NEURAL INFORMATION PROCESSING SYSTEMS

Why Go Full? Elevating Federated Learning Through Partial Network Updates

Haolin Wang[◇†], **Xuefeng Liu**[◇♡], **Jianwei Niu**[◇♡‡], **Wenkai Guo**[◇†], **Shaojie Tang**[♠]

◇ State Key Laboratory of Virtual Reality Technology and Systems,
School of Computer Science and Engineering, Beihang University
♠ Center for AI for Business Innovation, School of Management, University at Buffalo.
♡ Zhongguancun Laboratory
{wanghaolin, liu_xuefeng, niujianwei, kyeguo}@buaa.edu.cn
shaojiet@buffalo.edu

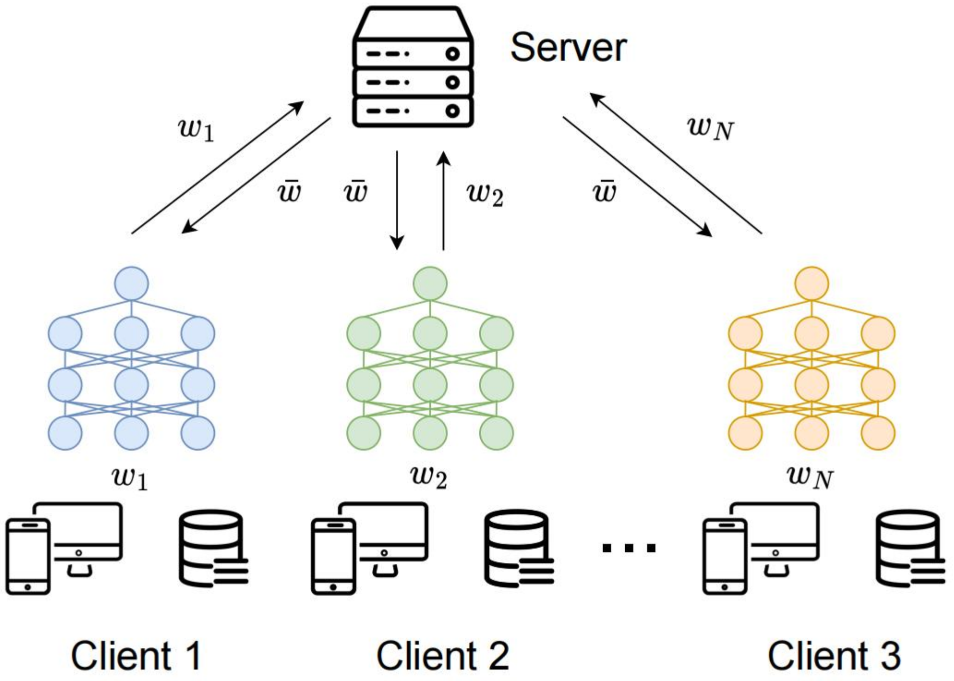
University at Buffalo
The State University of New York

The Paradigm of Federated Learning

NEURAL INFORMATION PROCESSING SYSTEMS

Target:

- Clients collaboratively train a global model without sharing private data.



Process:

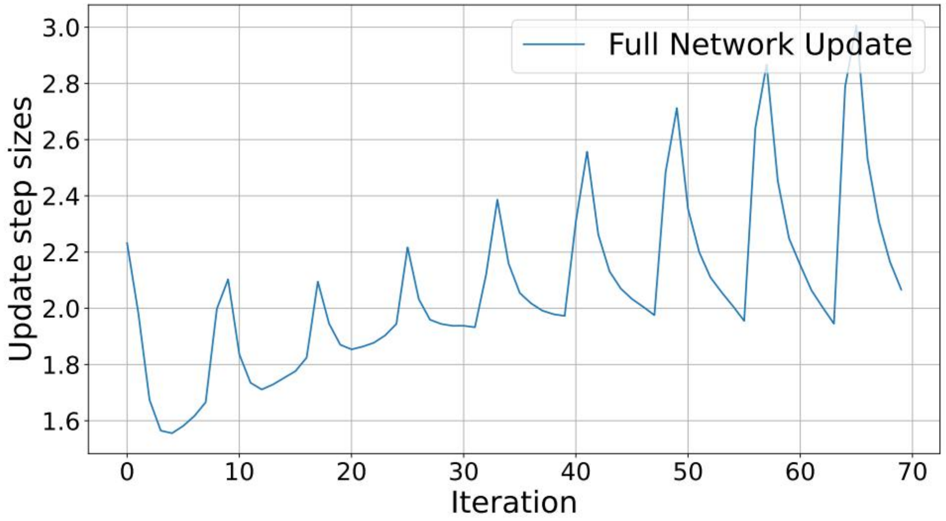
- Server broadcasts the global model.
- Clients locally update the model.
- Server aggregates all local updates to obtain an updated global model.
- Repeat the process.

