mathbf{y}}_i$ from a uniform distribution. Then the corresponding $\mathbf{z}_i$ is probabilistic concept embedding sampled from the distribution $\mathcal{N}(\boldsymbol{\mu}_{\hat{\mathbf{y}}_i}, \boldsymbol{\Sigma}_{\hat{\mathbf{y}}_i})$ of the $\hat{\mathbf{y}}_i$th class built in Eq. (2), and is fed into the generator:

$$\hat{\mathbf{x}}_i = \mathcal{G}(\mathbf{z}_i, \mathrm{W}_{\mathcal{G}}), \quad \mathbf{z}_i \sim \mathcal{N}(\boldsymbol{\mu}_{\hat{\mathbf{y}}_i}, \boldsymbol{\Sigma}_{\hat{\mathbf{y}}_i}), \tag{9}$$

where $\hat{\mathbf{y}}_i$ is the label of the sample $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{y}}_i \in [1, K]$. Compared with the distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$, the concept embedding distributions $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ are organized and class anchor-centered, making it easier for the generator to learn the mapping between the latent embedding space and the input space of global model. It is worth mentioning that the training of the generator only relies on the global model and concept embedding distributions, without requiring the private data of clients, thereby minimizing the communication and computational load.

3.4. Optimization and theoretical derivation

We present the pseudo code of the proposed FedBM framework in Algori Algorithm 1. The training for FedBM includes the optimization of the generator and local models. The generator $\mathcal{G}$ is supervised by the following hybrid loss function, including the three regularization terms:

$$\mathcal{L}_{generator} = \mathcal{L}_{sem} + \lambda_{div}\mathcal{L}_{div} + \lambda_{dis}\mathcal{L}_{dis}. \tag{10}$$

With these regularization terms, the generator is able to generate diverse data to capture the global data distribution of the global model. In experiments, we update the generator with multiple rounds of communication as a cycle to reduce computational and communication costs. The updated generator $\mathcal{G}$ is sent to all clients to synthesize pseudo data. Assuming that $\hat{D}^c$ is the local synthesized dataset at the cth client

Algorithm 1 The FedBM framework for heterogeneous FL.

```
Input: The client number $C$, the number $E$ of local epochs, the local
    datasets $\{\mathcal{D}^c\}_{c=1}^{C}$.
Server executes:
//Constructing the global classifier via LKCC
Collecting concepts from clients to form a set $\{P_k\}_{k=1}^{K}$,
Using PLMs to obtain a set of concept embeddings $\boldsymbol{E}$,
Building concept embedding distributions via Eq. (2),
Defining the global classifier via Eq. (5),
//Broadcasting the global classifier only once
for each communication round do
  //Estimating global distribution via CGDE
  Training a generator $\mathcal{G}(\cdot)$ via Eq. (10),
  for each client $c = 1, 2, ..., C$ do
    $\mathrm{W}^c_{fe} \leftarrow$ ClientUpdate($\mathcal{G}, \mathrm{W}_{fe}, \mathrm{W}_{fc}$).
  end for
  Model aggregation: $\mathrm{W}_{fe} \leftarrow \sum_{c=1}^{C} \mathrm{W}^c_{fe}$.
end for
Client executes:
ClientUpdate($\mathcal{G}$, $\mathrm{W}_{fe}$, $\mathrm{W}_{fc}$):
   Using $\mathrm{W}_{fe}$ to initialize $\mathrm{W}^c_{fe}$,
   Using $\mathcal{G}$ to generate the synthesized dataset $\hat{\mathcal{D}}^c$,
   for each epoch $e = 1, 2, ..., E$ do
      $\min_{\mathrm{W}^c_{fe}} \mathbb{E}_{(\mathbf{x}_i,\mathbf{y}_i)\sim\mathcal{D}^c\cup\hat{\mathcal{D}}^c} \overline{\mathcal{L}}^{\infty}_{align}(\mathcal{F}(\mathbf{x}_i, \mathrm{W}^c_{fe}, \mathrm{W}_{fc}), \mathbf{y}_i)$.
   end for
Output: The global model $\mathcal{F}(\mathrm{W}_{fe}, \mathrm{W}_{fc})$.
```

and is updated in each round of training, we utilize $\hat{\mathcal{D}}^c$ and the local original data $\mathcal{D}^c$ to train the local feature extractor W^c_{fe} by minimizing the surrogate loss $\overline{\mathcal{L}}^{\infty}_{align}$:

$$\min_{\mathrm{W}^c_{fe}} \mathbb{E}_{(\mathbf{x}_i,\mathbf{y}_i)\sim\mathcal{D}^c\cup\hat{\mathcal{D}}^c} \overline{\mathcal{L}}^{\infty}_{align}(\mathcal{F}(\mathbf{x}_i, \mathrm{W}^c_{fe}, \mathrm{W}_{fc}), \mathbf{y}_i), \tag{11}$$

where $\mathbf{x}_i$ is sampled from the union of $\mathcal{D}^c$ and $\hat{\mathcal{D}}^c$. Since the generator contains global knowledge and is shared by all clients, it can guide the optimization of these clients towards a consistent direction. In addition, the generator is very lightweight and does not incur a big communication overhead.

4. Experiment

We present the experimental setup, the comparison results against previous methods, and the ablation results.

4.1. Experiment setup

4.1.1. Datasets

To investigate the effectiveness of our FedBM framework, we evaluated it in five public datasets.

OCT-C8 (Subramanian et al., 2022) consists of 24,000 retinal OCT images and is divided into eight (categoriesage-related macular degeneration, choroidal neovascularisation, diabetic macular edema, drusen, macular hole, diabetic retinopathy, central serous retinopathy and one for healthy class). Based on the official division, 18,400 images are used for training, 2800 for validation, and 2800 for testing.

Kvasir-v2 (Pogorelov et al., 2017) contains 8000 endoscopic images of the gastrointestinal tract. These images belong to 8 categories, *i.e.*, esophagitis, cecum, pylorus, Z-line, polyps, ulcerative colitis, dyed lifted polyp, dyed resection margin. We randomly partition these samples into training, validation, and test sets with a ratio of 7 : 1 : 2.

HAM1000 (Tschandl et al., 2018) has 10,015 dermatoscopic images, which are from different populations. These images belong to 8 categories: actinic keratoses or intraepithelial carcinoma (akiec), basal cell carcinoma, benign keratosis-like lesions, dermatofibroma, melanoma, melanocytic nevi, and vascular lesions. Following the existing work (Yang et al., 2023), 7007 images were used for training, 1003 for validation, and 2005 for testing.

PBC (Acevedo et al., 2020) contains a total of 17,092 microscopic peripheral blood cell images. The dataset is organized into the eight groups: neutrophils, eosinophils, basophils, lymphocytes, monocytes, immature granulocytes, erythroblasts and platelets or thrombocytes. Following the existing work (Yang et al., 2023), 11,959 images are used for training, 1712 for validation, and 3421 for testing.

FEMNIST (Caldas et al., 2018) is a part of the LEAF benchmark. It comprises 814,277 handwritten digit, lowercase, and uppercase letter images from 3597 users, belonging to 62 classes. To ensure data heterogeneity, we select users whose sample sizes are fewer than 150, resulting in 396 users. For each experiment, 50 users are randomly selected from this group to conduct federated training. They are treated as individual client and randomly divided into training set (35 users), validation set (5 users), and test set (10 users) with a ratio of 7: 1: 2.

