Adaptive Federated Learning for Large Models with Scarce and Non-IID Data in Cloud-Edge Networks

Benteng Zhang[a], Yingchi Mao[a], Peng Zhang[b], Haotian Zheng[a], Xiaoming He[c], Jiawen Kang[d], and Jie Wu[e]

[a] College of Computer Science and Software Engineering, Hohai University, Nanjing, China
[b] Research and Development Center of Science and Technology, Huaneng Lancang River Hydropower Inc., China
[c] College of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing, China
[d] Guangdong University of Technology, Guangzhou, China
[e] Center for Networked Computing, Temple University, Philadelphia, USA

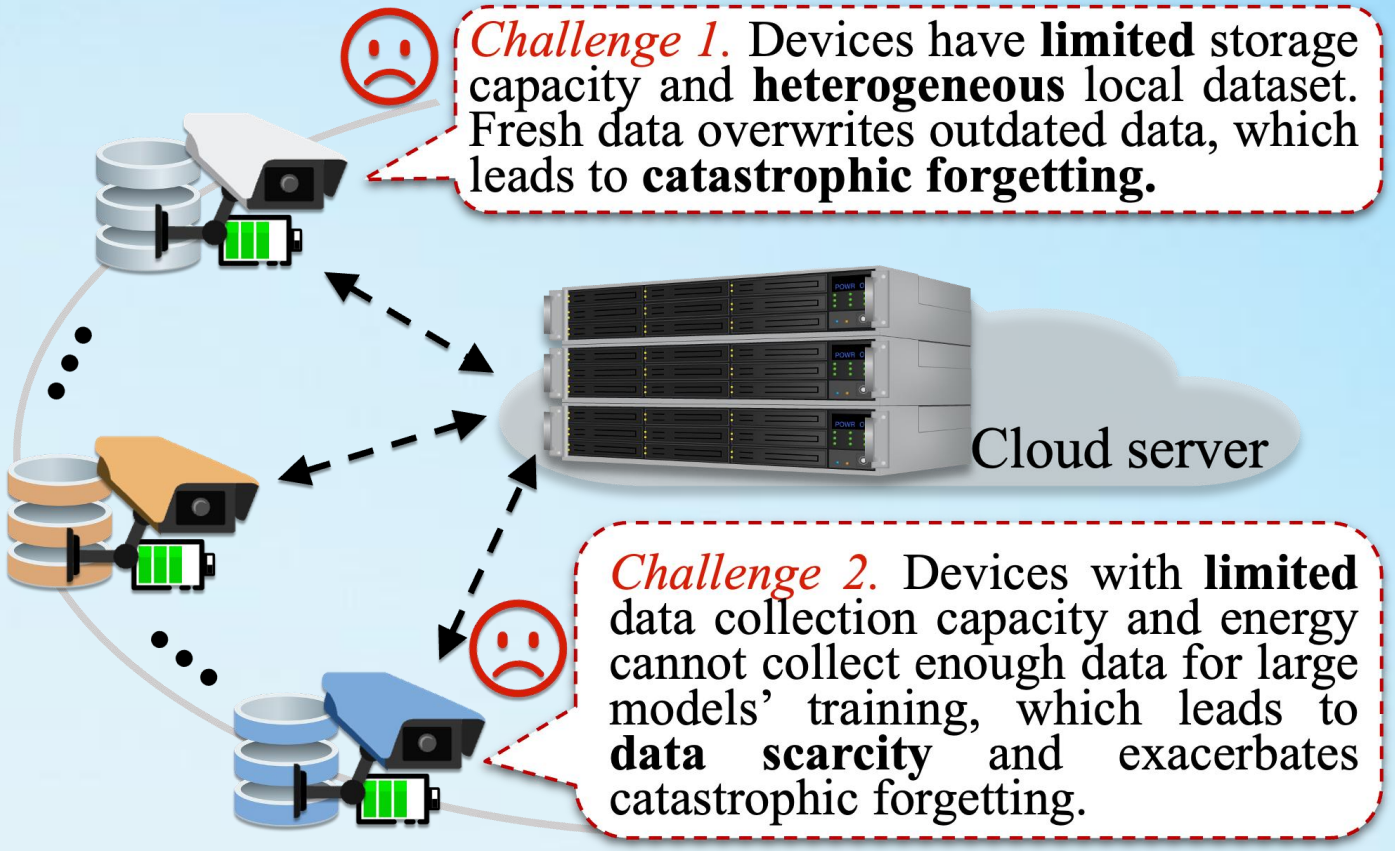
IEILM 2025

Contents

Introduction

■ Background

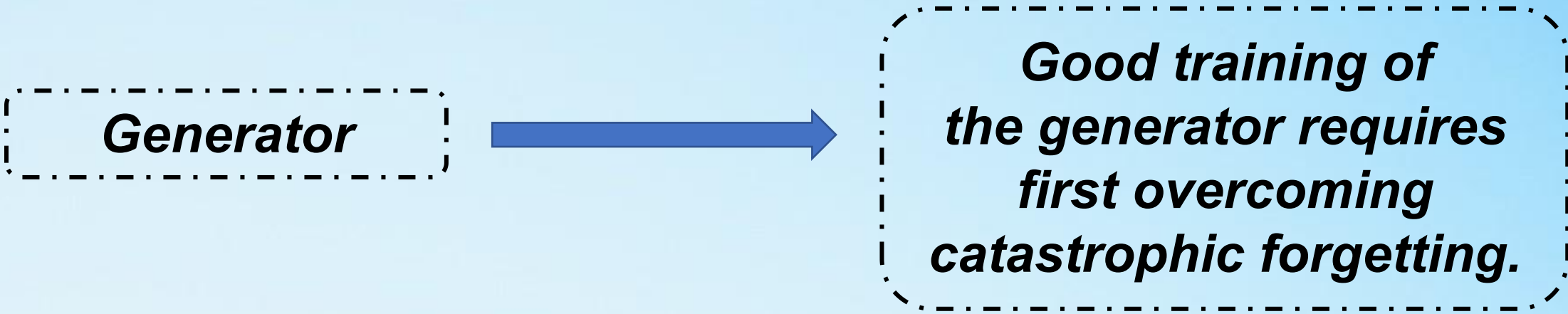
Federated Learning (FL) can coordinate numerous IoT devices to train large models collaboratively and provide intelligent services and applications. However, large models' training requires significant memory and data, which creates challenges for resource-constrained IoT devices.



Introduction

■ Background

To collect more training data, existing methods generate data for training. However, due to catastrophic forgetting, the discriminators may forget outdated data's characteristics, which undermines the stability of the generative network's training.



