Transparent Contribution Evaluation for Secure Federated Learning on Blockchain

Shuaicheng Ma
Emory University
Atlanta, US
sma30@emory.edu

Yang Cao
Kyoto University
Kyoto, Japan
yang@i.kyoto-u.ac.jp

Li Xiong
Emory University
Atlanta, US
lxiong@emory.edu

***Abstract*—Federated Learning is a promising machine learning paradigm when multiple parties collaborate to build a high-quality machine learning model. Nonetheless, these parties are only willing to participate when given enough incentives, such as a fair reward based on their contributions. Many studies explored Shapley value based methods to evaluate each party's contribution to the learned model. However, they commonly assume a semi-trusted server to train the model and evaluate the data owners' model contributions, which lacks transparency and may hinder the success of federated learning in practice. In this work, we propose a blockchain-based federated learning framework and a protocol to transparently evaluate each participant's contribution. Our framework protects all parties' privacy in the model building phase and transparently evaluates contributions based on the model updates. The experiment with the handwritten digits dataset demonstrates that the proposed method can effectively evaluate the contributions.**

***Index Terms*—Blockchain, Federated Learning, Contribution Evaluation, Transparency, Privacy**

I. Introduction

Federated learning (FL) [1] is a promising technology that allows multiple data owners to jointly train a model without sharing the raw data. One of the challenges in federated learning is the incentive problem. Data owners may not be willing to participate without proper incentives, e.g., a fair reward based on their contributions to the model. Recently, Shapley value (SV)-based contribution evaluation protocols [2]–[5] are proposed to help fairly allocate the profit among the participants in federated learning.

However, the existing studies on contribution evaluation in federated learning commonly assume a semi-trusted server to train the model and evaluate the data owners' model contributions. For the cross-silo federated learning that we consider, the organizations (e.g., banks) are mutually untrusted and potentially competing with each other. It is challenging to assume such a semi-trusted server. The lack of transparency in contribution evaluation may prevent data owners' collaboration and hinder the success of federated learning in practice.

A potential solution to solve the transparency issue is to resort to blockchain techniques. One may put all intermediate results of the training on a blockchain and implement the contribution evaluation in a smart contract on the blockchain. However, this solution introduces two new challenges: 1) There is a privacy issue if all blockchain data is public to all participants. It has been proven that the intermediate data (e.g., weight updates) in federated learning can leak data owners' private information [6]. 2) To address the above privacy issue, we can adopt the well-studied privacy-preserving federated learning, such as secure aggregation (which masks updates) [7]. However, in that case, the existing Shapley Value based contribution evaluation methods are not compatible since they cannot work on the masked updates (which determine the contributions).

Corresponding author: Yang Cao.

We make the following contributions in this work to address the above challenges.

- We design a blockchain-based secure federated learning framework that adopts secure aggregation to protect data owners' privacy during the training.
- We propose a new group-based SV computation framework that is compatible with secure aggregation.
- We implement the framework on the blockchain, and our experiments on real datasets demonstrate the proposed method can effectively evaluate the contributions.

II. Related Work

A. Contribution Evaluation in Federated/Machine Learning

Ghorbani *et al.* [2] and Jia *et al.* [3] have established that Shapley value (See Sect. IV.A) satisfies several important properties (balance, symmetry, zero elements, and additivity) to measure the value of data points in a dataset [8]. The main focus of their works is to reduce the time complexity of computing SV. Song *et al.* [4] and Wang *et al.* [5] demonstrate that SV method in FL setting has comparable performance to the centralized ML setting. However, they all assume that a semi-trusted server will honestly evaluate the contribution of each participant.

B. Privacy-preserving Federated Learning

There are two lines of works in the area to ensure client-level privacy: 1) Local Differential Privacy (LDP) based approach [9]–[11]: Adding calibrated noises to the gradients before sending to the server. However, the accumulated noises make the model not very useful. It is still an open problem of achieving a good balance between privacy and utility in LDP-based FL. 2) Cryptography-based approach:

Secure Multi-Party Computation (MPC) and Trusted execution environments (TEE) [12]. LDP adds noise to the final model and possibly degenerates a data owner's contribution. The cryptography-based approach can perfectly protect the intermediate data during model building, but it comes with an expensive computation. MPC is known for several decades that any function can be securely computed; however, it is usually not applicable when the number of parties is large, which is not the case in our setting (See Sect. III). TEE requires extra hardware support and additional engineering steps. Hence, we adopt MPC in this paper. Since the native SV method cannot be computed on the masked data, we design an MPC-compatible method to calculate SV. We recognize the final model privacy is an orthogonal problem to our setting and can be addressed by adding distributed DP noise during MPC computation [13].