4.1.2. Implementation details

The proposed FedBM and comparison methods are implemented with PyTorch library. We adopt the ResNet-18 (He et al., 2016) as the backbone network of all methods. The number of clients is set to 12 for OCT-C8 and PBC, and 10 for Kvasir-v2 and HAM1000 datasets, respectively. The numbers of local epochs and communication rounds are set to 2 and 200 for OCT-C8 and Kvasir-v2 datasets, 1 and 200 for HAM1000 dataset, 10 and 50 for PBC dataset, and 5 and 200 for FEMNIST dataset, respectively. For all datasets, we utilize the Adam (Kingma and Ba, 2014) optimizer with the initial learning rate of 1×10^{-2}. The batch size is set to 8 and the learning rate decays at a rate of 0.99 per epoch. The client sampling ratio is 0.5 except for FEMNIST (0.1). Similar to existing FL works (Luo et al., 2021; Li et al., 2023), we use Dirichlet distribution on label ratios to simulate the Non-IID data distribution among clients. We set the Dirichlet parameter β as 0.05 and 0.1 to ensure the high data heterogeneity. The weight λ_{div} is set to 1. λ_{dis} and the batchsize of local synthesized datasets are selected from sets $[0.1, 1]$ and $[8, 16, 32]$ by grid search, respectively. The predetermined prompts can refer to the work (Zhu et al., 2024c). Two commonly-used metrics, accuracy, and F1 score, are used to measure the classification performance. In all the experiments, we conduct three trials for each setting and present the mean and the standard deviation.

4.2. Comparison with state-of-the-art methods

We compare our FedBM framework with the state-of-the-art FL approaches in four datasets, including FedAvg (McMahan et al., 2017), FedDyn (Acar et al., 2021), FedProx (Li et al., 2020b), FedREP (Collins et al., 2021), FedROD (Chen and Chao, 2021), FedNH (Dai et al., 2023), FedProto (Tan et al., 2022), FedETF (Li et al., 2023) and Scaffold (Karimireddy et al., 2020). For a fair comparison, these methods are implemented in the standard FL framework, using the same split as FedBM for each dataset. The common hyperparameters for all methods are consistent and are determined by FedAvg (McMahan et al., 2017), such as learning rate, number of training round and so on. Following FedETF, we set μ to 0.001 in FedPROX and 0.01 in FedDyn as suggested in their official implementations, and set $\gamma = 1$ in FedROD. For FedNH, the smoothing hyperparameter $\rho = 0.9$ as suggested in the original paper.

Comparison Results on OCT-C8: In Table 1, we show the classification performance of different methods on OCT-C8 dataset to validate the proposed FedBM. Although both FedProx and FedDyn introduce constraints to the parameters of local models, the latter achieves better performance. Our FedBM suppresses FedProx with significant performance increments for two cases, such as 9.65% in F1-score for $\beta = 0.05$ and 12.52% in Accuracy for $\beta = 0.1$. Furthermore, in contrast to

Table 1
The performance comparison of the proposed method and existing methods on OCT-C8 dataset.

Methods	$\beta = 0.05$		$\beta = 0.1$	
	Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
Central Learning	93.85 ± 0.24	93.84 ± 0.22	93.85 ± 0.24	93.84 ± 0.22
FedAvg (McMahan et al., 2017)	74.82 ± 5.98	72.34 ± 6.87	78.64 ± 5.44	76.25 ± 7.23
FedDyn (Acar et al., 2021)	70.79 ± 1.94	65.88 ± 5.12	73.46 ± 5.49	69.86 ± 7.96
FedProx (Li et al., 2020b)	76.60 ± 5.43	74.49 ± 6.12	78.37 ± 5.76	75.64 ± 7.68
FedREP (Collins et al., 2021)	43.87 ± 7.24	32.98 ± 8.83	59.37 ± 12.81	55.20 ± 12.56
FedROD (Chen and Chao, 2021)	70.20 ± 4.17	64.24 ± 4.21	79.11 ± 5.62	77.65 ± 7.17
FedNH (Dai et al., 2023)	80.08 ± 4.23	78.61 ± 5.25	82.53 ± 2.57	82.25 ± 2.82
FedProto (Tan et al., 2022)	28.86 ± 2.15	16.95 ± 1.79	37.07 ± 1.40	24.88 ± 2.32
FedETF (Li et al., 2023)	77.79 ± 5.17	74.47 ± 8.64	82.81 ± 3.76	81.67 ± 5.31
Scaffold (Karimireddy et al., 2020)	72.96 ± 5.22	70.55 ± 7.80	79.34 ± 3.97	78.38 ± 4.78
FedBM	**84.84** ± 2.77	**84.55** ± 2.78	**88.32** ± 1.49	**88.16** ± 1.44

Table 2
The performance comparison of the proposed method and existing methods on Kvasir-v2 dataset.

Methods	$\beta = 0.05$		$\beta = 0.1$	
	Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
Central Learning	77.15 ± 0.03	77.01 ± 0.10	77.15 ± 0.03	77.01 ± 0.10
FedAvg (McMahan et al., 2017)	60.10 ± 5.76	54.07 ± 9.58	67.02 ± 1.72	63.93 ± 2.49
FedDyn (Acar et al., 2021)	55.20 ± 3.03	49.84 ± 3.86	63.18 ± 2.19	60.98 ± 1.41
FedProx (Li et al., 2020b)	59.68 ± 2.02	53.11 ± 3.26	68.77 ± 1.45	66.81 ± 2.53
FedREP (Collins et al., 2021)	33.06 ± 15.71	23.88 ± 13.72	48.95 ± 0.62	39.71 ± 2.50
FedROD (Chen and Chao, 2021)	61.79 ± 2.72	58.83 ± 3.40	70.10 ± 3.70	68.01 ± 5.71
FedNH (Dai et al., 2023)	61.08 ± 5.68	54.15 ± 8.68	69.10 ± 1.74	65.71 ± 3.32
FedProto (Tan et al., 2022)	26.79 ± 1.97	14.99 ± 1.37	35.32 ± 1.59	23.33 ± 1.30
FedETF (Li et al., 2023)	63.77 ± 3.73	60.12 ± 6.34	69.70 ± 3.56	67.15 ± 6.64
Scaffold (Karimireddy et al., 2020)	59.06 ± 5.41	54.25 ± 5.41	66.16 ± 1.80	64.88 ± 1.46
FedBM	**71.31** ± 2.81	**69.51** ± 4.11	**74.14** ± 1.89	**73.75** ± 1.79

FedETF, which employs orthogonal initialization to build local classifiers, our method obtains superior performance with a remarkable increments of 2.66% ($\beta = 0.05$) and 2.89% ($\beta = 0.1$) in F1-score. Noticeably, FedBM outperforms the second-best method, i.e., FedNH, by tremendous performance gaps for two cases, including 5.39% in F1-score ($\beta = 0.05$) and 5.79% (P-value < 0.05) in Accuracy ($\beta = 0.1$).

