Eliminating Domain Bias for Federated Learning in Representation Space

Jianqing Zhang[1] Yang Hua[2] Jian Cao[1] Hao Wang[3]

Tao Song[1] Zhengui Xue[1] Ruhui Ma[1] Haibing Guan[1]

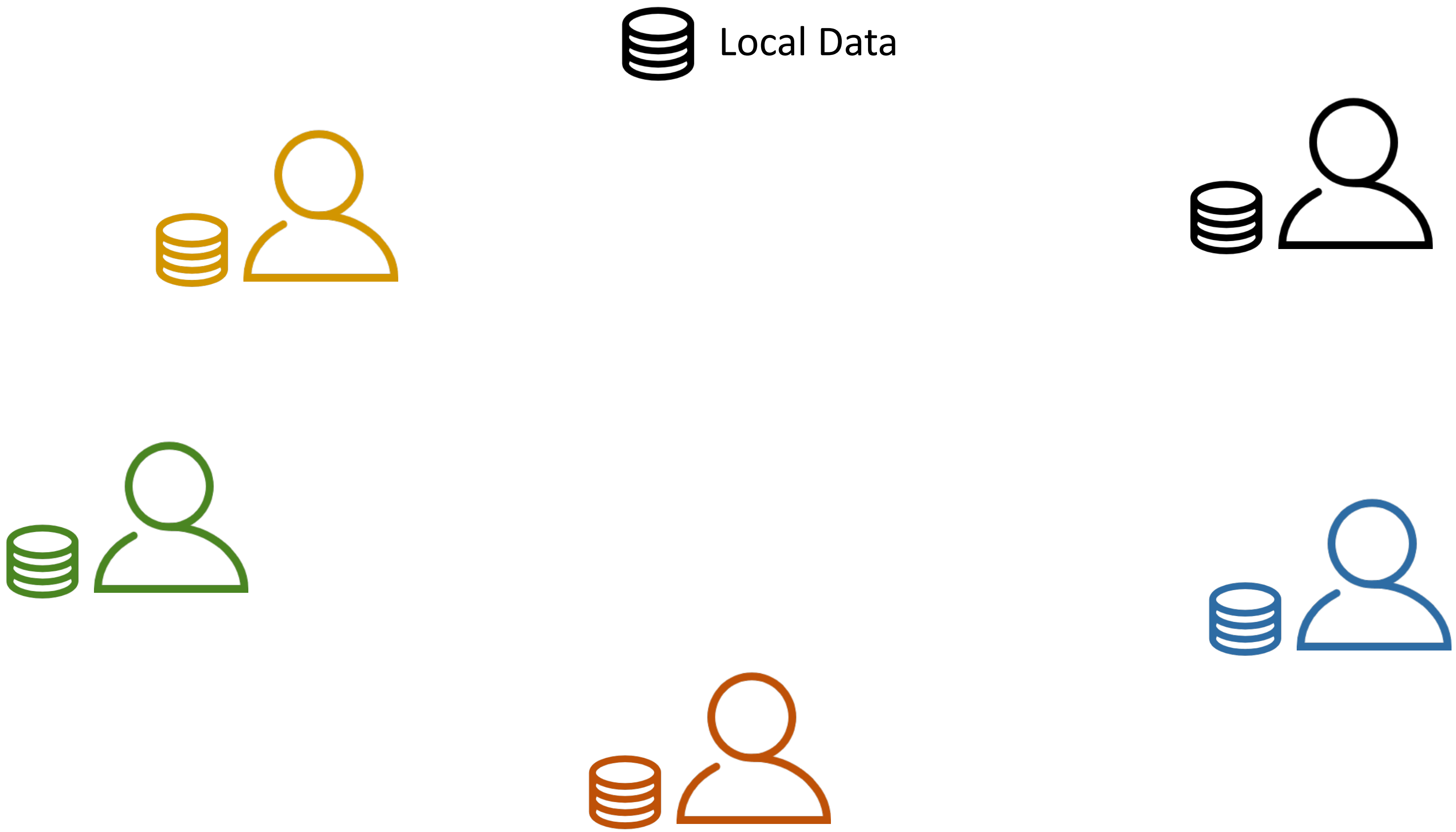
1 上海交通大学 Shanghai Jiao Tong University

2 Queen's University Belfast

3 LSU

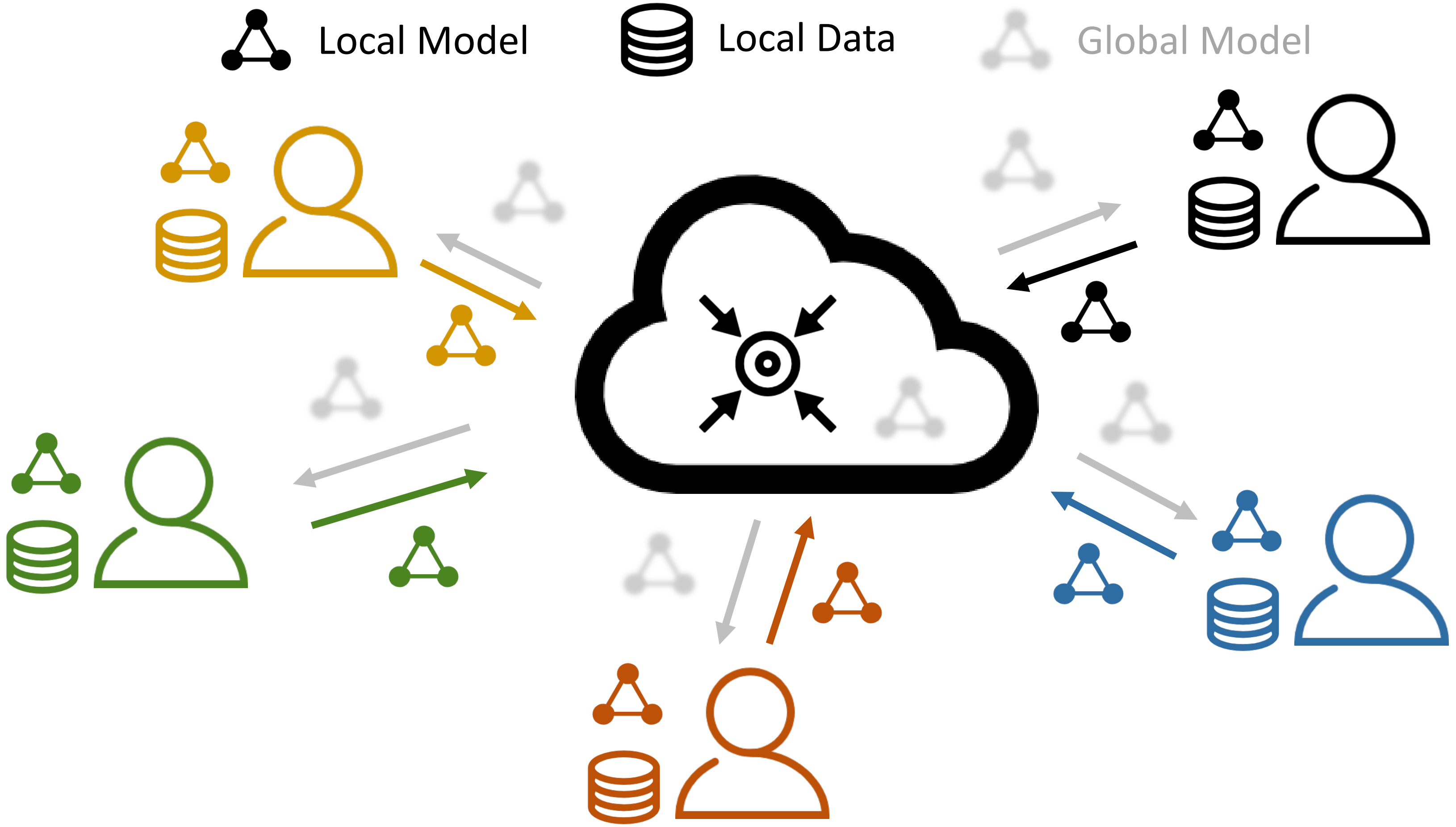
Federated Learning (FL)

