Lockdown: Backdoor Defense for Federated Learning with Isolated Subspace Training

Tiansheng Huang, Sihao Hu, Ka-Ho Chow, Fatih Ilhan, Selim Furkan Tekin, Ling Liu

School of Computer Science
Georgia Institute of Technology

Backdoor attack on FL

Trigger ■

Target Cat

Server

Apply update

Poisoned global model

Exhibit backdoor behavior

Serving/inference

Cat

Cat

Bird

Dog   Bird   Bird

Cat   Cat

👹 Malicious client   🧑‍💼 Benign client   Gradient update from clients
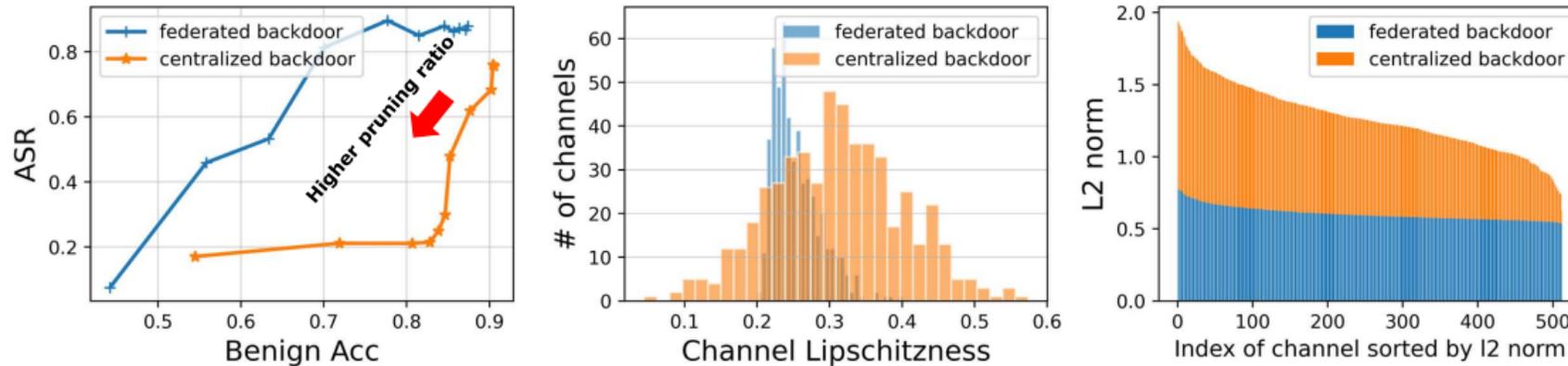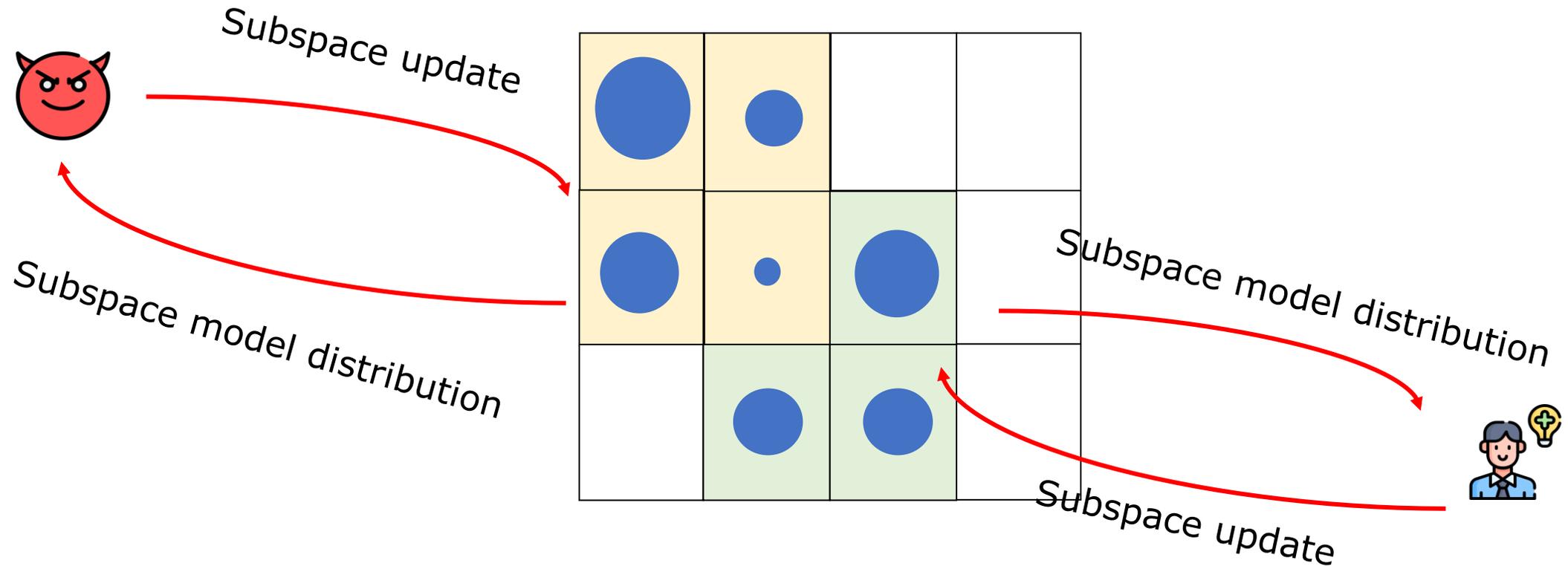
A poison coupling effect

