Backdoor Attacks on Federated Meta-Learning

Chien-Lun Chen, Leana Golubchik, Marco Paolieri

University of Southern California, Los Angeles, USA

I. Motivation

- Conventional federated learning is vulnerable to poisoning backdoor attacks
  - Existing defenses rely on a third-party to examine all user updates → can leak users' private data
- Poisoning backdoor attacks on federated *meta*-learning have not been investigated
  - Meta-learning allows users to train on different output classes → more practical in learning federation
  - The trained model can be *adapted to unseen/new tasks very quickly* (in only a few shots)
  - It is unclear whether meta-learning's *fast adaptation ability* can "forget" backdoors quickly

II. Main Contributions

- Investigate poisoning backdoor attacks on federated meta-learning, and show that
  - Effects of a one-shot attack can persist
  - Fine-tuning cannot effectively remove backdoors
- Propose a local defense mechanism that
  - Can remove backdoor effects *successfully*
  - Is *privacy-preserving*: does **not** require a (potentially untrustworthy) 3rd-party to examine user updates

III. Backdoor Attacks

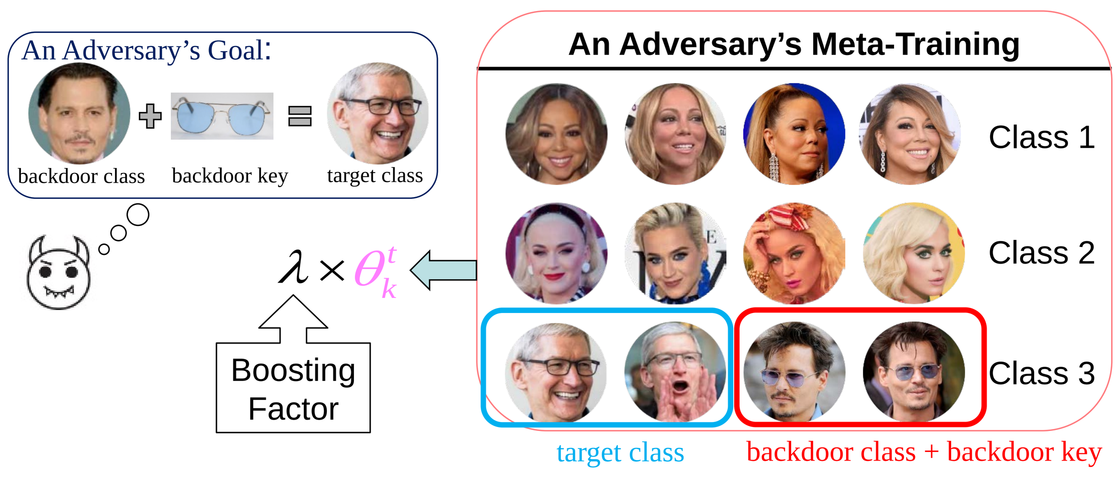
Open Questions

- (*Federated meta-training*) How quickly would updates from benign users dampen the effects of a one-shot backdoor attack?
- (*Meta-testing*) During fine-tuning, would meta-learning's adaptability help remove backdoors from a poisoned model?

IV. Experimental Setup

Threat Model – an illustrative example:

- Suppose an adversary wants Johnny Depp with glasses to be misclassified as Tim Cook



- During N-way K-shot meta-training, for the adversary, one of the N classes is always the target class, including some backdoor examples with a backdoor key
- The local update after poisoned training is then boosted before uploading to the server

Attack Evaluation – an illustrative example:

- We consider three different scenarios and two different test sets for evaluating the attack performance, illustrated as follows

Three different scenarios for evaluating attacks:

	Case (a)	Case (b)	Case (c)
During Federated Meta-Training	Benign clients do **NOT** have backdoor class	Benign clients have backdoor classes	Benign clients have backdoor classes
During Meta-Testing	Fine-Tuning / Test: No backdoor class is present during FT	Fine-Tuning / Test: No backdoor class is present during FT	Fine-Tuning / Test: Backdoor classes are present during FT

Two different test sets for evaluating attacks:

: attack training set (backdoor examples used by the attacker)