Comparison Results on Kvasir-v2: The performance of previous methods and FedBM on Kvasir-v2 dataset are demonstrated in Table 2. It can be observed that all previous methods implement poor performance. FedREP and FedProto even fail to converge when the data of clients are seriously heterogeneous ($\beta = 0.05$). The proposed FedBM exceeds the second-best approach, FedETF, with the overwhelming performance advantages in two cases, such as 7.54% (P-value < 0.06) and 9.39 (P-value < 0.01), 4.44% and 6.60% in Accuracy and F1-score for $\beta = 0.05$ and $\beta = 0.1$, respectively. Meanwhile, FedETF also shows a larger performance drop (5.93% in Accuracy and 7.03% in F1-score) from $\beta = 0.1$ to 0.05, while FedBM only undergoes 2.83% in Accuracy and 4.24% in F1-score.

Comparison Results on HAM1000: Table 3 presents the performance of existing methods and FedBM on HAM1000 dataset. Among existing methods, FedROD obtains a relatively good performance with 67.99% in Accuracy and 46.08% in F1-score when β is 0.1. However, when data becomes more heterogeneous ($\beta = 0.05$), FedROD suffers from a significant performance degradation, merely achieving 60.89% in Accuracy and 37.90% in F1-score, with 7.10% and 8.18% of drops, respectively. By comparison, FedBM outperforms FedROD with remarkable performance improvements in two cases, such as 5.56% and 11.86% in Accuracy, 3.41% and 7.53% in F1-score for $\beta = 0.05$ and $\beta = 0.1$ (P-value < 0.05), respectively. It is worth noting that FedBM only experiences a slight performance drops (0.79% in Accuracy and 4.06% in F1-score).

Comparison Results on PBC: Table 4 shows the performance of existing approaches and FedBM on PCB dataset. We can observe that FedDyn, FedREP and FedProto are difficult to converge, yielding very low performance. FedProx and FedETF achieve good performance (87.37% and 87.86% in Accuracy, 85.39% and 84.33% in F1-score, respectively) when the heterogeneity parameter β is 0.1. They only show a small performance gap compared with FedBM (89.88% and 87.71% in Accuracy and F1-score). FedBM remarkably surpasses FedProx and FedETF when β is 0.05, with prominent increments of 8.05% and 6.99% in Accuracy, 5.39% and 10.76% in F1-score, respectively. Although Scaffold exceeds FedBM when $\beta = 0.1$, it undergoes a severe performance decline when β becomes 0.05, showing the vulnerability to data heterogeneity.

Comparison Results on FEMNIST: Table 5 presents the classification performance of different approaches on FEMNIST. It can be observed that FedProto obtains the worst performance, since only transmitting prototypes cannot handle the data heterogeneity problem when each client has limited data. FedNH performs better than FedProto because it not only transmits prototypes to calibrate classifiers but also aggregates feature extractors across clients. In comparison, the proposed FedBM outperforms FedNH by a considerable margin in both Accuracy (1.56%) and F1-score (2.43%). Among all methods, Scaffold yields the second-best accuracy score (75.98%) but obtains the inferior result in F1-score. FedBM shows the best performance in both Accuracy (75.98%) and F1-score (54.77%).

These experimental results on the five datasets demonstrate the remarkable performance advantage of our method over state-of-the-art FL methods under different heterogeneous settings. Additionally, FedBM also shows more stable performance against the data heterogeneity than existing approaches.

4.3. Ablation study

4.3.1. Evaluation of different modules

LKCC and CGDE are two indispensable components of our FedBM framework to alleviate the data heterogeneity issue. To evaluate their contributions, we individually remove them to observe the performance of FedBM. As illustrated in Table 6, FedBM experiences significant performance decline once we remove LKCC (w/o LKCC), with decrements

Table 3
The performance comparison of the proposed method and existing methods on HAM1000 dataset.

Methods	$\beta = 0.05$		$\beta = 0.1$	
	Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
Central Learning	74.05 ± 2.19	51.03 ± 0.91	74.05 ± 2.19	51.03 ± 0.91
FedAvg (McMahan et al., 2017)	67.43 ± 3.59	31.84 ± 1.90	68.77 ± 3.32	41.25 ± 1.86
FedDyn (Acar et al., 2021)	67.84 ± 1.14	20.33 ± 0.70	66.48 ± 4.01	35.79 ± 1.14
FedProx (Li et al., 2020b)	67.56 ± 2.64	35.53 ± 6.32	69.42 ± 0.77	38.03 ± 7.17
FedREP (Collins et al., 2021)	61.74 ± 2.57	25.67 ± 1.99	59.98 ± 4.04	30.57 ± 2.04
FedROD (Chen and Chao, 2021)	60.89 ± 1.03	37.90 ± 4.77	67.99 ± 0.61	46.08 ± 0.87
FedNH (Dai et al., 2023)	64.95 ± 4.62	26.45 ± 5.48	58.28 ± 9.37	35.97 ± 5.00
FedProto (Tan et al., 2022)	36.62 ± 2.53	11.50 ± 1.15	41.06 ± 1.12	13.72 ± 1.28
FedETF (Li et al., 2023)	67.36 ± 1.16	33.67 ± 4.33	64.67 ± 2.26	43.45 ± 0.84
Scaffold (Karimireddy et al., 2020)	67.88 ± 1.36	27.19 ± 5.05	66.96 ± 2.83	36.99 ± 4.35
FedBM	**72.75** ± 0.88	**45.43** ± 3.13	**73.55** ± 0.28	**49.49** ± 1.74

Table 4
The performance comparison of the proposed method and existing methods on PBC dataset.

Methods	$\beta = 0.05$		$\beta = 0.1$	
	Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
Central Learning	97.43 ± 0.06	97.26 ± 0.09	97.43 ± 0.06	97.26 ± 0.09
FedAvg (McMahan et al., 2017)	63.11 ± 8.13	47.98 ± 9.37	78.61 ± 12.65	71.73 ± 16.50
FedDyn (Acar et al., 2021)	21.18 ± 3.89	6.19 ± 2.83	28.90 ± 6.28	15.76 ± 5.04
FedProx (Li et al., 2020b)	79.51 ± 0.96	72.49 ± 3.33	87.37 ± 5.34	85.39 ± 7.05
FedREP (Collins et al., 2021)	37.49 ± 2.61	26.47 ± 3.18	42.74 ± 7.21	33.66 ± 9.70
FedROD (Chen and Chao, 2021)	75.89 ± 1.61	67.89 ± 2.03	79.81 ± 12.38	77.87 ± 13.18
FedNH (Dai et al., 2023)	74.36 ± 5.16	64.22 ± 5.63	84.69 ± 7.32	82.44 ± 8.47
FedProto (Tan et al., 2022)	30.24 ± 2.23	18.17 ± 1.70	41.15 ± 2.52	28.28 ± 1.88
FedETF (Li et al., 2023)	77.91 ± 2.23	69.78 ± 4.90	87.86 ± 4.19	84.33 ± 6.73
Scaffold (Karimireddy et al., 2020)	58.03 ± 7.06	45.55 ± 4.84	**93.65** ± 2.42	**93.35** ± 2.52
FedBM	**84.90** ± 3.29	**80.54** ± 5.53	89.88 ± 4.47	87.71 ± 6.09

Table 5
The performance comparison of the proposed method and existing methods on FEMNIST dataset.