Figure 2: Properties of two models trained with centralized backdoor and federated backdoor. Left: ASR and benign accuracy with CLP defense in (Zheng et al., 2022). Middle: Channel lipschitzness of last convolutional layer of two models. Right: L2 norm of last convolutional layer of two models.

Model poisoned by Federated backdoor is difficult to cured by pure pruning method

[5] Zheng R, Tang R, Li J, et al. Data-free backdoor removal based on channel lipschitzness[C]//Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V. Cham: Springer Nature Switzerland, 2022: 175-191.

Proactive defense for poisoned decoupling

Subspace: a set of parameters with constant size

Attacker can only poison a subspace of model, therefore mitigating the coupling effect

Pro-active local procedures:

- Isolated subspace training

- Mask searching

Poisoning removal via parameters pruning

- Consensus fusion

Isolated subspace training
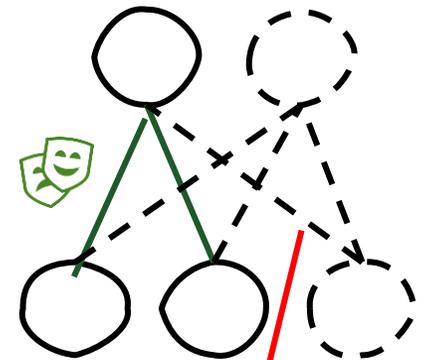
Initial sparse model before the first local step:

Each client's own subspace
(a binary mask)

$$\boldsymbol{w}_{i,t,0} = \boxed{\boldsymbol{m}_{i,t}} \odot \boldsymbol{w}_t$$

Keep the sparse structure according to the client's subspace

$$\boldsymbol{w}_{i,t,k+1} = \boldsymbol{w}_{i,t,k} - \eta \boxed{\boldsymbol{m}_{i,t}} \odot \nabla f_i(\boldsymbol{w}_{i,t,k}; \boldsymbol{\xi})$$

③Isolated subspace training

Keep sparse during K steps of local training

Dynamic subspace searching

Goal: Each client progressively involve the most important parameters within its subspace
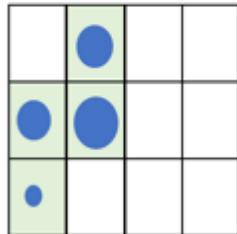
- Subspace initialization (each client has the same subspace)
- Subspace Pruning (criterion: absolute weight value)
- Subspace Recovery (criterion: gradient magnitude)

Dynamic subspace searching

Goal: Each client progressively involve the most important parameters within its subspace

- Subspace initialization (each client has the same subspace)
- Subspace Pruning (criterion: absolute weight value)
- Subspace Recovery (criterion: gradient magnitude)