- In practice, clients generate their specific private data, as shown by the colorful icons here.



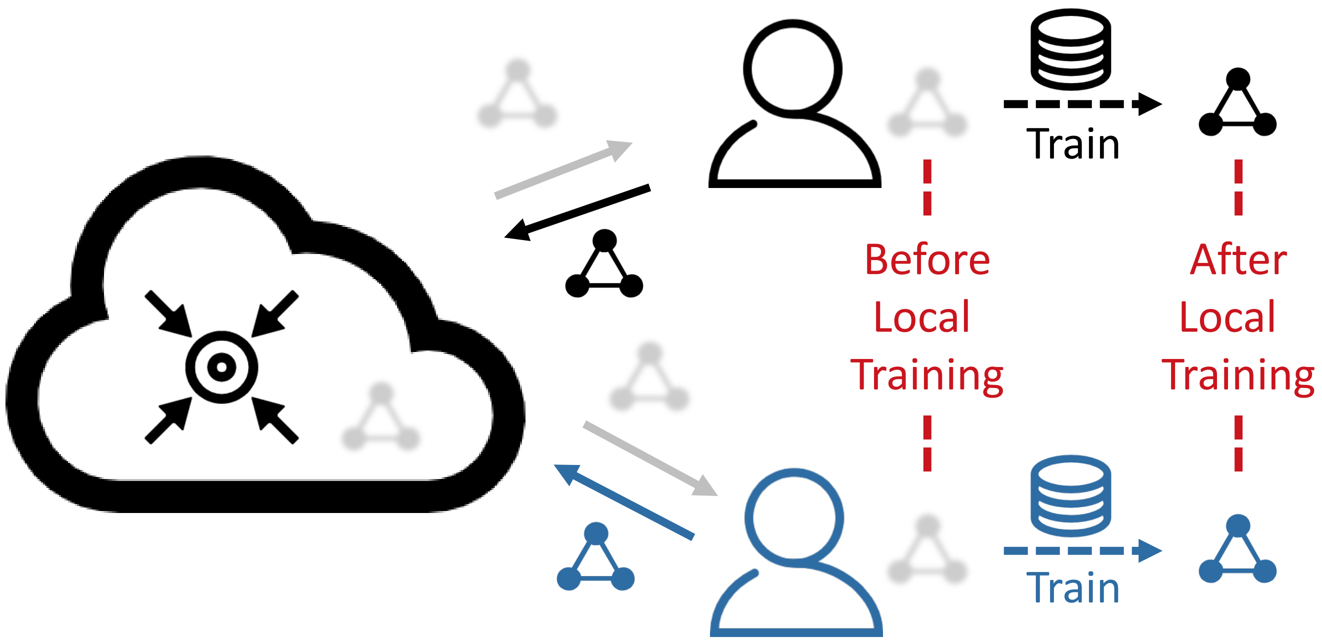
Statistical Heterogeneity Issue

- Client-specific private data brings the *statistical heterogeneity* issue

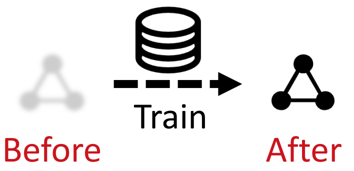


Statistical Heterogeneity Issue

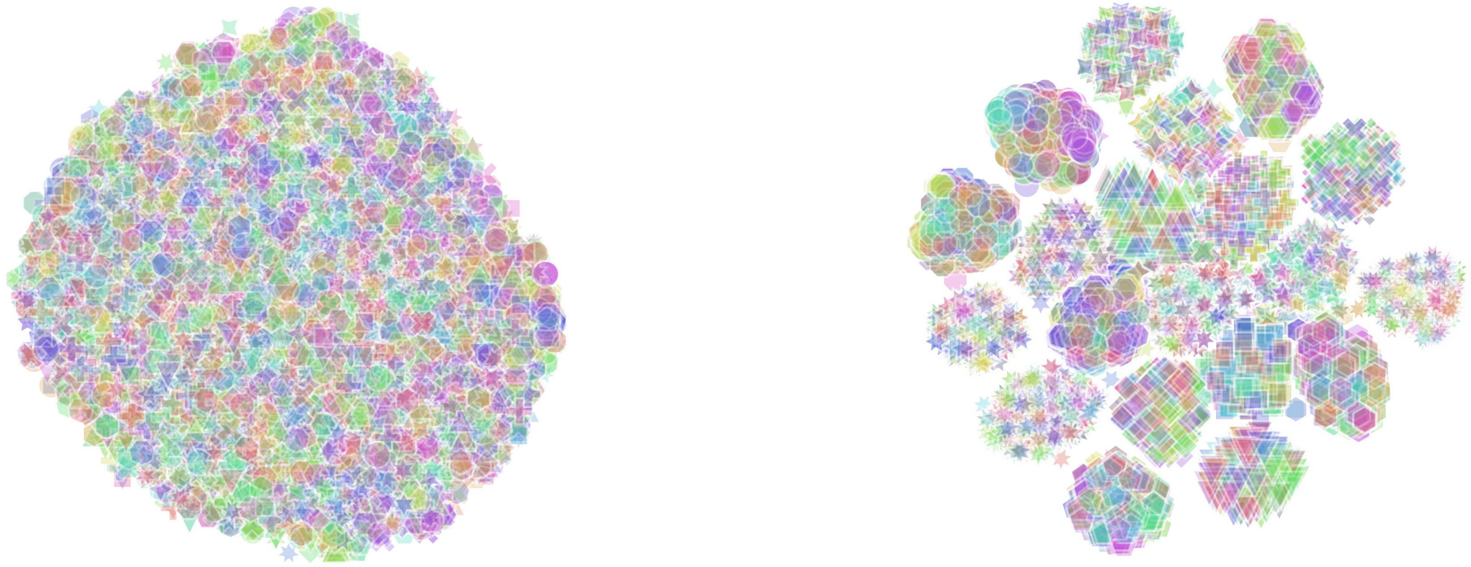
- With heterogeneous data, clients' local training turns the received global model to client-specific local models



Representation bias phenomenon



- After local training, the feature representations are ***biased*** to client-specific domains

