Secure Shapley Value for Cross-Silo Federated Learning

Shuyuan Zheng (Kyoto Univ.),
Yang Cao (Hokkaido Univ.),
Masatoshi Yoshikawa (Osaka Seikei Univ.)
Overview

• **Background:**
  • 1. Cross-silo FL solves the data silo problem.
  • 2. Contribution evaluation is important to cross-silo FL.

• **Motivation:**
  • 1. SV is a celebrated contribution metric widely adopted in collaborative ML
  • 2. Existing FL systems cannot support secure SV calculation

• **Challenges:**
  • 1. Need to additionally protect test data than secure federated training
  • 2. NP-hard to compute SVs
    • Existing estimation methods work poorly in cross-silo FL because no. of clients is small

• **Our proposal:** to facilitate secure SV calculation for secure contribution evaluation
Data Silo Problem

• Data are decentralized across organizations (e.g., banks and hospitals) as silos and **hardly shared** due to some reasons.
  • E.g., privacy concerns, strict data regulations, data as assets

• Data silos prevent organizations from obtaining accurate machine learning (ML) models to improve products and services.
  • Large amounts of training data required for modern neural networks.
Cross-silo federated learning

• Traditional collaborative ML: **uploading local datasets** for training.
• Cross-silo FL: **uploading local models** for training

\[ \theta \leftarrow \text{Train}(D_{1}^{\text{train}}, \ldots, D_{n}^{\text{train}}) \]

\[ \theta^{t+1} \leftarrow \text{Aggr}(\theta_{1}^{t}, \ldots, \theta_{n}^{t}) \]

Traditional collaborative ML

Cross-silo federated learning
Contribution evaluation

• Clients' contributions might be diverse.
  • Data silos vary in size, quality, and distribution
  • Different participation levels (e.g., number of training rounds)
  • Free-riding or malicious clients exist

• Shapley value (SV) [CTG53] for contribution evaluation
  • Widely adopted in collaborative ML
    • E.g., model rewards [ICML20], monetary rewards [NIPS22], client selection [AAAI21]
  • Measures the expected model accuracy improvement by each client
  • Privacy risk: SV calculation requires access to local models and test data.

[CTG53] LS Shapley. "A value for n-person games." Contributions to the Theory of Games, pages 307-317, 1953.
[ICML20] Sim et al. "Collaborative Machine Learning with Incentive-Aware Model Rewards." ICML 2020.
[NIPS22] Nguyen et al., "Trade-off between payoff and model rewards in Shapley-fair collaborative machine learning." NIPS 2022.
[AAAI21] Nagalapatti et al. "Game of gradients: Mitigating irrelevant clients in federated learning." AAAI 2021.
Secure federated training

- [TIFS18]: using **homomorphic encryption (HE)** to make federated training secure.
  - HE: supports arithmetic operations on encrypted data.
  - **Encrypted local models** are uploaded for model aggregation.

[TIFS18] Phong et al. “Privacy-preserving deep learning via additively homomorphic encryption.” TIFS, 13(5):1333-1345, 2018.
Secure Shapley value

• For SV calculation, no secure systems proposed
• Our proposal: secure SV calculation for secure contribution evaluation
  • Follows [TIFS18] to train models using FL + HE.
  • More challenging than [TIFS18]: test data should be protected additionally.

[TIFS18] Phong et al. "Privacy-preserving deep learning via additively homomorphic encryption." TIFS, 13(5):1333-1345, 2018.
Problem formulation

• Assumptions:
  • All the parties are honest-but-curious.
  • Test data $D_i$ and model parameters $\theta_i^t$ are private.
  • The model structure is public.
  • Focus on neural networks and classification tasks.

• Goal: the server can compute SVs $\phi_1^t, \ldots, \phi_n^t$, while no party can learn other parties’ private information.
  • $\phi_i^t = \mathbb{E}_{S \subseteq \{1, \ldots, n\} \setminus \{i\}} \left[ U\left( \theta_{S \cup \{i\}} \right) - U(\theta_S) \right]$
    • $U(\theta_S)$: accuracy of model $\theta_S$
    • NP-hard to compute: need to test $O(2^n)$ models
Protocol overview

• Baseline: HESV (one-server)
  • Secure model testing: HE for both models and data [IJCAI18]
  • Secure MatMult: Matrix Squaring (extension of SOTA [SIGSAC18])
    • SOTA [SIGSAC18] cannot support large-sized neural networks
  • Problem: multiplications between ciphertexts are inefficient

• Advanced: SecSV (two-server)
  • Secure model testing: HE for models, secret sharing for data
  • Secure MatMult: Matrix Reducing (more efficient than Matrix Squaring)
  • SV estimation: SampleSkip

[IJCAI18] Gelu-net: A globally encrypted, locally unencrypted deep neural network for privacy-preserved learning
[SIGSAC18] Secure outsourced matrix computation and application to neural networks.
HESV

• Secure model testing scheme: HE for both models and data [IJCAI18]
  • Linear layers (i.e., matrix multiplications) evaluated under HE
  • Nonlinear activations (e.g., softmax) evaluated in plaintext
    • HE cannot support nonlinear operations

• Problem: **multiplications between ciphertexts are inefficient**
Hybrid model testing scheme for SecSV