: attack validation set (backdoor examples NOT used by the attacker)

Datasets:

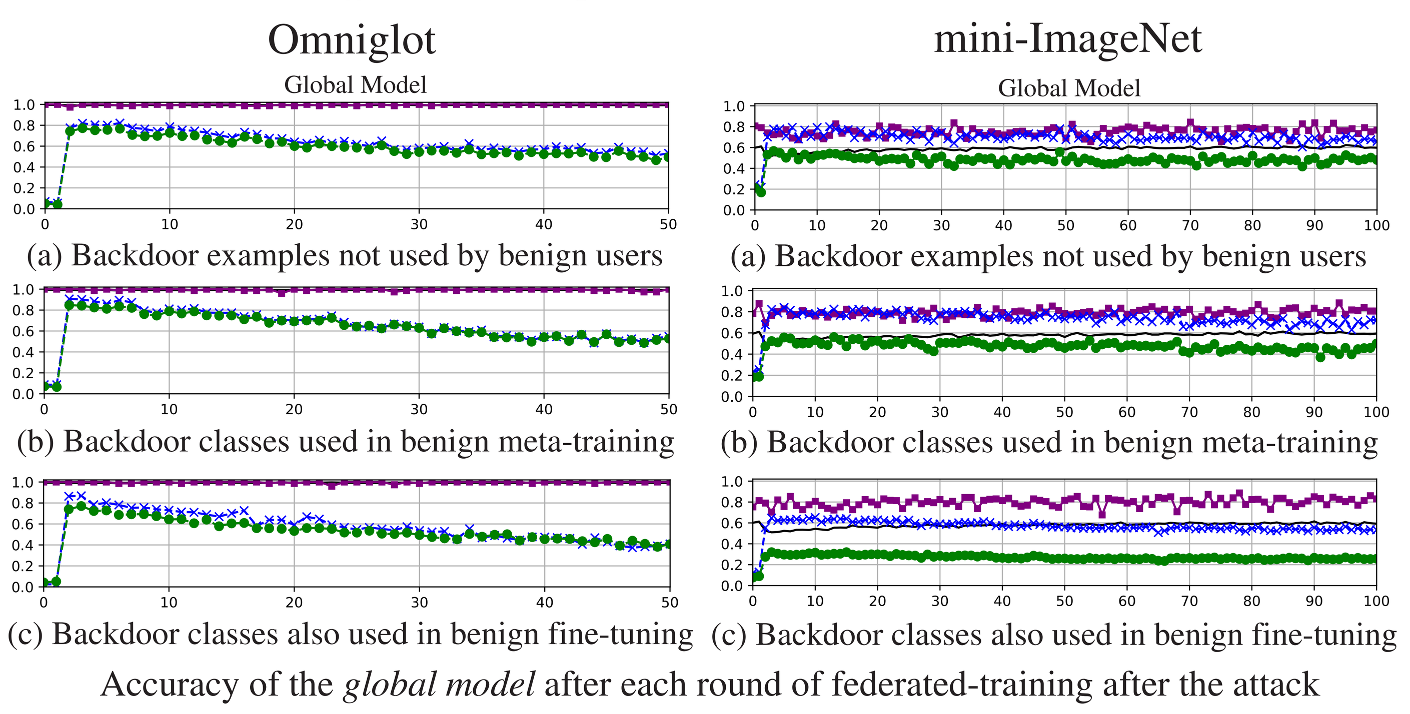
	Omniglot	mini-ImageNet
Backdoor Attack		
Target Class		
Backdoor Classes		
Backdoor Key		
Attack Training Set		

Federated Learning:

- 1 server and 4 clients (1 malicious adversary (Client 1); 3 benign clients (Clients 2, 3, 4))
- The server updates the global model when it receives 3 updates from clients

