KIPS Trans. Comp. and Comm. Sys.
Vol.9, No.12 pp.321~332
ISSN: 2287-5891 (Print), ISSN 2734-049X (Online)

https://doi.org/10.3745/KTCCS.2020.9.12.321

Practical Concerns in Enforcing Ethereum Smart Contracts as a Rewarding Platform in Decentralized Learning

Sandi Rahmadika† · Muhammad Firdaus† · Seolah Jang†† · Kyung-Hyune Rhee†††

ABSTRACT

Decentralized approaches are extensively researched by academia and industry in order to cover up the flaws of existing systems in terms of data privacy. Blockchain and decentralized learning are prominent representatives of a deconcentrated approach. Blockchain is secure by design since the data record is irrevocable, tamper-resistant, consensus-based decision making, and inexpensive of overall transactions. On the other hand, decentralized learning empowers a number of devices collectively in improving a deep learning model without exposing the dataset publicly. To motivate participants to use their resources in building models, a decent and proportional incentive system is a necessity. A centralized incentive mechanism is likely inconvenient to be adopted in decentralized learning since it relies on the middleman that still suffers from bottleneck issues. Therefore, we design an incentive model for decentralized learning applications by leveraging the Ethereum smart contract. The simulation results satisfy the design goals. We also outline the concerns in implementing the presented scheme for sensitive data regarding privacy and data leakage.

Keywords : Blockchain, Data Privacy, Decentralized Learning, Incentive Mechanism, Smart Contract

연합학습의 인센티브 플랫폼으로써 이더리움 스마트 컨트랙트를 시행하는 경우의 실무적 고려사항

Sandi Rahmadika† · Muhammad Firdaus† · 장 설 아†† · 이 경 현†††

요 약

탈중앙화 접근법은 기존 시스템의 데이터 프라이버시 결함을 보완하기 위해 산·학계에서 폭넓게 연구되고 있다. 블록체인은 기록된 데이터는 위조할 수 없으며 합의를 기반으로 의사결정을 이루고 전반적인 거래의 비용은 저렴한 특징을 가지고 있다. 연합학습은 데이터 집합을 공개적으로 노출하지 않고 다수의 장치를 집합적으로 사용 함으로서 딥러닝 모델을 개선할 수 있게 한다. 모델 구축을 위해서는 자원을 사용하도록 참여자들의 동기 부여를 위한 적절하고 참여 비율에 합당한 인센티브 제도가 필수적이다. 그러나 중앙집중화된 인센티브 메커니즘은 중간 계층에 의존하고 여전히 병목현상을 유발하기 때문에 연합학습에 적용하기에는 어려움이 있다. 따라서, 우리는 이더리움 스마트컨트랙트를 활용하여 연합학습 어플리케이션을 위한 인센티브 모델을 제안한다. 구현 결과는 설계 목표를 충족하였고, 마지막 절에서 연합학습에서 프라이버시 및 데이터 유출과 관련된 민감 데이터에 대한 본 구현을 실행할 때 발생할 수 있는 사항들을 설명한다.

키워드 : 블록체인, 데이터 프라이버시, 연합학습, 인센티브 모델, 스마트 컨트랙트

1. Introduction

Decentralized learning is a popular machine learning mechanism that enables a new paradigm where a shared global model from participating devices (Clients) is trained in a decentralized approach that is harmonized by a synapsis server. The concept of decentralized learning refers to federated learning [1] introduced by the Google AI Team with to form a distributed training data across a large number of client devices in order to update a global model collaboratively while protecting the data privacy of the clients.

In contrast to traditional centralized learning where user data is aggregated and processed centrally in

※ This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2018R1D1A1B07048944) and partially was supported by Pukyong National University Research Fund in 2019.
※ 이 논문은 2020년 한국정보처리학회 춘계학술발표대회에서 "Merging Collaborative Learning and Blockchain: Privacy in Context"의 제목으로 발표된 논문을 확장한 것임.
† 준 회 원 : 부경대학교 인공지능융합학과 박사과정
†† 준 회 원 : 부경대학교 인공지능융합학과 석사과정
††† 종신회원 : 부경대학교 IT융합응용공학과 교수
Manuscript Received : August 4, 2020
Accepted : September 6, 2020
* Corresponding Author : Kyung-Hyune Rhee(khrhee@pknu.ac.kr)

the data center, decentralized learning allows all clients involved to train their data locally without revealing the data used to optimize the model [2]. In order to prevent data leakage, their data are only used to compute an update to the current global model maintained by the model provider. The model provider only provides an initial model that will be used by clients using their private data (Locally trained). The cycle continuous in multiple rounds until achieving the desired parameter.

The usage of a decentralized approach is significantly expanding along with the increasing concern of the drawbacks in a centralized system. Google's Gboard [3] is one of the successful applications that implement the advantages of this approach to advance the model of next-word prediction. In addition, several research efforts to improve security [4] of decentralized learning in various fields with restricted data such as financial and healthcare [5]. Comprehensive research is continuing on decentralized learning field. However, several problems threaten the security risk from dishonest clients to disrupt global model updates by submitting incorrect information. This sort of behavior may jeopardize the entire decentralized training scheme. Thus, it is vital to provide an adequate incentive mechanism to motivate the clients to behave honestly by maintaining fairness during the collaboration training process [6], so that more clients can actively contribute to improving the global models.