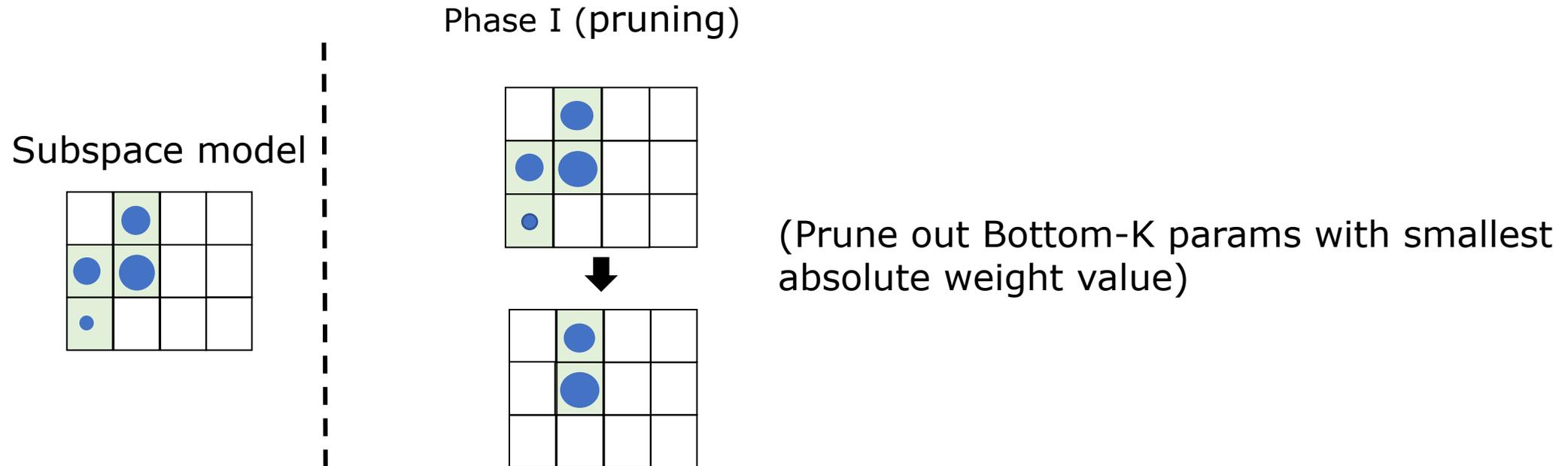Subspace model

Dynamic subspace searching

- Subspace initialization (each client has the same subspace)
- Subspace Pruning (criterion: absolute weight value)
- Subspace Recovery (criterion: gradient magnitude)

Phase I (pruning)

Subspace model

(Prune out Bottom-K params with smallest absolute weight value)