C. Machine Learning on the blockchain

Chen *et al.* [14] propose a blockchain-based machine learning method using an l-nearest aggregation algorithm to protect the system from potential Byzantine attacks. Nevertheless, they do not address the contribution evaluation problem. Cai *et al.* [15] introduce a framework to record model updates and contributions on the blockchain. However, the evaluation process is done off-chain, so it is unverifiable, and the framework does not have any privacy guarantee. Our on-chain evaluation method is fully transparent, verifiable, and protects data owners' privacy.

III. Problem Formulation

Our target scenario is cross-silo FL, and the dataset is horizontally partitioned. Moreover, we assume all data owners participate in all training rounds. Our protocol is agnostic to blockchain implementation. Any blockchain with Smart contract capability (e.g., Ethereum, Hyperledger) is compatible with our protocol. Smart contract is a transaction protocol that runs in the blockchain to execute program logic. Indeed, in our setting, Smart contract builds the FL model and evaluates the contribution.

In our framework, any data owner can have two roles: FL model trainer and blockchain miner. The latter role constructs a P2P network with the blockchain protocol that conceptually replaces the traditional centralized server in FL. The protocol has two parts: 1) The leader selection protocol periodically selects a leader to propose a set of transactions. 2) A verification protocol requires all other miners to re-execute the proposed transactions. If the re-execution results are the same as the proposed, the miners accept them; otherwise, they wait for another leader to propose. The blockchain protocol guarantees the transactions' truthfulness as long as the majority of miners are honest.

A. Threat/Trust Model

In the FL model-building phases, we assume the data owners are semi-honest. They submit honest local weight updates to the blockchain, but they are curious about others' private information since they most likely are competitors or have competing interests. The study [6] has shown that a curious user can re-build the others' local model if he/she has access to others' local updates, and he/she will be able to retrieve private information from the individual local model. In the contribution evaluation phases, the selected data owner (a.k.a leader) may be fraudulent, and he/she will try to maximize his/her contribution by proposing incorrect evaluation results. However, when the majority of miners are honest, only truthful results are accepted by the blockchain. Our framework can effectively replace the semi-trusted server in the traditional FL setting.

IV. Our Approach

A. Preliminaries

1) Secure aggregation: We adopt Google's work on secure aggregation [7]. Readers may refer to the original work for the security and the privacy guarantee. The secure aggregation is based on discrete logarithm cryptography. Here we use a simple example to illustrate the procedure.

- User A, B, C generate the private key a, b, c and broadcast their public key g^a, g^b, g^c to the blockchain.
- User A, B, C computes the Diffie–Hellman key g^{ab}, g^{bc}, g^{ac} based on others' public key.
- At each round r, each user uses a pseudorandom number generator $PRNG(\cdot)$ to generate a paired masks based on Diffie–Hellman key and submits masked local model updates w_i to the blockchain. For user A: $PRNG(g^{ab}, r) \rightarrow m^r_{ab}$, $PRNG(g^{ac}, r) \rightarrow m^r_{ac}$. It submits $w_A + m^r_{ab} - m^r_{ac}$.
- At each round r, the blockchain aggregates all masked updates and the masks are cancelled out. $w_A + m^r_{ab} - m^r_{ac} + w_B + m^r_{bc} - m^r_{ab} + w_C + m^r_{ac} - m^r_{bc} = w_A + w_B + w_C$

Using secure aggregation, the local updates on the blockchain are masked. The curious user cannot learn others' models.

2) Shapley value: Shapley value, named in honor of Lloyd Shapley, is a solution concept from cooperative game theory [8]. The SV of user i with respect to the utility function $u(\cdot)$, denoting as v_i, is defined as the average of marginal contribution of i to coalition S over all $S \subseteq I \setminus i$. The native method to calculate SV is:

$$v_i = \frac{1}{n} \sum_{S \subseteq I \setminus \{i\}} \frac{1}{\binom{n-1}{|S|}} [u(S \cup \{i\}) - u(S)] \tag{1}$$

where n is the total number of users.