The conventional incentive mechanism, such as the client's reputation-based [7], and payment-based incentive scheme [8, 9], generally use a centralized trusted authority to prevent fraudulent client behaviors, but they failed to offer fairness settlement [10]. Blockchain through smart contracts can be a solution to accommodate a proper incentive mechanism in decentralized learning. Blockchain is secure by design since the data record is irrevocable, tamper-resistant, consensus-based decision making, and inexpensive of overall transactions. In order to support the implementation of blockchain, smart contracts are utilized as a self-executing contract that can facilitate transparent, irreversible, and traceable transactions [11], which are then stored in distributed ledger blockchain. It can improve security and efficiency communication among clients [12] in an immutable manner. Hence, blockchain and smart contracts preserve reliable transactions for the users [13], while also provide the efficiency of a decentralized learning system.

The main contributions are summarized as follows:

1) We design an incentive model for decentralized learning applications by leveraging the Ethereum smart contract. The objective of this scheme is to empower a number of devices collectively in improving a deep learning model without exposing the dataset publicly. A decent and proportional incentive system can motivate clients to jointly contribute to the decentralized training process by using their private resources.
2) We implement a decentralized learning prototype and evaluate the performance of the decentralized training and smart contracts based on the result of the simulation.
3) We analyze the practical concerns in implementing our scheme especially for sensitive data regarding privacy and data leakage in decentralized learning activities.

2. Problem Statement

2.1 Conventional Training Models

In the traditional training approach, a centralized cloud-server able to train and process the model using the data that are uploaded by the client's devices [15]. These methods involve a simple data transaction, where the clients able to collect and upload their data to the cloud-based server. Then, the server aggregates the uploaded data to obtain useful information for future events such as detection, classification, and prediction [16]. Furthermore, the centralized cloud server has visibility into the training data that are generated from multiple client devices.

The centralized cloud server model allows several benefits for the clients since it does not overburden their device. In this sense, clients only send the data regularly to the centralized server, then the cloud server conducts several tasks for the training process. The cloud server creates the log right after the clients uploading their data. The data from the clients are used to create a dashboard model. Later, the clients download the model and start to collect the data to improve the model in the cloud server. Then, the cloud-server aggregates and updates the gradients of

the model for further use.

Deep learning improves the accuracy of the large-scale database by providing an end-to-end learning system that allows classifiers and features are learned in parallel [17, 18]. The ability of large deep neural networks (DNN) increases the capability of deep learning in advance [19]. In order to train the model on-device deep learning application, either academia or industry provide two solutions [20]: (1) compress the size of large DNNs, and (2) subdivide the large DNNs across the client machines and cloud-based. The first solution aims to abbreviate the size of DNNs become suitable to train on client devices. On the other hand, the second solution allows the client devices to conduct the little portions of the DNN training model while large and complex parts of DNN are training on the cloud-based server. The lack of a client's device issues can be conquered by the cloud-based server solution, particularly on the training model.

Despite the prosperity of the cloud-based server solution in a centralized framework, it also causes several critical challenges consists of a server security risk and overhead issues in collecting and storing the training data. During the training process, the server gathers the data clients that lead to the vulnerable phase of the network model from adversaries. Once the adversaries snatch the server, the client's sensitive data may reveal, such as medical health records, financial, biometric data, racial or ethnic origin, and so on. Hence, the client's privacy issue becomes a crucial problem since the clients share their data without controlling the learning objective.

A malicious cloud-based server provider may leak the client data by using a neuron that is deliberately released to engage in training model updates. Therefore, collaborative model training with a decentralized framework offers a potential solution to address the aforementioned issues.

2.2 Incentive Mechanism with Centralized Frameworks

The incentive mechanism aims to motivate clients to contribute and share their data honestly to improve the system in general. The existing scheme of incentive mechanism mainly includes the client's reputation-based and payment-based still rely on a centralized framework to prevent fraudulent client behaviors. The client's reputation-based scheme [7] evaluate the trustworthiness of the clients in certain activities according to their past experiences. On the other hand, the payment-based mechanism design a quality-based incentive to avoid inefficient contributions and unnecessary rewards. This scheme motivates clients to contribute and receive electronic money as rewards. A centralized management structure of the incentive mechanism enables simple transactions between clients and trusted authority. The trusted authority guarantees direct communication and secure transaction. Hence, the reliability, equity, and quality of the system are influenced by the central trusted authority as a service provider.

The centralized incentive mechanism is facing the risk of a single point of failure. The authorized party requires high-security protection to guarantee a secure transaction in the system. It worth noting that a single mistake affects the entire system's orchestration. In the context of a user privacy point of view, a central service provider may also expose the client's private data, or trade it for personal benefit involving collision attacks. Therefore, the client's privacy issue needs to be considered to form the fairness incentive mechanism.

2.3 Requirements of Fairness Incentive Mechanism

Incentive mechanisms used to form collaborative fairness among decentralized clients. The objectives of the paper are to design the appropriate incentive mechanism to protect client's data privacy in a decentralized learning framework (collaborative learning). The requirements of fairness incentive mechanism provided as follows:

- **Privacy and confidentiality.** Since the server of model provider potential to emerge the risk of the client's privacy data leakage, the clients unwilling to share their data in a collaborative training process. Accordingly, the decentralized learning framework required to preserve the privacy data clients and to encourage the more contributions of clients in the training process. To protect the confidentiality of the private data, clients able to train the model on-device to avoid the

data are abused by the model provider. In this sense, the clients train the model locally using their data before uploads to improve the global model collaboratively. Therefore, the data of clients never leave their device while the server has no visibility into the client's local data.

