Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Blockchain and homomorphic encryption based privacy-preserving model aggregation for medical images

Rajesh Kumar a, Jay Kumar c, Abdullah Aman Khan d,b, Zakria e, Hub Ali f, Cobbinah M. Bernard d, Riaz Ullah Khan a, Shaoning Zeng a,*

a Yangtze Delta Region Institute (Huaxia), University of Electronic Science and Technology of China, Huaxia 313001, China
b Sichuan Artificial Intelligence Research Institute, Yibin 644000, China
c Institute for Big Data Analytics – Dalhousie University, 6299 South St, Halifax NS B3H4R2, Canada
d School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
e School of Software Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
f Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

MSC:
41A05
41A10
65D05
65D17

Keywords:
Blockchain
Federated-learning
Privacy-preserved
COVID-19

Abstract

Medical healthcare centers are envisioned as a promising paradigm to handle the massive volume of data for COVID-19 patients using artificial intelligence (AI). Traditionally, AI techniques require centralized data collection and training models within a single organization. This practice can be considered a weakness as it leads to several privacy and security concerns related to raw data communication. To overcome this weakness and secure raw data communication, we propose a blockchain-based federated learning framework that provides a solution for collaborative data training. The proposed framework enables the coordination of multiple hospitals to train and share encrypted federated models while preserving data privacy. Blockchain ledger technology provides decentralization of federated learning models without relying on a central server. Moreover, the proposed homomorphic encryption scheme encrypts and decrypts the gradients of the model to preserve privacy. More precisely, the proposed framework: (i) train the local model by a novel capsule network for segmentation and classification of COVID-19 images, (ii) furthermore, we use the homomorphic encryption scheme to secure the local model that encrypts and decrypts the gradients, (iii) finally, the model is shared over a decentralized platform through the proposed blockchain-based federated learning algorithm. The integration of blockchain and federated learning leads to a new paradigm for medical image data sharing over the decentralized network. To validate our proposed model, we conducted comprehensive experiments and the results demonstrate the superior performance of the proposed scheme.

1. Introduction

The drastic spread of the novel coronavirus (COVID-19) around the globe caused a large number of deaths within a year. The COVID-19 virus causes acute respiratory disease, which directly infects the human lungs, resulting in intensive breathing difficulties. Due to its highly contagious nature, COVID-19 detection remains among the highest-priority tasks. Currently, various artificial intelligence (AI) techniques are being explored to discover better solutions for detection (Kumar et al., 2021a,b; Deng et al., 2021; Khan et al., 2020). Last year, a significant portion of the research focused on CT scan-based detection, which has proven to be a more reliable source. However, these techniques often require a large amount of data from a single source (hospital or research center) to train the classification model to predict more accurately. In contrast, data from a single source lacks the variance in feature distribution. The less variation in the data, the greater the sampling error and model loss, which consequently, affects the diagnosis performance. The data variation problem can be solved if many hospitals can share data. However, data confidentiality and privacy concerns confine hospitals from sharing the data to train the model. Due to this issue, traditional learning (where only local data is considered) may not generalize properly and perform well in real-world situations. In contrast, transfer learning enables sharing the model weights instead of sharing the actual data. Transfer learning exploits a general pre-trained model and fine-tunes the pre-trained.
model parameters (weights) (Das et al., 2020; Pathak et al., 2020). The sensitivity of a local model depends on the quality of the pre-trained model. For example, a hospital in a rural area probably has insufficient data to train the model. However, the hospital can collaborate with another hospital while considering the same goal without sharing the actual data. However, transfer learning still confines the base model to increase robustness while taking advantage of local data from the hospital. For this reason, hospitals are still unable to fully benefit from AI-based medical analysis.

A possible solution to overcome the data privacy issue is federated learning. Federated learning is capable of collaboratively training a common model without physically exchanging the actual data. The collaborative model training solves the problem of data variance and enables the evolution of the model over time for all hospitals, i.e., the model can be updated for the latest mutation samples, etc. Generally, federated learning is a collaborative learning framework that allows multiple collaborators to train their local model and share the learned weights for the aggregation process. This aggregation process helps in realizing a robust model that is up to date regarding the latest mutations and samples. Such procedure of gaining knowledge is in the form of a consensus model without moving the actual patients' data beyond the firewalls of the parent data centers (hospitals or research centers). To this point, the learning process occurs locally at each participating institution, and only the model parameters are shared using a federated server for global model aggregation. Originally, federated learning was developed for different domains i.e., distributed learning, edge devices, and mobile computing. Due to its vast scope of applications, it has gained considerable research attention for healthcare applications (Blanquer et al., 2020; Yang et al., 2021; Thwal et al., 2021; Malekzadeh et al., 2021; Li et al., 2020; Baheti et al., 2020; Brisimi et al., 2018).

Recent research has proven that the models trained by federated learning can achieve comparable levels of performance to the ones trained with centrally hosted medical data center (Can and Ersoy, 2021; Dinh et al., 2021; Cheng et al., 2020; Yang et al., 2020). However, there exist some privacy and security concerns with federated learning (Shokri and Shmatikov, 2015) where gradients of local models related to users can be shared without compromising the security and privacy of the data. To this end, previous methodologies are easily accessible to passive attackers, thus making them vulnerable (Dai et al., 2019; Tang et al., 2018) to attacks.

To tackle the security and scalability issues, the blockchain provides a ledger technology that enables model decentralization i.e., without involving any central server. In particular, blockchain provides the facility to collect the data model securely from various points or locations (i.e., Japan, China, Pakistan, USA, and UK) to train the global model. Recent research focuses on federated learning through the use of a central server topology (Blanquer et al., 2020; Yang et al., 2021; Thwal et al., 2021; Malekzadeh et al., 2021; Li et al., 2020; Baheti et al., 2020; Brisimi et al., 2018; Can and Ersoy, 2021). However, none of these studies explored decentralized blockchain-based federated learning for global model aggregation for medical image analysis. Nevertheless, Kumar et al. (2021a) proposed the blockchain-based federated learning-based image detection technique but they did not secure the gradients. Therefore, there is still a gap between secure federated learning (without the dependency of a central server) to provide secure model sharing and trust issues for data providers (e.g., hospitals).