B. Procedure

In secure aggregation, the masking technique randomize each user's updates to protect users' privacy during the training phase. However, this method prevents us from computing SV using native SV calculation (i.e., Eq. 1) since we are not able to calculate the correct utility of excluding a single user (e.g., $u(S \cup \{i\}) - u(S)$). We propose a group-based SV method, GroupSV. First, we partition users I into m groups based on a permutation sample at round r with random seed e (line 1-2), then use secure aggregation to jointly train a group model

Algorithm 1: Group SV: Contribution Evaluation

Input: users $i \in I$, masked user local weights $[w_i]$, random seed e, round number r, utility function $u(\cdot)$, number of groups m.

Output: users contributions $[v_i^r]$, global model W_G

// e.g., I is $[A, B, C, D, E, F, G, H, I]$

// e.g., $\pi = A, E, H, B, F, I, C, G, D$

1 $\pi \leftarrow permutation(e, r, I)$;

// $grouping(\cdot)$ assigns users into groups based π and m.

// G_j denotes the j^{th} group (e.g., $m = 3$, G_1 is $[A, E, H]$).

2 $G = \{G_j | j \in [1, m]\} \leftarrow grouping(\pi, m)$;

3 **for** $j = 1$ ***to*** m **do** $W_j = \frac{1}{||G_j||} \sum_{i \in G_j} w_i$;

// $\mathbb{P}(\cdot)$ is a powerset operation.

4 **for** $S \subseteq \mathbb{P}(G)$ **do** $W_S = \frac{1}{||S||} \sum_{j \in S} W_j$;

5 **for** $j = 1$ ***to*** m **do**

// V_j denotes the SV of j^{th} group.

6 $V_j \leftarrow \sum_{S \subseteq G \setminus \{j\}} \frac{1}{m\binom{m-1}{|S|}} [u(W_{S \cup \{j\}}) - u(W_S)]$;

7 **for** $i \in G_j$ **do** $v_i^r \leftarrow \frac{1}{||G_j||} V_j$;

for each group (line 3). From those group models, we build coalition models W_S of different subsets of groups G using plain aggregation (line 4). Last, we compute each group's SV and assign it to its members (line 5-7). In other words, each data owner's SV is approximated by its group's SV. In following section, we describe the procedure of our protocol.

- At the off-chain setup stage, users reach a consensus on FL parameters (e.g., FL algorithm), secure aggregation parameters (e.g., generator g) , and contribution evaluation parameters (e.g., permutation seed e, group size m, utility function u) and submit them to the blockchain.
- At each round r, users train their models and send masked updates to the blockchain. Then, users download the new global model (an aggregation of all the group model) and start the next round until reaching the maximum number of rounds.
- In the end, all users share an aggregated global model W_G, and the total SV for a user is $v_i = \sum_{r=0}^{R} v_i^r$, where R is the total number of the rounds.

Group SV is configurable with different number of groups, m. When m is the maximum, $m = n$ (where $n = |I|$), all users are assigned to a single group, and their SVs are evaluated independently based on their per round local model, so SVs have the highest resolution. However, the model parameters are revealed. In general, given the number of groups m, the average model parameters for each group of size n/m is revealed, in some sense similar to (n/m)-anonymity. Hence, the larger the m, the less private. When m decreases, one group has multiple users, and their SVs are computed based on the averaged model, so the resolution decreases. When computing SV using the native method, we need to train additional $2^n - 1$ coalition models. For our group model aggregation, we adopt a similar approach from [4]. The coalition models are aggregated from users' local model updates in a FL fashion.

V. Experiments

A. Setup

*1) **Dataset**:* We experiment on the *Optical Recognition of Handwritten Digits* dataset [16], which is normalized bitmaps of handwritten digits. It contains 5620 instances, 64 attributes, and 10 classes. We randomly split the dataset into a training dataset and a testing dataset with a ratio of 8:2 and randomly split the training dataset into 9 subsets to simulate 9 data owners, $D = \{d_i\}_{i=0}^{8}$. To simulate different data quality of each data owner, we add Gaussian noise with an increasing sigma, $d_i = d_i + \mathcal{N}(0, \sigma * i)$. As a result, d_0 has the best data quality, d_1 has worse data quality, and so on. When $\sigma = 0$, all data owners have similar data quality; the higher σ leads to more diverse data quality.

