2021 IEEE International Conference on Image Processing (ICIP) | 978-1-6654-4115-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICIP42928.2021.9506589

FROM GRADIENT LEAKAGE TO ADVERSARIAL ATTACKS IN FEDERATED LEARNING

Jia Qi Lim and Chee Seng Chan

Centre of Image and Signal Processing, Faculty of Computer Science and Information Technology,
Universiti Malaya, 50603 Kuala Lumpur, Malaysia
{jiaqi0602@gmail.com, cs.chan@um.edu.my}

ABSTRACT

Deep neural networks (DNN) are widely used in real-life applications despite the lack of understanding on this technology and its challenges. Data privacy is one of the bottlenecks that is yet to be overcome and more challenges in DNN arise when researchers start to pay more attention to DNN vulnerabilities. In this work, we aim to cast the doubts towards the reliability of the DNN with solid evidence particularly in Federated Learning environment by utilizing an existing privacy breaking algorithm which inverts gradients of models to reconstruct the input data. By performing the attack algorithm, we exemplify the data reconstructed from inverting gradients algorithm as a potential threat and further reveal the vulnerabilities of models in representation learning. Pytorch implementation are provided at `https://github.com/Jiaqi0602/adversarial-attack-from-leakage/`

***Index Terms*—** Gradient Leakage, Federated Learning, Adversarial Learning

1. INTRODUCTION

Deep neural networks (DNN) based solutions have pervaded into our daily lives because of their impressive success across various machine learning problems [1]. However, this achievement is heavily depending on having sufficient number of image-label pairs to train the DNN models. Now, this process is further complicated with the recently announced data protection legislation such as the General Data Protection Regulation[1] (EU GDPR) which aims to safeguard the privacy of data.

For this purpose, Federated Learning [2–4] - a decentralized model training protocol was proposed. The core idea of federated learning is privacy-preserving model training in heterogeneous, distributed networks where one of the most successful use case is the Google Mobile Keyboard [5]. However, in a recent study, Zhu et al [6] questioned the trustworthiness of this decentralized protocol in federated learning by demonstrating the possible leakage of the private input data if the shared gradients are accessible by malicious actors.

[1]https://gdpr-info.eu/

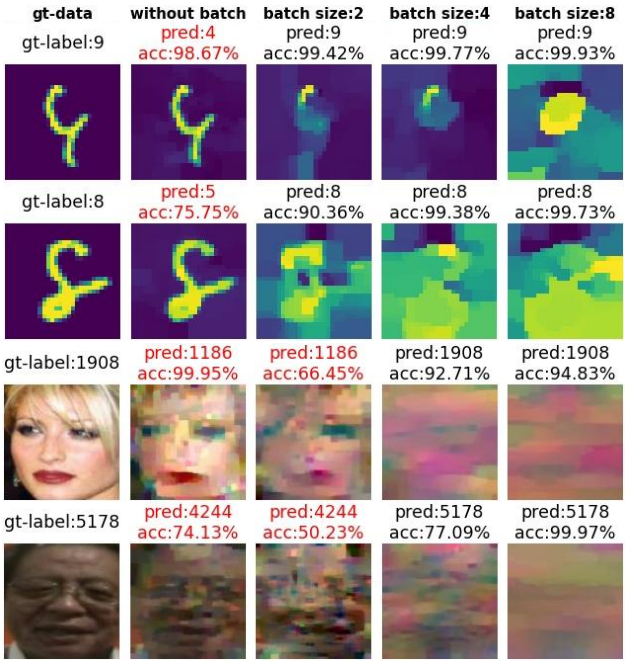

Fig. 1: It can be noticed the images reconstructed with a larger batch size have resulted in a correct prediction with high confidence in respective datasets (top 2: MNIST; bottom 2: VGGFace2) despite these reconstructed images from federated learning protocol are heavily distorted and unrecognizable by human eyes.

This issue leads us to the following question: how much can we trust federated learning protocol? In this paper, we provide a comprehensive study on how different hyperparameter configurations of the existing federated learning privacy breaking method - inverting gradients [7] that may impact the effectiveness of the privacy leakage. To our surprise, we found out that not just it is possible to reconstruct the ground-truth data from the shared gradient, but with a carefully engineered hyperparameters, it is also possible to use the shared gradient to launch an adversarial attack as shown in Fig. 1. As a summary, our contributions are as follows:

- We provide a more detailed study and analyze the impact of inverting shared gradients from federated learn-

ing protocol, and

- We provide a new perspective on viewing the reliability of feature learning in DNN with data generated from inverting gradients method.