Introduction

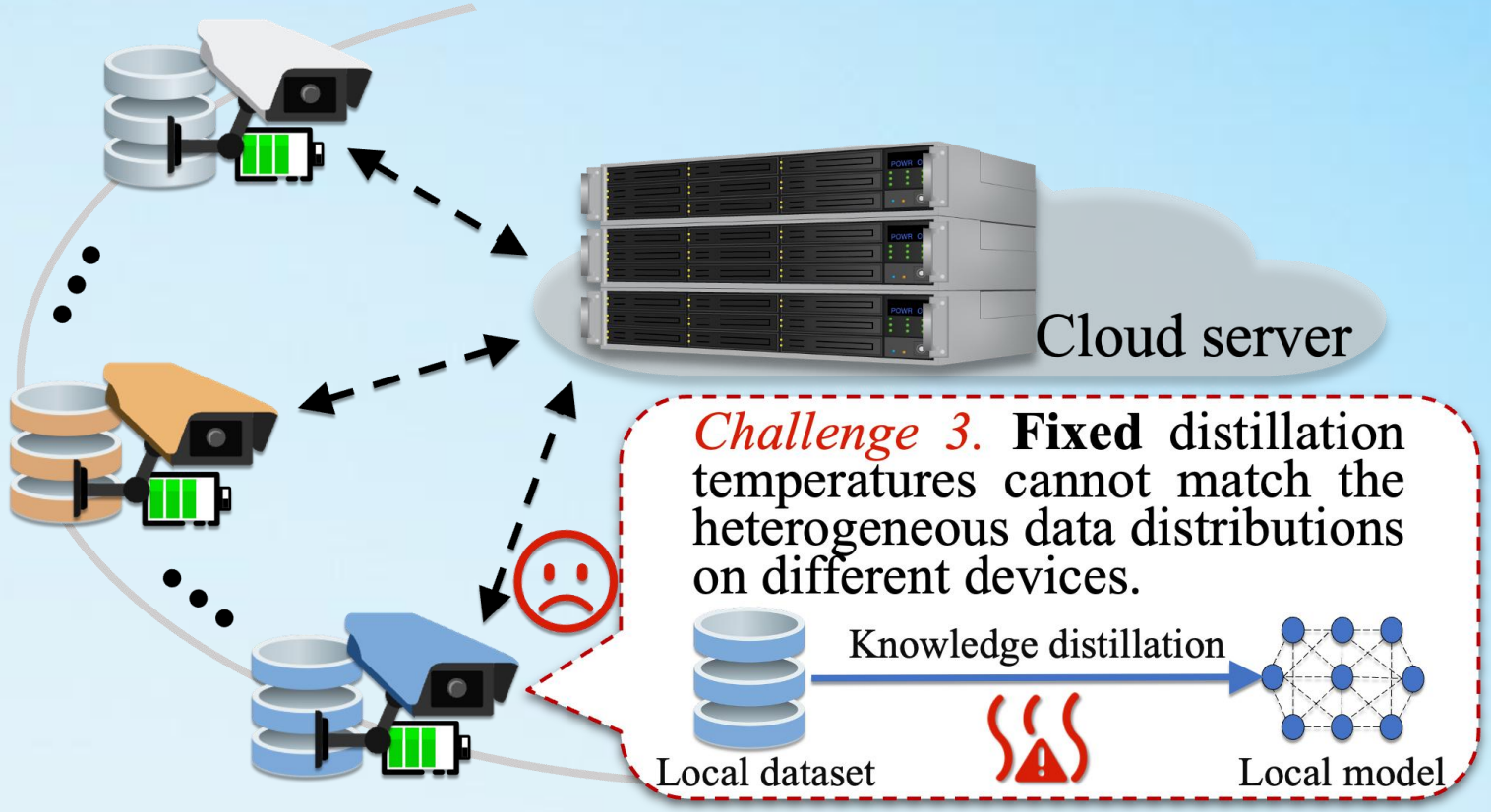
■ Background

Furthermore, Introducing Knowledge Distillation into FL to extract and combine characteristics from both fresh and outdated data can effectively guide local training and mitigate catastrophic forgetting.

Introduction

Background

Existing Knowledge Distillation methods use fixed distillation temperatures across different devices, which ignores that fixed