pFedDef: Grey-box Defense for Personalized Federated Learning

Taejin Kim, Nikhil Madaan, Shubhranshu Singh, Carlee Joe-Wong
Carnegie Mellon University, Contact: tkim2@andrew.cmu.edu

Background: Evasion Attacks and Federated Learning

Adversarial Evasion Attack
• An adversarial example is an altered input to a neural network with perturbations undetectable to a human, but causes misclassification to a neural network [1]
• Often, gradient information is used to perturb the input, leading to either a targeted attack to a specific label, or an untargeted attack to any label
• Success measured by misclassification rate

Federated Learning
• Federated learning is a machine learning technique that trains a single model across multiple devices holding local data samples while maintaining data privacy [2]
• Personalized federated learning utilizes a similar training procedure to train slightly different models at each client that fit local data better, this paper uses FedEM algorithm for personalization [3]

Problem Statement: Internal Grey-Box Evasion Attacks

<table>
<thead>
<tr>
<th>Data set</th>
<th>Method</th>
<th>Acc.</th>
<th>Adv. Acc.</th>
<th>Target Hit</th>
</tr>
</thead>
<tbody>
<tr>
<td>(CIFAR)</td>
<td>Local</td>
<td>0.52</td>
<td>0.38</td>
<td>0.06</td>
</tr>
<tr>
<td></td>
<td>FedEM</td>
<td>0.64</td>
<td>0.10</td>
<td>0.46</td>
</tr>
<tr>
<td></td>
<td>FedAvg</td>
<td>0.81</td>
<td>0.00</td>
<td>0.85</td>
</tr>
<tr>
<td>(Celeba)</td>
<td>Local</td>
<td>0.57</td>
<td>0.19</td>
<td>0.48</td>
</tr>
<tr>
<td></td>
<td>FedEM</td>
<td>0.85</td>
<td>0.13</td>
<td>0.52</td>
</tr>
<tr>
<td></td>
<td>FedAvg</td>
<td>0.80</td>
<td>0.01</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Table 1. Test accuracy, accuracy against untargeted attacks (Adv. Acc.), and success of targeted attacks (Target Hit) for 40 clients given different training algorithms: local learning, federated learning, and personalized federated learning for CIFAR-10 and Celeba.

Grey-Box Attacks
• Clients have full ( federated learning, white-box attack) information of models at other clients that can be used to create adversarial examples with higher success attack rate [1]
• E.g., spam filter developed through federated learning, malicious clients have knowledge to bypass spam filter of other clients
• Our problem scenario is different from poisoning attacks that compromise models during training phase [4]

Solution: pFedDef Algorithm and Robustness Propagation

Algorithm 1 pFedDef Training
1: Input: Adv. Prop. \( G \), Dataset Update Freq. \( Q \), 
    PGD steps \( K \), Client resource \( R_c \)
2: for \( i \) \in \{1, \ldots, R \} do
3: if \( \% Q = 0 \) then
4: \( F \leftarrow \text{adv-prod}(G, R_c) \)
5: for \( c \in \{C\} \) do
6: update adv. dataset(c, \( K, F_c \))
7: end for
8: end if
9: federated_adversarial_training()
10: end for

pFedDef: Personalized Federated Defense
• Each client \( c \in \{C\} \) sets a local adversarial proportion \( F_c \) for the local data set that will be turned into adversarial data points, while staying within resource constraints \( F_c \leq R_c \) (line 4)
• Clients perform adversarial training over adversarial data set at local clients and perform personalized federated learning aggregation (Line 9)
• Adversarial training was originally proposed for increasing robustness in a single model [1]

Empirical Evaluation of pFedDef

Test Parameters
• 40-80 clients, 200 rounds
• Random resource availability at each client
• L2 Norm attacks with perturbation budget \( \epsilon = 4.5 \)

Legend
• Triangles – adv. trained model
• Circles – non-adv. models
• Solid –grey-box attacks
• Hollow –external attacks

Selected References