2. RELATED WORK

Federated learning protocol [2–4] was proposed to enable multiple parties to jointly improve the performance of a DNN without sharing of any users' data and reduces computational costs in model optimization process. That is to say, each party will compute his/her own gradients locally, and then transmits the gradients to a central server. The gradients will be aggregated and averaged before updating the weights with synchronous stochastic gradient descent (SGD) in the resulting model. At the same time, the local model of each party takes a step of gradient descent to update his/her own local model. Many works in federated learning have adapted synchronous SGD because of guarantee convergence to an optimal solution.

However, a recent study [6] showed that there is a potential reconstruction (deep gradient leakage) of the private input data using the gradients shared across the devices in an existing federated learning environment. The core idea of the work is to match the actual shared gradients with dummy gradients that were generated by a random initialized dummy input. Euclidean loss between the dummy gradients and the actual gradients is computed and used to update the dummy input, when the optimization of the gradients loss finishes, the actual input is revealed. An improved version was proposed by Zhao et al [8] where the work further showed that it is able to extract the ground-truth labels of the input data. Yet, the cost function chosen in [6, 8] is not robust enough to adapt complex model architectures, specifically architectures with rectifier linear unit (ReLU) activation function. Without changing the core idea of DLG [6], [7] proposed cosine similarity as the cost function with an additional of total variation in optimizing the loss between the gradients to solve the problem in [6].

3. GRADIENT LEAKAGE REVISIT

The idea of federated learning is a distributed learning paradigm where it will only share the gradients $\bigtriangledown_\theta L_\theta(x, y)$ instead of the original data (x, y). Supposedly, this kind of distributed learning where user privacy is crucial is safe to be deployed. This is also stated in [5] that in federated learning protocol, original data is considered impossible to recover. However, Zhu et al [6] proved this wrong where it is possible to unveil the original data with inverting gradient as shown in Fig. 2. Gradients inverting starts with a randomly initialized dummy input, x^* as the input of the target model, then dummy gradients, $\bigtriangledown_\theta L_\theta(x^*, y)$ are generated from the derivative of loss, L w.r.t the model's parameters, θ. With a specific cost function and iterations, k, the dummy input, x^* is updated with the loss of actual gradients, $\bigtriangledown_\theta L_\theta(x, y)$ and dummy gradients, $\bigtriangledown_\theta L_\theta(x^*, y)$. At the end of the optimization process after k iterations, the content of actual input, x would be revealed in the dummy input, x^*.

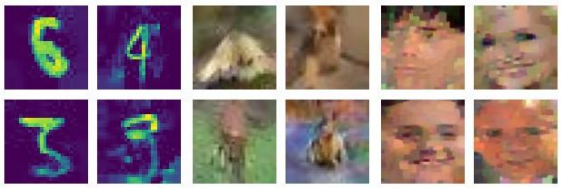

Fig. 2: Examples of reconstructed images using inverting gradients algorithm for different datasets.

In our paper, we would like to extend this and study how difference hyperparameters will affect the reconstructed images, in particular we would like to know if the reconstructed images from the gradients can be correctly recognised by a deep model. For this purpose, we chose cosine similarity proposed by [7] for images reconstruction as shown below:

$$\arg\min_{x\in(0,1)} 1 - \frac{\langle \bigtriangledown_\theta L_\theta(x, y), \bigtriangledown_\theta L_\theta(x^*, y)\rangle}{\|\bigtriangledown_\theta L_\theta(x, y)\| \, \|\bigtriangledown_\theta L_\theta(x^*, y)\|} + \alpha TV(x) \tag{1}$$

where dummy gradients, $\bigtriangledown_\theta L_\theta(x^*, y)$ of the target model are generated from randomly initialized dummy input, x^* and $\alpha TV(x)$ as the regularization parameter in the optimization process. By utilizing this framework, we provide a comprehensive study on how different hyperparameter configurations of this inverting gradients method may impact the effectiveness of the privacy leakage.