Given the current situation, the model needs to be updated continuously to deal with new types of COVID-virus mutations while considering the previously discussed problems. In this paper, we propose a framework that integrates privacy-preserving federated learning over the decentralized blockchain. For federated learning, we designed a novel capsule network for local hospital training that utilizes image segments for classifying COVID-19 images. Our segmentation network aims to extract nodules from the patients’ chest CT scan images. For every locally trained model, gradients are encrypted using a homomorphic encryption technique to preserve the privacy of each hospital. In this encryption technique, hospitals are assigned the same secret key for reducing the communication overhead of high-dimensional data in neural networks. In this way, the client or user side encryption knowledge, which guarantees user privacy using blockchain, ensures the data’s reliability. The task of aggregation and learning the global model is carried out over the blockchain. We use the Direct Acyclic Graph (DAG) to improve the blockchains’ computation efficiency. The following are the main contributions of this paper:

1. We propose a blockchain-based federated-learning framework that enables collaborative data training and decentralization of federated learning models without the involvement of a central server.
2. We utilized a homomorphic encryption scheme for encrypting the weights of the locally trained model which ensures the privacy of the hospital data.
3. We designed a blockchain-based federated learning algorithm to build data models and share the data models instead of actual data. The algorithm aggregates the local model weights to realize the global model.
4. For local model training, we propose a Capsule Network for segmenting pneumonia infection regions and automatically classifying the COVID-19 chest CT images.
5. The proposed framework is continuously updated to deal with new mutations of COVID-virus and quickly exchange the most recent information with hospitals around the world.

2. Preliminaries

This section presents a brief introduction to deep learning, federated learning, homomorphic encryption, and blockchain-based federated learning followed by the system model. The main mathematical notations used in this article are listed in Table 1.

### 2.1. Deep learning

Usually, deep learning extracts features using deeper convolutional networks to extract features. Further, deep learning models utilize feedforward and backpropagation algorithms to train the model using the obtained features. A general deep learning model with a feedforward neural network is shown in Fig. 1. The feedforward function can be defined as \( f(x, w) = y \), where \( x \) shows the input vector and \( w \) represents the parameter vector. The \( D = (x_i, y_i) : i \in I \) is the training dataset for each instance of \( (x_i, y_i) \). Moreover \( I \) is the loss function, whereas the training dataset \( D \) on loss function defined as \( L(D, w) = \frac{1}{|D|} \sum_{(x_i, y_i) \in D} L(y_i, f(x_i, w)) \). However, the backpropagation phase utilizes methods such as stochastic gradient descent (SGD) for updating the parameters.

### Table 1

Summary of the mathematical notations.

| Notation | Description |
|----------|-------------|
| \( W(l) \) | Local model weights |
| \( m_i(t) \) | Local model learned by devices |
| \( CW \) | The cumulative weight of transactions |
| \( W \) | Weights of the model |
| \( P_{x,y} \) | The transition probability of transactions |
| \( \lambda \) | The 0 and 1 selection state |
| \( E_{\text{ct}} \) | Plaintext space |
| \( A \rightarrow Z_{ct} \) | Matrix |
| \( \sigma \) | Gradients vector for the model |
| \( pk_i / sk_i \) | Public/private key |
| \( \odot \) | Product between two ciphertexts |
There are four stages comprising two transformer blocks for each stage. \( \mathcal{L}_n \) presents the output of SW-MSA. Furthermore, MLP stands for multi-layer perceptron. We utilize the size of 2 × 2 × 2 with a dimension of 2 × 2 × 2 × 4. In total, there are four stages comprising two transformer blocks for each stage. At the first stage, linear embedding layer creates \( \frac{H 	imes W 	imes D}{2} \) tokens. To further decrease the feature representation resolution, a patch merging layer is utilized with a factor of 2 (at the end of each stage). In other words, a patch merging layer groups and concatenates patches resulting in a 4C sized feature embedding. Further, at each stage, the feature size is reduced by 2C.

The decoder part utilizes the feature representations extracted by the encoder using skip connections for each resolution. For each stage \( n \) of the encoder and the bottleneck the output is reshaped into \( \frac{H}{2} \times \frac{W}{2} \times \frac{D}{2} \) and further a residual block of two \( 3 \times 3 \times 3 \) convolutional layers is utilized along with normalization. By using this mechanism, the size of the feature map is increased, by a factor of 2, with the help of deconvolutional layers. Moreover, the outputs are concatenated with outputs of the previous stage. Later, another residual block is utilized. The final segmentation is computed by a 1 × 1 × 1 convolutional layer with sigmoid activation. Further details about Swin UNetR can be found in Hatamizadeh et al. (2022).

2.2. Swin UNetR segmentation

For the segmentation part, we utilized a Swin UNet Transformer network i.e., (Swin UNetR) (Hatamizadeh et al., 2022). This network helps in the segmentation of lung CT scans. The Swin UNetR encoder extracts features as inputs. It outputs an unencrypted version of the same data with a ciphertext without decryption. The new encrypted data matches the result of the operation performed on the unencrypted data after decryption. We utilize the BGV (Brakerski et al., 2014) encryption scheme algorithm, which takes the secret key with large noise and a ciphertext as inputs. It outputs an unencrypted version of the same data with a
of the users. Therefore, we use federated learning with the blockchain from multiple sources without leaking the privacy and authentication. The function \( \phi \) maintains a low-rank functionality. The function \( Z \) given by:

\[
\begin{bmatrix}
\phi_{11} & \phi_{12} & \cdots & \phi_{1S} \\
\phi_{21} & \phi_{22} & \cdots & \phi_{2S} \\
\vdots & \vdots & \ddots & \vdots \\
\phi_{S1} & \phi_{S2} & \cdots & \phi_{SS}
\end{bmatrix}
\]

This key is only accessible to the users/participants who share the mini-batch data.

\[
\begin{bmatrix}
Z_{t(1)} \\
Z_{t(2)} \\
\vdots \\
Z_{t(S)}
\end{bmatrix} =
\begin{bmatrix}
\phi_{11} & \phi_{12} & \cdots & \phi_{1S} \\
\phi_{21} & \phi_{22} & \cdots & \phi_{2S} \\
\vdots & \vdots & \ddots & \vdots \\
\phi_{S1} & \phi_{S2} & \cdots & \phi_{SS}
\end{bmatrix}
\begin{bmatrix}
Z_{t(1)} \\
Z_{t(2)} \\
\vdots \\
Z_{t(S)}
\end{bmatrix}
\]

where \( Z(t) \) shows the vector data of the \( t \) node of the blockchain ledger. The \( \otimes \) operator shows the product between two ciphertexts given by:

\[
Z_{t(i)} = \phi_{t(i)} Z_{i(1)} + \phi_{t(2)} Z_{i(2)} + \cdots + \phi_{t(S)} Z_{i(S)}
\]

Fig. 3 shows the homomorphic encryption function with the linear transformation of a matrix. In this way, the linear transformation maintains a low-rank functionality. The function \( \phi \in \{0, 1\} \), and \( \sum_{i=0}^{S} w_{i} = 1 \) shows the homomorphic encryption with private key.