- **Equity and sustainability.** The design of the incentive mechanism should be able to provide the fairness of reward schemes among clients based on their contributions. Smart contracts through the Ethereum platform are exploited to design the sustainable incentive mechanism.
- **Exhaustiveness.** The system design should be able to ensure the completeness of the transaction process.

3. Rewarding Model for Contributed Clients

3.1 Decentralized Learning Pipelines

The synapsis server Ag_{svr} along with the number of devices involved $Dx_1, Dx_2, Dx_3, ..., Dx_n$ are the core in the decentralized learning system. Every device holds a certain amount of valuable private data δ_n. The synapsis server Ag_{svr} also acts as a provider for a global model, which is used by devices to train the model $\xi_{ver\_0}^{gb_n}$, in the decentralized learning. The updated models genuinely can be understood as the average value of the training results derived from the total number of devices combined as can be seen in (1). The devices with their associated private dataset can be denoted as follows:

$$\sum_{n=0}^{x_n} \{ Dx_1, Dx_2, \dots, Dx_n \parallel \delta_n \} ; \text{for} \, \xi_{ver_0}^{gb_n} \rightarrow \xi_{ver_n}^{up_n} \qquad (1)$$

The data and its type may vary depending on the models' requirements. For instance, the data in the healthcare area which is profoundly sensitive [21] is denoted by a feature vector for every device $Dx_1, Dx_2, Dx_3, ..., Dx_n$; with $|Dx_n|$. Eventually, the margin between classes of data can be maximized by implementing several protocols such as Support Vector Machine (SVM).

The devices $|Dx_n|$ independently use a global model to produce an updated gradient value. Within a certain timeframe, the server collects all the gradient values $\sum_{n=0}^{x_n} \{ Dx_1, Dx_2, \dots, Dx_n \parallel \delta_n \}$ so that the latest model $\xi_{ver\_n}^{up_n}$ is derived and ready to be distributed again across the network. In the initial process for each round, the synapsis server roughly mapping the available devices to be set, and also sets several dynamic rules r_{dc} that every device is obligated to meet such as network latency, bandwidth, memory, storage, and to name a few. However, for ease of the presentation, we set the device is appropriate to be used to conduct a training.

$$\partial_i > 0; \in \xi_{ver_i}^{gb_i} \rightarrow |Dx_n \| \delta_n| \qquad (2)$$

There is a maximum waiting time Mx_t in the server-side for each round of training model by devices as shown in (2). The Mx_t for a round of activity $Mx_t > 0$ is considered adequate to the server Ag_{svr} in order to collect the gradient information from the devices involved. This Mx_t makes the system more organized with a standard for every execution carried out. Within the maximum time Mx_t, the devices $|Dx_n \| \delta_n|$ are allowed to access the global model sending the updated gradient to the server back and forth. In this sense, the dataset on devices remains confidential.

Eventually, the decentralized learning process runs continuously as long as the model providers consider the updated gradient values from devices have a significant contribution to improve the model. The provider is able to change the dynamic rules since he is the owner of the model. Furthermore, the model provider can state the amount of cryptocurrency given for the devices that refer to data that is possessed. The reward is proportionally designed by the model provider, and it is stated arbitrary in the Ethereum smart contracts.

3.2 Decentralized Rewarding - Resistance to Failure

The decentralized rewarding mechanism is developed as a reward mechanism to the parties involved in the decentralized learning. The parties receive two types of rewards in general, depending upon their roles. The devices with their private training data are the main focus in the system. Whilst miners' rewards follow the mechanism of blockchain smart contracts

by design. The objective of the decentralized rewarding scheme is to deliver rewards proportionally in a highly secure and reliable manner without having a single point of failure.

Each device $Dx_1, Dx_2, \ldots, Dx_n \parallel \delta_n$ that is incorporated into model $\xi_{ver_i}^{gb_i}$ produces a different gradient value that refers to various personal datasets. In this sense, every device has a local weight updated which is denoted by $\psi^x(\triangle W_n)$ that consist of a cipher ψ^x to encrypt the information. Conclusively, the value of updated aggregation from several devices can be defined in equation (3) as follows:

$$\psi^x(W_{(n+1)}) = \frac{1}{\forall Dx_n}\psi^x(W_n)\Pi_{(n+1)}^x\psi_n^x(\triangle W_n) \qquad (3)$$

Where $\psi^x W_n$ is an encrypted weight value for a round of activity $Mx_t > 0$, while $\psi^x(W_{(n+1)})$ is a collective version of values derived from multiple devices within maximum waiting time Mx_t. Devices with their respective data $Dx_n \| \delta_n$ are required to submit a statement in smart contract regarding data ownership used to train a selected model $\xi_{ver\_0}^{gb_n}$. All of the information is specified in a transaction through a smart contract notated as T_{dx}^{δ}.