Methods	Accuracy (%)	F1-score (%)
Central Learning	79.13 ± 1.27	58.89 ± 1.35
FedAvg (McMahan et al., 2017)	73.06 ± 2.09	53.83 ± 3.58
FedDyn (Acar et al., 2021)	69.17 ± 2.41	44.04 ± 2.44
FedProx (Li et al., 2020b)	73.60 ± 1.70	50.99 ± 1.65
FedREP (Collins et al., 2021)	50.92 ± 2.58	22.71 ± 1.74
FedROD (Chen and Chao, 2021)	71.66 ± 0.53	51.48 ± 2.35
FedNH (Dai et al., 2023)	74.42 ± 2.02	52.34 ± 2.01
FedProto (Tan et al., 2022)	28.57 ± 1.16	10.19 ± 0.22
FedETF (Li et al., 2023)	73.48 ± 1.17	53.98 ± 1.13
Scaffold (Karimireddy et al., 2020)	74.55 ± 1.19	51.61 ± 1.37
FedBM	**75.98** ± 2.15	**54.77** ± 0.81

Table 6
The performance of the proposed FedBM framework with different modules.

Datasets	Methods	$\beta = 0.05$		$\beta = 0.1$	
		Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
OCT-8	w/o LKCC	83.02 ± 2.45	82.94 ± 2.47	85.20 ± 3.54	84.92 ± 3.72
	w/o CGDE	79.14 ± 3.77	77.13 ± 5.10	85.00 ± 2.66	84.56 ± 2.99
	FedBM	84.84 ± 2.77	84.55 ± 2.78	88.32 ± 1.49	88.16 ± 1.44
Kvasir-v2	w/o LKCC	28.04 ± 9.74	21.18 ± 7.12	23.68 ± 5.79	16.07 ± 7.13
	w/o CGDE	65.00 ± 4.36	61.67 ± 6.61	70.90 ± 1.97	68.73 ± 3.36
	FedBM	71.31 ± 2.81	69.51 ± 4.11	74.14 ± 1.89	73.75 ± 1.79

Table 7
The performance of LKCC with different proportions of prompts. LKCC (25%) indicates that only 25% of prompts are used.

Methods	$\beta = 0.05$		$\beta = 0.1$	
	Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
LKCC (25%)	76.64 ± 6.26	76.05 ± 6.48	83.71 ± 2.83	83.45 ± 3.05
LKCC (50%)	76.82 ± 5.41	76.54 ± 4.88	82.40 ± 3.17	82.23 ± 3.30
LKCC (75%)	77.70 ± 3.53	75.71 ± 4.51	84.12 ± 4.18	83.39 ± 4.79
LKCC (100%)	79.14 ± 3.77	77.13 ± 5.10	85.00 ± 2.66	84.56 ± 2.99

of 1.61% ($\beta = 0.05$) and 3.24% ($\beta = 0.1$) on OCT-8, and 48.33% ($\beta = 0.05$) and 57.68% ($\beta = 0.1$) on Kvasir-v2 in F1-score. Global model with a biased classifier leads to low-quality generated samples. Exploiting these samples to train local models incurs performance degradation. Moreover, we can observe that discarding CGDE (w/o CGDE) also leads to a substantial performance drop, with decrements of 5.70% ($\beta = 0.05$) and 3.32% ($\beta = 0.1$) on OCT-8, and 6.31% ($\beta = 0.05$) and 3.24% ($\beta = 0.1$) on Kvasir-v2 in Accuracy. The experimental results highlight the importance of aligning local feature extractors. The best results are obtained when FedBM is equipped with LKCC and CGDE, which can corroborate the effectiveness of the two modules.

4.3.2. Ablative experiments on LKCC module

The Impact of Prompt Number The number of prompts is related to the quality of the global classifier. To study its impact, we only equip FedBM with the LKCC module and adjust the proportion of prompts to observe the performance change on the OCT-8 dataset. As shown in Table 7, our method with different number of prompts presents different performance. Reducing the number of prompts probably leads to a

Table 8
The performance of different methods on OCT-C8 dataset.

Methods	$\beta = 0.05$		$\beta = 0.1$	
	Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
Baseline	74.82 ± 5.98	72.34 ± 6.87	78.64 ± 5.44	76.25 ± 7.23
Random (Freezing)	76.11 ± 6.31	73.07 ± 7.57	82.67 ± 3.40	82.03 ± 3.80
Embedding (Averaging)	76.75 ± 5.00	76.34 ± 4.38	84.22 ± 2.38	83.81 ± 2.64
Embedding (Distribution)	79.14 ± 3.77	77.13 ± 5.10	85.00 ± 2.66	84.56 ± 2.99

decline in performance. When all prompts are used to construct local classifiers, FedBM achieves the highest performance. The experiment results demonstrate the importance of the number of prompts.

The Impact of Classifier Construction Method To study the impact of the classifier construction method, we only equip FedBM with the LKCC module and change the method of classifier construction. The vanilla FedAvg is regarded as the baseline. As shown in Table 8, freezing randomly-initialized local classifiers obtains better performance than the baseline. The results indicate that sharing a fixed classifier across clients is a feasible path to alleviate the classifier bias problem. Using averaged concept embedding as local classifiers outperforms the random initialization strategy, highlighting the effectiveness of linguistic knowledge. Notably, concept embedding distribution is superior to averaged concept embedding in different heterogeneous settings. This is because using embedding distribution as local classifiers can help the model capture the semantic diversification of image representations.

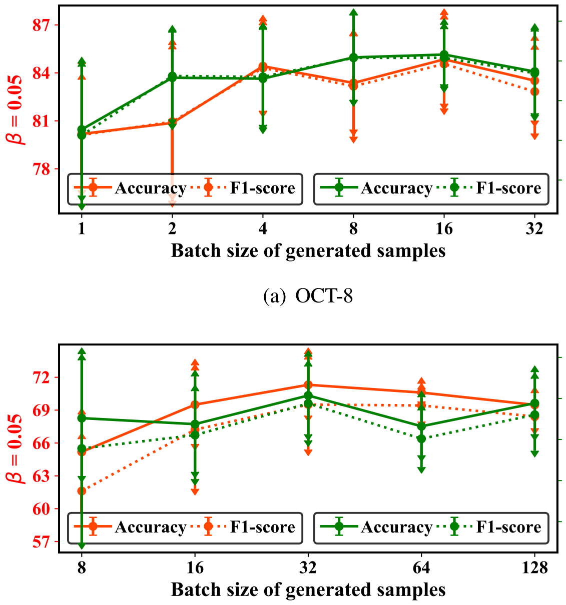
4.3.3. Ablative experiments on CGDE module

Evaluation of Key Components Concept conditions, $\mathcal{L}_{div}$ and $\mathcal{L}_{dis}$ are the key components of the CGDE module. We remove them individually to observe the performance of our method on OCT-8 and Kvasir-v2 datasets. As shown in Table 9, our method obtains the worst performance if we do use Gaussian noises instead of concept conditions on two datasets. Moreover, both removing $\mathcal{L}_{div}$ and $\mathcal{L}_{dis}$ result in performance degradation of the proposed method. The best performance is achieved by our method possessing three components simultaneously. Therefore, these experimental results confirm the importance of these components for the CGDE module.

