Better Together: Attaining the Triad of Byzantine-robust Federated Learning via Local Update Amplification

Liyue Shen, Yanjun Zhang, Jingwei Wang, Guangdong Bai

The University of Queensland, Deakin University
Federated Learning

Local Data -> Local Model -> Global Model
Poisoning attacks in Federated Learning

Untargeted attack

Targeted attack

An example: The decision boundary of the classifier is significantly impacted if just one training sample is changed [1]

Byzantine-robust Methods against Poisoning Attacks

Typical AGR strategies:
- Distance Based [1, 2]
- Prediction Based [3]
- Trust Bootstrapping Based [4]

Triad of Byzantine-robust Federated Learning [1]

Robustness
The method shall minimize the decrease of the global model's test accuracy caused by malicious updates.

Efficiency
The method shall retain the computation efficiency and scale in large-scale FL training.

Fidelity
The method shall not harm the performance of the global model when there is no malicious updates.

Better Together: Attaining the Triad of Byzantine-robust Federated Learning via **Local Update Amplification**

**AgrAmplifier**

- Collect gradients $G$ from participants
- Re-organize gradients into patches
- Select the maximum value for each patch of the feature map
- Produce amplified gradients $G_{amp}$
Achieving Robustness

Since the most activated features are extracted, the local updates become more distinguishable after the amplification.

(a) Untargeted Attack  (b) Targeted Attack

Figure: Gradients projected to 2-D surface using PCA. We plot 50 local updates at the 70th epoch of the training on the LOCATION30 dataset. Red dots (16/50) are malicious updates and green ones are benign. The untargeted attack is implemented via label flipping, and the targeted attack is the Scaling attack proposed by Cao et al.
Achieving Fidelity and Efficiency

Fidelity
AgrAmplifier provides invariance to distortion. Similar to a max pooling layer in typical CNNs, it can ignore the small changes. This makes AgrAmplifier a noise canceller when no attack is present.

Efficiency
The significant dimension reduction in the feature space can greatly benefit the efficiency of AGR.
Equip AgrAmplifier

Universally compatible with existing AGRs regardless of the underlying aggregation rules.
Evaluate AgrAmplifier

- **Datasets**
  - CIFAR-10, MNIST, Location30, Purchase100 (non-iid), Texas100 (non-iid) [1]

- **Evaluated Attacks**
  - Data poisoning via label-flipping (L-flip)
  - Model poisoning via gradient manipulation (G-asc)
  - \(<\text{L-flip}>+<\text{G-asc}\>
  - Optimized and adaptive attack (S&H Attack) [2]
  - Scaling attack, Targeted (T-Scal) [3]

- **Evaluated Defence**
  - Distance-based: Cosine similarity / Euclidean distance / Merged distance combined with the Density measurement
  - Prediction-based: Fang defence [4]
  - Trust Bootstrapping-based: FLTrust [3]

Robustness Performance against Untargeted Attack

\[ L = [\text{Non-attacked Test Accuracy}] - [\text{Attacked Test Accuracy}] \]

(a) Distance-based  (b) Prediction-based  (c) Trust bootstrapping-based
Robustness Performance against Targeted Attack

\[ A = \text{[attacked samples predicted as the attacker wish]} - \text{[other predictions]} \]
Fidelity Performance

![Bar chart showing the averaged performance loss $\mathcal{L}$ for different models: CosDen_Amp+, CosDen, Fang_Amp+, Fang, EuDen_Amp+, EuDen, MgDen_Amp+, MgDen, FLTrust_Amp+, FLTrust. The x-axis represents the averaged performance loss, ranging from 0.00 to 1.75. The y-axis lists the model names. The red bars represent the Base Aggregator, and the blue bars represent the AgrAmplifier.](image)
Efficiency Performance

Time consumption (in second) of aggregation

<table>
<thead>
<tr>
<th>Hidden Layer</th>
<th>C</th>
<th>C+</th>
<th>F</th>
<th>F+</th>
<th>T</th>
<th>T+</th>
</tr>
</thead>
<tbody>
<tr>
<td>512</td>
<td>3110.7</td>
<td>3038.6</td>
<td>3916.8</td>
<td>3822.6</td>
<td>3334.1</td>
<td>3283.6</td>
</tr>
<tr>
<td>1024</td>
<td>6722.9</td>
<td>6479.4</td>
<td>8612.4</td>
<td>7588.0</td>
<td>6672.6</td>
<td>6339.5</td>
</tr>
<tr>
<td>2048</td>
<td>12984.6</td>
<td>10367.1</td>
<td>17039.5</td>
<td>15883.0</td>
<td>13610.2</td>
<td>11084.4</td>
</tr>
</tbody>
</table>

† In this table, we use C to stand for CosDen, C+ for CosDen_Amp+, F for Fang, F+ for Fang_Amp+, T for FLTrust, T+ for FLTrust_Amp+

The difference is more obvious when the neural network architecture is larger, which benefits Byzantine-robust mechanisms towards learning on large/deep models.
Conclusion

- AgrAmplifier benefits Byzantine-robust mechanisms on all three properties
- We recommend COSDEN_Amp+ (distance based) as it achieves a desirable trade-off among the three. It performs the best in term of fidelity, and the joint best in robustness and efficiency
Thanks!
Does anyone have any questions?
liyue.shen@uqconnect.edu.au
UQ-TrustLab

CREDITS: This presentation template was created by Slidesgo, including icons by Flaticon, and infographics & images by Freepik