Dynamic subspace searching

- Subspace initialization (each client has the same subspace)
- Subspace Pruning(criterion: absolute weight value)
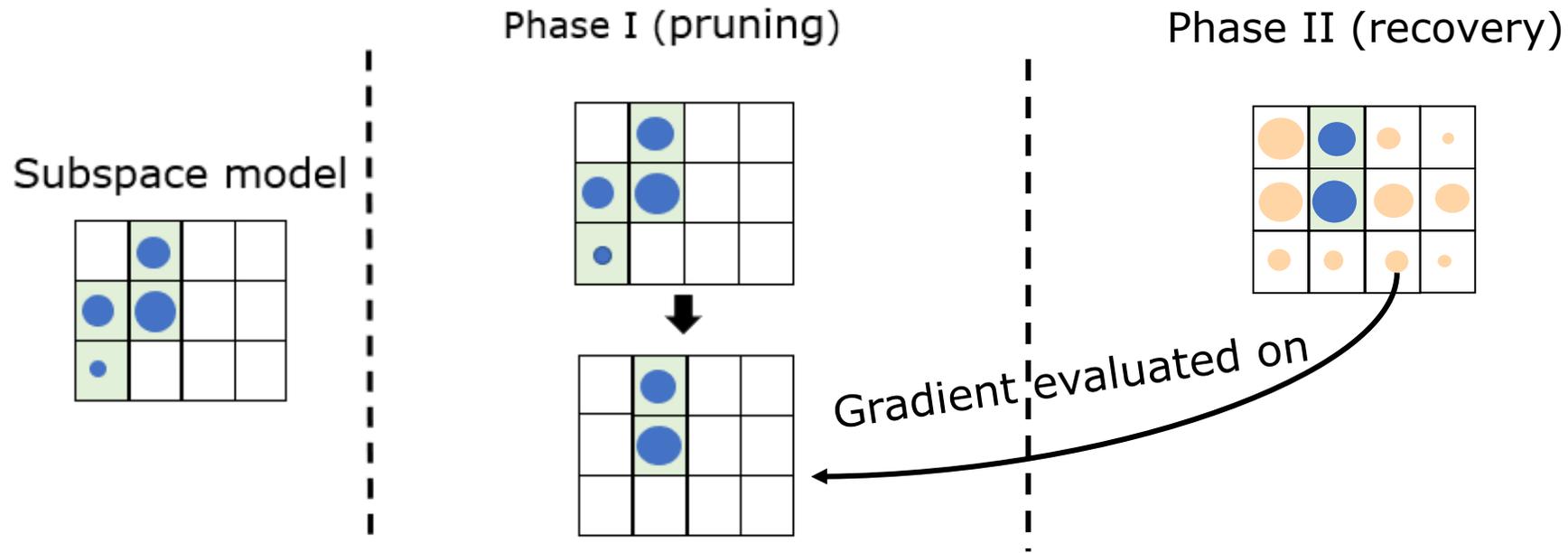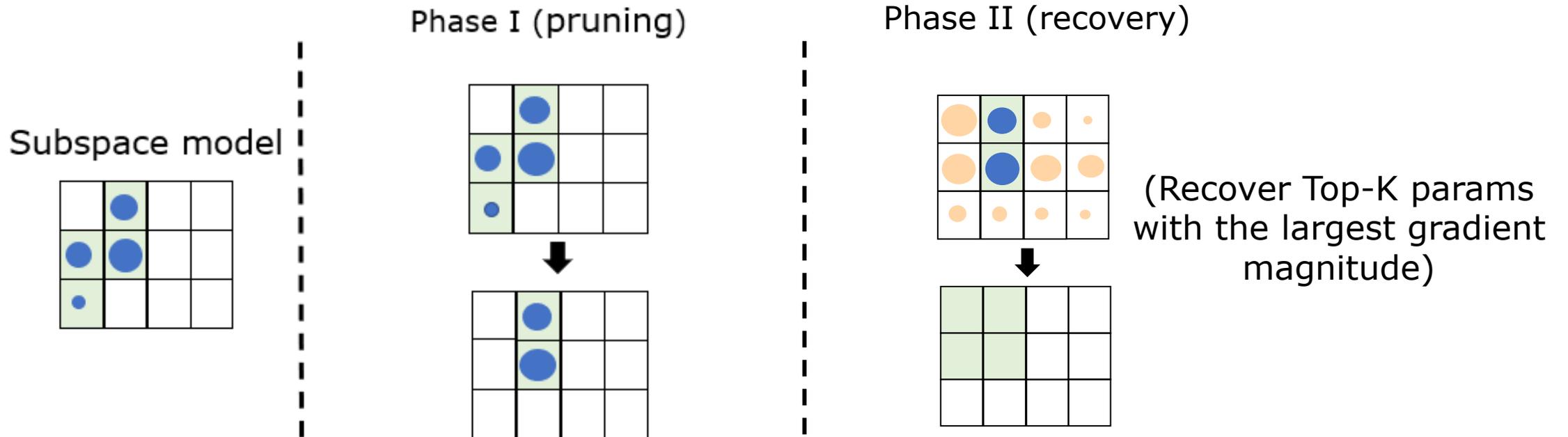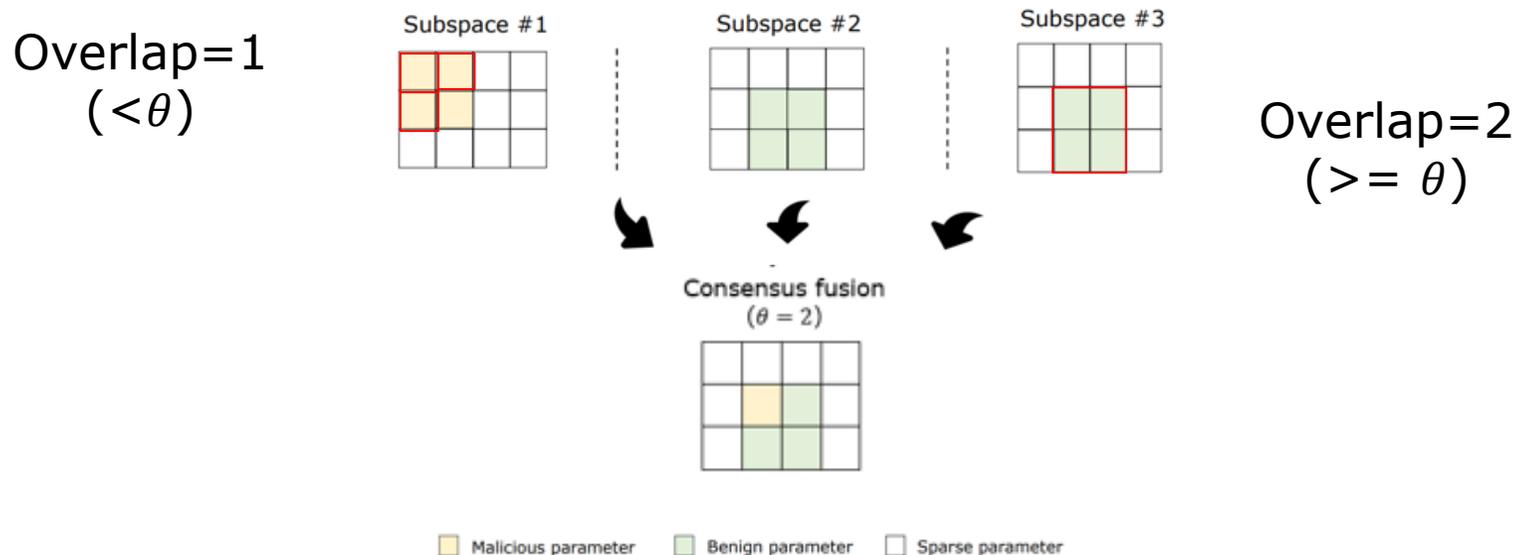- Subspace Recovery (criterion: gradient magnitude)