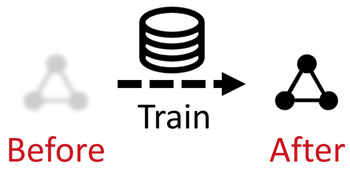


(a) Before local training

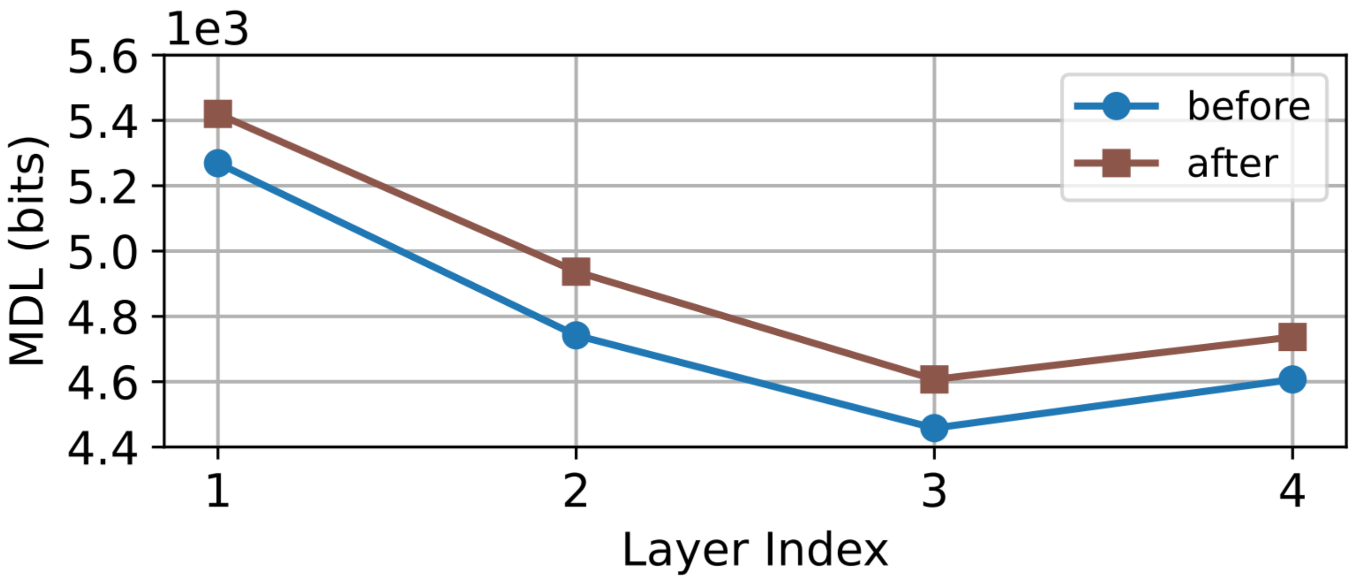
(b) After local training

t-SNE visualization for representations before/after local training in FedAvg.
We use *color* and *shape* to distinguish *labels* and *clients*, respectively.
Representations form client-specific domains after local training.

Representation degeneration phenomenon



- At the same time, representations' quality is also ***degenerated***



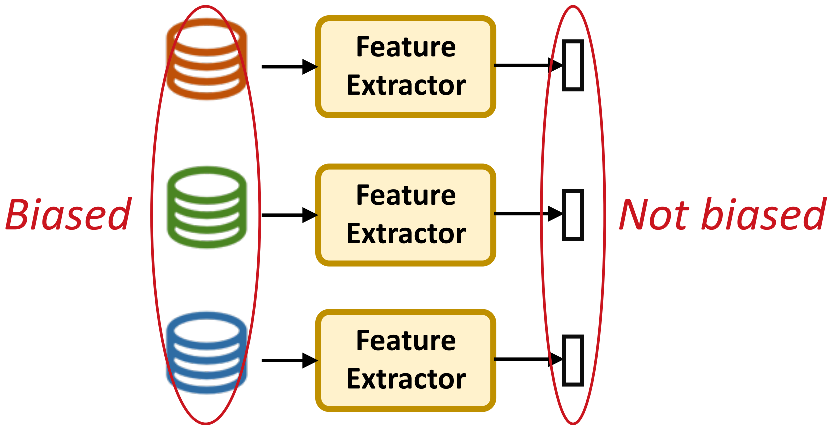
Per-layer MDL (bits) for representations before/after local training in FedAvg.
A large MDL value means low representation quality.

