Enhanced Security with Encrypted Vision Transformer in Federated Learning

1st Aso Rei *Tokyo Metropolitan Univercity* Tokyo, Japan aso-rei@ed.tmu.ac.jp 2nd Shiota Sayaka Tokyo Metropolitan Univercity Tokyo, Japan shiota@tmu.ac.jp 3rd Kiya Hitoshi *Tokyo Metropolitan Univercity* Tokyo, Japan kiya@tmu.ac.jp

Abstract—Federated learning is a learning method for training models over multiple participants without directly sharing their raw data, and it has been expected to be a privacy protection method for training data. In contrast, attack methods have been studied to restore learning data from model information shared with clients, so enhanced security against attacks has become an urgent problem. Accordingly, in this article, we propose a novel framework of federated learning on the bases of the embedded structure of the vision transformer by using the model information encrypted with a random sequence. In image classification experiments, we verify the effectiveness of the proposed method on the CIFAR-10 dataset in terms of classification accuracy and robustness against attacks.

Index Terms—Federated Learning, Vision Transformer, Privacy Preserving

I. INTRODUCTION

Deep neural networks (DNNs) have been deployed in various applications. Training high-performance DNN models requires a huge amount of training data, and training data include sensitive information such as personal information in general. Accordingly, it is difficult to prepare an amount of data to train DNN models, so privacy-preserving methods for deep learning have become an urgent problem [1], [2]. Federated learning (FL) has been excepted as one of the solutions [3]. FL is a type of distributed machine learning. It is a model learning method that reflects all clients' data by sharing only the updated information of each local model without directly sharing each client's training data. However, it has been pointed out to be vulnerable to state-of-the-attacks [4]–[6]. In particular, vision transformer (ViT) models [7], which are known to have a high performance, are highly vulnerable as discussed in [6].

Therefore, various privacy-preserving methods have been proposed to enhance security in FL so far. Differential privacy [8] is one of the state-of-the-art, in which the values of model parameters are hidden by adding noise with a specific distribution. However, there is a trade-off relation between the level of privacy protection and model performance, so if we want to strongly protect model parameters, the use of differential privacy degrades the performance of models.

Accordingly, in this paper, we propose a novel framework for enhancing the security of ViT models in FL. In the proposed framework, focusing on the embedding structure of ViT, each updated local model is encrypted by using a random matrix generated with a secret key, which has been inspired by privacy-preserving deep learning with encrypted images for ViT [9]–[12]. Encrypted local-model information is extracted from each encrypted model, and the encrypted local-model information is shared with clients to perform model integration directly in the encrypted domain. In experiments, the proposed method is demonstrated not only to maintain the same accuracy as that of FL without any encryption but to also enhance robustness against an attack called Attention Privacy Leakage (APRIL) [6], which aims to restore the visual information.

II. RELATED WORK

A. Federated Learning

Before discussing the proposed method, we summarize the general procedure of FL.

- i) A server provider distributes an initial global model to all clients.
- ii) Each client updates the global model with their training data.
- iii) Each client sends the updated model to the server.
- iv) The server provider integrates the model information received from all clients and updates the global model.
- v) The server provider sends the updated global model to all clients.
- vi) Repeat ii) to v) multiple times.

FedAvg (Federated Averaging) [3] and FedSGD (Federated Stochastic Gradient Descent) [3] are typical methods for updating global models. In FedAvg, each client computes the gradient of an image and updates the local model with the gradient. After that, the client sends the parameters of the local model to the server. The server integrates the model parameters from clients to update the global model. In FedSGD, the client computes gradients from an image and sends them to the server. The server updates the model based on the gradients from each client. The proposed method can be adapted to both methods.

Identify applicable funding agency here. If none, delete this.



B. Vision Transformer

ViT is mainly used for image classification tasks and is known to have high classification performance [7]. As shown in Figure 1, it consists of three components: embedding layer, transformer encoder, and MLP (Multi-Layer Perceptron) Head. In this paper, we focus on the embedding layer, which is a layer for converting an image into a feature vector.