The Impact of Batch Size of Generated Samples In Eq. (11), the larger batch size of generated samples indicates the stronger constraint on local updates. We fix the batch size of the original local data and compare the performance of FedBM with various batch sizes of generated samples. As presented in Fig. 3(a), FedBM obtains the lowest performance when the batch size of generated samples is 1. As the batch size increases from 1 to 16 on OCT-8, the performance of FedBM exhibits an increasing trend both in Accuracy and F1-score for different β. In Fig. 3(b), when the batch size of generated samples increases from 8 to 128, the performance of FedBM first rises to the highest point (the batch size is 32) and then shows a decreasing trend on Kvasir-v2. From these results, we can find that the constraint from a small batch size of generated samples is too weak to calibrate local updates, while a too-large batch size causes overcorrection of local updates.

4.3.4. The impact of hyperparameters λ_{div} and λ_{dis}

λ_{div} and λ_{dis} in Eq. (10) are two important hyper-parameters of FedBM. The former controls the diversity of generated samples and the latter improve the training stability of the generator. To investigate their impact, we adjust $\lambda_{div} \in [0.1, 1, 10]$ and $\lambda_{dis} \in [0.1, 1, 10]$ to observe the performance of FedBM. As shown in Table 10, FedBM achieves the best performance on two datasets when λ_{div} is set to 1. Therefore, we fix $\lambda_{div} = 1$ for all experiments. In Table 11, the highest performance is observed when λ_{dis} is set to 0.1 or 1, so we present the best performance by selecting λ_{dis} from [0.1, 1] in all experiments.



(a) OCT-8

(b) Kvasir-v2

Fig. 3. The performance of our method with different batch sizes of generated samples on OCT-C8 and Kvasir-v2 datasets.

4.3.5. The impact of different PLMs

To investigate the effect of pre-trained large language models, we equip the proposed FedBM framework with different PLMs, including Bert (Devlin, 2018), RoBERTa (Liu, 2019), and the text encoder of BiomedCLIP (Zhang et al., 2024b). As shown in Table 12, FedBM with different PLMs dose not present significant performance differences. Overall, BiomedCLIP yields better performance on two datasets compared with Bert and RoBERTa. The core reason may be that the text encoder of BiomedCLIP is trained on a medical text corpus, while Bert and RoBERTa are trained on text corpora that partially contain medical data.

4.3.6. The impact of client number

To compare the performance of different methods across different numbers of clients, we set the data heterogeneity $\beta = 0.05$ and divide the training data of OCT-8 and Kvasir-v2 datasets into C clients, respectively. In Figs. 4(a) and 4(b), the performance of previous methods display sharp fluctuations with the client number on OCT-8 dataset. By comparison, FedBM presents a more stable performance trend than these approaches. For Kvasir-v2 dataset, except for FedDyn and FedProx, although the performance of other existing methods is steady with respect to the number of clients, these methods achieve the limited performance. The proposed FedBM framework significantly surpasses all existing approaches for any client number. The experimental results can prove that the proposed FedBM is more robust against client numbers than existing methods.

5. Conclusion

In this paper, we propose a Federated Bias eliMinating (FedBM) framework to solve local learning bias problem in heterogeneous federated learning, which contains Linguistic Knowledge-based Classifier Construction (LKCC) and Concept-guided Global Distribution Estimation (CGDE). LKCC can remove classifier bias by exploiting class concepts and pre-trained language models (PLMs) to construct a high-quality global classifier. CGDE is able to get rid of the learning bias of local feature extractors. It is based on probabilistic concept embeddings to learn a conditional generator. The generator is shared by all

Table 9
The performance of the proposed FedBM framework with different modules.

Datasets	Methods	$\beta = 0.05$		$\beta = 0.1$	
		Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
OCT-8	w/o Concept Conditions	81.19 ± 3.25	80.18 ± 2.85	84.09 ± 4.28	83.43 ± 4.99
	w/o $\mathcal{L}_{div}$	84.26 ± 2.36	83.65 ± 2.58	87.60 ± 1.88	87.65 ± 1.90
	w/o $\mathcal{L}_{dis}$	82.13 ± 2.30	81.54 ± 2.29	85.91 ± 1.67	85.99 ± 1.65
	FedBM	84.84 ± 2.77	84.55 ± 2.78	88.32 ± 1.49	88.16 ± 1.44
Kvasir-v2	w/o Concept Conditions	64.29 ± 1.56	63.05 ± 1.87	71.16 ± 2.50	70.25 ± 3.12
	w/o $\mathcal{L}_{div}$	70.00 ± 3.12	68.14 ± 4.10	74.52 ± 2.13	74.03 ± 2.32
	w/o $\mathcal{L}_{dis}$	67.95 ± 2.10	65.39 ± 2.91	72.10 ± 0.78	71.54 ± 0.83
	FedBM	71.31 ± 2.81	69.51 ± 4.11	74.14 ± 1.89	73.75 ± 1.79

Table 10
The performance of the proposed FedBM with different λ_{div}.

Datasets	λ_{div}	$\beta = 0.05$		$\beta = 0.1$	
		Accuracy	F1-score	Accuracy	F1-score
OCT-C8	0.1	83.90 ± 2.71	83.64 ± 2.71	86.92 ± 1.82	87.10 ± 1.68
	1.0	$\mathbf{84.84} \pm 2.77$	$\mathbf{84.55} \pm 2.78$	$\mathbf{88.32} \pm 1.49$	$\mathbf{88.16} \pm 1.44$
	10	83.26 ± 3.02	82.68 ± 3.02	87.51 ± 1.94	87.39 ± 2.03
Kvasir-v2	0.1	69.31 ± 2.96	67.40 ± 3.89	73.77 ± 1.13	73.26 ± 1.06
	1.0	$\mathbf{71.31} \pm 2.81$	$\mathbf{69.51} \pm 4.11$	74.14 ± 1.89	$\mathbf{73.75} \pm 1.79$
	10	69.62 ± 2.35	67.90 ± 3.86	$\mathbf{74.35} \pm 1.69$	73.71 ± 2.07

Table 11
The performance of the proposed FedBM with different λ_{dis}.

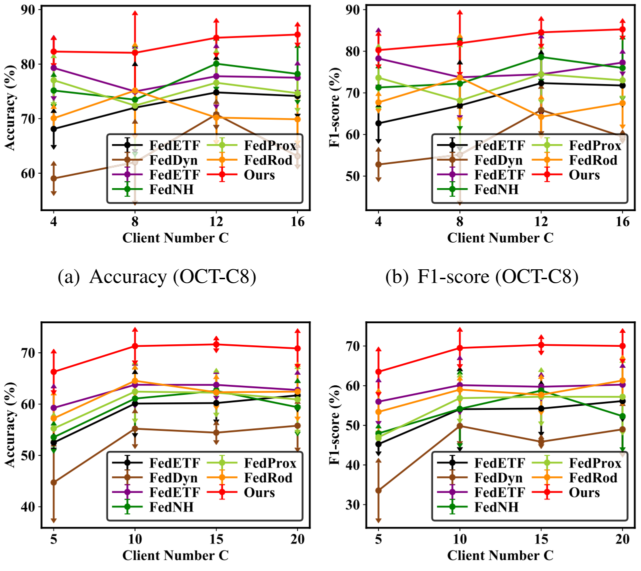
Datasets	λ_{dis}	$\beta = 0.05$		$\beta = 0.1$	
		Accuracy	F1-score	Accuracy	F1-score
OCT-C8	0.1	$\mathbf{84.84} \pm 2.77$	$\mathbf{84.55} \pm 2.78$	$\mathbf{88.32} \pm 1.49$	$\mathbf{88.16} \pm 1.44$
	1	83.50 ± 4.00	83.33 ± 3.93	87.05 ± 3.12	87.12 ± 3.18
	10	83.00 ± 2.60	82.46 ± 2.75	86.17 ± 3.58	86.33 ± 3.47
Kvasir-v2	0.1	70.14 ± 2.76	67.72 ± 4.34	$\mathbf{74.14} \pm 1.89$	$\mathbf{73.75} \pm 1.79$
	1	$\mathbf{71.31} \pm 2.81$	$\mathbf{69.51} \pm 4.11$	72.33 ± 1.81	71.50 ± 2.04
	10	69.97 ± 2.03	68.55 ± 2.65	72.39 ± 2.95	71.39 ± 3.33