Core Observation: Layer Mismatch

NEURAL INFORMATION PROCESSING SYSTEMS

Our question:

- Are global models really optimal for local tasks?
- Traditional View: Yes. The strong test results suggest better generalization.
- Our View: Probably not. Analysis of gradient norms in each iteration reveals potential issues.



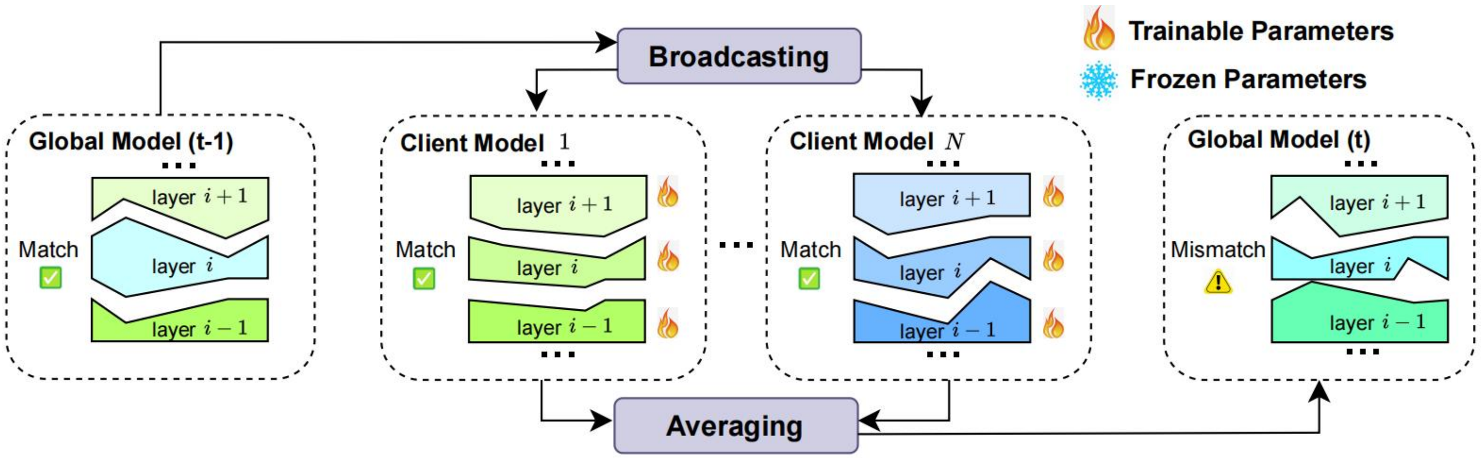
Observations:

- In federated learning, the gradient norm spikes after each aggregation.
- This can lead to training instability.
- Indicates that global models may not always suit individual local tasks.

Understanding the Phenomenon

Hypothesis:

- During back-propagation, the gradient for one layer depends on the parameters of the next.
- We believe parameter mismatch between layers causes this instability.



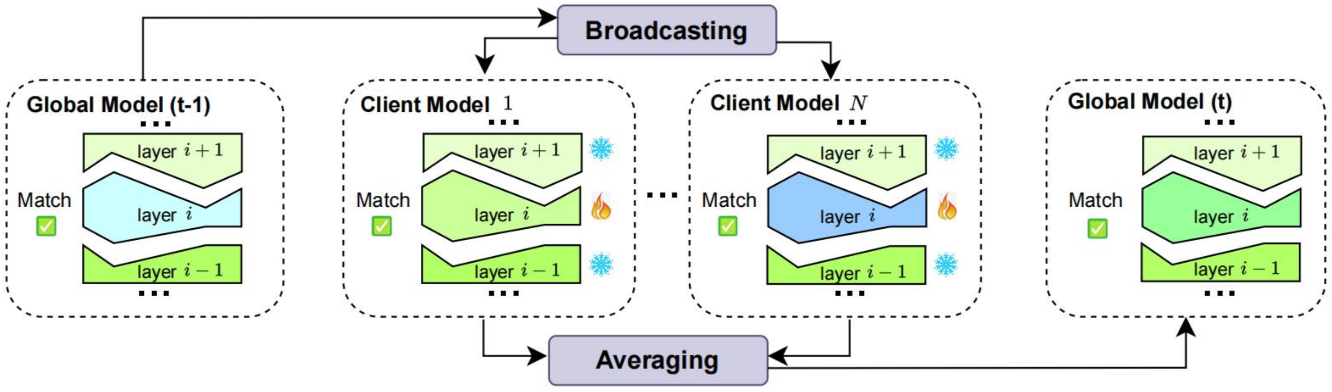
(a) FedAvg: Full network updates lead to layer mismatch.