distillation temperatures cannot match the heterogeneous data distributions on devices. This leads to poor distillation performance and fails to mitigate catastrophic forgetting.



Introduction

■ Optimization object

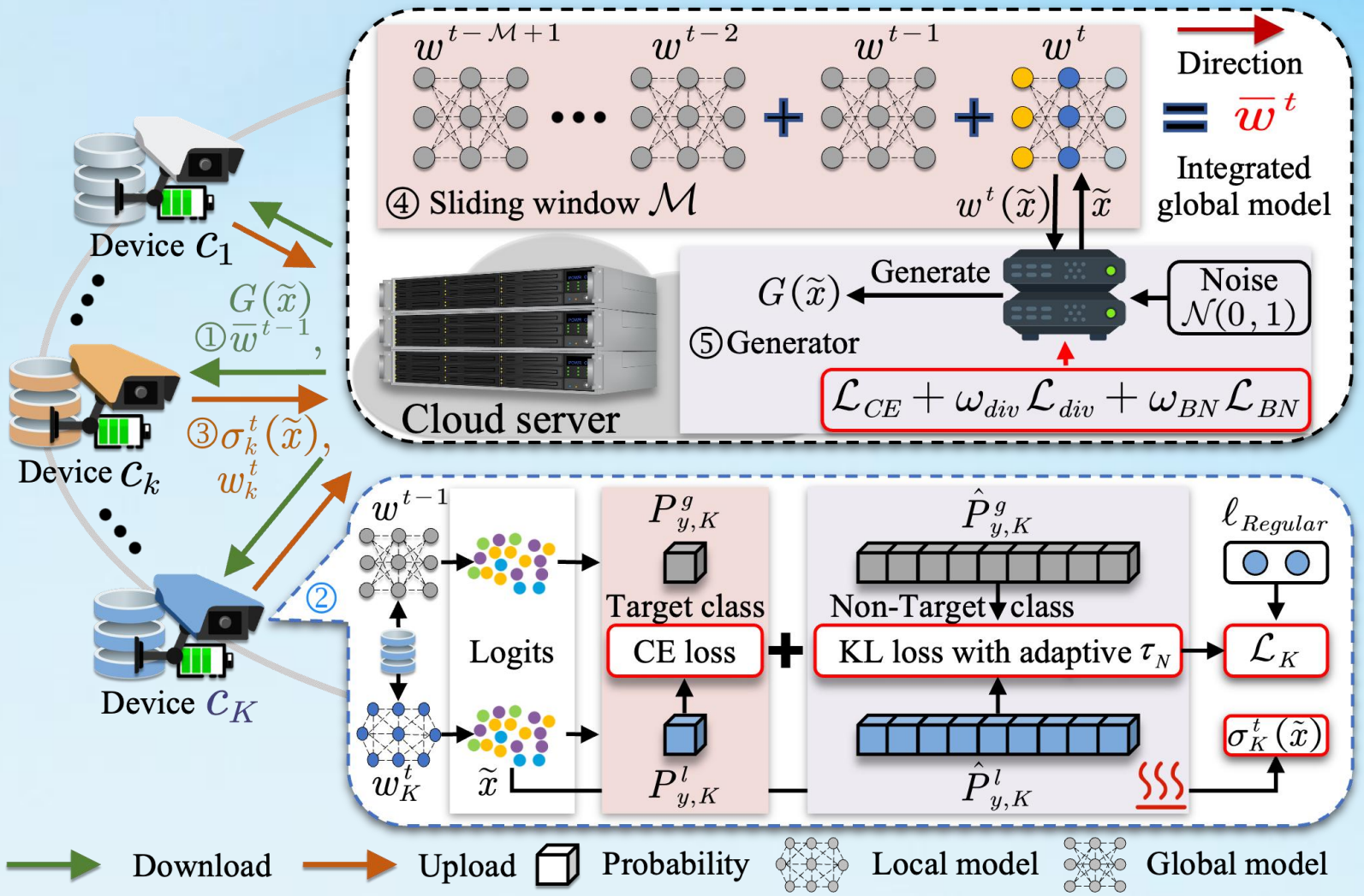
Dynamically adjust distillation temperatures for devices based on the heterogeneous data distribution to overcome catastrophic forgetting and stabilize the generator's training.