2.5. Blockchain-enabled federated learning

Training a better AI model for the industry requires collecting data from multiple sources without leaking the privacy and authentication of the users. Therefore, we use federated learning with the blockchain distributed ledger to update the global AI model. The blockchain collects the data model from different nodes and aggregates the local and global models. The smart contract then uploads the weights and updates the models. The proposed architecture integrates blockchain with federated learning for full decentralization and enhanced security. Also, decentralization provides higher accuracy of the model and enables the poisoning-attack-proof.

Some issues are not resolved for federated learning, i.e., insufficient incentives, poisoning attacks, etc. Therefore, some authors (Lu et al., 2020b; Qu et al., 2020) design the blockchain with federated learning. Similarly, Pokhrel and Choi (2020) designed a technique to protect privacy. The major issue with the previous papers was that they did not include the encryption technique with the blockchain model gradient sharing. Therefore, this paper uses the directed acyclic graph with the Proof-of-Work (PoW) consensus algorithm for the aggregation of gradients. Additionally, this work is fully decentralized and trains an accurate model without leaking the privacy of the user.

3. Secure data sharing

This section provides an introduction to the high-level architecture of the system and technical details in Fig. 4. Our proposed scheme consists of multiple users sharing the data securely using federated learning with blockchain. The proposed architecture has multiple phases.

Local model:

1. Input COVID-19 images to train the model.
2. Learn the local model and calculate the local gradients.
3. Encrypt the gradients of the local model.

Send the weights to the blockchain network for aggregation model:

1. Aggregate all user weights ciphertext i.e., \( 1_u W_{i(a)} + \sum_{i=5}^{1} D_i W_{i(a)} \)

Broadcast the weights:

1. Update the deep learning model based on global weights.
2. Upload the local model updates.

3.1. Local model training

In this section, we provide the details regarding training the local model for the detection of COVID-19. The main model is divided into three parts: (i) Segmentation Network (ii) Classification (iv)Probabilistic Grad-CAM Saliency Map Visualization.

3.1.1. Segmentation process

We obtained the ground-truth lung masks and extracted lung region using a learning method (Liao et al., 2019; LaLonde and Bagci, 2018). We removed the unnecessary or failed data manually, and the remain- ing segmentation data is taken as ground-truth masks. The 3D Lung mask also serves as input together with the whole image for training and testing data. The training objective is to adopt the capsule network segmentation. Where \( r_{f(i)} \) is the routing coefficient, \( b_{f(i)} \) shows the
Architecture of blockchain-based secure data sharing using homomorphic scheme. Step 1: training and segmentation of CT scans using capsule network, Step 2: Encrypt the gradients using the homomorphic scheme, Step 3: use a blockchain-based federated learning model for training the global model.

Pixel of images, $s$ and $y$ shows the ground truth label. The segmentation is represented as follows:

$$\tilde{u}_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})}$$

To determine the final output of the segmentation using the non-linear squashing function, we have:

$$v_{xy} = \frac{\|p_{xy}\|^2}{1 + \|p_{xy}\|^2}$$

where $v_{xy}$ is the output of the segmented image with the spatial location $(x, y)$ and $p_{xy}$ is the final input.

### 3.1.2. Classification

We designed a Capsule Network due to the nature of its inverse graph, which helps to detect medical images and achieve high performance. Therefore, the capsule network is capable of predicting the instantiating parameters for any medical image or object. The estimated probability of an object is represented via the length of a vector. However, the Capsule Network technique provides augmented transformations (i.e., rotation, stretching, skewed, thickness, etc.) to improve the performance on a smaller amount of data. The possible probability of the length of the vector is between 1 and 0 using the squashing function of the capsule network. Each layer is connected to the previous layer, and the previous layer is the output of the next layer. Capsule Network does not use a dot product to make the prediction and improve the accuracy. Instead of using dot products, they used the path in a hierarchy or a dynamic routing mechanism to find multiple objects in an image, so they could recognize them. The Capsule Network contains four layers: (i) convolution layers, (ii) primary capsule, (iii) DigitCaps (second capsule), and (iv) fully connected layers. Each layer of the network is composed of multiple capsules in terms of convolution layers (i.e., Conv1 and Conv2); ReLU (rectified linear unit) is adopted for activating the convolution layers. As an outcome, each capsule (i.e., Conv1 and Conv2) generates different feature maps. Similarly, the second layer of DigitCaps presents the output layer of the capsule. The loss is calculated after the digits cap. Finally, the fully connected layer is used to reconstruct the images. $W_{ij}$ are a pair of weighted matrices. A pose vector $U_i$ is rotated and translated by the weighted matrix to a vector $\tilde{u}_i$ for each component. The instantiation parameters of capsules at higher levels are predicted by the transformation matrix at the same capsule level.

$$\tilde{u}_i = W_{ij}u_j$$

In contrast, the prediction vector is defined as follows:

$$\hat{u}_i = W_{ij}u_j$$

Predictions from each lower-level capsule are combined to form the next higher level capsule ($s_j$) the total sum of predictions layers is $c_{ij}$ represented as a coefficient of $c_{ij}$ with coupling defined as:

$$S_j = \sum_i c_{ij}\hat{u}_{ij}$$

Here, $c_{ij}$ is a routing softmax function defined as:

$$c_{ij} = \frac{e^{b_{ij}}}{\sum_k e^{b_{ik}}}$$

As shown in Fig. 4, the parameter $c_{ij}$, a squashing function is applied for scaling the output probabilities in the range of 0 and 1 and...
3.1.3. CAM map visualization