The transaction T_{dx}^{δ} is deployed by a pseudo-public key address generated by devices in advance in the form of $Pk_{dx}^{psu} \in \{g_{dx_1}^{Sk}; g_{dx_2}^{Sk}; g_{dx_n}^{Sk}\}$ with $g_{dx_n}^{Sk}$ is a public key generated from devices' private key S_k from a generator g (pre-specified parameter). Likewise, the private key is generated in advance $S_k \in Z^*$ that relies on the collision-resistant hash function in the element non-negative integer numbers. Within T_{dx}^{δ} there is a state of "knowledge" to describe the essential information of data used such as the size, data format, fields, description, and the samples. To summarize, we depict a high-level process of decentralized rewarding mechanisms by relying on blockchain smart contracts in Algorithm 1.

4. Implementation and Evaluation

The synapsis server provides a model based on a convolutional neural network (CNN or ConvNet) with a standard two-dimensional convolution layer. PyTorch build (stable version 1.5) and PySyft library are adopted to build the model conjointly with a Python-based programming language. Each device uses the same dataset gathered from the Modified National Institute of Standards and Technology database that consists of 60,000 samples to be performed in the global model in our system [22]. To facilitate information gathering, we use several interfaces such as *Ganache (Truffle suite)*, and *metamask* as an extension for accessing Ethereum enabled distributed applications (DApps).

The account addresses of the parties are generated by *Ganache (Truffle suite)* that consists of the public address and private key. *Ganache* uses *ethereumjs* to simulate full client behavior. We notice that private keys are 64 characters long (it must be input as a 0x-prefixed hex string). The amount of Ether for each party is assigned to be 100 Ether by default"-*defaultBalanceEther*". For the "*blockTime*" is set to be automatic mining, the system will immediately mine a new block for every coming transaction. In the meantime, the gas price is 20000000000 wei or equal to (20 Gwei), with the block gas limit defaults to 0x6691b7. The gas limit is also set for *eth_call* and

Algorithm 1: *Decentralized Rewarding Mechanism.*

1: // *Incentivized by relying blockchain smart contract*
2: **procedure** *Decentralized_Revenue* Γ_{dx}^{v};
3: Ag_{svr} collects the info of devices
4: $Dx_1, Dx_2, Dx_3, \cdots, Dx_n$;
5: Dx_n deploys T_{dx}^{δ} respects to the δ_n;
6: Ag_{svr} checks for $\psi^x(\triangle W_n)$; where $|Dx_n \| \delta_n|$;
7: Ag_{svr} calculate an aggregation value for $\xi_{ver\_n}^{up_n}$ in $\forall D_{xn}$;
8: //*when the calculation is complete, then:*
9: Ag_{svr} publishes new $\xi_{ver\_n}^{up_n}$ & check contribution $\forall D_{xn}$;
10: //*trigger SM directly*
11: **for** Active miners (by system) $m_1, m_2, m_3, \cdots, m_n$;
12: //m_n *validate the results till it gets confirmed*
13: Γ_{dx}^{v} is distributed to $\forall D_{xn}$;
14: **end procedure**

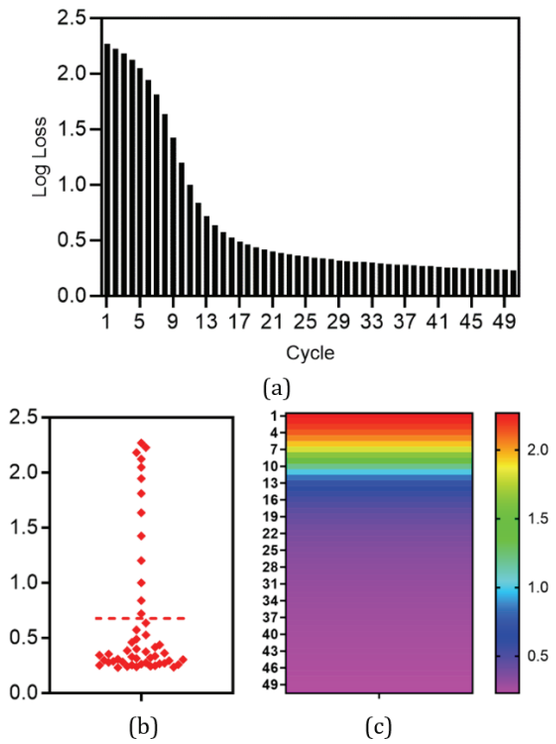

Fig. 1. The Performance of Decentralized Learning

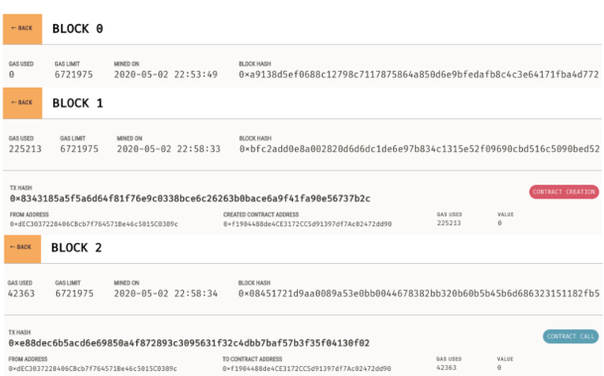

Fig. 2. The Information of the First Three Blocks

eth_estimateGas calls. It is specified as a *hex string*.