■ Goal

Improve distillation performance, mitigate catastrophic forgetting and stabilize the generator's training.

Approach: FedFKD

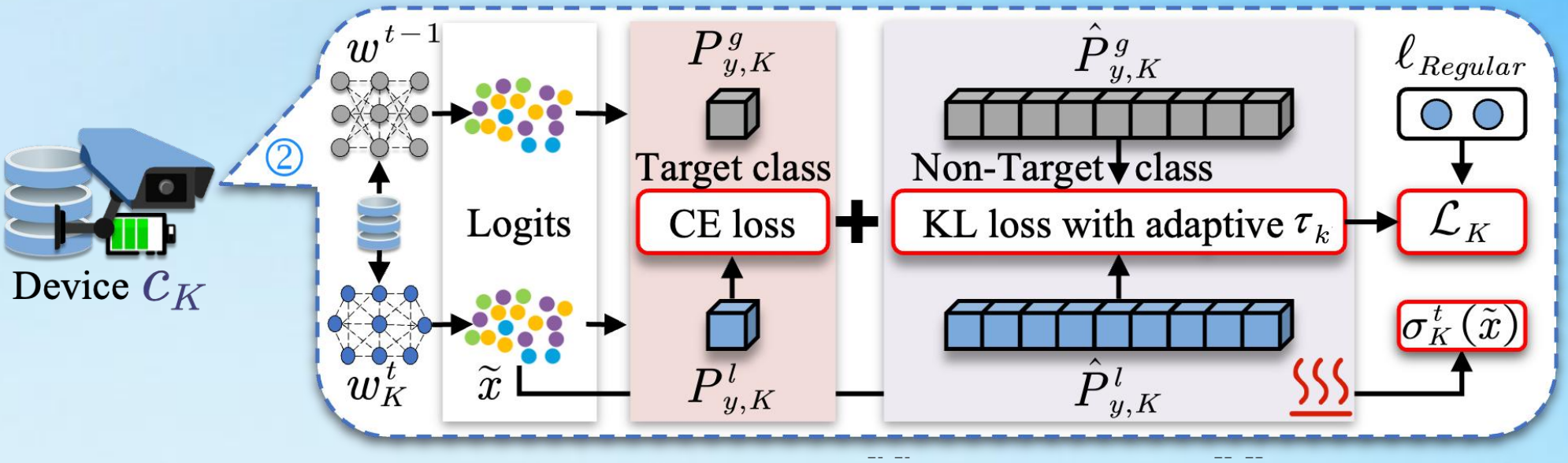
Federated Fine-tuning Adaptive Aggregation via Knowledge Distillation (FedFKD).



1. Global model delivery.
2. Decoupled distillation and local training.
3. Upload parameters.
4. Global aggregation and model integration.
5. Generate data.

Approach: DP-FedAGS

(1) Adjusting Distillation Temperature and Decoupled Distillation



$$\tau_k = \tau[s(w_k^t, [\mathcal{D}_k]_b)],$$

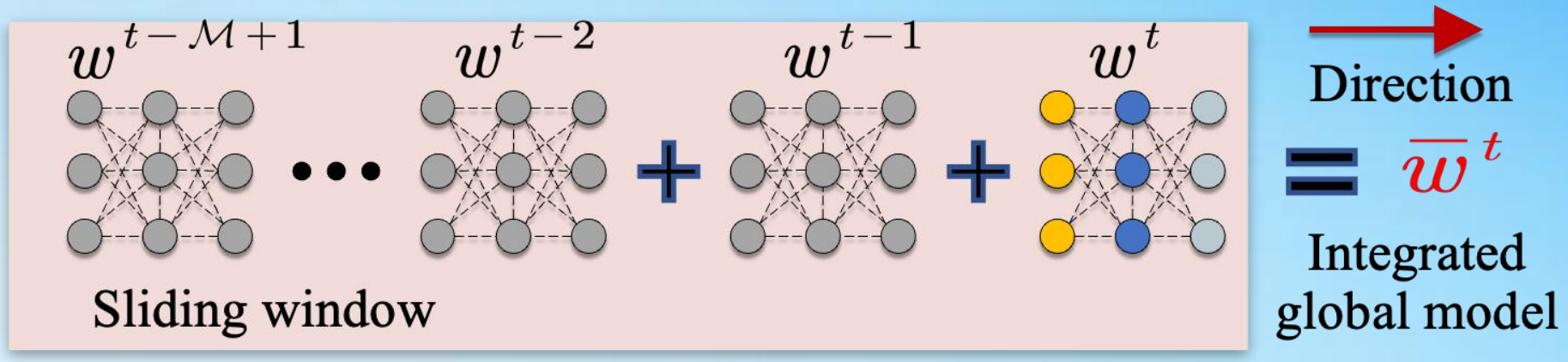
$\tau[\cdot]$: the standard deviation function