An input image $x \in \mathbb{R}^{H \times W \times C}$ is first divided into patches with a size of $P \times P$ where H, W, and C are the height, width, and number of channels of the image. The number of patches N is given as $N = (W/P) \times (H/P)$ as an integer. After that, each patch is flattened as $x_p^i = [x_p^i(1), x_p^i(2), ..., x_p^i(L)]$, where $L = P^2C$. Finally, a sequence of embedded patches is given as

$$Z_0 = [x_{\text{class}}; x_p^1 E_{\text{pat}}; \cdots x_p^i E_{\text{pat}}; \cdots x_p^N E_{\text{pat}}] + E_{\text{pos}} \quad (1)$$

where,

$$E_{pos} = \left(\left(e_{pos}^{0} \right)^{T} \cdots \left(e_{pos}^{i} \right)^{T} \cdots \left(e_{pos}^{N} \right)^{T} \right)^{I}$$
$$x_{class} \in \mathbb{R}^{D}, x_{p}^{i} \in \mathbb{R}^{L}, e_{pps}^{i} \in \mathbb{R}^{D},$$
$$E \in \mathbb{R}^{L \times D}, E_{pos} \in \mathbb{R}^{(N+1) \times D}.$$

 x_{class} is the classification token, E_{pat} is the embedding (patch embedding) to linearly map each patch to dimensions D, E_{pos} is the embedding (position embedding) that gives position information to patches in the image, e_{pos}^0 is the information of the classification token, and e_{pos}^i , i = 1, ..., N, is the position information of each patch.

III. PROPOSED METHOD

A. Overview

The proposed method aims to prevent the visual information of plain training images from being restored from model parameters sent from each client to the server. Figure 2 shows the framework of the method, where we assume that ViT is used and a secret key for encryption is shared with all clients. The procedure of the method is summarized below.

 A server provider distributes an initial global model to all clients.



Fig. 2. Framework of proposed method

- Each client updates the global model with their training data.
- 3 Each client encrypts the parameters of the updated model with a common key and sends it to the server.
- ④ The server provider integrates the model information received from clients and updates the global model in the encrypted domain.
- (5) The server provider sends the updated global model to all clients.
- 6 Each client decrypts the global model with a common key.
- 7 Repeat 2 to 6 multiple times.

In this framework, a malicious external third party and cloud provider (untrusted) have no secret key, so they cannot restore training data from updated model information sent from clients. The main contribution of this paper is to propose a method that allows us to update a global model in the encrypted domain for the first time.

B. Model Encryption

Model encryption is carried out in (3). In the method, patch embedding E_{pat} and position embedding E_{pos} in Eq.(1) are encrypted by using random matrices, respectively.

1) Patch Embedding Encryption: The following transformation matrix E_a is used to encrypt patch embedding E_{pat} .

$$E_{a} = \begin{pmatrix} k_{(1,1)} & k_{(1,2)} & \cdots & k_{(1,L)} \\ k_{(2,1)} & k_{(2,2)} & \cdots & k_{(2,L)} \\ \vdots & \vdots & k_{(i,j)} & \vdots \\ k_{(L,1)} & k_{(L,2)} & \cdots & k_{(L,L)} \end{pmatrix},$$
(2)

where

$$E_{\mathbf{a}} \in \mathbb{R}^{L \times L}, \det E_{\mathbf{a}} \neq 0,$$

 $k(i, j) \in \mathbb{R}, i, j \in \{1, ..., L\}.$

Note that the element values of E_a are randomly decided, but E_a has to have an inverse matrix.

Then, by multiplying E_{pat} by E_{a} , an encrypted patch embedding \hat{E}_{pat} is given as

$$\widehat{E}_{\text{pat}} = E_{\text{a}} E_{\text{pat}}.$$
(3)

2) Position Embedding Encryption: Position Embedding E_{pos} is encrypted as below.

1 Generate a random integer vector with a length of N as

$$l_t = [l_e(1), l_e(2), \dots, l_e(i), \dots, l_e(N)], \quad (4)$$

where

$$l_e(i) \in \{1, 2, ..., N\},\$$

$$l_e(i) \neq l_e(j) \quad if \quad i \neq j,\$$

$$i, j \in \{1, 2, ..., N\}.$$

2 Calculate $m_{(i,j)}$ as

$$m_{(i,j)} = \begin{cases} 1 & (j = l_e(i)) \\ 0 & (j \neq l_e(i)). \end{cases}$$
(5)

3 Define a random matrix as

$$E_{b} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & m_{(1,1)} & m_{(1,2)} & \cdots & m_{(1,N)} \\ 0 & m_{(2,1)} & m_{(2,2)} & \cdots & m_{(2,N)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & m_{(N,1)} & m_{(N,2)} & \cdots & m_{(N,N)} \end{pmatrix}$$
(6)

where

$$E_{\mathbf{h}} \in \mathbb{R}^{(N+1) \times (N+1)}$$

4 Transform E_{pos} to \hat{E}_{pos} as

$$\widehat{E}_{\rm pos} = E_{\rm b} E_{\rm pos}.\tag{7}$$

C. Global Model Update

The cloud server updates the global model by using the model information received from each client.