V. Experimental Results: Backdoor Attacks

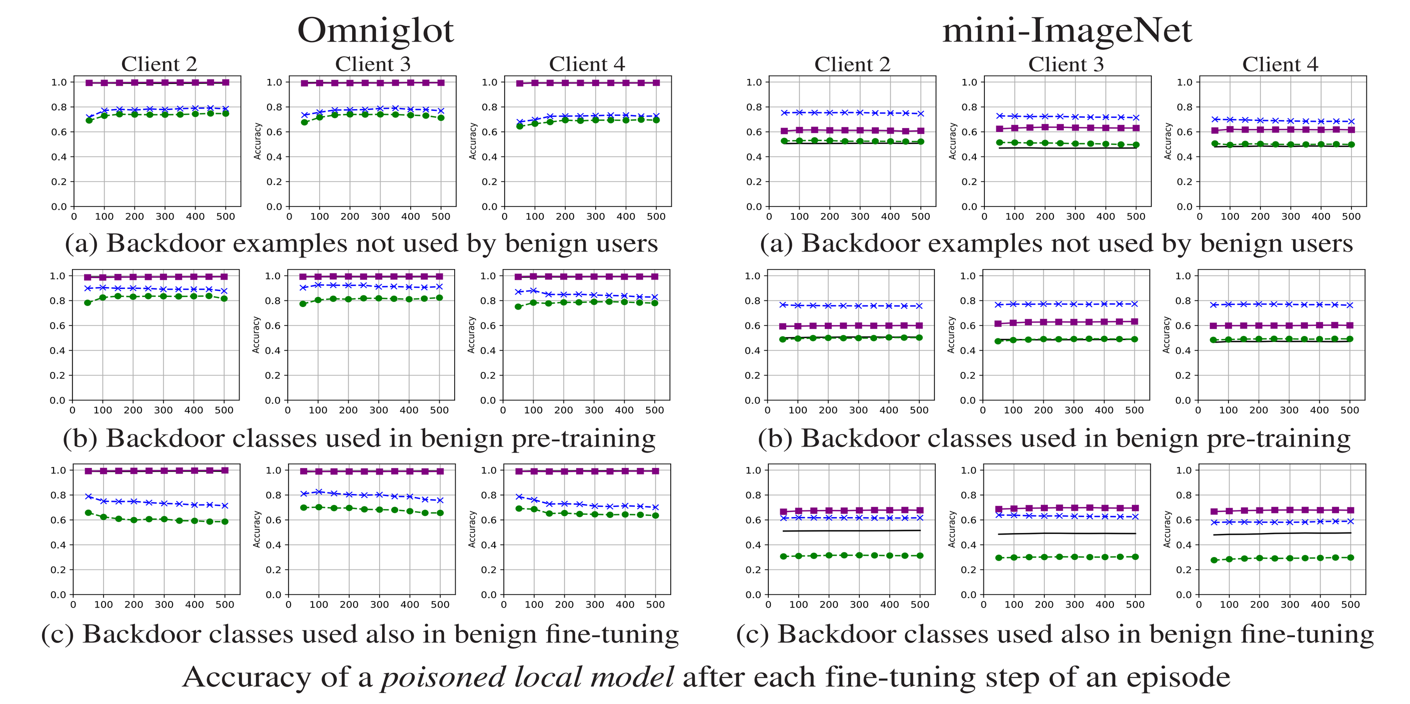
(*Federated meta-training*) How quickly would updates from benign users dampen the effects of a one-shot poisoning backdoor attack?



Accuracy of the *global model* after each round of federated-training after the attack

- Key Results
  - Default federated meta-training dampens backdoor effects slowly; backdoor effects persist tens to hundreds of rounds
  - Backdoor attacks are more successful on the attack training set
  - When benign examples of backdoor classes are used for fine-tuning (Case (c)), backdoor attacks are less successful

(*Meta-testing*) During fine-tuning, would meta-learning's adaptability help remove backdoors from a poisoned model?