$[\mathcal{D}_k]_b$: the *b*-th mini-batch of data

$s(w_k^t, x)$: The Logits vector output

Approach

(2) Model Weighted Aggregation and Integration



Aggregation weight

$$\sigma_k^t(\tilde{x}) = Var(s(w_k^t, \tilde{x})),$$

$$\alpha_k^t(\tilde{x}) = \sigma_k^t(\tilde{x}) / \sum^{K} \sigma_k^t(\tilde{x}).$$

Model aggregation

$$w^t = \sum^{K} \alpha_k^t(\tilde{x}) w_k^t.$$

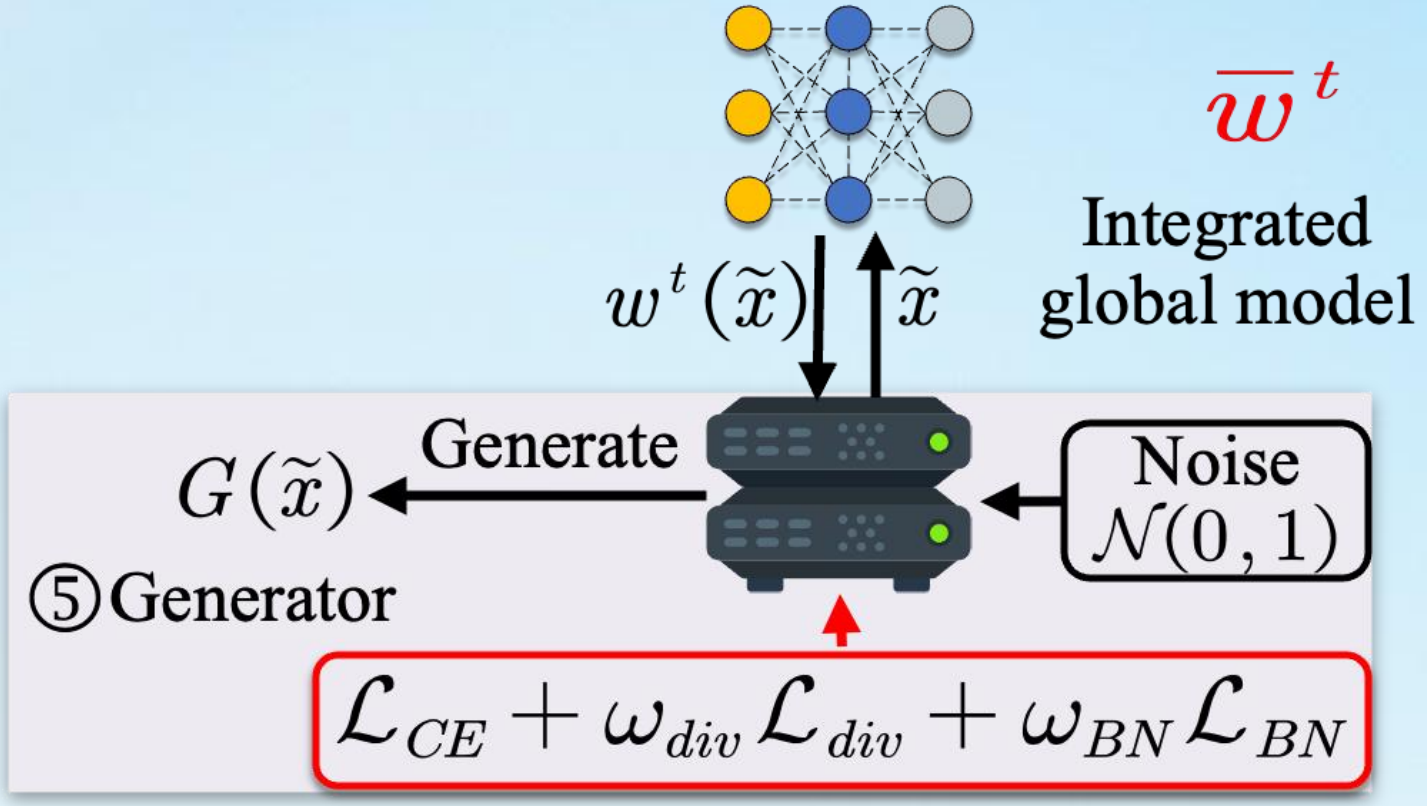
Model integration

$$\overline{w}^t = \frac{1}{\mathcal{M}} \sum_{m=1}^{\mathcal{M}} w^{t-m+1}.$$

Approach

(3) Data Generator

The goal of *G* is to generate data that is similar to the real data distribution.