First, we record the performance of the multiple devices in building a decentralized *ConvNet* model provided by the synapsis server. Every device holds the same dataset with the equal capability to train a model. The devices also meet the fundamental requirements of the dynamic rules. The global model is gradually built to achieve better accuracy by upgrading the gradient value derived from the devices. The global model is trained by adopting the back-propagation algorithm. The performance of the designed system is depicted in Fig. 1.

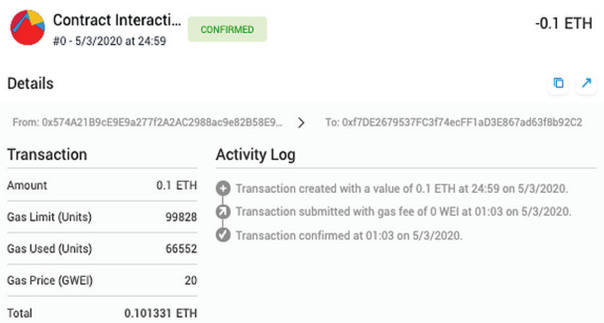

Fig. 3. Accessing Ethereum Enabled Distributed Applications

The number of devices conjointly building the global model is set to be 25 devices. For ease of presentation, we calculate the average value of the total performance from devices to be displayed in this paper. In the 1st-10th percentile, the average loss is around 1.8856. As nature in the deep learning process, the training process is getting better as an increasing cycle in training as shown in Fig. 1(a). The performance shows that the last 10th of total percentile achieve 82.33% - 92.64% accuracy. The distribution points of loss value are occurred in the 45th percentile up to 88th percentile as displayed in Fig. 1(b) with the heatmap distribution in Fig. 1(c).

As the transactions on the blockchain, the genesis block does not refer to the previous block. It is created with the label *Block 0* with the gas used is 225213 units (gas limit is set up to be 6721975 units). The *contract creation* function is added in Block 1 (created contract address into the blockchain), where the system spends 225213 units to run this function. Meanwhile, *Block 2* is the result of the execution of the *contract call* function which is arranged by the model provider or synapsis server. For the contract call function, the gas consumed is slightly high compare to the previous functions mentioned. The information of the first three blocks can be seen in Fig. 2.

When the claim contribution from the device through a transaction T_{dx}^{δ} is successfully confirmed by the model provider, then an amount of Ether is distributed to the devices proportionally to their resources and contributions. We use the *metamask* interface as a crypto wallet and gateway to blockchain application as well as to manage various addresses used in transactions as shown in Fig. 3. With a

certain number of contributions to the resources that the devices owned, the model provider provides 0.1 Ether to a device in which this transaction consumes 66552 Units of gas, with 99828 Units gas limits. The total cost for this transaction is equal to 0.101331 Ether. All these transactions are recorded in a blockchain sequence, which can be seen on the *ganache truffle* interface.

The provider can modify the dynamic rules frequently within the smart contracts, yet it can deactivate the previous version of the contract that has been approved by every device. Moreover, the model provider should migrate the contract in order to deploy the new contracts to the Ethereum network. The *truffle* migrates function is responsible for staging the provider deployment tasks that can be changed over time. For instance, to run the truffle migration function in our model costs 225213 Units gas, with the total cost 0.00450426 Ether, and the transaction is recorded in block 32 (See Fig. 4).

```
Replacing 'Migrations'
----------------------
   > transaction hash    : 0xb56ec502e3e387009d368f3ae1ee5f4d6d2d382bc
   > Blocks              : 0
   > contract address    : 0x13AEB9Af3c98a161889eb297eAB0a58bd668a736
   > block number        : 32
   > block timestamp     : 1588449967
   > account             : 0xdEC3037228406CBcb7f764571Be46c5015C0389c
   > balance             : 99.97408892
   > gas used            : 225213 (0x36fbd)
   > gas price           : 20 gwei
   > value sent          : 0 ETH
   > total cost          : 0.00450426 ETH
   > Saving migration to chain.
   > Saving artifacts
   -------------------------------------
   > Total cost:          0.00450426 ETH

Replacing 'Smart_Contract'
--------------------------
    > transaction hash:    0x73f6c7072732e332fbed143311ced209be1f8237d
    > Blocks: 0            Seconds: 0
    > contract address:    0x16878be30E005b92537AFf15846E7150F17F2dEd
    > block number:        34
    > block timestamp:     1588449968
    > account:             0xdEC3037228406CBcb7f764571Be46c5015C0389c
    > balance:             99.95996168
    > gas used:            663999 (0xa21bf)
    > gas price:           20 gwei
    > value sent:          0 ETH
    > total cost:          0.01327998 ETH
    > Saving migration to chain.
    > Saving artifacts
    -------------------------------------
    > Total cost:          0.01327998 ETH
```

Fig. 4. Initial Migration and Deploying a Contract

In respects to the total amount of Ether, every device receives the Ether proportionately to their resources by design, yet the devices are required to conduct a first move by tendering transaction to the model provider. The size of each transaction varies depending on the contributions from the device.