By visualizing COVID-19 slices, we find the interpretability of the proposed capsule network. The most widely used method is GRAD-CAM (Selvaraju et al., 2017). The GRAD-CAM map takes input as an image using the following equation:

\[ f'(x) = U\text{p} \text{output} \left\{ \sigma \left( \sum_{M} a_{M}f^{M}(x) \right) \right\} \in I \]

(19)

Upsampling the input image \( m \times n \) with the feature vector \( u \times v \) produces a \( a \) that is defined as the ReLU layer. However, the probability is determined by:

\[ \left[ r_{p\text{output}}(x) \right]_{i} = \frac{1}{M} \left[ \sum_{M=1}^{M} \left( x_{M} \right) Q_{M} \left( f' \left( x_{M} \right) \right) \right] \]

(20)

where \( K \) is the slice of the each image \( x \) pixel, \( f' \left( x_{M} \right) \) computed the GRAD-CAM by using Eq. (19) with respect to frequency of the image. \( M \) is computed after the softmax layer of the capsule network. Eq. (20) shows the average probability of each pixel of the class for the global saliency map.

3.2. Architecture of gradients encryption & decryption

The data provider \( P \), which holds the private medical images \( I \), trains the local model and encrypts the local model vector. Then send it to the blockchain network \( B \). The blockchain federated learning model aggregates the encrypted vector using the global federated learning model. Moreover, the gradient encryption & decryption techniques for secure weight sharing were proposed by ElGamal (1985) based on the Ring-LWE scheme. Suppose \( \Phi_{\phi}(x) \) is the reducible polynomial function. The degree of the polynomial function \( \phi(n) \), \( R_{p} = R/pR \) and \( R = \mathbb{Z}[X]/(\Phi_{\phi}(X)) \) is the polynomial ring. The samples \((a, b = s \cdot a + e)\) of the Ring-LWE, where \( s, e \) indicates the Gaussian distribution. We utilized the (Hao et al., 2019) homomorphic scheme for the blockchain ledger.

We define a ciphertext and plaintext space. Ring \( R_{p} = \mathbb{Z}[X]/(\Phi_{\phi}(X)) \) defined as plaintext with the modulus \( p \). Similiarly, \( R_{q} = (\mathbb{Z}/q\mathbb{Z})[X]/(\Phi_{\phi}(X)) \) defined as ciphertext.

In the architecture of gradients encryption for external ciphertext:

1. Set \( g_{d_{1}} = Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{1}) \in R_{q} \).
2. Draw \( e_{0}, e_{1} \leftarrow \mathcal{G}(\phi(n), \sigma) \) and \( v \leftarrow \mathcal{G}(\phi(n)) \).
3. Compute \( \tilde{c}_{0} = \tilde{b} \cdot v + q \cdot \tilde{c}_{0} + g_{d_{1}} \) and \( \tilde{c}_{1} = \tilde{b} \cdot v + q \cdot e_{1} \) for modulus \( p_{1} \).
4. Output internal ciphertext \( \tilde{c}_{1} = (\tilde{c}_{0}, \tilde{c}_{1}) \in R_{p_{1}} \times R_{p_{1}} \).
5. Recover the sum of RBGV ciphertext by computing \( v = \sum_{i} v_{i} = G \cdot e \bmod p_{1} \in \mathbb{Z}_{p_{1}}^{(a)} \).
6. Split the vector \( v = (e_{0}, e_{1}) \in \mathbb{Z}_{p_{1}}^{(a)} \times \mathbb{Z}_{p_{1}}^{(a)} \).
7. Set \( \tilde{c}_{0} = Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{0}) \in R_{p_{1}} \) and \( \tilde{c}_{1} = Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{1}) \in R_{p_{1}} \).
8. Invoke algorithm \( \text{Scale} \left( \tilde{c}_{0}, \tilde{c}_{1}, p_{1} \right) \) to switch modulus and produce the scaled ciphertext \( \tilde{c}_{0}, \tilde{c}_{1} \) modulo \( p_{0} \).
9. Decrypt the ciphertext and produce the sum of plaintext by \( g_{d} = \sum_{i=1}^{q} g_{d_{i}} = \tilde{c}_{0} \cdot SK_{C_{1}}, \tilde{c}_{1} \bmod q \in R_{q} \).
10. Set \( g_{d'} = Map_{g_{d'}}^{Z_{\phi(n)}}(g_{d}) \).
11. Broadcast the global gradients \( g_{d'} \).

3.2.2. Gradients encryption

The following mappings are used during the encryption phase to connect the vector \( Z^{	ext{a}} \) and the ring \( R \) encryption phases:

1. \( Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{1}) : \) A coefficient representation of an input ring elements of \( n \) entities.
2. \( Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{1}) : \) A matrix over the same ring as the vector containing the coefficients representations of the vector.

3.2.3. The architecture of gradients encryption for external ciphertext

1. Set \( v_{i} = Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{i,1}) \in \mathbb{Z}_{p_{1}}^{(a)} \).
2. Invoke Algorithm 1 to sample \( e_{i} \in \mathbb{Z}_{p_{1}}^{(a)} \) subject to the distribution \( \mathcal{L}_{v}^{(a)}(G) \), where \( e_{i} = \text{sample} \left( v_{i,1}, \sigma \right) \).
3. Set \( e_{i} = \left( Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{i,1}) \right) \).
4. Compute \( c_{i} = a \cdot s_{i} + e_{i} \in R_{p_{1}} \).
5. Send final ciphertext \( c_{i} \) to the blockchain network.
6. Aggregate all the ciphertexts \( e = \sum_{i} c_{i} = a \cdot s_{i} + \sum_{i} e_{i} \in R_{p_{1}} \).