$$\mathcal{L}_{CE} = CE(\overline{w}^t(\tilde{x}), \tilde{y}).$$

$$\mathcal{L}_{div} = -\mathcal{L}_{KL}(G(\tilde{x}), w^t(\tilde{x})).$$

$$\mathcal{L}_{BN} = \sum_l (\|\mu_l(\tilde{x}) - \mu_l\| + \|\epsilon_l^2(\tilde{x}) - \epsilon_l^2\|),$$

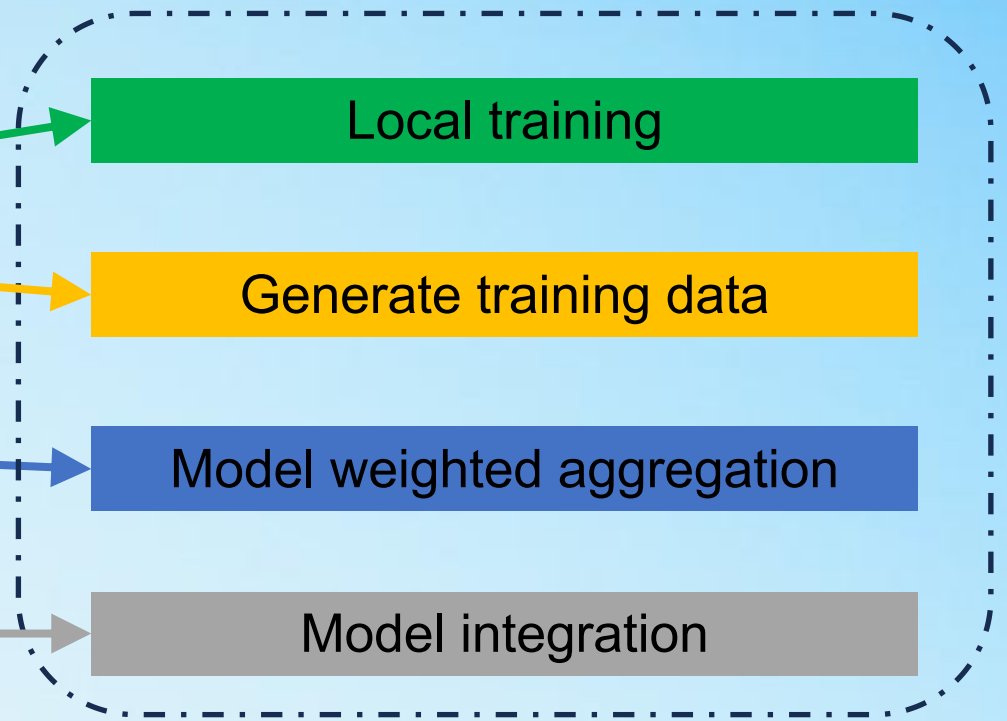
Approach

FedFKD

Algorithm 1: FedFKD

Input: N, T, E, $\mathcal{M}$, r, β, θ, ω_{div}, ω_{BN}
Output: Global model w

Initial w^0 and w_G
begin
  for *each training round* $t = 1, 2, 3, ..., T$ **do**
    for *device* c_k *in parallel* **do**
      Save global model w^{t-1} and $w_k^t \leftarrow w^{t-1}$
      $w_k^t \leftarrow ClientUpdate(w_k^t, \mathcal{D}_k)$
      $\tilde{x} = Generate(w^{t-1})$
      $\sigma_k^t(\tilde{x}) = Var(s(w_k^t, \tilde{x}))$
    end
    $\alpha_k^t(\tilde{x}) = \sigma_k^t(\tilde{x}) / \sum_k \sigma_k^t(\tilde{x})$
    $w^t = \frac{1}{K} \sum_{k=1}^{K} \alpha_k^t(x) w_k^t$
    if $t < \mathcal{M}$ **then**
      Send w^t to devices
    else
      Send $w^t = \overline{w}^t = \frac{1}{\mathcal{M}} \sum_{m=1}^{\mathcal{M}} w^{t-m+1}$ to devices
    end
  end
  return w
end