Table 12
The performance of the proposed FedBM framework with different PLMs.

Datasets	PLMs	$\beta = 0.05$		$\beta = 0.1$	
		Accuracy (%)	F1-score (%)	Accuracy (%)	F1-score (%)
OCT-8	Bert	84.83 ± 2.44	84.35 ± 2.26	87.82 ± 2.06	87.73 ± 2.10
	RoBERTa	83.59 ± 1.72	82.78 ± 1.98	$\mathbf{88.40} \pm 2.48$	$\mathbf{88.35} \pm 2.53$
	BiomedCLIP	$\mathbf{84.84} \pm 2.77$	$\mathbf{84.55} \pm 2.78$	88.32 ± 1.49	88.16 ± 1.44
Kvasir-v2	Bert	69.48 ± 3.25	67.64 ± 4.42	73.83 ± 1.81	73.19 ± 1.58
	RoBERTa	70.73 ± 1.58	$\mathbf{69.99} \pm 1.69$	73.25 ± 2.20	71.52 ± 3.51
	BiomedCLIP	$\mathbf{71.31} \pm 2.81$	69.51 ± 4.11	$\mathbf{74.14} \pm 1.89$	$\mathbf{73.75} \pm 1.79$

clients and produces pseudo data to calibrate updates of local feature extractors. The experimental results on five public datasets show the superior performance of FedBM in contrast to state-of-the-art methods under different heterogeneous settings. Extensive ablation experiments prove the effectiveness of submodules of FedBM.

The proposed FedBM has achieved promising performance on various medical tasks, yet there are several limitations: (1) Although current PLMs are trained on large-scale datasets and show the strong generalization ability, some extreme cases may exist, such as highly similar concepts and open classes, for which our method is not suitable; (2) Theoretically, more diverse prompts can enable the proposed method to achieve the better performance. Our work is expected to provide a new respective to the research community for addressing data heterogeneity. The number of prompts in our experiments is not necessarily optimal. Both how to obtain diverse prompts and how to select the optimal number of prompts are two open directions worth exploring in the future; (3) In the proposed method, the generator is trained using the global model obtained by averaging the models from participating clients. Although there are not techniques available to recover the user-level privacy information for the generator, the copyright of client data may be infringed since other participants can use the generator to produce data for unintended purposes. A feasible solution to this problem is to train a generator that produces intermediate-layer feature maps rather than images.



(a) Accuracy (OCT-C8) (b) F1-score (OCT-C8)

(c) Accuracy (Kvasir-v2) (d) F1-score (Kvasir-v2)

Fig. 4. The performance of our method with different client numbers on Kvasir-v2 and OCT-C8 datasets.

CRediT authorship contribution statement

Meilu Zhu: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Formal analysis, Conceptualization. **Qiushi Yang:** Writing – review & editing, Methodology, Formal analysis, Conceptualization. **Zhifan Gao:** Writing – review & editing. **Yixuan Yuan:** Writing – review & editing, Supervision, Funding acquisition, Formal analysis, Conceptualization. **Jun Liu:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Hong Kong Research Grants Council under Grant 11212321, 11217922, and ECS-21212720, the HKSAR Innovation and Technology Commission (ITC) under ITF Project

MHP/109/19, ITS/229/22, and the Science, Technology and Innovation Committee of Shenzhen under Grant SGDX20210823104001011.

Appendix A

We present the theoretical derivation of $\overline{\mathcal{L}}^{\infty}_{align}$ in Eq. (4).

$$
\begin{aligned}
\mathcal{L}^{\infty}_{align} &= \frac{1}{N_c}\sum_{i=1}^{N_c}\mathbb{E}_{e^{(\mathbf{y}_i)}\sim\mathcal{N}^{(\mathbf{y}_i)}}\left(-\log\frac{e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}}}{e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}}+\sum_{k\neq\mathbf{y}_i}^{K}\mathbb{E}_{e^{(k)}\sim\mathcal{N}^{(k)}}e^{\tau\mathbf{h}_i e^{(k)}}}\right)\\
&= \frac{1}{N_c}\sum_{i=1}^{N_c}\mathbb{E}_{e^{(\mathbf{y}_i)}\sim\mathcal{N}^{(\mathbf{y}_i)}}\left(\log(e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}}+\sum_{k\neq\mathbf{y}_i}^{K}\mathbb{E}_{e^{(k)}\sim\mathcal{N}^{(k)}}e^{\tau\mathbf{h}_i e^{(k)}})\right)\\
&\quad -\frac{1}{N_c}\sum_{i=1}^{N_c}\mathbb{E}_{e^{(\mathbf{y}_i)}\sim\mathcal{N}^{(\mathbf{y}_i)}}\left(\log(e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}})\right)\\
&\quad \text{//using the Jensen's inequality: } \mathbb{E}[\log(X)]\leq\log(\mathbb{E}X)\\
&\leq \frac{1}{N_c}\sum_{i=1}^{N_c}\left[\log\mathbb{E}_{e^{(\mathbf{y}_i)}\sim\mathcal{N}^{(\mathbf{y}_i)}}\left(e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}}+\sum_{k\neq\mathbf{y}_i}^{K}\mathbb{E}_{e^{(k)}\sim\mathcal{N}^{(k)}}e^{\tau\mathbf{h}_i e^{(k)}}\right)\right]\\
&\quad -\frac{1}{N_c}\sum_{i=1}^{N_c}\left[\mathbb{E}_{e^{(\mathbf{y}_i)}\sim\mathcal{N}^{(\mathbf{y}_i)}}\log(e^{\tau\mathbf{h}_i e^{(\mathbf{y}_i)}})\right]\\
&= \frac{1}{N_c}\sum_{i=1}^{N_c}\left[\log(\sum_{k=1}^{K}\mathbb{E}_{e^{(k)}\sim\mathcal{N}^{(k)}}e^{\tau\mathbf{h}_i e^{(k)}})-\tau\mathbf{h}_i\boldsymbol{\mu}_{(\mathbf{y}_i)}\right]\\
&\quad \text{//using the moment generation function for Gaussian}\\
&\quad \text{variable } X:\ \mathbb{E}[e^{\mathbf{h}X}]=e^{\mathbf{h}\boldsymbol{\mu}+\frac{1}{2}\mathbf{h}^2\boldsymbol{\Sigma}}\\
&= \frac{1}{N_c}\sum_{i=1}^{N_c}\left[\log(\sum_{k=1}^{K}e^{\tau\mathbf{h}_i\boldsymbol{\mu}_k+\frac{1}{2}\tau^2\mathbf{h}_i^2\boldsymbol{\Sigma}_k})-\tau\mathbf{h}_i\boldsymbol{\mu}_{(\mathbf{y}_i)}\right]\\
&\quad \text{//Let } \mathcal{F}(\mathbf{h},y)=\tau\mathbf{h}\boldsymbol{\mu}_{(y)}+\frac{1}{2}\tau^2\mathbf{h}^2\boldsymbol{\Sigma}_{(y)}\\
&= \frac{1}{N_c}\sum_{i=1}^{N_c}\left[\log(\sum_{k=1}^{K}e^{\tau\mathbf{h}_i\boldsymbol{\mu}_k+\frac{1}{2}\tau^2\mathbf{h}_i^2\boldsymbol{\Sigma}_k})-\mathcal{F}(\mathbf{h}_i,\mathbf{y}_i)+\mathcal{F}(\mathbf{h}_i,\mathbf{y}_i)\right]\\
&\quad -\frac{1}{N_c}\sum_{i=1}^{N_c}\left[\tau\mathbf{h}_i\boldsymbol{\mu}_{(\mathbf{y}_i)}\right]\\
&= \frac{1}{N_c}\sum_{i=1}^{N_c}\left[-\log\frac{e^{\mathcal{F}(\mathbf{h}_i,\mathbf{y}_i)}}{\sum_{k=1}^{K}e^{\mathcal{F}(\mathbf{h}_i,k)}}+\mathcal{F}(\mathbf{h}_i,\mathbf{y}_i)-\tau\mathbf{h}_i\boldsymbol{\mu}_{(\mathbf{y}_i)}\right]\\
&= \frac{1}{N_c}\sum_{i=1}^{N_c}\left[-\log\frac{e^{\mathcal{F}(\mathbf{h}_i,\mathbf{y}_i)}}{\sum_{k=1}^{K}e^{\mathcal{F}(\mathbf{h}_i,k)}}+\frac{1}{2}\tau^2\mathbf{h}_i^2\boldsymbol{\Sigma}_{(\mathbf{y}_i)}\right]\\
&= \overline{\mathcal{L}}^{\infty}_{align}.
\end{aligned}
$$