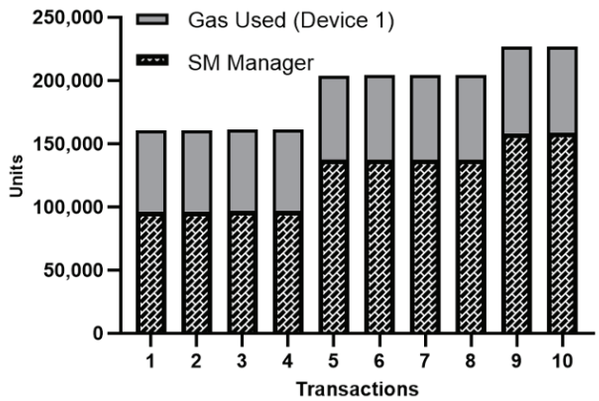

Fig. 5. The comparison Amount of Gas Used

The more notable a contribution, the bigger Ether is obtained. Yet, the gas is relatively higher compared to transactions with minimum contributions in building the AI model. This point can be seen in Fig. 5, where the Device 1 with the number of contributions varied from the minimum to the maximum, the amount of gas consumed is increasing in order to submit a transaction. Meanwhile, the amount of gas spent by the smart contract manager does not change significantly since there is not many changes made in the smart contract.

5. Leassons Learned

5.1 Privacy Concerns in Implementing

Up to the extent, we have performed a blockchain-based incentive scheme that relies on the Ethereum smart contract. The rewards are distributed in a secure manner with clear communication among parties (eliminating miscommunication), real-time tracking performance (it brings tremendous cost savings), and the model removes any possibility of manipulation, bias, and error during transactions. In short, the system preserves privacy for the devices with an efficient rewarding mechanism that can solve problems that commonly occur among parties.

Transaction in the Ethereum smart contract is transparent to the core. This platform leverages this transparency as part of Ethereum security. In this sense, the Ethereum platform ensures that users cannot falsify data and transactions. Apart from transparency and privacy on the Ethereum platform, there is another startup based on the Ethereum called *Enigma* which is dedicated to building an

Table 1. The Prominent Existing Protocols

Protocol & Implementation	Techniques Used	Info.
Confidential Transaction	Pedersen Commitments [23]	*Provides strong confidentiality *The hash-based commitment scheme
Zcash [24]	zk-SNARKS	*Provides confidentiality for users *Obscure amount of a transaction
Ring Signatures [25]	*Ring confidential *Group of members' public key	*Preserves privacy for the sender *No group manager *Without an approval from members
RingCT	*Ring confidential transaction *Outputs of previous transaction *Members of RingCT	*Disguise the exact amount of a transaction *Ring confidential transaction of previous output of transactions
CryptoNote Protocol [26]	*Ring signature *Stealth Address *RingCT	*strong privacy for sender, recipient, and amount of a transaction *Freely used without an approval of members *Observer cannot tell the activities that occur
Stealth Addresses	*Public key (pair)	*Privacy for the recipient *One-time pair public key *Diffie-Helman principle

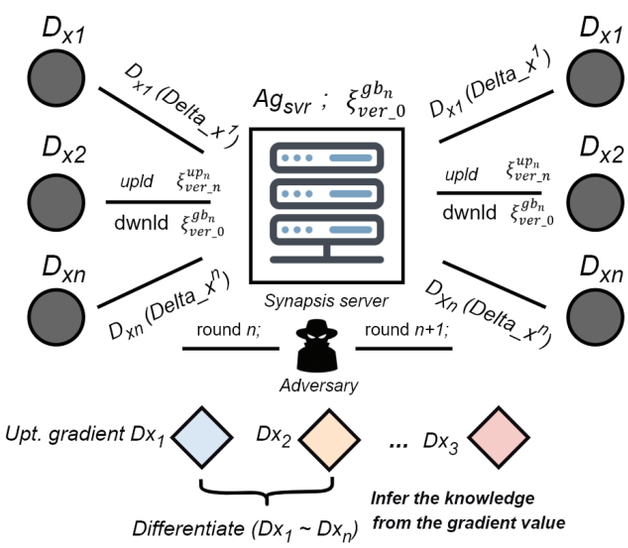

Fig. 6. Inferring the Knowledge of the Devices' Information

off-chain computational setting for different sorts of data privacy for the users. Nevertheless, the terms of privacy in the private decentralized learning system, it is going beyond the concept that the system can protect the anonymity or the learning activity with the capability to do computations on encrypted data.

Concerns in implementing the Ethereum smart contract in a decentralized learning system as a whole are briefly shown in Fig. 6. Privacy issues are not remarked when the device conducts training locally for the selected global model $\xi^{gb_n}_{ver\_0}$, thus the data is kept confidential during training. However, when the device tenders the T^{δ}_{dx} to the synapsis server that consists of $\psi^{x}(\triangle W_n)$ with a cipher ψ^{x} to encrypt the information. The device exposes his information publicly through the T^{δ}_{dx} transaction in order to meet the requirements of the dynamic rules designed by the provider. With a moderate assumption, the adversary can find the different values from the set of gradient

values for a particular round, so that he can conclude the information held by the targeted device.

The adversary also can make a connection between certain transactions through the address for each device, since the device uses the same public address for each transaction conducted in the decentralized learning. Additionally, the amount of cryptocurrency can be seen by every party in the Ethereum blockchain. With the combination of knowledge possessed by the adversary, he can associate the amount of cryptocurrency (Ether) received by the device with the estimated amount of data used in training. Moreover, every activity in this scheme can be tracked even though the information is encrypted with cryptographic algorithms. On this matter, by enforcing Ethereum smart contracts as an incentive platform in decentralized learning can break privacy, it is also linkable, and traceable publicly. This point is a concern if applied to a system with sensitive data.