- Secure model testing scheme: **HE for models, secret sharing for data**
  - High efficiency because multiplications between ciphertexts are avoided
- Assumption: two non-colluding servers
  - Example: two large companies who care about their business reputation.
  - Each evaluates one share of data

\[
\begin{align*}
\hat{Y}^{(i)} & = \text{lin}(\theta^{(i)}, X^{(i)}) \\
\hat{Y}'^{(i)} & = \hat{Y}^{(i)} \oplus \hat{Y}''^{(i)} \\
\end{align*}
\]
Matrix Reducing

- Matrix Reducing: much more efficient than Matrix Squaring (extension of SOTA [SIGSAC18])
  - Homomorpheric rotation (HRot) is computationally-expensive
  - Matrix Squaring: many homomorphic rotations needed
  - Matrix Reducing: no homomorphic rotations needed

| Batch size \( m \) | Matrix Squaring \( m \leq \min\{d_{in}, \lfloor \sqrt{N} \rfloor \} \) | Matrix Reducing \( m \leq \lfloor N/d_{out} \rfloor \) |
|-----------------|-----------------|-----------------|
| Complexity of HMult | \( O(d_{in} \cdot d_{out}/\sqrt{N}) \) | \( O(d_{in}) \) |
| Complexity of HRot | \( O(d_{in}/(d_{out} \mod \sqrt{N})) \) | 0 |

[SIGSAC18] Secure outsourced matrix computation and application to neural networks.
SampleSkip

- **Insight:** A sample correctly predicted by two models also be correctly predicted by their aggregated model.
  - Proven to be true for linear models.
  - Almost to be true for nonlinear models.
- SampleSkip can be combined with other SV estimation methods
  - E.g., Permutation Sampling (PS) [ICML19], Group Testing (GT) [ICML19], Kernel SHAP (KS) [NIPS17]
  - SampleSkip is sample-skipping, while they are model-skipping.

[ICML19] Towards efficient data valuation based on the shapley value.
[NIPS17] A unified approach to interpreting model predictions.
Experiments

- RQ1: How efficient are SecSV and HESV for secure SV calculation?
- A1: SecSV with (without) SampleSkip speeds up HESV by $7.2$-$36.6$ ($4.2$-$21.4$) times.

| Dataset (model) | Method  | Speedup w.r.t. HESV | Error ($\times 10^{-2}$) |
|-----------------|---------|----------------------|--------------------------|
|                 | SampleSkip off/on | SampleSkip off/on |
| AGNEWS (LOGI)   | SecSV   | 4.2 × 7.2×           | 0.10 0.10                |
|                 | SecSV+PS | 4.2 × 7.2×           | 2.00 2.01                |
|                 | SecSV+GT | 3.5 × 5.5×           | 3.41 3.39                |
|                 | SecSV+KS | 5.3 × 8.6×           | 17.63 17.63              |
|                 | SecSV+PS | 21.4 × 36.6×         | 0.09 0.09                |
|                 | SecSV+GT | 8.9 × 10.8×          | 3.40 3.40                |
|                 | SecSV+KS | 27.0 × 44.1×         | 7.67 7.66                |
| BANK (LOGI)     | SecSV   | 21.3 × 36.5×         | 1.25 1.24                |
|                 | SecSV+PS | 7.0 × 25.8×          | 0.09 0.64                |
|                 | SecSV+GT | 6.9 × 25.3×          | 2.69 2.88                |
|                 | SecSV+KS | 9.0 × 27.2×          | 3.58 3.80                |
| MNIST (CNN)     | SecSV   | 7.0 × 25.8×          | 2.09 2.64                |
|                 | SecSV+PS | 6.9 × 25.3×          | 2.69 2.88                |
|                 | SecSV+GT | 9.0 × 27.2×          | 3.58 3.80                |
|                 | SecSV+KS | 15.46 15.65          |                          |
| miRNA-mRNA (RNN)| SecSV   | 5.3 × 11.8×          | 1.70 1.82                |
|                 | SecSV+PS | 5.3 × 11.8×          | 3.03 3.25                |
|                 | SecSV+GT | 5.3 × 11.8×          | 3.67 3.50                |
|                 | SecSV+KS | 7.0 × 14.0×          | 20.77 20.49              |
Experiments

- Q2: How much can SampleSkip accelerate SV calculation?
- A2: **67.05-90.77% of test samples skipped.**
Experiments

• Q3: How many test samples are wrongly skipped by SampleSkip?
• A3: 0.00% for linear models; 0.16%-0.22% for nonlinear models.
Experiments

• Q4: How efficient are Matrix Reducing for secure MatMult?
• A4: Matrix Reducing speeds up Matrix Squaring by **1.69-11.39 times**.

Table 4: Speedup of Matrix Reducing w.r.t. Matrix Squaring in the time per sample spent on HE computations for evaluating $AB$. The shape of matrix $A$ is varied. "Full" means both $A$ and $B$ are encrypted, whilst "Half" means only $A$ is encrypted.
Conclusion

• Contribution: the first study on secure SV calculation in collaborative ML.

• Limitations:
  • 1. SecSV requires noncolluding servers.
  • 2. Protocols tailored for horizontal FL.
    • Clients have different samples with the same attributes.
  • 3. Only neural networks and classification tasks considered.

• Future work:
  • 1. More efficient one-server protocol.
  • 2. Secure SV calculation for vertical FL.
    • Clients have different attributes of the same samples.
  • 3. Consider more types of models and ML tasks.
Thank you for listening.
Welcome to visit our poster in range 71-75!