3.2.4. The architecture of gradients encryption for inner ciphertext

1. Set \( g_{d_{1}} = Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{1}) \in R_{q} \).
2. Draw \( e_{0}, e_{1} \leftarrow \mathcal{G}(\phi(n), \sigma) \) and \( v \leftarrow \mathcal{G}(\phi(n)) \).
3. Compute \( \tilde{c}_{0} = \tilde{b} \cdot v + q \cdot \tilde{c}_{0} + g_{d_{1}} \) and \( \tilde{c}_{1} = \tilde{b} \cdot v + q \cdot e_{1} \) for modulus \( p_{1} \).
4. Output internal ciphertext \( \tilde{c}_{1} = (\tilde{c}_{0}, \tilde{c}_{1}) \in R_{p_{1}} \times R_{p_{1}} \).
5. Recover the sum of RBGV ciphertext by computing \( v = \sum_{i} v_{i} = G \cdot e \bmod p_{1} \in \mathbb{Z}_{p_{1}}^{(a)} \).
6. Split the vector \( v = (e_{0}, e_{1}) \in \mathbb{Z}_{p_{1}}^{(a)} \times \mathbb{Z}_{p_{1}}^{(a)} \).
7. Set \( \tilde{c}_{0} = Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{0}) \in R_{p_{1}} \) and \( \tilde{c}_{1} = Map_{g_{d_{1}}}^{Z_{\phi(n)}}(c_{1}) \in R_{p_{1}} \).
8. Invoke algorithm \( \text{Scale} \left( \tilde{c}_{0}, \tilde{c}_{1}, p_{1} \right) \) to switch modulus and produce the scaled ciphertext \( \tilde{c}_{0}, \tilde{c}_{1} \) modulo \( p_{0} \).
9. Decrypt the ciphertext and produce the sum of plaintext by \( g_{d} = \sum_{i=1}^{q} g_{d_{i}} = \tilde{c}_{0} \cdot SK_{C_{1}}, \tilde{c}_{1} \bmod q \in R_{q} \).
10. Set \( g_{d'} = Map_{g_{d'}}^{Z_{\phi(n)}}(g_{d}) \).
11. Broadcast the global gradients \( g_{d'} \).

3.3. Consent in permissioned blockchain federated learning

The main goal of this section is to exaggerate the global model with the blockchain DAG mechanism. The local DAG is responsible for synchronous global training via federated learning. Consequently, the storage capability of the model by using DAG is improved. Based on the federated learning and permissioned blockchain, the following steps are taken to adjust the decentralized model for aggregation. Firstly, we select the users’ nodes and then perform local training and encrypt the weights. Then, we aggregate the weights in the global model. The consensus (i.e., POW) for data sharing is high cost. To address the problem, we proposed a hybrid DAG-based scheme that is provided in Algorithm 2. We combine the update weight process of federated learning with the quality verification process using the blockchain DAG. Algorithm 12 shows the global aggregation of the model gradients for federated learning.
Algorithm 1: Global Federated Learning aggregation algorithm.

1. \( \theta_0 \) of global model;
2. \( \left\{ G_{i(t)} \right\}_{i=1}^n \) — local gradient vectors;
3. \( \theta_{global} \leftarrow 0; \)
4. \( l \leftarrow 0; \)
5. for \( k=1,2,...,m \) do
6.  
7.  
8.  
9.  
10.  
11.  
12. end

Fig. 5. Communication graph.

3.3.3. Confirmation and consensus

The transactions are confirmed or validated based on the cumulative weights. This article utilizes the weighted walk method based on credibility, which can validate the transaction by selecting the unverified transactions. When a new transaction is generated, two walkers will be added to the blockchain DAG to select the transaction. More transaction has been passed for verification to achieve a higher cumulative weight for verification.

\[
P_{xy} = \sum_{z \in \mathbb{Z}} e^{CW(x)-CW(z)}
\]

where \( P_{xy} \) is the transition probability towards the unverified transaction \( x \) and \( y \). \( z \) is the neighboring node of a transaction belonging to \( x \) and \( y \). In this approach, the PoW is faster than a traditional PoW because of the reduction in complexity.

Algorithm 2: Federated Learning Empowered with Blockchain Network

1. \( D_1 \leftarrow \{M_1,m_2,\ldots,m_k\} \) dataset;
2. \( \theta_0 \leftarrow \) Initialize global weights with the permissioned blockchain BC and DAG;
3. \( r_0 \leftarrow \) select the users to \( M_p \in M_1 \) by the node selection \( \{r_1, r_2, \ldots, r_k\} \);
4. for \( t \in \) [episode] do
5.  
6.  
7.  
8.  
9.  
10.  
11.  
12. end
13. end
14. end
15. \( r_0 \leftarrow \frac{\sum C_{W}(m_p)}{\sum C_{W}} \) DAG blockchain updated the model, and averaging the models into \( M(r_0) \);
16. \( r_0 \) broadcasts model \( M(r) \) to other nodes for verification, and add all the transactions into the blockchain ledger; \( r_0 \) include the \( M(r) \) global model form the blockchain ledger;

4. Security and performance analysis

4.1. Dataset

We collected the dataset from five different hospitals with various types of CT scanners. The total number of patients was 170 and 6 different CT scanners from Chengdu city, Sichuan Province, China. In addition, to validate the proposed method, we combine the open-source dataset with the collected data. All the patients conformed to the antibody tests or nucleic acid tests. However, this research collects the dataset from different sources due to federated learning and blockchain. One dataset is collected from CC-19 Dataset (https://github.com/abdikhanstc/Covid-19) which contains 170 patients from 6 hospitals shown in Table 4 and another dataset is downloaded from Dataverse HARVARD repository to validate the model(see Table 2).

4.2. Security analysis

The use of permissioned blockchain distributed technology achieved a secure mechanism for the various devices. We integrate the consensus blockchain process with federated learning to address the trust of the security threats and privacy of the data.
Table 2

| CT scanner ID | A   | B   | C   | D   | E   | F   |
|-------------|-----|-----|-----|-----|-----|-----|
| CT scanner  | 1   | 2   | 3   | 4   | 5   | 6   |
| iCT         | Philips | Siemens | Philips | Philips | Siemens | Siemens |
| Definitanion Edge | 16P iCT | CT scanner | scope | scope | scope | scope |
| Number of patients | 17 | 3 | 5 | 55 | 50 | 10 |
| Number of slices | 128 | 16 | 16 | 120 | 256 | 64 |
| Matrix | 512*512 | 512*512 | 512*512 | 512*512 | 512*512 | 512*512 |
| Tube voltage (K vp) | 100 | 140 | 120 | 120 | 120 | 120 |
| Lung window width (HU) | 128*0.6 | 128*0.62 | 128*0.62 | 500 | 550 | 550 |
| Pitch | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| Slice thickness (mm) | 2 | 2 | 5 | 5 | 5 | 5 |
| Slice increment (mm) | 2 | 2 | 5 | 5 | 5 | 5 |
| Tube voltage (K vp) | 140 | 120 | 110 | 120 | 120 | 120 |
| Infection annotation | ROI - level | ROI - level | Voxel-level | Voxel-level | Voxel-level | Voxel-level |

1. **Differential Privacy:** According to the privacy of users, our proposed protocol is used to generate indistinguishable random values. We select the random vector for the generation of the ciphertext \( \tilde{c} \) using the BGV scheme (Brakerski et al., 2014). Where \( K \) is an indistinguishable security parameter for the random values. Then \( v_i \) is transferred from the polynomial \( \tilde{c} \) (for random values).

\[
\Pr[F(x) = S] \leq e^\epsilon
\]

A function that satisfies differential privacy is often called a mechanism. We say that a mechanism \( F \) satisfies differential privacy if all neighboring datasets \( x \) and \( x' \) have possible outputs \( S \).