5.2 Plausible Solutions

Privacy and transparency in Ethereum blockchain are the keys barrier when applying decentralized learning to a real business environment. The presented scheme satisfies the efficiency, accuracy, and clear communication, yet the privacy of users is not fully protected by system. Therefore, the decentralized learning scheme needs to be combined with several protocols in order to cover up the aforementioned issues. For instance, *Pedersen Commitments* can be adopted in the system to provide confidential transactions with the hash-based commitment scheme included within the protocol. Likewise, the *Zcash* protocol with *zk-SNARKS* techniques can be implemented within the decentralized learning to obscure the amount of transaction provided by the synapsis server.

To disguise the activities during training, *Ring Signature* protocol might be suitable since devices can submit a transaction by using the ring member signature. By doing so, the observer cannot distinguish the actual signer. Since the adversary can impose the devices' information through the amount of Ether sent by the provider, *RingCT* protocol is likely can resolve this issue. *RingCT* disguises the actual amount of Ether by combining it with prior outputs of Ether. Compact implementation of these protocols refers to the CryptoNote protocol that leverages Ring Signature, Stealth Address, and RingCT protocols as the core idea to preserve privacy. The summary is shown in Table 1.

6. Opportunities and Challenges

In this section, we present the opportunities and challenges in adopting decentralized learning with Ethereum smart contract as a platform for distributing rewards to the contributed devices. Overall, we emphasize the role of a centralized synapsis server to be replaced with the decentralized computing parties, thus the system can be fully decentralized. Therefore, the decentralized learning scheme needs to be combined with several protocols in order to cover up the aforementioned issues.

6.1 Opportunities

The presented decentralized learning scheme is still relying on the centralized synapsis server to calculate the gradient values which are derived from multiple clients. In this regard, the system is not entirely in a decentralized form. Centralized synapsis server becomes a concern since it is inseparable from a single point of failure (SPoF). Moreover, with an increasing number of activities in the decentralized learning scheme, it can burden the synapsis server in handling the task.

By looking at the blockchains' merits, the role of the synapsis server can be replaced by adopting a decentralized blockchain approach. The updated aggregation values can be calculated by more than one authorized validator incorporated in the blockchain network. The prominent consensus such as *Practical Byzantine Fault Tolerance* (PBFT) is suitable to be implemented. It can defend against system failures with or without symptoms in order to reach an agreement among the validators. Furthermore, blockchain can play a role in storing the record version of the models (off-chain), while the record of the transaction is stored in the on-chain. By doing so, the clients allow accessing the model that they desire. We put forward these points as the part concept of our future research. Eventually, the plausible solutions described in Section 5.2.

6.2 Challenges

Decentralized learning with a commensurate incentive scheme faces several concerns in general, especially in a complicated communication scheme. The large number of parties involved, and the complexity of the communication system can be a challenge in implementing this scheme. The willingness of the end-users to provide their private data during training can be an obstacle as well. Even though the end-users are incentivized proportionately, and the system is secure by design, but the decentralized learning scheme is likely still suffer from privacy.

The attack in the collaborative learning scheme is under the assumption where the malicious clients or server has white-box access to the model updates. The adversary can actively bias the model to leak property by sending crafted updates. However, the performance of the adversary decreases as the number of clients in the system increases. In short, the system should be able to tackle these issues e.g. by adopting several protocols, anomaly detection, and so forth.

7. Conclusion

In this paper, we have presented the practical concerns in implementing Ethereum smart contracts as a rewarding platform in decentralized learning. Decentralized learning scheme combined with blockchain technology can enhance the privacy of the parties by design. However, the reward distribution by relying on the Ethereum smart contract breaks the privacy since the devices are frequently sending the training information to the model provider via a smart contract. Furthermore, the address of the parties is exposed publicly. The observer can link the transactions by sniffing the information sent and received by parties. This is a concern if the data is private and sensitive. Therefore, an in-depth analysis is needed to implement the presented scheme in the real world. In addition, various protocols such as *ring signature, stealth address*, and *CryptoNote* protocol can be a plausible solution to tackle the concerns. We consider this point as an indispensable element of our further research.

References

[1] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," pp.1-10, 2016, [Online]. Available: http://arxiv.org/abs/1610.05492.

[2] S. Lugan, P. Desbordes, E. Brion, L. X. Ramos Tormo, A. Legay, and B. Macq, "Secure architectures implementing trusted coalitions for blockchained distributed learning (TCLearn)," *IEEE Access*, Vol.7, pp.181789-181799, 2019.

[3] Hard, Andrew, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage, "Federated learning for mobile keyboard prediction," arXiv preprint arXiv:1811.03604, 2018.

[4] Bonawitz, Keith, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth, "Practical secure aggregation for privacy-preserving machine learning," In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp.1175-1191, 2017.

[5] Yang, Qiang, Yang Liu, Tianjian Chen, and Yongxin Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, Vol.10, No.2, pp.1-19, 2019.

[6] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang and W. Luo, "Deepchain: Auditable and privacy- preserving deep learning with blockchain-based incentive," *IEEE Transactions on Dependable and Secure Computing*, 2019.