Personalized FL (pFL)

- pFL methods learn personalized modules, but

Personalized FL (pFL)

- pFL methods learn personalized modules, but
- feature extractors are still trained with only biased local data domains *on clients*, leading to



Before local training

Personalized FL (pFL)

- pFL methods learn personalized modules, but
- feature extractors are still trained with only biased local data domains *on clients*, leading to
- ***representation bias*** and ***representation degeneration*** during local training.

Before local training

After local training

Our Domain Bias Eliminator (DBE)

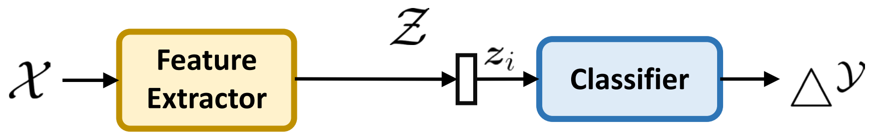
- Thus, we propose **DBE** to *eliminate domain bias in representation space* via two modules:

Our Domain Bias Eliminator (DBE)

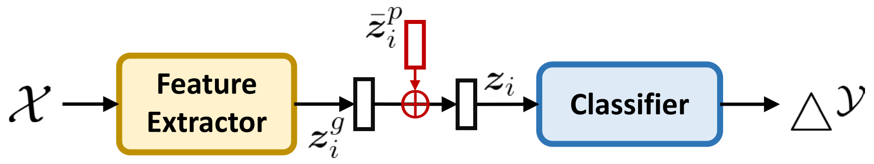
- Thus, we propose **DBE** to *eliminate domain bias in representation space* via two modules:
  - Personalized Representation Bias Memory (PRBM)
  - Mean Regularization (MR)

Personalized Representation Bias Memory (PRBM)

- PRBM stores personalized (***biased***) representation information ($\bar{z}_i^p$) for each client, and
- make the remaining information (z_i^g) to be global.



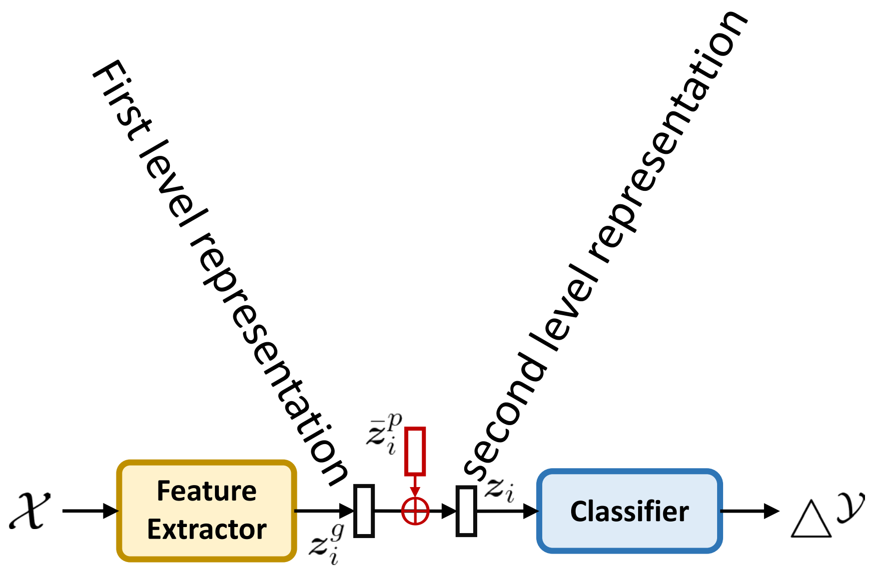
Local model (original)



Local model (with PRBM)

Personalized Representation Bias Memory (PRBM)

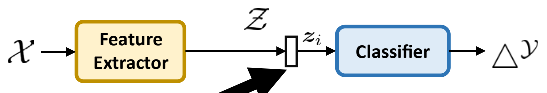
- PRBM stores personalized (***biased***) representation information ($\bar{z}_i^p$) for each client, and
- make the remaining information (z_i^g) to be global.



Local model (with PRBM)

Personalized Representation Bias Memory (PRBM)

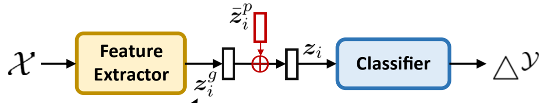
- PRBM stores personalized (***biased***) representation information ($\bar{z}_i^p$) for each client, and
- make the remaining information (z_i^g) to be global.
- Formally,



Local loss (original): $$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}) := \mathbb{E}_{(\boldsymbol{x}_i, y_i) \sim \mathcal{D}_i}[\ell(h(f(\boldsymbol{x}_i; \boldsymbol{\theta}^f); \boldsymbol{\theta}^h), y_i)]$$

Personalized Representation Bias Memory (PRBM)

- PRBM stores personalized (***biased***) representation information ($\bar{z}_i^p$) for each client, and
- make the remaining information (z_i^g) to be global.
- Formally,



Local loss (original):

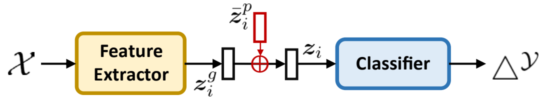
$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f);\boldsymbol{\theta}^h), y_i)]$$

Local loss (with PRBM):

$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f) + \bar{\boldsymbol{z}}_i^p;\boldsymbol{\theta}^h), y_i)]$$

Personalized Representation Bias Memory (PRBM)

- PRBM stores personalized (***biased***) representation information ($\bar{z}_i^p$) for each client, and
- make the remaining information (z_i^g) to be global.
- Formally,



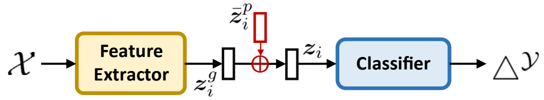
Local loss (original): $$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f);\boldsymbol{\theta}^h), y_i)]$$

Local loss (with PRBM): $$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f) + \bar{\boldsymbol{z}}_i^p;\boldsymbol{\theta}^h), y_i)]$$

View the PRBM as a personalized translation transformation $\texttt{PRBM} : \mathcal{Z} \mapsto \mathcal{Z}$:

Personalized Representation Bias Memory (PRBM)

- PRBM stores personalized (***biased***) representation information ($\bar{z}_i^p$) for each client, and
- make the remaining information (z_i^g) to be global.
- Formally,