2. **Data Access:** The proposed technique uses federated learning with blockchain technology. The core idea is to develop the privacy of the data. The proposed model achieves data privacy by aggregating the encrypted technique with blockchain, which grants the privacy protection of the data.

3. **Trust:** To aggregate the sum of weights, the blockchain, and the local model client provide the security as follows:

3.1 **Setup:** In the first step, the security algorithm generates the public parameters for the model.

3.2 **Encrypt:** client specify the parameter \((i, m)\), where \(i\) is the index of the entity and \(m\) is the plaintext. Finally, it returns the \( Enc(i, m) \) value to the model.

3.3 **Compromise:** The model comprises an \( i \) entity, then the aggregate model returns the secret keys \( SK_{\tilde{c}} \), this phase repeat many times.

3.4 **Challenge:** It is allowed only once throughout the entire cycle. Generate and send two plain text messages, \( m_1 \) and \( m_2 \), for every \( i \in K \). If bit is equal to zero, compute \( c_i = Enc(m_i) \). Otherwise, it will be encrypted in the same way and sent as \( c_i \).

3.5 **Guess:** The final output is 1 or 0

4. **Removing Centralized Trust:** It removes third-party trust and allows users or hospitals to comment on the network.

5. **Secure Data Management:** To ensure the model’s reliability, only the trusted data provider uploads the data to the network. Also, the cryptography algorithm ensures the security of data.

6. **Guarantee Quality Model:** To ensure the quality of the model, the consensus process guarantees the quality of the learned data.

4.3. **Performance analysis**

To evaluate the proposed methods’ performance, we adopted the federated learning model as a classifier to conduct the experimentation.

We analyze and evaluate our model in terms of accuracy. The deep learning model contains fully connected convolutional layers where each of the layers consists of 128 neurons. Two factors affect the accuracy and running time of the federated learning model i.e., the number of hospitals and gradients per hospital. We examined both factors for different values ranges as shown in Fig. 6 and Fig. 7 respectively. It shows the execution time and accuracy with a different number of iterations. Here, the number of iterations indicates the update of parameters. We compared the effect with different numbers of gradients per hospital, and we distributed data over six hospitals. We assume no user has dropped out to conduct the experimentation in the basic setting. It can be seen that increasing the number of gradients per hospital leads to higher performance. Whereas, it leads to a computation overhead as shown in Fig. 6(b). Therefore, to reduce the computation overhead in a practical environment, an appropriate number of gradients can be empirically chosen. In terms of model iterations, it can be observed that model accuracy converges after a certain number of iterations.

The required time to train the local model (local gradients) also depends on the size of the data and the number of selected hospitals. We analyzed the accuracy of different users to train the model. The classification accuracy and execution time can be seen in Fig. 7. Similar to the previous observation, naturally increasing numbers of iterations and hospitals consume high computation costs. However, due to independent gradient computation for each user, the number of hospitals leads to high accuracy. The data is split into many chunks as per the hospital. Therefore, the local gradient for A will be calculated and combined to produce high accuracy.

4.4. **Local model analysis**

In this section, we analyze the local deep learning models, which are divided into three parts. (i) Segmentation, (ii) Classification, and (iii) Visualization of the Attention Map.

4.4.1. **Segmentation network results**

Capsule network-based localization of the lung’s COVID-19 region is shown in Fig. 8. We extracted the region of the lung of COVID-19 patients. We fix the parameters of the blockchain-based federated learning where total communication costs \( T = 300 \) and validate each model in every round to select the best local model from the blockchain nodes. Moreover, we set the Adam optimizer learning rate at 0.0001. Each round contains 300 iterations with a batch size of 4. Table 3 shows the federated learning model for the five hospitals. The first three rows show the hospitals (I/II/III). We compute the average of five hospitals’ accuracy in the “global test average”. This measure shows the global model and blockchain nodes as the major metrics for performance evaluation. Additionally, Table 5 compares the segmentation performance stage-wise.
4.4.2. Comparison of the global and local model

This paper presents results from global and local deep learning models i.e., Local I, Local II, Local III, Fed AVG, FedGlobal, and FedProxy. We compared the performance and adopted deep learning models with different layers. Moreover, Fig. 9 shows the performance comparison concerning the segmentation task. Additionally, we evaluate the performance comparison of the capsule network concerning accuracy. Fig. 9 demonstrates the local and global models; the global model achieves high-level detection performance through the network.

4.4.3. Visualizations of the attention map regions

For a better understanding, we computed the probabilistic CAM for each CT image of COVID-19. The capsule network visualizes the patient’s CT images from the normal and COVID-19 classes, and a
Table 3
COVID-19 lesion segmentation. The global test average shows the Federated Learning global model. n spices the number of patients.

| Parameters | Local - I | Local - II | Local III | FedAvg | FedAvg - Blockchain | FedProx |
|------------|-----------|------------|-----------|--------|----------------------|---------|
| I (n = 40) | 80.2      | 64.12      | 57.0      | 82.13  | 78.93                | 82.53   |
| II (n = 20) | 84.02     | 82.15      | 74.74     | 85.99  | 86.51                | 87.18   |
| III (n = 17) | 74.00    | 72.38      | 88.05     | 82.72  | 87.18                | 82.65   |
| Global test avg | 85.99    | 82.15      | 73.16     | 83.61 ± 0.18 | 84.21 ± 0.43   | 84.12 ± 0.58 |
| Local avg  | 84.07     | 84.67      | 84.44     | 61.99  |
| Local gen  | 70.99     | 81.0       | 81.48     | 80.53  |

Table 4
Federated learning segmentation performance at early, progressive, and severe stages.