[7] R. Jurca and B. Faltings, "An incentive compatible reputation mechanism," 2003.

[8] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," 2003.

[9] B. Bin Chen and M. C. Chan, "MobiCent: A credit-based incentive system for disruption tolerant network," 2010.

[10] U. Shevade, H. H. Song, L. Qiu, and Y. Zhang, "Incentive-aware routing in DTNs," 2008.

[11] A. Singh, R. M. Parizi, Q. Zhang, K. K. R. Choo, and A. Dehghantanha, "Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities," *Computers and Security*, 2020.

[12] Rahmadika, Sandi, and Kyung-Hyune Rhee, "Toward privacy-preserving shared storage in untrusted blockchain p2p networks," *2019 Wireless Communications and Mobile Computing*, pp.1-13, 2019.

[13] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," 2016.
[14] M. Vukolić, "Rethinking permissioned blockchains," 2017.
[15] Lim, Wei Yang Bryan, Nguyen Cong Luong, Dinh Thai Hoang, Yutao Jiao, Ying-Chang Liang, Qiang Yang, Dusit Niyato, and Chunyan Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, 2020.
[16] Wang, Shiqiang, Tiffany Tuor, Theodoros Salonidis, Kin K. Leung, Christian Makaya, Ting He, and Kevin Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, Vol.37, No.6, pp.1205-1221, 2019.
[17] Dean, Jeffrey, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Marc'aurelio Ranzato et al., "Large scale distributed deep networks," *Advances in Neural Information Processing Systems*, Vol.25, pp.1223-1231, 2012.
[18] J. Zhao, Y. Chen, and W. Zhang, "Differential privacy preservation in deep learning: Challenges, opportunities and solutions," *IEEE Access*, 2019.
[19] Lane, Nicholas D., Sourav Bhattacharya, Petko Georgiev, Claudio Forlivesi, Lei Jiao, Lorena Qendro, and Fahim Kawsar, "Deepx: A software accelerator for low-power deep learning inference on mobile devices," *In 2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pp.1-12. IEEE, 2016.
[20] J. Wang, X. Zhu, J. Zhang, B. Cao, W. Bao, and P. S. Yu, "Not just privacy: Improving performance of private deep learning in mobile cloud," 2018.
[21] Rahmadika, Sandi, and Kyung-Hyune Rhee, "Blockchain technology for providing an architecture model of decentralized personal health information," *International Journal of Engineering Business Management*, Vol.10, pp.11-12, 2018.
[22] Deng, Li, "The mnist database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Processing Magazine*, Vol.29, No.6, pp.141-142, 2012.
[23] S. F. Sun, M. H. Au, J. K. Liu, and T. H. Yuen, "RingCT 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency Monero," 2017.
[24] Sasson, Eli Ben, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza, "Zerocash: Decentralized anonymous payments from bitcoin," *In 2014 IEEE Symposium on Security and Privacy*, pp.459-474. IEEE, 2014.
[25] Rivest, Ronald L., Adi Shamir, and Yael Tauman, "How to leak a secret," *In International Conference on the Theory and Application of Cryptology and Information Security*, pp.552-565. Springer, Berlin, Heidelberg, 2001.
[26] N. Van Saberhagen, "CryptoNote v 2.0," Self-published, pp. 1-20, 2013.

Sandi Rahmadika
https://orcid.org/0000-0002-7848-6579
e-mail : sandika@pukyong.ac.kr
He received his Master of Engineering degree from the dual master degree programs between Institut Teknologi Bandung (ITB), Indonesia, and Pukyong National University (PKNU), South Korea in 2016. He is currently a Ph.D. student in the Laboratory of Information Security and Internet Applications (LISIA), PKNU. His research interests include applied cryptography, privacy-preserving in the decentralized system, and AI with blockchain integration.

Muhammad Firdaus
https://orcid.org/0000-0003-0104-848X
e-mail : mfirdaus@pukyong.ac.kr
He received his Master of Engineering degree in Telematics and Telecommunication Networks from Institut Teknologi Bandung (ITB), Indonesia. He is currently a Ph.D. student in the Laboratory of Information Security and Internet Applications (LISIA), Pukyong National University. His research interests include communication security, applied cryptography, and blockchain with AI inegration.

Seolah Jang
https://orcid.org/0000-0002-4636-5027
e-mail : seolaang1020@pukyong.ac.kr
She received a B.S. degree in the Department of Global Business, Dong-A University, Busan, South Korea, in 2020. She is currently a Master course student in the Interdisciplinary Graduate Program of Artificial Intelligence on Computer, Electronic and Mechanical Engineering, Pukyong National University, Busan, South Korea. Her research interests include Blockchain, IoT Security, and Artificial Intelligence.

Kyung-Hyune Rhee
https://orcid.org/0000-0003-0466-8254
e-mail : khrhee@pknu.ac.kr

He received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Republic of Korea from 1985 to 1993. He also worked as a visiting scholar in the University of Adelaide, the University of Tokyo, and the University of California, Irvine. He has served as a Chairman of Division of Information and Communication Technology, Colombo Plan Staff College for Technician Education in Manila, the Philippines. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Republic of Korea. His research interests center on security and the evaluation of blockchain technology, key management and its applications, and AI-enabled security evaluation of cryptographic algorithms.