Our Solution

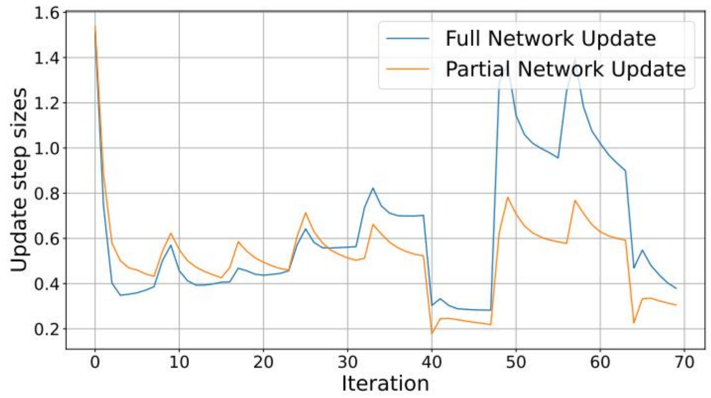
NEURAL INFORMATION PROCESSING SYSTEMS

Proposal:

- Train only part of the network in each round.
- Keep some layers frozen as "anchors" to enhance stability across updates.



(b) FedPart: Partial network updates help to reduce layer mismatch.



Reflection

NEURAL INFORMATION PROCESSING SYSTEMS

Advantages of Partial Network Training:

- Mitigates the layer mismatch issue.
- Reduces communication and computation costs.

Disadvantages:

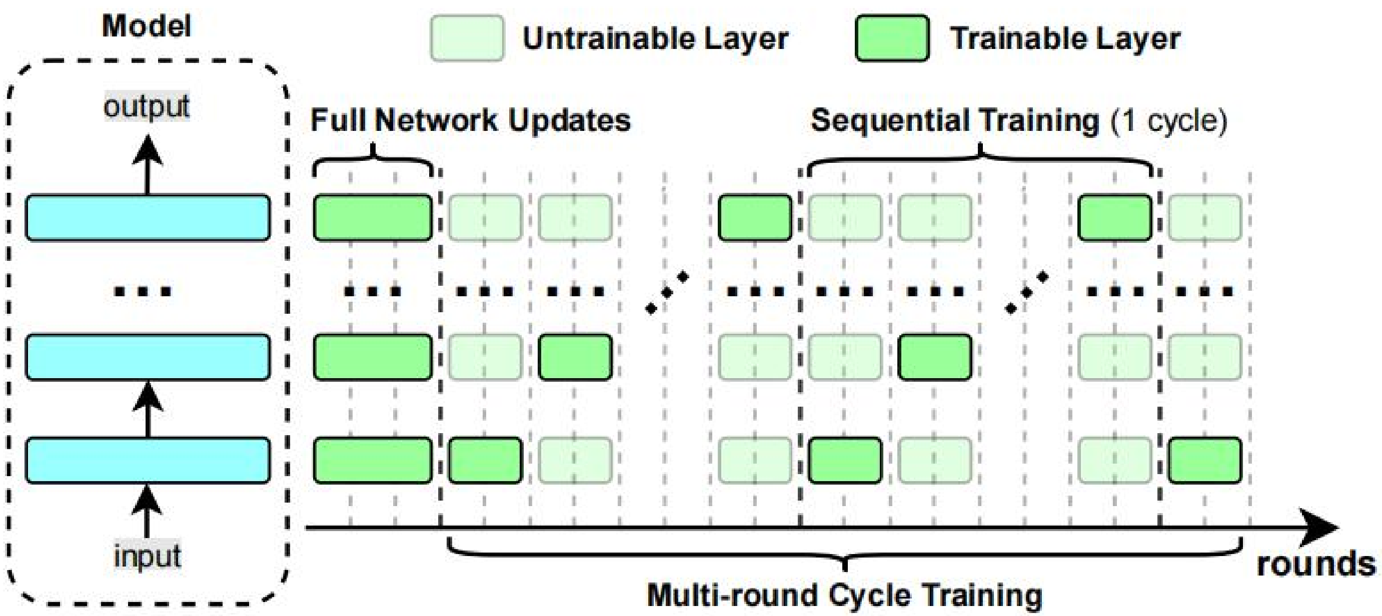
- Slower learning efficiency: Reduced convergence speed and performance.
- Less effective knowledge sharing: Communication between clients is limited.

Improvements: Selecting Trainable Layers

NEURAL INFORMATION PROCESSING SYSTEMS

Core ideas:

- Full parameter training is still essential at times.
- Implement partial network training sequentially.
- Repeat sequential training across multiple rounds for better results.