| Method                  | Early (75%) | Progressive (15%) | Severe (10%) | RMSE | Recall | Dice | Worst-case |
|-------------------------|-------------|-------------------|--------------|------|--------|------|------------|
| FedAvg - Blockchain     | 0.895       | 0.925             | 0.943        | 0.025| 0.789  | 0.795| 0.577      |
| FedAvg                  | 0.769       | 0.896             | 0.882        | 0.082| 0.802  | 0.573| 0.125      |
| FedProx                 | 0.799       | 0.912             | 0.924        | 0.028| 0.702  | 0.692| 0.032      |
| DeeplabV3               | 0.726       | 0.820             | 0.868        | 0.048| 0.759  | 0.896| 0.175      |
| UNet                    | 0.758       | 0.805             | 0.855        | 0.076| 0.625  | 0.459| 0.087      |

Table 5
Federated learning time and memory consumption details.

| Method                  | Prediction (s) | Training (h) |
|-------------------------|----------------|--------------|
| FedAvg - Blockchain     | 12             | 5.5          |
| FedAvg                  | 13             | 5.4          |
| FedProx                 | 15             | 6            |
| DeeplabV3               | 12             | 3.5          |
| UNet                    | 10             | 2            |

Table 6
Ablation study on the effect of number of users dropout on our proposed blockchain-based federated learning (FedAvg-Blockchain).

| Dropout no. | Early (75%) | Progressive (15%) | Severe (10%) | RMSE | Recall | Dice |
|-------------|-------------|-------------------|--------------|------|--------|------|
| 1           | 0.882       | 0.912             | 0.935        | 0.039| 0.780  | 0.776|
| 2           | 0.851       | 0.901             | 0.918        | 0.080| 0.703  | 0.734|
| 3           | 0.823       | 0.870             | 0.881        | 0.124| 0.680  | 0.690|

Table 7
Tradeoff between global average accuracy and privacy.

| Privacy budget (c) | Global test avg |
|--------------------|-----------------|
| 0.10               | 54 ± 0.75       |
| 0.20               | 60 ± 0.92       |
| 0.30               | 73 ± 0.02       |
| 0.40               | 76 ± 0.83       |
| 0.50               | 84 ± 0.21       |

noticeable activation map is shown in Fig. 10. Moreover, the applied CAM (LaLonde and Bagci, 2018; Liao et al., 2019) function visualizes each image slice.

4.5. Trade off between accuracy and privacy

For this experiment, we measure the global test average of the proposed model corresponding to different privacy budgets, and in each case, we set the sensitivity of the additional noise added to the model’s weights as the optimal probability that increases the privacy of the framework. Here, the privacy budget indicates the overall end-to-end privacy loss for the participating user and smaller implies higher privacy. As shown in Table 7, by increasing the privacy budget $\epsilon < 1$ value from 0.10 to 0.50, the corresponding test average keeps increasing, matching the intuition that higher privacy corresponds to lower accuracy.
privacy and creates trust in the data training process. However, the help build an intelligent model without leakage of the data providers’ both local model training and secure global model training. We secure multiple hospitals for the internet of things applications that includes 5. Conclusion training, the performance reduction is insignificant.

reduction in performance as the number of users dropped out from the federated learning round, it indirectly decreases the total number of global model (Sattler et al., 2019). Also, when users drop out from the likely due to the divergence of weights of the local models from the consistent with the literature. The reason for the performance drop is method reduces as the number of drop-out users increases, which is In Table 6, it could be realized that the performance of our proposed model (as we can observe), the capsule network achieved 98% accuracy in detecting the COVID-19 CT scans. Although Han et al. also achieve 98% accuracy, they do not consider the data sharing techniques. Furthermore, we compare our scheme with the security analysis shown in Table 8. Moreover, Bonawitz et al. (2017) adopted federated learning and proposed a framework to secure the aggregation of gradients. Whereas, Zhang et al. (2017) introduced the scheme of homomorphic encryption (HE) and threshold secret sharing to secure the gradients. The main problem with sharing the model is uncertainty about user authenticity. In other words, there is still a lack of trust among various groups. Thus, the proposed approach bridges this gap and achieves the desired result of trust between parties.

4.6. Comparison with other methods

To prove the local models’ accuracy and effectiveness, the proposed model reduces the number of drop-out users increases, which is consistent with the literature. The reason for the performance drop is likely due to the divergence of weights of the local models from the global model (Sattler et al., 2019). Also, when users drop out from the federated learning round, it indirectly decreases the total number of data instances to aid the training process. Even though there was a reduction in performance as the number of users dropped out from the training, the performance reduction is insignificant.

4.7. Ablation study

We further performed additional experiments to ascertain the effect of user dropout for our proposed federated blockchain framework for medical images. Users participating in the federated learning task may drop from the federated learning systems at any time due to several reasons, such as low device battery, poor connectivity, etc., which may affect the performance of the model. For each dropout setting, we randomly drop out \( n \in \{1, 2, 3\} \) number of users for each training round. In Table 6, it could be realized that the performance of our proposed method reduces as the number of drop-out users increases, which is consistent with the literature. The reason for the performance drop is likely due to the divergence of weights of the local models from the global model (Sattler et al., 2019). Also, when users drop out from the federated learning round, it indirectly decreases the total number of data instances to aid the training process. Even though there was a reduction in performance as the number of users dropped out from the training, the performance reduction is insignificant.

5. Conclusion

This paper proposes a secure data sharing scheme for distributed multiple hospitals for the internet of things applications that includes both local model training and secure global model training. We secure the local model through the homomorphic encryption scheme which helps build an intelligent model without leakage of the data providers’ privacy and creates trust in the data training process. However, the blockchain-based algorithm aggregates the local model updates and provides the authentication of the data. The experiment results confirm the performance and effectiveness of the model. In future work, we aim to enhance the latency of the blockchain and provide a more cost-effective solution.

CRediT authorship contribution statement

Rajesh Kumar: Writing – original draft, Conceptualization, Methodology. Jay Kumar: Data curation, Visualization. Abdullah Aman Khan: Data curation, Visualization. Zakria: Draft, Review, Validation. Hub Ali: Formal analysis. Cobbinah M. Bernard: Software, Supervision. Riaz Ullah Khan: Writing – review & editing, Validation. Shaoning Zeng: Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant U2033212.