*2) **Environment**:* We conduct experiments on Ubuntu 18.04.4 with Intel(R) i7-6700K CPU @ 4.00GHz and 64GB main memory. The codes are implemented on Python 3.7 with NumPy. We use logistic regression with gradient descent in local train epoch and FedAvg [1] in global train epoch. The contribution evaluation phase occurs in a simulated blockchain.

B. Results

We next analyze the three experimental results.

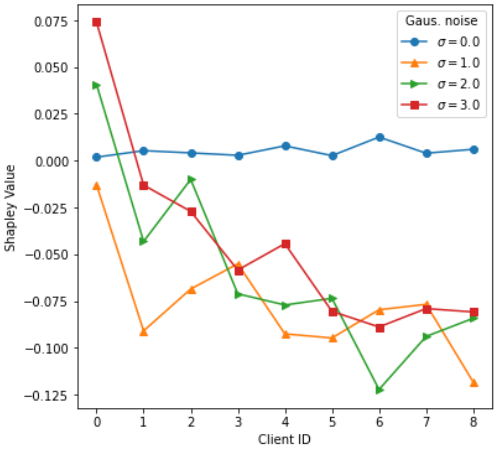

Fig. 1. Ground Truth SV distribution over users w.r.t. different σ.

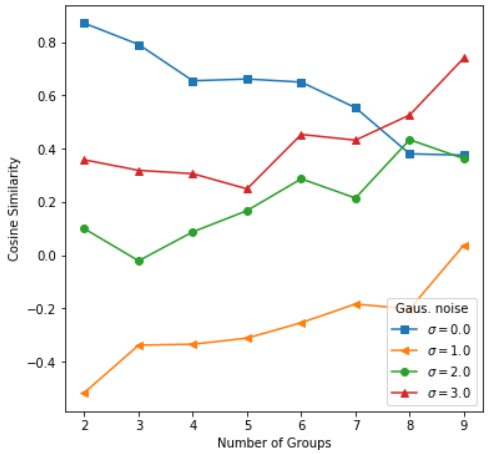

Fig. 2. The approximation accuracy between Group SV method and Native SV method with different Gaussian noise.

TABLE I
TIME COMPARISON BETWEEN GROUPSV AND NATIVESV

	GroupSV								NativeSV
# of Groups	2	3	4	5	6	7	8	9	9
Time/s	2	3	4	7	11	20	39	77	316

*1) **Ground Truth**:* First, we build 2^n models based on the data coalitions, $\{M_S | S \subseteq \mathbb{P}(I)\}$, then establish the ground truth SV using the native SV method (Eq. 1). We emphasize that native SV cannot be computed with privacy protection on the blockchain.

Fig.1 shows users' SVs with respect to different Gaussian noise. When there is no noise added ($\sigma = 0$), all users have a close to zero SV. We split the dataset uniformly at random, so the subsets have similar distributions, and the marginal contributions towards the final model are negligible. As a result, they contribute almost equally to the final model. When $\sigma \neq 0$, the good (less noisy) quality dataset has a higher SV than the bad quality dataset. As we expected, SV can distinguish different data owners' contribution according to their quality.

*2) **Accuracy Comparison**:* We use cosine similarity, i.e., $similarity = \cos\theta = \frac{\vec{u}\cdot\vec{v}}{|\vec{u}||\vec{v}|}$, to measure the distance between the Shapley values calculated by our group SV and the ground truth SV. Fig. 2 shows the change of cosine similarity with respect to the number of groups. We can see that, when $\sigma = 0$ (i.e., all data owners have similar data quality), the similarity is decreasing along with the number of groups. The reason is that, when $\sigma = 0$, the ground truth SV value distribution is close to uniform over the participants. Note that, our group-based SV calculation allocates contribution uniformly to each participant within the group. Hence, when the group size is large (i.e., the number of groups is small), most users have the similar SV, so the cosine similarity in this case is high. On the other hand, we observe that, when $\sigma \neq 0$, the similarity increases with the number of groups. It is because, when the number of groups is large, our group based method is closer to the native one. For the same reason, the cosine similarity increases with respect to increasing σ (i.e., more diverse data quality).