For example, when using FedSGD, a global model is updated below.

Let M be the number of clients, $W^{(t)}$ be the parameters of the global model after t updates, $\theta_i^{(t)}$ be the model update information (gradients) computed by client i and τ be the learning rate. In this case, the global model is updated as follows;

$$W^{(t+1)} = W^{(t)} - \tau \frac{1}{M} \sum_{i=1}^{M} \theta_i^{(t)}.$$
(8)

Since the model is updated independently in each layer, model parameters in patch embedding and position embedding can be updated from Eq.(8) as

$$W_{\text{pat}}^{(t+1)} = W_{\text{pat}}^{(t)} - \tau \frac{1}{M} \sum_{i=1}^{M} E_{\text{pat},i}^{(t)} \quad , \tag{9}$$

$$W_{\rm pos}^{(t+1)} = W_{\rm pos}^{(t)} - \tau \frac{1}{M} \sum_{i=1}^{M} E_{{\rm pos},i}^{(t)} .$$
 (10)

 $E_{{\rm pat},i}^{(t)}$ and $E_{{\rm pos},i}^{(t)}$ are the parameters of patch and position embeddings updated by client i.

According to Eqs.(3) and (9), the parameters of patch embedding are updated as

$$\widehat{W}_{\text{pat}}^{(t+1)} = \widehat{W}_{\text{pat}}^{(t)} - \tau \frac{1}{M} \sum_{i=1}^{M} \widehat{E}_{\text{pat},i}^{(t)}$$
$$= E_{\text{a}} \left(W_{\text{pat}}^{(t)} - \tau \frac{1}{M} \sum_{i=1}^{M} E_{\text{pat},i}^{(t)} \right)$$
$$= E_{\text{a}} W_{\text{pat}}^{(t+1)}.$$
(11)

According to Eqs.(7) and (10), the parameters of position embedding are updated as

$$\widehat{W}_{\text{pos}}^{(t+1)} = \widehat{W}_{\text{pos}}^{(t)} - \tau \frac{1}{M} \sum_{i=1}^{M} \widehat{E}_{\text{pos},i}^{(t)}$$
$$= E_{\text{b}} \left(W_{\text{pos}}^{(t)} - \tau \frac{1}{M} \sum_{i=1}^{M} E_{\text{pos},i}^{(t)} \right)$$
$$= E_{\text{b}} W_{\text{pos}}^{(t+1)}.$$
(12)

Eqs.(9) and (10) show that the global model on the cloud server can be updated in the encrypted domain.

D. Model Decryption

An encrypted global model is decrypted by each client as shown in ⁽⁶⁾.

To decrypt patch embedding, $\widehat{W}_{\text{pat}}^{(t+1)}$ is multiplied by the inverse of matrix E_{a} used for model encryption as

$$E_{\rm a}^{-1}\widehat{W}_{\rm pat}^{(t+1)} = E_{\rm a}^{-1}E_{\rm a}W_{\rm pat}^{(t+1)} = W_{\rm pat}^{(t+1)}.$$
 (13)

In contrast, to decrypt position embedding, $\widehat{W}_{\text{pos}}^{(t+1)}$ is multiplied by the inverse of matrix E_{b} used for model encryption as

$$E_{\rm b}^{-1}\widehat{W}_{\rm pos}^{(t+1)} = E_{\rm b}^{-1}E_{\rm b}W_{\rm pos}^{(t+1)} = W_{\rm pos}^{(t+1)}$$
(14)

From the above equations, we see that the updated parameters of the unencrypted model can be obtained from the updated parameters of the encrypted model. Accordingly, the proposed method is verified to obtain the same parameters as those of models trained without any encryption.

IV. EXPERIMENT RESULTS

A virtual server and five clients were set up on a single machine to verify the effectiveness of the proposed FL. All experiments were carried on an open source framework called Flower [13], and FedSGD [3] was used as an aggregation algorithm. The CIFAR10 dataset, which consists of 50,000 training and 10,000 test color images with a seize of 32×32 , was also used to fine-tune the ViT model pre-trained with Image-Net. In experiments, each client was given randomly selected 10,000 images as training data without duplicates,

where images were resized from $32 \times 32 \times 3$ to $224 \times 224 \times 3$ to fit the size of images to that of ViT. We evaluated the classification accuracy by inputting 10,000 test images to the final global model. In the setting of ViT, the patch size P in patch embedding was set to 16, the number of split patches in an input image was N = 196, and the dimensionality of output feature vectors was D = 384.