Appendix B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.media.2025.103524.

Data availability

Data will be made available on request.

References

Acar, D.A.E., Zhao, Y., Navarro, R.M., Mattina, M., Whatmough, P.N., Saligrama, V., 2021. Federated learning based on dynamic regularization. arXiv preprint arXiv:2111.04263.

Acevedo, A., Merino González, A., Alférez Baquero, E.S., Molina Borrás, Á., Boldú Nebot, L., Rodellar Benedé, J., 2020. A dataset of microscopic peripheral blood cell images for development of automatic recognition systems. Data Brief 30 (105474).

Bang, D., Shim, H., 2021. Mggan: Solving mode collapse using manifold-guided training. In: ICCV. pp. 2347–2356.

Caldas, S., Duddu, S.M.K., Wu, P., Li, T., Konečnỳ, J., McMahan, H.B., Smith, V., Talwalkar, A., 2018. Leaf: A benchmark for federated settings. arXiv preprint arXiv:1812.01097.

Chen, H.-Y., Chao, W.-L., 2021. On bridging generic and personalized federated learning for image classification. In: ICLR.

Chen, Z., Zhu, M., Yang, C., Yuan, Y., 2021. Personalized retrogress-resilient framework for real-world medical federated learning. In: MICCAI. Springer, pp. 347–356.

Collins, L., Hassani, H., Mokhtari, A., Shakkottai, S., 2021. Exploiting shared representations for personalized federated learning. In: ICML. pp. 2089–2099.

Dai, Y., Chen, Z., Li, J., Heinecke, S., Sun, L., Xu, R., 2023. Tackling data heterogeneity in federated learning with class prototypes. In: AAAI. Vol. 37, pp. 7314–7322.

de Luca, A.B., Zhang, G., Chen, X., Yu, Y., 2022. Mitigating data heterogeneity in federated learning with data augmentation. arXiv preprint arXiv:2206.09979.

Devlin, J., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Diao, Y., Li, Q., He, B., 2023. Towards addressing label skews in one-shot federated learning. In: ICLR.

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A., 2020. Shortcut learning in deep neural networks. Nat. Mach. Intell. 2 (11), 665–673.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. NeurIPS 27.

Guo, Y., Guo, K., Cao, X., Wu, T., Chang, Y., 2023a. Out-of-distribution generalization of federated learning via implicit invariant relationships. In: ICML. pp. 11905–11933.

Guo, Y., Tang, X., Lin, T., 2023b. FedBR: Improving federated learning on heterogeneous data via local learning bias reduction. In: ICML. pp. 12034–12054.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: CVPR. pp. 770–778.

Ho, J., Jain, A., Abbeel, P., 2020. Denoising diffusion probabilistic models. NeurIPS 33, 6840–6851.

Huang, W., Liu, Y., Ye, M., Chen, J., Du, B., 2024. Federated learning with long-tailed data via representation unification and classifier rectification. IEEE Trans. Inf. Forensics Secur. 19, 5738–5750.

Huang, W., Ye, M., Shi, Z., Li, H., Du, B., 2023. Rethinking federated learning with domain shift: A prototype view. In: CVPR. pp. 16312–16322.

Jeong, E., Oh, S., Kim, H., Park, J., Bennis, M., Kim, S.-L., 2018. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. arXiv preprint arXiv:1811.11479.

Karimireddy, S.P., Kale, S., Mohri, M., Reddi, S., Stich, S., Suresh, A.T., 2020. Scaffold: Stochastic controlled averaging for federated learning. In: ICML. pp. 5132–5143.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Li, Q., He, B., Song, D., 2021a. Model-contrastive federated learning. In: CVPR. pp. 10713–10722.

Li, X., Jiang, M., Zhang, X., Kamp, M., Dou, Q., 2021b. FedBN: Federated learning on non-IID features via local batch normalization. In: ICLR.

Li, T., Sahu, A.K., Talwalkar, A., Smith, V., 2020a. Federated learning: Challenges, methods, and future directions. IEEE Signal Process. Mag. 37 (3), 50–60.

Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Smith, V., 2020b. Federated optimization in heterogeneous networks. In: MLSys. Vol. 2, pp. 429–450.

Li, Z., Shang, X., He, R., Lin, T., Wu, C., 2023. No fear of classifier biases: Neural collapse inspired federated learning with synthetic and fixed classifier. In: ICCV. pp. 5319–5329.

Li, Z., Shao, J., Mao, Y., Wang, J.H., Zhang, J., 2022. Federated learning with gan-based data synthesis for non-iid clients. In: International Workshop on Trustworthy Federated Learning. pp. 17–32.

Liu, Y., 2019. Roberta: A Robustly Optimized Bert Pretraining Approach. Vol. 364, arXiv preprint arXiv:1907.11692.

Liu, Y., Zhang, W., Wang, J., 2021. Source-free domain adaptation for semantic segmentation. In: CVPR. pp. 1215–1224.