Approach

FedFKD

function $ClientUpdate(w_k^t, \mathcal{D}_k)$
begin
  for *local epoch* $e = 1, ..., r$ **do**
    for *batch* $b \in B$ **do**
      $\tau_k = \tau[s(w_k^t, [D_k]_b)]$
      $\mathcal{L}_k = \mathcal{L}_{DeD}(w_k^t, w^{t-1}, [D_k]_b) + \theta\mathcal{L}_{Regular}(w_k^t, [D_k]_b)$
      $w_k^t \leftarrow w_k^{t,0} - \frac{\eta}{|[D_k]_b|} \sum_{e=0}^{r-1} \sum_{b=1}^{B} \nabla\mathcal{L}_k$
    end
  end
  return w_k^t to server
end
function $Generate(w^{t-1})$
begin
  Input noise $z \sim \mathcal{N}(0, 1)$ and $w_G \leftarrow w^{t-1}$
  Generate $\tilde{x}$ with label $\tilde{y}$
  Update w_G by $\min_G \mathcal{L}_{CE} + \omega_{div}\mathcal{L}_{div} + \omega_{BN}\mathcal{L}_{BN}$
  return $\tilde{x}$
end

- Calculate the distillation temperature
- Decoupled Distillation
- Local model update
- Generate training data
- update generator

Experiment

Dataset & Model

Dataset	Target Model
CIFAR-10	2 convolutional layers and 2 fully connected layers
CIFAR-100	
Tiny-ImageNet	ResNet-18

Parameter Settings

Parameter	Value
Clients	10/100
Batch size	50
Learning rate	0.1 for devices, 0.01 for generator
Training round	200
Others	$\beta = 1.0$, $\theta = 0.1$, $\omega_{div} = 1$, $\omega_{BN} = 0.1$, $\mathcal{M} = 5$

Experiment

Non-IID datasets

***Dirichlet* function** $Dir(\alpha)$ $\alpha = \{0.05, 0.5, 1\}$

Baselines

- **FedAvg**
- **FedCurv**
- **FedProx**
- **FedProxGAN**

Metrics

- **Global model accuracy** Acc_g
- **Global model training loss** $Loss_g$

Experiment

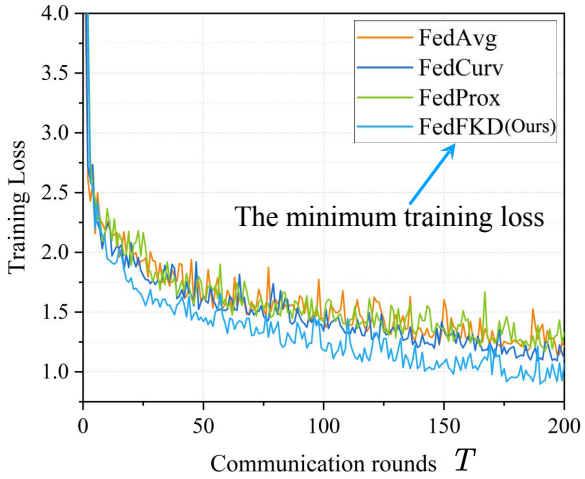
(1) Global model accuracy

GLOBAL MODEL ACCURACY Acc_g OF DIFFERENT TRAINING METHODS (%)

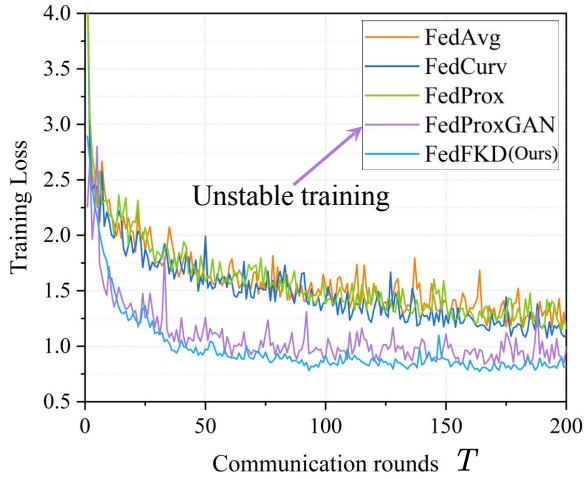
Method	CIFAR-10			CIFAR-100			Tiny-ImageNet			Average
	$Dir(0.05)$	$Dir(0.5)$	$Dir(1)$	$Dir(0.05)$	$Dir(0.5)$	$Dir(1)$	$Dir(0.05)$	$Dir(0.5)$	$Dir(1)$	
FedAvg	35.13	66.97	73.25	31.31	37.32	38.62	13.62	16.93	17.70	36.76
FedCurv	34.51	67.34	72.82	30.85	35.99	39.16	14.26	17.97	19.62	36.95
FedProx	38.16	63.98	62.65	32.13	32.10	35.70	16.73	18.17	21.57	35.69
FedProxGAN	-	69.73	74.38	35.33	40.23	**43.31**	20.11	23.15	26.43	41.58
FedFKD (Ours)	**43.96**	**71.70**	**74.65**	**36.90**	**42.50**	42.56	**21.24**	**24.72**	**28.36**	**42.95**