A. Classification Performance

In an image classification task, we verified the model performance of the proposed method in terms of image classification accuracy, compared with a standard FL method without any encryption.

Table 1 shows the comparison between the proposed method (encrypted) and the standard one (plain). From the table, the method was verified to maintain the same accuracy as the standard one. Accordingly, the proposed method did not cause any performance degradation even when using encryption.



B. Evaluation of Robustness against Restoration Attack

To verify the effectiveness of the proposed method in terms of enhancing security, a visual information restoration attack was performed on the model information sent from each client. Attention privacy leakage attack (APRIL) [6], which is a method proposed for ViT and is known to restore the original image with high accuracy, was used as an attack method in the experiment. It was pointed out that the parameters of FL learned under the use of ViT have privacy vulnerability, and a method for analytically restoring images from model information was proposed in [6].



(a)original image (b)without encrypted

(c)proposed

Fig. 3. Result of images restored from model parameter with APRIL

Figure 3 shows the results of the experiment. (a) is the original image, and (b) is the image reconstructed from model information (gradients) by APRIL attack in FL without any encryption. From the result, the visual information was confirmed to be restored. In contrast, (c) is the image restored from the model information protected by the proposed method. When applying the proposed method, the visual information was not restored by using APRIL.

V. CONCLUSION

In this paper, we proposed a novel framework of FL based on the embedded structure of ViT for enhancing the security of FL. In the proposed method, the model information shared between the cloud server and each client is encrypted with a secret key that the cloud provider does not know, and a global model is updated in the encrypted domain. In the experiments, the effectiveness of the proposed was confirmed in terms of image classification accuracy and robustness against an image restoration attack called APRIL.

ACKOWLEDGMENT

This study was partially supported by JSPS KAKENHI (Grant Number JP21H01327) and JST CREST (Grant Number JPMJCR20D3).

References

- H. Kiya, A. MaungMaung, Y. Kinoshita, S. Imaizumi, and S. Shiota, "An overview of compressible and learnable image transformation with secret key and its applications," *APSIPA Transactions on Signal and Information Processing*, vol.11, no.1, e11, 2022.
- [2] W. Sirichotedumrong, T. Chuman, S. Imaizumi, and H. Kiya, "Grayscale-based block scrambling image encryption for social networking services," in 2018 IEEE International Conference on Multimedia and Expo (ICME), 2018, pp. 1–6.
- [3] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, vol. 54, 2017, pp. 1273–1282.
- [4] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in Advances in Neural Information Processing Systems, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.
- [5] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients - how easy is it to break privacy in federated learning?" in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS'20. Red Hook, NY, USA: Curran Associates Inc., 2020.
- [6] J. Lu, X. S. Zhang, T. Zhao, X. He, and J. Cheng, "APRIL: Finding the achilles' heel on privacy for vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (CVPR), June 2022, pp. 10051–10060.
- [7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv:2010.11929, 2020.
- [8] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014. [Online]. Available: http: //dx.doi.org/10.1561/0400000042
- [9] H. Kiya, R. Iijima, A. MaungMaung, and Y. Kinoshita, "Image and model transformation with secret key for vision transformer," *IEICE Transactions on Information and Systems*, vol. E106.D, no. 1, pp. 2–11, 2023.
- [10] A. MaungMaung and H. Kiya, "Privacy-preserving image classification using an isotropic network," *IEEE MultiMedia*, vol. 29, no. 2, pp. 23–33, 2022.
- [11] Z. Qi, A. MaungMaung, Y. Kinoshita, and H. Kiya, "Privacy-preserving image classification using vision transformer," in 2022 30th European Signal Processing Conference (EUSIPCO), 2022, pp. 543–547.
 [12] H. Kiya, T. Nagamori, S. Imaizumi, and S. Shiota, "Privacy-
- [12] H. Kiya, T. Nagamori, S. Imaizumi, and S. Shiota, "Privacypreserving semantic segmentation using vision transformer," *Journal* of *Imaging*, vol. 8, no. 9, 2022. [Online]. Available: https: //www.mdpi.com/2313-433X/8/9/233
- [13] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, J. Fernandez-Marques, Y. Gao, L. Sani, K. H. Li, T. Parcollet, P. P. B. de Gusmão, and N. D. Lane, "Flower: A friendly federated learning research framework," 2022.