Local loss (original):

$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(f(\boldsymbol{x}_i; \boldsymbol{\theta}^f); \boldsymbol{\theta}^h), y_i)]$$

Local loss (with PRBM):

$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(f(\boldsymbol{x}_i; \boldsymbol{\theta}^f) + \bar{\boldsymbol{z}}_i^p; \boldsymbol{\theta}^h), y_i)]$$

View the PRBM as a personalized translation transformation $\mathrm{PRBM} : \mathcal{Z} \mapsto \mathcal{Z}$:

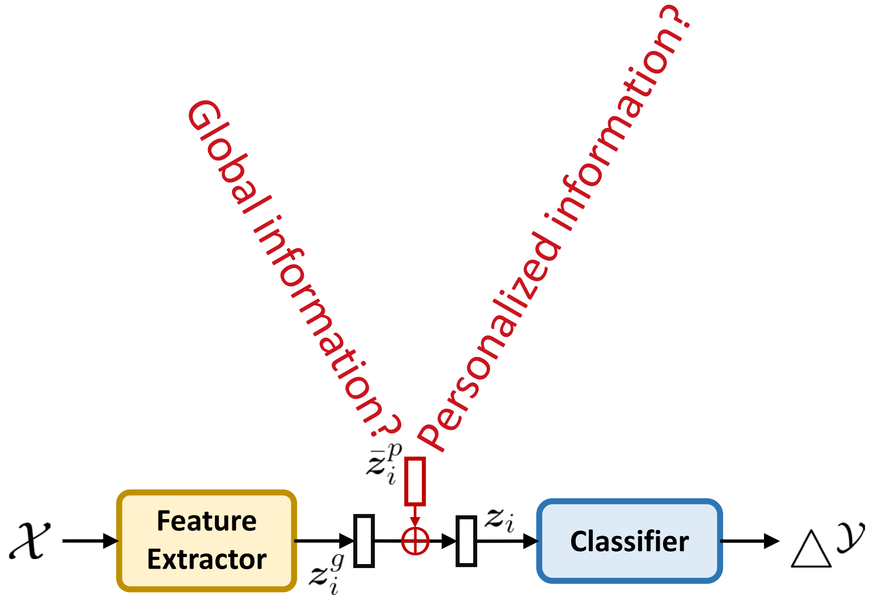
$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(\mathrm{PRBM}(f(\boldsymbol{x}_i; \boldsymbol{\theta}^f); \bar{\boldsymbol{z}}_i^p); \boldsymbol{\theta}^h), y_i)]$$

Personalized Representation Bias Memory (PRBM)

- We make PRBM to be trainable to learn personalized representation information

Personalized Representation Bias Memory (PRBM)

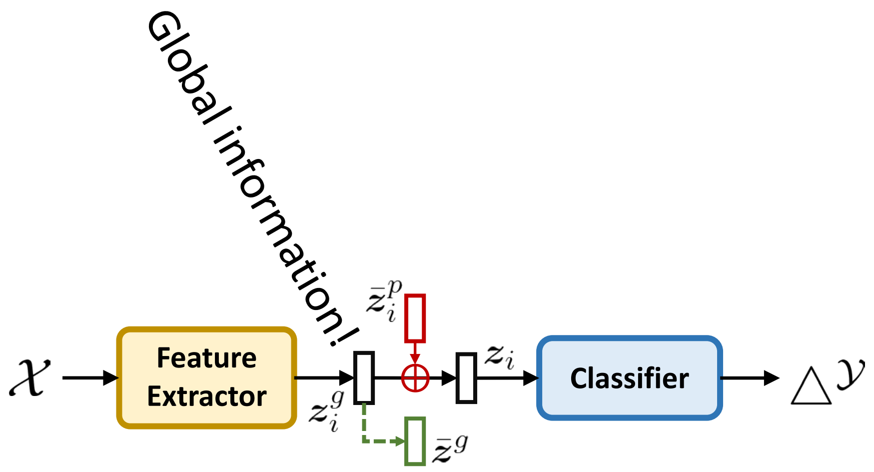
- We make PRBM to be trainable to learn personalized representation information
- However, trainable PRBM requires guidance to recognize the global and personalized information



Local model (with PRBM)

DBE: PRBM + Mean Regularization (MR)

- MR explicitly guides the local feature extractor to generate z_i^g with global information, by
- further regularize z_i^g to a globally shared ***client-invariant mean*** $\bar{z}^g$

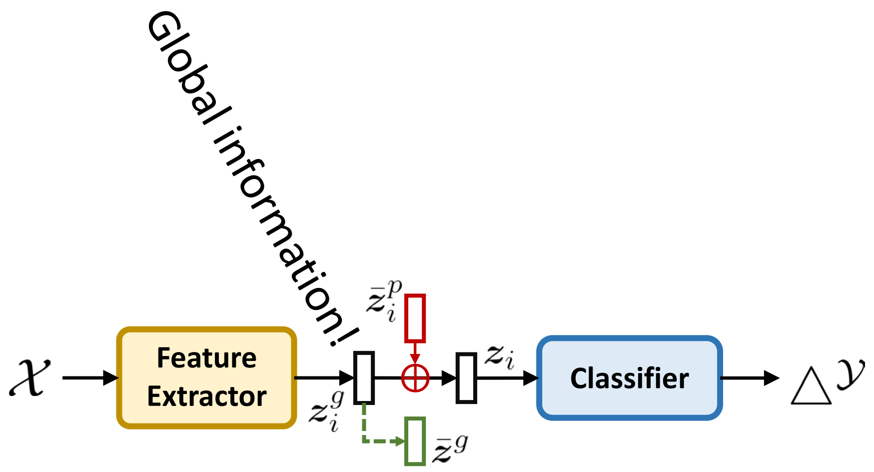


Local model (with PRBM and MR)

DBE: PRBM + Mean Regularization (MR)

- MR explicitly guides the local feature extractor to generate z_i^g with global information, by
- further regularize z_i^g to a globally shared ***client-invariant mean*** $\bar{z}^g$