Experimental Results

NEURAL INFORMATION PROCESSING SYSTEMS

Settings:

- Baseline methods: FedAvg, FedProx, FedMoon.
- Our method shows significant improvements.
- Communication cost: ↓ 72%, Computation cost: ↓ 27%

Data	C	FedAvg FNU	FedAvg FedPart	FedProx FNU	FedProx FedPart	FedMoon FNU	FedMoon FedPart	Comm. FNU	Comm. FedPart	Comp. FNU	Comp. FedPart
CIFAR-10	1	56.0 (±1.1)	57.7 (±0.5)	54.4 (±2.1)	57.5 (±0.6)	58.9 (±0.5)	57.8 (±0.4)	4.83	1.35	4.38	3.21
	2	58.6 (±1.6)	60.2 (±0.4)	60.2 (±1.5)	59.9 (±0.5)	61.1 (±0.1)	59.4 (±0.2)	9.65	2.70	8.76	6.43
	3	59.6 (±1.7)	61.7 (±0.3)	62.3 (±0.7)	61.3 (±0.1)	62.3 (±0.4)	59.8 (±0.1)	14.5	4.05	13.2	9.64
	4	60.7 (±1.3)	**62.8 (±0.2)**	**62.8 (±1.1)**	62.3 (±0.1)	**62.3 (±0.4)**	60.5 (±0.6)	19.3	**5.40**	17.5	**12.9**
CIFAR-100	1	30.9 (±0.4)	31.0 (±0.5)	30.6 (±0.3)	30.9 (±0.5)	31.0 (±0.5)	30.9 (±0.4)	4.92	1.38	4.39	3.22
	2	32.9 (±0.3)	34.8 (±0.5)	33.6 (±0.5)	34.7 (±0.4)	33.2 (±0.9)	35.1 (±0.4)	9.65	2.75	8.78	6.44
	3	34.3 (±0.2)	36.1 (±0.5)	34.5 (±0.5)	36.7 (±0.4)	34.6 (±1.1)	36.5 (±0.6)	14.8	4.13	13.2	9.66
	4	35.6 (±0.3)	37.0 (±0.6)	35.8 (±0.2)	37.1 (±0.4)	35.0 (±1.0)	37.2 (±0.6)	19.7	5.51	17.6	12.9
	5	35.6 (±0.3)	**37.2 (±0.7)**	36.2 (±0.5)	**37.5 (±0.2)**	35.4 (±0.8)	**37.6 (±0.5)**	24.6	**6.88**	21.9	**16.1**
Tiny-ImageNet	1	15.6 (±0.6)	17.1 (±0.2)	15.8 (±0.4)	16.8 (±0.2)	17.5 (±0.6)	17.3 (±0.3)	5.02	1.40	17.5	12.9
	2	17.0 (±0.8)	20.3 (±0.1)	17.2 (±1.0)	20.1 (±0.2)	17.5 (±0.6)	20.5 (±0.0)	10.0	2.81	35.1	25.7
	3	17.6 (±0.4)	20.8 (±0.2)	18.0 (±0.5)	20.7 (±0.1)	18.4 (±0.8)	21.1 (±0.1)	15.1	4.21	52.6	38.6
	4	17.7 (±0.4)	21.1 (±0.1)	18.2 (±0.7)	21.2 (±0.1)	18.4 (±0.8)	21.5 (±0.1)	20.1	5.62	70.1	51.4
	5	17.7 (±0.4)	**21.4 (±0.2)**	18.4 (±0.8)	**21.5 (±0.2)**	18.4 (±0.8)	**21.7 (±0.1)**	25.1	**7.02**	87.7	**64.3**

Ablation Studies

NEURAL INFORMATION PROCESSING SYSTEMS

Q1: Can we enhance models already converged using full training?

A1: Yes. This further validates that our approach reduces the layer mismatch problem.

Table 6: Impact of the warm-up rounds.

Dataset	State	0 init.	5 init.	60 init.
CIFAR-10	bef.	0	41.56	58.92
	aft.	58.48	61.25	66.18
CIFAR-100	bef.	0	20.38	34.16
	aft.	29.53	33.59	36.65
Tiny-ImageNet	bef.	0	9.11	16.25
	aft.	16.81	20.69	19.99

Q2: Why is sequential training of layers effective?

A2: Likely because deeper layers build upon more fundamental, shallower ones.

Table 7: Impact of training sequences

Dataset	C	Seq.	Rev.	Ran.
CIFAR-10	1	58.80	58.53	59.62
	2	60.46	59.76	59.97
	3	**61.25**	60.19	60.23
CIFAR-100	1	30.07	27.84	29.58
	2	32.53	29.41	30.92
	3	**33.59**	31.79	31.44
Tiny-ImageNet	1	16.00	13.15	15.91
	2	19.25	15.62	17.71
	3	**20.69**	18.33	18.99

NEURAL INFORMATION PROCESSING SYSTEMS

Thank you for listening!

University at Buffalo
The State University of New York

Lab Homepage

Source Code