*3) **Runtime Comparison**:* Tab. I shows the time performance of our group SV and the native SV method. Since coalition models are aggregated directly from users' local updates, the number of models required to train to compute SV reduces from 2^n to n. Consequently, we see an order of magnitude improvement in time in our group SV method.

VI. CONCLUSION & FUTURE WORK

This paper addresses transparent contribution evaluation challenges for different data owners in a cross-silo horizontal FL setting. We propose a blockchain-based method to measure data owners' SV-based contributions with configurable resolution without sacrificing their privacy. Our experiments on the handwriting digits dataset demonstrate that our method can effectively evaluate contributions.

Although we proposed core algorithms and demonstrated their effectiveness, several important future works are needed to complete this study. First, we will investigate the proposed method's applicability to the existing blockchain platforms (such as Ethereum or Hyperledger Fabric). We will pinpoint the potential bottlenecks (such as transaction throughput) of implementing secure federated learning with the blockchain. Second, we will study the effects of adversarial participants on the Shapley value calculation since the proposed group-based SV method may be influenced by the number of groups and the participants' adversarial behavior. Third, we will thoroughly examine the trade-offs between privacy, transparency, and security in the proposed system.

ACKNOWLEDGMENT

This work is supported by National Science Foundation (NSF) CNS-1952192, National Institutes of Health (NIH) R01GM118609, JSPS KAKENHI Grant No. 19K20269, CCF-Tencent Open Fund WeBank Special Fund.

REFERENCES

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*, PMLR, 2017.

[2] A. Ghorbani and J. Zou, "Data shapley: Equitable valuation of data for machine learning," *arXiv preprint arXiv:1904.02868*, 2019.

[3] R. Jia, D. Dao, B. Wang, F. A. Hubis, N. Hynes, N. M. Gürel, B. Li, C. Zhang, D. Song, and C. J. Spanos, "Towards efficient data valuation based on the shapley value," in *Proceedings of Machine Learning Research* (K. Chaudhuri and M. Sugiyama, eds.), vol. 89 of *Proceedings of Machine Learning Research*, pp. 1167–1176, PMLR, 16–18 Apr 2019.

[4] T. Song, Y. Tong, and S. Wei, "Profit allocation for federated learning," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 2577–2586, 2019.

[5] T. Wang, J. Rausch, C. Zhang, R. Jia, and D. Song, "A principled approach to data valuation for federated learning," 2020.

[6] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Advances in Neural Information Processing Systems* (H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), vol. 32, pp. 14774–14784, Curran Associates, Inc., 2019.

[7] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, (New York, NY, USA), 2017.

[8] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, vol. 2, no. 28, pp. 307–317, 1953.

[9] N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, and G. Yu, "Collecting and analyzing multidimensional data with local differential privacy," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pp. 638–649, IEEE, 2019.

[10] R. Liu, Y. Cao, M. Yoshikawa, and H. Chen, "Fedsel: Federated sgd under local differential privacy with top-k dimension selection," in *International Conference on Database Systems for Advanced Applications*, pp. 485–501, Springer, 2020.

[11] R. Liu, Y. Cao, H. Chen, R. Guo, and M. Yoshikawa, "Flame: Differentially private federated learning in the shuffle model," *AAAI*, 2020.

[12] V. Costan and S. Devadas, "Intel sgx explained." Cryptology ePrint Archive, Report 2016/086, 2016. https://eprint.iacr.org/2016/086.

[13] S. Goryczka and L. Xiong, "A comprehensive comparison of multiparty secure additions with differential privacy," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 5, pp. 463–477, 2017.

[14] X. Chen, J. Ji, C. Luo, W. Liao, and P. Li, "When machine learning meets blockchain: A decentralized, privacy-preserving and secure design," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018.

[15] H. Cai, D. Rueckert, and J. Passerat-Palmbach, "2cp: Decentralized protocols to transparently evaluate contributivity in blockchain federated learning environments," in *2020 IEEE International Conference on Blockchain*, 2020.

[16] D. Dua and C. Graff, "UCI machine learning repository," 2017.