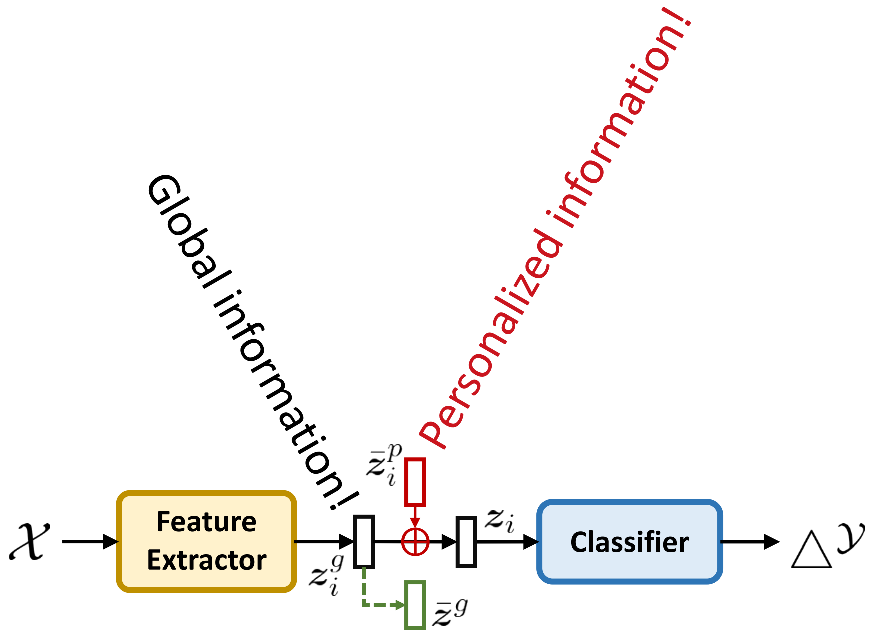
A **consensus** obtained during the **initialization period before FL**



Local model (with PRBM and MR)

DBE: PRBM + Mean Regularization (MR)

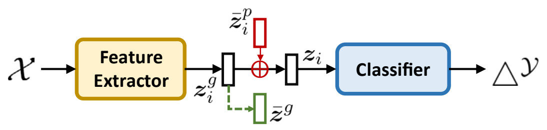
- MR explicitly guides the local feature extractor to generate z_i^g with global information, by
- further regularize z_i^g to a globally shared ***client-invariant mean*** $\bar{z}^g$



Local model (with PRBM and MR)

DBE: PRBM + Mean Regularization (MR)

- MR explicitly guides the local feature extractor to generate z_i^g with global information, by
- further regularize z_i^g to a globally shared ***client-invariant mean*** $\bar{z}^g$
- Formally,

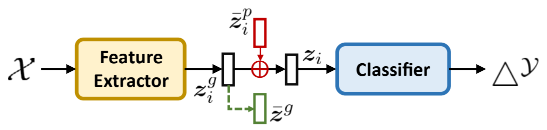


Local loss (with PRBM):

$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(\mathrm{PRBM}(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f);\bar{\boldsymbol{z}}_i^p);\boldsymbol{\theta}^h), y_i)]$$

DBE: PRBM + Mean Regularization (MR)

- MR explicitly guides the local feature extractor to generate z_i^g with global information, by
- further regularize z_i^g to a globally shared ***client-invariant mean*** $\bar{z}^g$
- Formally,



Local loss (with PRBM):

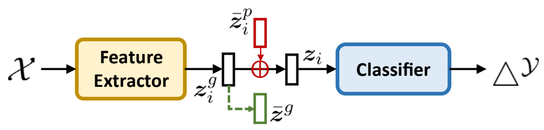
$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(\mathtt{PRBM}(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f);\bar{\boldsymbol{z}}_i^p);\boldsymbol{\theta}^h), y_i)]$$

Local loss (with PRBM and MR):

$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(\mathtt{PRBM}(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f);\bar{\boldsymbol{z}}_i^p);\boldsymbol{\theta}^h), y_i)] + \kappa \cdot \mathtt{MR}(\bar{\boldsymbol{z}}_i^g, \bar{\boldsymbol{z}}^g)$$

DBE: PRBM + Mean Regularization (MR)

- MR explicitly guides the local feature extractor to generate z_i^g with global information, by
- further regularize z_i^g to a globally shared ***client-invariant mean***
- Formally,



Local loss (with PRBM):

$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(\mathtt{PRBM}(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f);\bar{\boldsymbol{z}}_i^p);\boldsymbol{\theta}^h), y_i)]$$

Local loss (with PRBM and MR):

$$\mathcal{L}_{\mathcal{D}_i}(\boldsymbol{\theta}_i) := \mathbb{E}_{(\boldsymbol{x}_i, y_i)\sim\mathcal{D}_i}[\ell(h(\mathtt{PRBM}(f(\boldsymbol{x}_i;\boldsymbol{\theta}^f);\bar{\boldsymbol{z}}_i^p);\boldsymbol{\theta}^h), y_i)] + \kappa \cdot \mathtt{MR}(\bar{\boldsymbol{z}}_i^g, \bar{\boldsymbol{z}}^g)$$

Final loss for client *i*

Improved Bi-directional Knowledge Transfer

- **DBE** can promote ***bi-directional knowledge transfer*** between server and client with
- ***Theoretical guarantee***

Improved Bi-directional Knowledge Transfer

- **Local-to-global** knowledge transfer

Corollary 1. *Consider a local data domain $\mathcal{D}_i$ and a virtual global data domain $\mathcal{D}$ for client i and the server, respectively. Let $\mathcal{D}_i = \langle \mathcal{U}_i, c^* \rangle$ and $\mathcal{D} = \langle \mathcal{U}, c^* \rangle$, where $c^* : \mathcal{X} \mapsto \mathcal{Y}$ is a ground-truth labeling function. Let $\mathcal{H}$ be a hypothesis space of VC dimension d and $h : \mathcal{Z} \mapsto \mathcal{Y}, \forall\, h \in \mathcal{H}$. When using `DBE`, given a feature extraction function $\mathcal{F}^g : \mathcal{X} \mapsto \mathcal{Z}$ that shared between $\mathcal{D}_i$ and $\mathcal{D}$, a random labeled sample of size m generated by applying $\mathcal{F}^g$ to a random sample from $\mathcal{U}_i$ labeled according to c^*, then for every $h^g \in \mathcal{H}$, with probability at least $1 - \delta$:*