Experiment

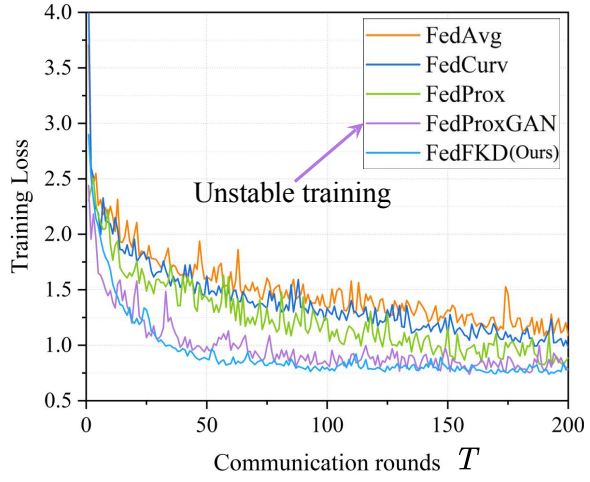
(2) Global model training loss



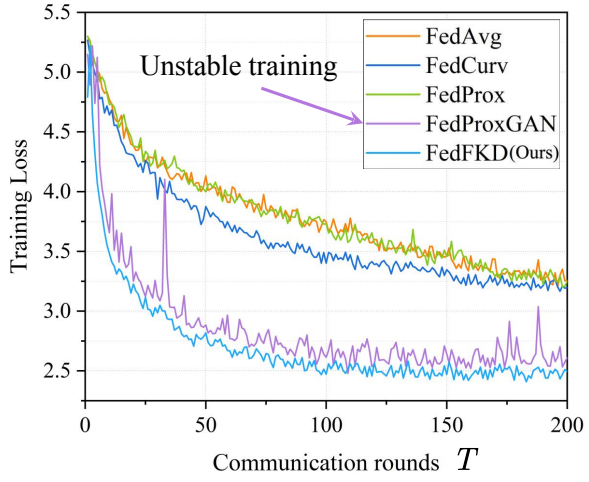
(a) $CIFAR-10,\ \alpha=0.05$



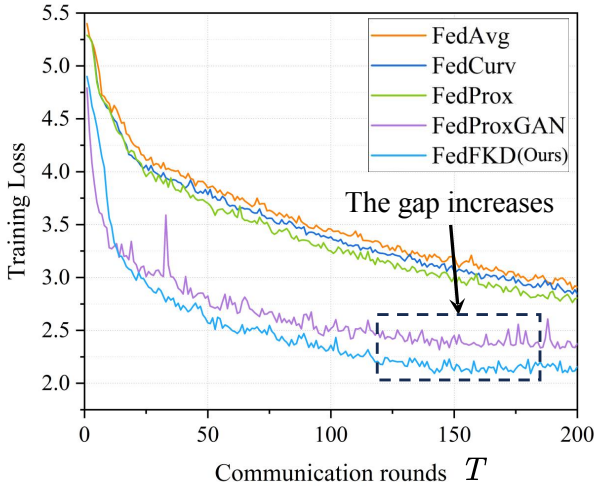
(b) $CIFAR-10,\ \alpha=0.5$



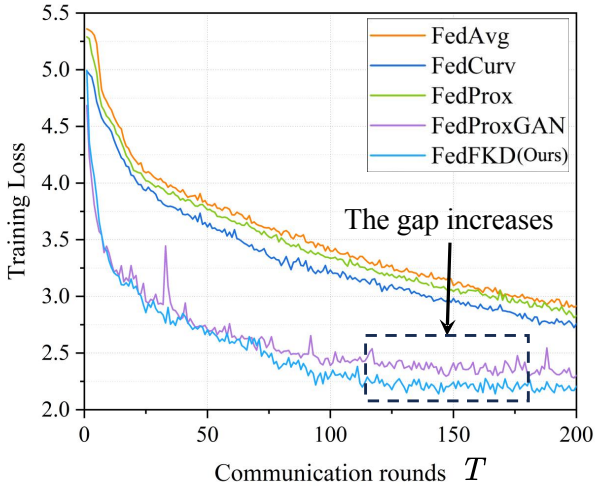
(c) $CIFAR-10,\ \alpha=1$



(d) $CIFAR-100,\ \alpha=0.05$



(e) $CIFAR-100,\ \alpha=0.5$



(f) $CIFAR-100,\ \alpha=1$

Conclusion

Conclusion

- In each training round, FedFKD utilizes aggregated global model to aid the generator's training.
- FedFKD dynamically assigns aggregation weights to guide the global model to update in the optimal direction.
- FedFKD builds an adaptive distillation temperature-aware mechanism to adjust distillation temperatures for each device dynamically.

Future Work

- Convergence analysis of FedFKD.

At Last

Thanks for your attention!