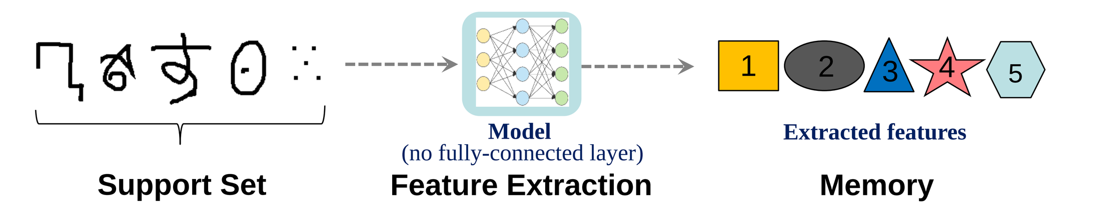
Accuracy of a *poisoned local model* after each fine-tuning step of an episode

- Key Results
  - When benign examples of backdoor classes are not used for fine-tuning (Cases (a) and (b)), fine-tuning doesn't dampen backdoor effects (even after 500 iterations)
  - When benign examples of backdoor classes are used for fine-tuning (Case (c)), backdoors are not removed effectively

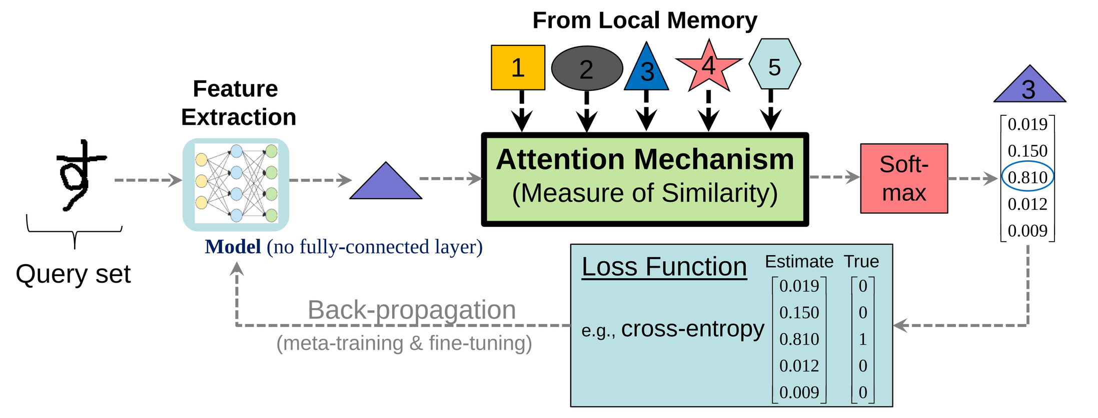
VI. Defense Mechanism Performed Locally

Matching Networks:

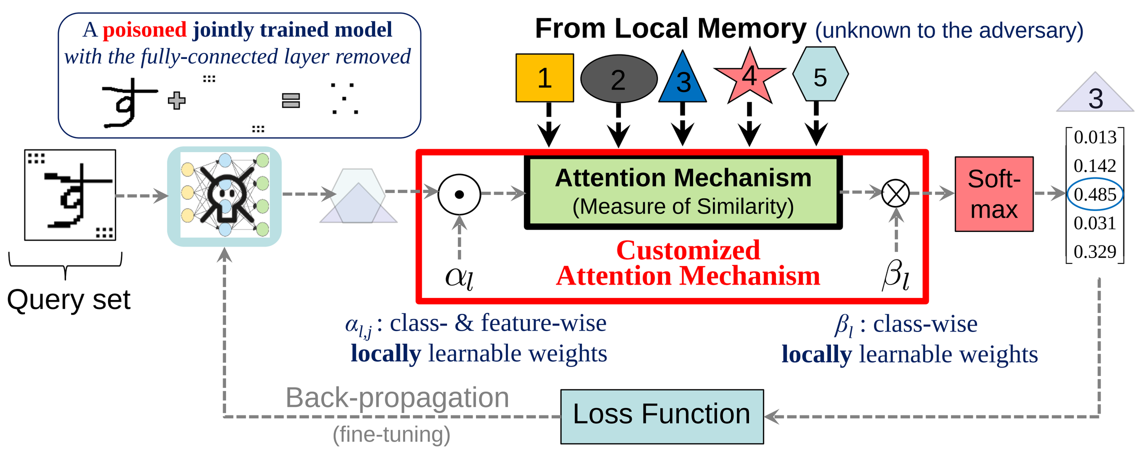
- Step 1: Each user saves features of a local *support set* of training examples in memory