$$\mathcal{L}_{\mathcal{D}}(h^g) \leq \mathcal{L}_{\hat{\mathcal{D}}_i}(h^g) + \sqrt{\frac{4}{m}(d \log \frac{2em}{d} + \log \frac{4}{\delta})} + \boxed{d_{\mathcal{H}}(\tilde{\mathcal{U}}_i^g, \tilde{\mathcal{U}}^g)} + \lambda_i,$$

where $\mathcal{L}_{\hat{\mathcal{D}}_i}$ is the empirical loss on $\mathcal{D}_i$, e is the base of the natural logarithm, and $d_{\mathcal{H}}(\cdot, \cdot)$ is the $\mathcal{H}$-divergence between two distributions. $\lambda_i := \min_{h^g} \mathcal{L}_{\mathcal{D}}(h^g) + \mathcal{L}_{\mathcal{D}_i}(h^g)$, $\tilde{\mathcal{U}}_i^g \subseteq \mathcal{Z}$, $\tilde{\mathcal{U}}^g \subseteq \mathcal{Z}$, and $\boxed{d_{\mathcal{H}}(\tilde{\mathcal{U}}_i^g, \tilde{\mathcal{U}}^g) \leq d_{\mathcal{H}}(\tilde{\mathcal{U}}_i, \tilde{\mathcal{U}})}$. $\tilde{\mathcal{U}}_i^g$ and $\tilde{\mathcal{U}}^g$ are the induced distributions of $\mathcal{U}_i$ and $\mathcal{U}$ under $\mathcal{F}^g$, respectively. $\tilde{\mathcal{U}}_i$ and $\tilde{\mathcal{U}}$ are the induced distributions of $\mathcal{U}_i$ and $\mathcal{U}$ under $\mathcal{F}$, respectively. $\mathcal{F}$ is the feature extraction function in the original FedAvg without `DBE`.

Improved Bi-directional Knowledge Transfer

- **Global-to-local** knowledge transfer

Corollary 2. *Let $\mathcal{D}_i$, $\mathcal{D}$, $\mathcal{F}^g$, and λ_i defined as in Corollary $\boxed{1}$. Given a translation transformation function $\texttt{PRBM} : \mathcal{Z} \mapsto \mathcal{Z}$ that shared between $\mathcal{D}_i$ and virtual $\mathcal{D}$, a random labeled sample of size m generated by applying $\mathcal{F}'$ to a random sample from $\mathcal{U}_i$ labeled according to c^*, $\mathcal{F}' = \texttt{PRBM} \circ \mathcal{F}^g : \mathcal{X} \mapsto \mathcal{Z}$, then for every $h' \in \mathcal{H}$, with probability at least $1 - \delta$:*

$$\mathcal{L}_{\mathcal{D}_i}(h') \leq \mathcal{L}_{\hat{\mathcal{D}}}(h') + \sqrt{\frac{4}{m}(d \log \frac{2em}{d} + \log \frac{4}{\delta})} + \boxed{d_{\mathcal{H}}(\tilde{\mathcal{U}}', \tilde{\mathcal{U}}_i')} + \lambda_i,$$

where $\boxed{d_{\mathcal{H}}(\tilde{\mathcal{U}}', \tilde{\mathcal{U}}_i') = d_{\mathcal{H}}(\tilde{\mathcal{U}}^g, \tilde{\mathcal{U}}_i^g) \leq d_{\mathcal{H}}(\tilde{\mathcal{U}}, \tilde{\mathcal{U}}_i) = d_{\mathcal{H}}(\tilde{\mathcal{U}}_i, \tilde{\mathcal{U}})}$. $\tilde{\mathcal{U}}'$ and $\tilde{\mathcal{U}}_i'$ are the induced distributions of $\mathcal{U}$ and $\mathcal{U}_i$ under $\mathcal{F}'$, respectively.

Please refer to our paper for proofs.

Extensive Experiments

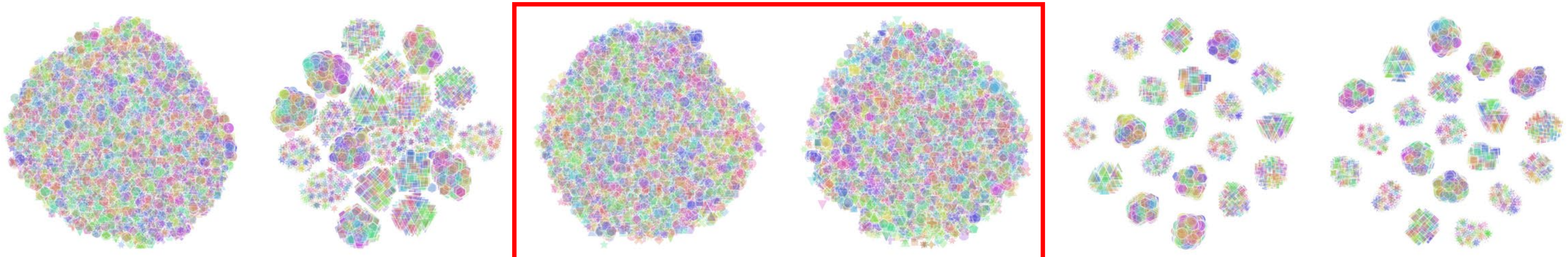
- How to Split the Model?

Table 1: The MDL (bits, ↓) of layer-wise representations, test accuracy (%, ↑), and the number of trainable parameters (↓) in PRBM when adding DBE to FedAvg on Tiny-ImageNet using 4-layer CNN in the practical setting. We also show corresponding results for the close pFL methods. For FedBABU, "[36.82]" indicates the test accuracy after post-FL fine-tuning for 10 local epochs.

Metrics	MDL				Accuracy	Param.
	CONV1→CONV2	CONV2→FC1	FC1→FC2	Logits		
FedPer [3]	5143	4574	3885	4169	33.84	—
FedRep [20]	5102	4237	3922	4244	37.27	—
FedRoD [14]	5063	4264	3783	3820	36.43	—
FedBABU [61]	5083	4181	3948	3849	16.86 [36.82]	—
Original (FedAvg)	5081	4151	3844	3895	19.46	0
CONV1→DBE →CONV2	4650 (-8.48%)	4105 (-1.11%)	3679 (-4.29%)	3756 (-3.57%)	21.81 (+2.35)	28800
CONV2→DBE →FC1	**4348 (-14.43%)**	3716 (-10.48%)	**3463 (-9.91%)**	**3602 (-7.52%)**	**47.03 (+27.57)**	10816
FC1→DBE →FC2	4608 (-9.31%)	**3689 (-11.13%)**	3625 (-5.70%)	3688 (-5.31%)	43.32 (+23.86)	512

Extensive Experiments

- Eliminate Representation Bias for the First Level of Representation ***after*** local training



(a) FedAvg (B). (b) FedAvg (A). (c) +DBE ($\boldsymbol{z}_i^g$, B). (d) +DBE ($\boldsymbol{z}_i^g$, A). (e) +DBE ($\boldsymbol{z}_i$, B). (f) +DBE ($\boldsymbol{z}_i$, A).