Dynamic subspace searching

- Subspace initialization (each client has the same subspace)
- Subspace Pruning(criterion: absolute weight value)
- Subspace Recovery (criterion: gradient magnitude)

(Recover Top-K params with the largest gradient magnitude)

Consensus fusion (after FL training)

- **Goal**: Prune out the poisoned parameters for the "subspace isolated poisoned" model

- **Intuition:** The poisoned parameters should not appear in the benign subspace.

- **How to prune?**

  ① Obtain the clean coordinates that have at least θ times overlap with others.

  ② Project the global model into the clean coordinates.

Overlap=1
$(<\theta)$

Subspace #1

Subspace #2

Subspace #3

Overlap=2
$(>= \theta)$

Consensus fusion
$(\theta = 2)$

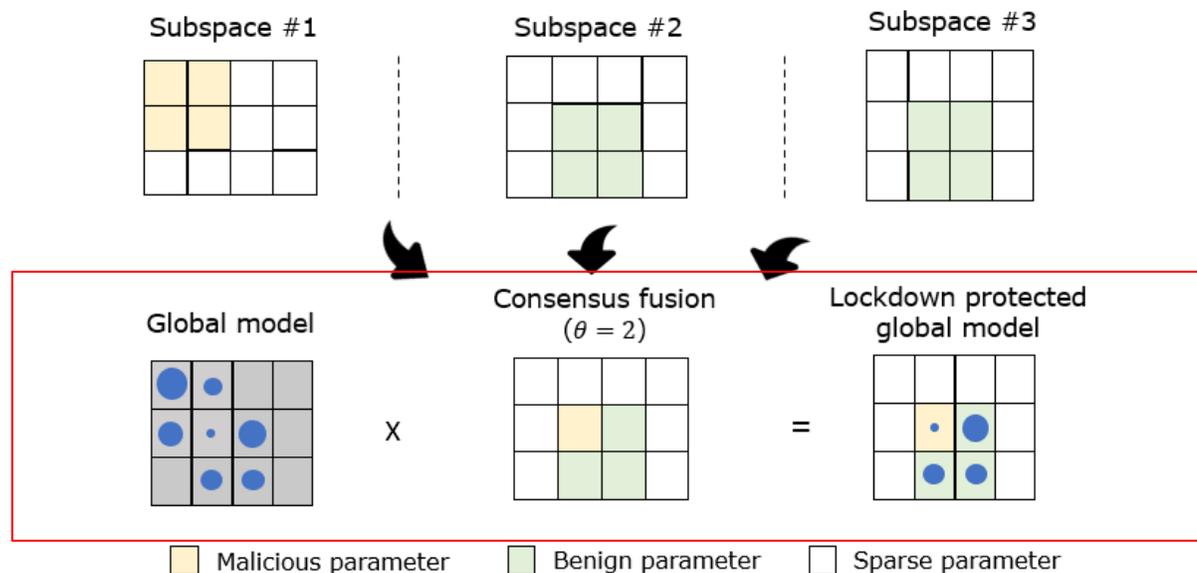☐ Malicious parameter   ☐ Benign parameter   ☐ Sparse parameter

Consensus fusion (after FL training)

- **Goal**: Prune out the poisoned parameters for the "subspace isolated poisoned" model

- **Intuition:** The poisoned parameters should not appear in the benign subspace.

- **How to prune?**

  ① Obtain the clean coordinates that have at least θ times overlap with others.

  ② Project the global model into the clean coordinates.

Experiment results

Attacker ratio: # of attackers / # of total clients (fix to 0.1)

Poison ratio p: Ratio of data being poisoned in an attacker

Methods	Benign Acc (%) ↑					ASR (%) ↓				
(IID)	clean	p=.05	p=.2	p=.5	p=.8	clean	p=.05	p=.2	p=.5	p=.8
FedAvg	**91.0**	**91.4**	**91.1**	**91.0**	**90.8**	**1.6**	12.4	19.9	66.1	94.8
RLR	86.8	86.7	86.6	86.3	85.5	2.3	**2.4**	**2.4**	**4.3**	25.1
Krum	76.3	78.0	75.6	76.4	75.8	4.7	3.9	4.3	4.3	4.9
RFA	90.9	91.2	91.1	90.8	90.7	**1.6**	15.8	20.7	83.7	99.3
Trimmed mean	91.0	90.6	91.1	90.9	90.8	1.7	5.0	20.7	61.7	96.2
Lockdown	90.0	90.0	89.9	90.1	90.0	1.8	3.6	2.5	7.1	**4.0**

Methods	Benign Acc (%) ↑					ASR (%) ↓				
(Non-IID)	clean	p=.05	p=.2	p=.5	p=.8	clean	p=.05	p=.2	p=.5	p=.8
FedAvg	**89.0**	**89.2**	**89.3**	**88.8**	**88.7**	1.7	17.3	54.4	86.4	96.7
RLR	74.4	74.4	73.6	72.9	72.5	5.8	15.0	40.2	29.5	82.5
Krum	42.7	37.4	45.2	43.4	45.1	10.0	**5.2**	10.4	11.1	10.6
RFA	88.8	88.8	88.8	88.3	88.3	2.0	21.4	52.8	90.8	98.7
Trimmed mean	88.5	88.4	88.2	88.3	88.3	1.9	25.2	48.4	84.6	96.0
Lockdown	85.6	86.2	86.7	86.1	86.6	**0.9**	7.6	**3.6**	**3.4**	**3.3**

ASR is lower up-to 93% compared with no defense, though with approx. 3% drop of benign acc

Defense efficacy is better for larger poison ratio!

Main Takeaway:

Proactive local mechanism is necessary for backdoor removal of a federated learning model.

Thank you!

Source code: https://github.com/git-disl/Lockdown