Long, Y., Xue, Z., Chu, L., Zhang, T., Wu, J., Zang, Y., Du, J., 2023. Fedcd: A classifier debiased federated learning framework for non-iid data. In: ACM MM. pp. 8994–9002.

Luo, M., Chen, F., Hu, D., Zhang, Y., Liang, J., Feng, J., 2021. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. NeurIPS 34, 5972–5984.

Luo, k., Wang, S., Fu, Y., Li, X., Lan, Y., Gao, M., 2024. DFRD: Data-free robustness distillation for heterogeneous federated learning. NeurIPS 36, 17854–17866.

McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A., 2017. Communication-efficient learning of deep networks from decentralized data. In: AISTATS. pp. 1273–1282.

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

Papyan, V., Han, X., Donoho, D.L., 2020. Prevalence of neural collapse during the terminal phase of deep learning training. Proc. Natl. Acad. Sci. 117 (40), 24652–24663.

Pogorelov, K., Randel, K.R., Griwodz, C., Eskeland, S.L., de Lange, T., Johansen, D., Spampinato, C., Dang-Nguyen, D.-T., Lux, M., Schmidt, P.T., Riegler, M., Halvorsen, P., 2017. Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection. In: MMSys. pp. 164–169.

Qi, Z., Meng, L., Chen, Z., Hu, H., Lin, H., Meng, X., 2023. Cross-silo prototypical calibration for federated learning with non-iid data. In: ACM MM. pp. 3099–3107.

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I., 2021. Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763.

Razzak, M.I., Naz, S., Zaib, A., 2018. Deep learning for medical image processing: Overview, challenges and the future. Classif. BioApps: Autom. Decis. Mak. 323–350.

Shi, Y., Shen, L., Wei, K., Sun, Y., Yuan, B., Wang, X., Tao, D., 2023. Improving the model consistency of decentralized federated learning. In: ICML. pp. 31269–31291.

Shin, M., Hwang, C., Kim, J., Park, J., Bennis, M., Kim, S.-L., 2020. Xor mixup: Privacy-preserving data augmentation for one-shot federated learning. arXiv preprint arXiv:2006.05148.

Subramanian, M., Shanmugavadivel, K., Naren, O.S., Premkumar, K., Rankish, K., 2022. Classification of retinal oct images using deep learning. In: ICCCI. pp. 1–7.

Tan, Y., Long, G., Liu, L., Zhou, T., Lu, Q., Jiang, J., Zhang, C., 2022. Fedproto: Federated prototype learning across heterogeneous clients. In: AAAI. Vol. 36, pp. 8432–8440.

Tschandl, P., Rosendahl, C., Kittler, H., 2018. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data 5 (1), 1–9.

Wang, J., Liu, Q., Liang, H., Joshi, G., Poor, H.V., 2020. Tackling the objective inconsistency problem in heterogeneous federated optimization. NeurIPS 33, 7611–7623.

Wang, Y., Pan, X., Song, S., Zhang, H., Huang, G., Wu, C., 2019. Implicit semantic data augmentation for deep networks. NeurIPS 32.

Xu, J., Tong, X., Huang, S.-L., 2023. Personalized federated learning with feature alignment and classifier collaboration. In: ICLR.

Yan, Z., Yang, X., Cheng, K.-T., 2018. Joint segment-level and pixel-wise losses for deep learning based retinal vessel segmentation. IEEE Trans. Biomed. Eng. 65 (9), 1912–1923.

Yan, Y., Zhu, L., 2023. A simple data augmentation for feature distribution skewed federated learning. arXiv preprint arXiv:2306.09363.

Yang, J., Shi, R., Wei, D., Liu, Z., Zhao, L., Ke, B., Pfister, H., Ni, B., 2023. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. Sci. Data 10 (1), 41.

Ye, R., Ni, Z., Xu, C., Wang, J., Chen, S., Eldar, Y.C., 2023. Fedfm: Anchor-based feature matching for data heterogeneity in federated learning. IEEE Trans. Signal Process. 71, 4224–4239.

Yin, H., Molchanov, P., Alvarez, J.M., Li, Z., Mallya, A., Hoiem, D., Jha, N.K., Kautz, J., 2020. Dreaming to distill: Data-free knowledge transfer via deepinversion. In: CVPR. pp. 8715–8724.

Yoon, T., Shin, S., Hwang, S.J., Yang, E., 2021. Fedmix: Approximation of mixup under mean augmented federated learning. arXiv preprint arXiv:2107.00233.

Yu, S., Chen, J., Han, H., Jiang, S., 2023. Data-free knowledge distillation via feature exchange and activation region constraint. In: CVPR. pp. 24266–24275.

Zhang, H., 2018. Mixup: Beyond empirical risk minimization. In: ICLR.

Zhang, J., Chen, C., Li, B., Lyu, L., Wu, S., Ding, S., Shen, C., Wu, C., 2022a. Dense: Data-free one-shot federated learning. NeurIPS 35, 21414–21428.

Zhang, J., Liu, Y., Hua, Y., Cao, J., 2024a. An upload-efficient scheme for transferring knowledge from a server-side pre-trained generator to clients in heterogeneous federated learning. In: CVPR. pp. 12109–12119.

Zhang, L., Shen, L., Ding, L., Tao, D., Duan, L.-Y., 2022b. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In: CVPR. pp. 10174–10183.

Zhang, S., Xu, Y., Usuyama, N., Xu, H., Bagga, J., Tinn, R., Preston, S., Rao, R., Wei, M., Valluri, N., Wong, C., Tupini, A., Wang, Y., Mazzola, M., Shukla, S., Liden, L., Gao, J., Lungren, M.P., Naumann, T., Wang, S., Poon, H., 2024b. BiomedCLIP: a multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs. arXiv:2303.00915.

Zhou, T., Zhang, J., Tsang, D.H., 2024. FedFA: Federated learning with feature anchors to align features and classifiers for heterogeneous data. IEEE Trans. Mob. Comput. 23 (6), 6731–6742.

Zhu, M., Chen, Z., Yuan, Y., 2021a. DSI-Net: Deep synergistic interaction network for joint classification and segmentation with endoscope images. IEEE Trans. Med. Imaging 40 (12), 3315–3325.

Zhu, M., Chen, Z., Yuan, Y., 2023. FedDM: Federated weakly supervised segmentation via annotation calibration and gradient de-conflicting. IEEE Trans. Med. Imaging 42 (6), 1632–1643.

Zhu, Z., Hong, J., Zhou, J., 2021b. Data-free knowledge distillation for heterogeneous federated learning. In: ICML. pp. 12878–12889.

Zhu, M., Liao, J., Liu, J., Yuan, Y., 2024a. FedOSS: Federated open set recognition via inter-client discrepancy and collaboration. IEEE Trans. Med. Imaging 43 (1), 190–202.

Zhu, M., Mao, A., Liu, J., Yuan, Y., 2024b. DEeR: Deviation eliminating and noise regulating for privacy-preserving federated low-rank adaptation. IEEE Trans. Med. Imaging.

Zhu, M., Yang, Q., Gao, Z., Liu, J., Yuan, Y., 2024c. Stealing knowledge from pre-trained language models for federated classifier debiasing. In: MICCAI. pp. 685–695.