Figure 3: t-SNE visualization for representations on Tiny-ImageNet (200 labels). "B" and "A" denote "before local training" and "after local training", respectively. We use *color* and *shape* to distinguish *labels* and *clients*, respectively. *Best viewed in color and zoom-in.*

Extensive Experiments

- **DBE** can greatly improve existing FL methods in both **generalization** and **personalization** abilities

Extensive Experiments

- **DBE** promotes traditional FL methods in both MDL and accuracy by at most
  - **-22.35%** in MDL (bits) and
  - **+32.30** in accuracy (%)

Table 4: The MDL (bits, ↓) and test accuracy (%, ↑) before and after adding DBE to traditional FL methods on Cifar100, Tiny-ImageNet, and AG News in the practical setting. TINY and TINY* represent using 4-layer CNN and ResNet-18 on Tiny-ImageNet, respectively.

Metrics	**MDL**				**Accuracy**			
Datasets	Cifar100	TINY	TINY*	AG News	Cifar100	TINY	TINY*	AG News
SCAFFOLD [38]	1499	3661	3394	1931	33.08	23.26	24.90	88.13
FedProx [46]	1523	3701	3570	2092	31.99	19.37	19.27	87.21
MOON [45]	1516	3696	3536	1836	32.37	19.68	19.02	84.14
FedGen [96]	1506	3675	3551	1414	30.96	19.39	18.53	89.86
SCAFFOLD+DBE	**1434**	**3549**	**3370**	**1743**	**63.61**	**45.55**	**45.09**	**96.73**
FedProx+DBE	**1439**	**3587**	**3490**	**1689**	**63.22**	**42.28**	**41.45**	**96.62**
MOON+DBE	**1432**	**3580**	**3461**	**1683**	**63.26**	**43.43**	**41.10**	**96.68**
FedGen+DBE	**1426**	**3563**	**3488**	**1098**	**63.26**	**42.54**	**41.87**	**97.16**

Extensive Experiments

- **DBE** greatly improves FedAvg at most **+47.40** on Cifar100 in the pathological setting and
- outperforms the SOTA pFL methods by up to **+11.36** on Cifar100†

Table 5: The test accuracy (%, ↑) of pFL methods in two statistically heterogeneous settings. Cifar100† represents the experiment with 100 clients and joining ratio $\rho = 0.5$ on Cifar100.

Settings	Pathological setting			Practical setting					
	FMNIST	Cifar100	TINY	FMNIST	Cifar100	Cifar100†	TINY	TINY*	AG News
Per-FedAvg [22]	99.18	56.80	28.06	95.10	44.28	38.28	25.07	21.81	87.08
pFedMe [67]	99.35	58.20	27.71	97.25	47.34	31.13	26.93	33.44	87.08
Ditto [47]	99.44	67.23	39.90	97.47	52.87	39.01	32.15	35.92	91.89
FedPer [3]	99.47	63.53	39.80	97.44	49.63	41.21	33.84	38.45	91.85
FedRep [20]	99.56	67.56	40.85	97.56	52.39	41.51	37.27	39.95	92.25
FedRoD [14]	99.52	62.30	37.95	97.52	50.94	48.56	36.43	37.99	92.16
FedBABU [61]	99.41	66.85	40.72	97.46	55.02	52.07	36.82	34.50	95.86
APFL [21]	99.41	64.26	36.47	97.25	46.74	39.47	34.86	35.81	89.37
FedFomo [89]	99.46	62.49	36.55	97.21	45.39	37.59	26.33	26.84	91.20
APPLE [52]	99.30	65.80	36.22	97.06	53.22	—	35.04	39.93	84.10
FedAvg	80.41	25.98	14.20	85.85	31.89	28.81	19.46	19.45	87.12
FedAvg+DBE	**99.74**	**73.38**	**42.89**	**97.69**	**64.39**	**63.43**	**43.32**	**42.98**	**96.87**

Extensive Experiments

- Other experiments also show the ***effectiveness*** and ***efficiency*** of our **DBE**.

Table 6: The test accuracy (%, ↑) and computation overhead (↓) of pFL methods.

Items	Heterogeneity			pFL+MR		Overhead	
	$\beta = 0.01$	$\beta = 0.5$	$\beta = 5$	Accuracy	Improvement	Total time	Time/iteration
Per-FedAvg [22]	39.39	21.14	12.08	—	—	121 min	3.56 min
pFedMe [67]	41.45	17.48	4.03	—	—	1157 min	10.24 min
Ditto [47]	50.62	18.98	21.79	42.82	10.67	318 min	11.78 min
FedPer [3]	51.83	17.31	9.61	41.78	7.94	83 min	1.92 min
FedRep [20]	55.43	16.74	8.04	41.28	4.01	471 min	4.09 min
FedRoD [14]	49.17	23.23	16.71	42.74	6.31	87 min	1.74 min
FedBABU [61]	53.97	23.08	15.42	38.17	1.35	811 min	1.58 min
APFL [21]	49.96	23.31	16.12	39.22	4.36	156 min	2.74 min
FedFomo [89]	46.36	11.59	14.86	29.51	3.18	193 min	2.72 min
APPLE [52]	47.89	24.24	17.79	—	—	132 min	2.93 min
FedAvg	15.70	21.14	21.71	—	—	365 min	1.59 min
FedAvg+DBE	**57.52**	**32.61**	**25.55**	—	—	171 min	1.60 min

Eliminating Domain Bias for Federated Learning in Representation Space

Paper with code: https://github.com/TsingZ0/DBE

E-mail: tsingz@sjtu.edu.cn

Paper with code

Thanks!