References

Aono, Y., Hayashi, T., Wang, L., Moriai, S., et al., 2017. Privacy-preserving deep learning via additively homomorphic encryption. IEEE Trans. Inf. Forensics Secur. 13 (5), 1333–1345.

Baheti, P., Sikka, M., Arya, K.V., Rajesh, R., 2020. Federated learning on distributed medical records for detection of lung nodules. In: Farinella, G.M., Radeva, P., Braz, J. (Eds.), Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISIGRAPP 2020. Volume 4: VISAPP, Valletta, Malta, February 27-29, 2020. SCITEPRESS, pp. 445–451.

Blanquer, I., Brasilheiro, F.V., Brito, A., Calatrava, A., Carvalho, A., Figueiredo, F., Guimaraes, R.P., Marinho, L.B., Jr., W.M., da Silva, A.S., Alberich-Bayarri, A., Camacho-Ramos, E., Jimenez-Pastor, A., Ribeiro, A.L.L., Nascimento, B.R., Silva, F., 2020. Federated and secure cloud services for building medical image classifiers on an intercontinental infrastructure. Future Gener. Comput. Synt. 110, 119–134.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H.B., Patel, S., Ramage, D., Segal, A., Seth, K., 2017. Practical secure aggregation for privacy-preserving machine learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 1175–1191.

Bottou, L., 2010. Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT’2010. Springer, pp. 177–186.

Brakerski, Z., Gentry, C., Vaikuntanathan, V., 2014. (Leveled) fully homomorphic encryption without bootstrapping. ACM Trans. Comput. Theory (TOCT) 6 (3), 1–36.

Brizim, T.S., Shen, R., Mela, T., Ohbeyrak, A., Paschalidis, I.C., Shi, W., 2018. Federated learning of predictive models from federated electronic health records. Int. J. Med. Inf. 112, 59–67.

Can, Y.S., Ersoy, C., 2021. Privacy-preserving federated deep learning for wearable IoT-based biomedical monitoring. ACM Trans. Internet Technol. 21 (1), 21:1–21:17.

Cheng, Y., Liu, Y., Chen, T., Yang, Q., 2020. Federated learning for privacy-preserving AI. Commun. ACM 63 (12), 33–36.

Dai, H.-N., Zheng, Z., Zhang, Y., 2019. Blockchain for internet of things: A survey. IEEE IoT J. 6 (5), 8076–8094.

Das, N.N., Kumar, N., Kaur, M., Kumar, V., Singh, D., 2020. Automated deep transfer learning-based approach for detection of COVID-19 infection in chest X-rays. Irbm.

Table 8

Comparison with the security analysis. Furthermore, DA represents data authentication, P/E represents Privacy/Encryption of data, DaA represents Data Access and CeT represents Centralized Trust.

| Study          | Blockchain | Server | DA | P/E | DaA | CeT |
|---------------|------------|--------|----|-----|-----|-----|
| Ours          | Yes        | No     | Yes| Yes | Yes | Yes |
| Kumar et al. (2021a)| Yes | No | Yes | No | Yes | Yes |
| Kim et al. (2019) | Yes | No | Yes | No | Yes | Yes |
| Lu et al. (2020b) | Yes | No | Yes | No | Yes | Yes |
| Lu et al. (2020a) | Yes | No | Yes | No | Yes | Yes |
| Xu et al. (2019) | No | Yes | No | Yes | No | No |
| Yang et al. (2014) | No | Yes | No | Yes | No | No |

Fig. 10. Visualizations of the attention map regions.
Malekzadeh, M., Hasircioglu, B., Mital, N., Katarya, K., Ozfatura, M.E., Gündüz, D., Lu, Y., Huang, X., Zhang, K., Maharjan, S., Zhang, Y., 2020b. Blockchain empowered federated learning for industrial artificial intelligence. IEEE Trans. Ind. Inf. 16 (10), 6532–6542.

Hamitazdehadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H., Xu, D., 2022. Swin UNETR: Swin transformer for semantic segmentation of breast tumors in MRI images. arXiv preprint arXiv:2201.01266.

Dinh, C.T., Tran, N.H., Nguyen, M.N.H., Hong, C.S., Bao, W., Zomaya, A.Y., Gramoli, V., 2021. Federated learning over wireless networks: Convergence analysis and resource allocation. IEEE/ACM Trans. Netw. 29 (1), 398–409.

ElGamal, T., 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inform. Theory 31 (4), 469–472.

Hao, M., Li, H., Luo, X., Xu, G., Yang, H., Liu, S., 2019. Efficient and privacy-enhanced federated learning. IEEE Trans. Comput. 68 (11), 3484–3495.

Dinh, C.T., Tran, N.H., Nguyen, M.N.H., Hong, C.S., Bao, W., Zomaya, A.Y., Gramoli, V., 2021. Federated learning over wireless networks: Convergence analysis and resource allocation. IEEE/ACM Trans. Netw. 29 (1), 398–409.

ElGamal, T., 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inform. Theory 31 (4), 469–472.

Hao, M., Li, H., Luo, X., Xu, G., Yang, H., Liu, S., 2019. Efficient and privacy-enhanced federated learning. IEEE Trans. Comput. 68 (11), 3484–3495.

Dinh, C.T., Tran, N.H., Nguyen, M.N.H., Hong, C.S., Bao, W., Zomaya, A.Y., Gramoli, V., 2021. Federated learning over wireless networks: Convergence analysis and resource allocation. IEEE/ACM Trans. Netw. 29 (1), 398–409.

ElGamal, T., 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inform. Theory 31 (4), 469–472.

Hao, M., Li, H., Luo, X., Xu, G., Yang, H., Liu, S., 2019. Efficient and privacy-enhanced federated learning. IEEE Trans. Comput. 68 (11), 3484–3495.

Dinh, C.T., Tran, N.H., Nguyen, M.N.H., Hong, C.S., Bao, W., Zomaya, A.Y., Gramoli, V., 2021. Federated learning over wireless networks: Convergence analysis and resource allocation. IEEE/ACM Trans. Netw. 29 (1), 398–409.

ElGamal, T., 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inform. Theory 31 (4), 469–472.

Hao, M., Li, H., Luo, X., Xu, G., Yang, H., Liu, S., 2019. Efficient and privacy-enhanced federated learning. IEEE Trans. Comput. 68 (11), 3484–3495.