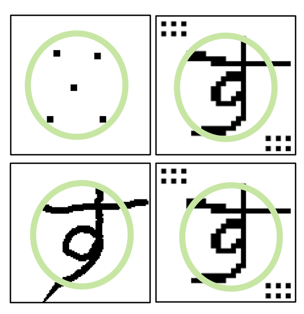
- Step 2: Features of a query input are matched against support set features in memory



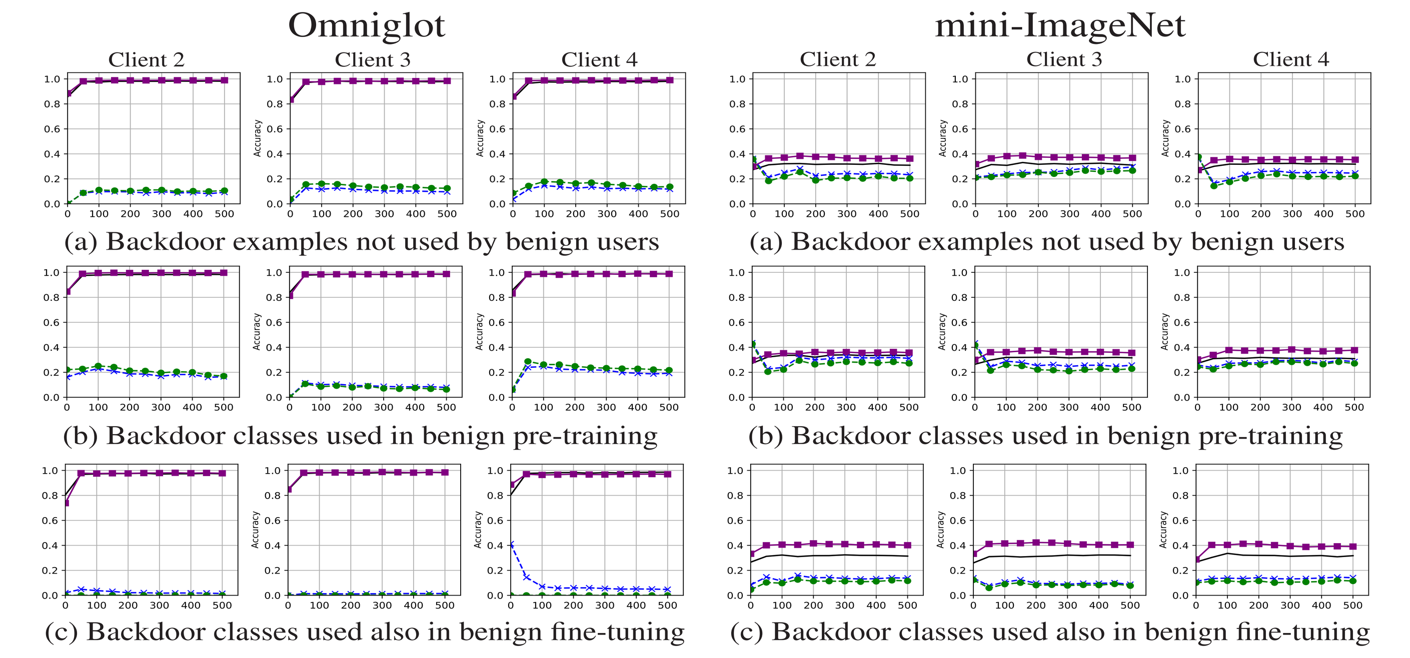
Defense Mechanism: Matching Networks with Customized Attention Mechanism



- Tradeoffs:
  - Matching of the extracted features should remove backdoor effects
  - Not using a shared fully-connected layer could hurt prediction accuracy



VII. Experimental Results: Defense Against Backdoors



Accuracy of a *poisoned local model* after each fine-tuning step of an episode

- Key Results
  - The defense mechanism *effectively* and *efficiently* removes backdoors
  - We are **first** to demonstrate feasibility of this defense *without a centralized approach* (that could leak users' private data)
  - Future work: enhance model performance for complex datasets (e.g., mini-ImageNet)

VIII. Acknowledgements

This material is based upon work supported by the National Science Foundation under grants number CNS-1816887 and CCF-1763747.