4. EVALUATION

4.1. Datasets and Network architecture

We conduct all the experiments on three popular datasets, that are **MNIST** [9] that consists of handwritten digits 0-9 images, **CIFAR10** [10] that consists of 10 classes of 32x32 colour images, and finally **VGGFace2** [11] - a face recognition dataset across pose and age. Also, we employed two different convolutional neural network (CNN) architectures: **ResNet-18** [12] and **DenseNet-121** [13] in this experiment.

Beside the studied hyperparameters settings that will be examined next, the rest of the hyperparameters used in gradients inverting algorithm are fixed for all datasets. For instance, the learning rate is 0.1 with a decay rate that reduces the learning rate by a factor of 0.1 after $\frac{3}{8}$, $\frac{5}{8}$ and $\frac{7}{8}$ iterations and the optimization algorithm to converge the signed gradient is Adam.

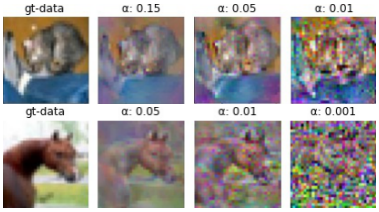

Fig. 3: This shows the effects of different total variation coefficients, α

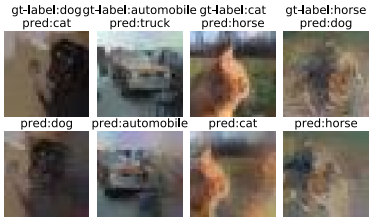

Fig. 4: This shows the effects of different total variation coefficients: first row is 0.05, second row is 0.15.

4.2. Coefficient of total variation and Magnitude of gradients

In computer vision, total variation is implemented to denoise and restore images. As of Eq. 1, $\alpha TV(x)$ term acts as a penalty or regularization term to avoid the occurrence of overshooting in the minimization especially when the magnitude of the gradients is relatively small or near to zero. Examples in Fig. 3 show the reconstructions of input from the ResNet-18 and DenseNet-121 model on CIFAR-10 with different coefficients of total variation, α. We can notice that when the total variation is getting smaller, the quality of the reconstructed images from gradient will be poorer. Another main reason of poor quality in reconstructed images is that since the magnitude of the gradients are relatively small, the gradients contain less information for reconstruction which results in some distortions in the reconstruction output.

4.3. From Deep Leakage to Adversarial Attacks

The quality of reconstructed image from shared gradient is highly depending on the magnitude of the gradients where gradients with larger magnitude are assumed to carry more information of the input data, therefore the quality of the reconstructed image will be better. However, this also implies that the reconstructed image will have a high chance of misclassification due to much larger prediction loss.

In this experiment, we employed the reconstructed images for a classification task. As shown in Fig. 4-5, unexpectedly, we can notice that a larger coefficient of total variation or gradient with the largest norm magnitude used for reconstruction has resulted in a better classification performance despite the ground-truth data was initially wrongly classified.

Fig. 5: This shows the effects of different gradients: first row is averaged gradients norms, second row is gradients with the largest norm magnitude

Empirically, larger total variation coefficient will cause minor degradation in the reconstructed image quality and thus better classification result is less favourable. Yet, gradient with the largest norm magnitude used for reconstruction surprisingly also shows better classification result despite the image reconstructed are highly distorted.

The surprising phenomena caused by the manipulations of total variation coefficient and gradients prompts us to further investigate in using gradients of data in different batches to increase the distortion in the reconstructed images.

4.3.1. Batch Size

As highlighted in [7], reconstruction using gradients of input data in large batch size will produce highly distorted reconstructed output images. So, we stacked a single input into different batch sizes and use the gradients derived from the model prediction for reconstruction, e.g. using batch size of 4 will output 4 reconstructed images from the single input and vice versa. Batch sizes chosen in our experiment are 2, 4 and 8 respectively, coefficient of total variation remains constant with a value of 0.1 throughout the experiment as the effect is less significant if batch size is larger than 1. Then, the reconstructed images will be inferred by the target model.

Stacking the ground-truth data into different batch sizes is to further understand that if larger batch sizes that result in higher distortions reconstructed images will have a correlation with classification performance as we found in Fig. 4- 5. As shown in Fig. 1, for batch size = 2, it can be seen that most of the reconstructed images are out of shape and could barely identify in comparison to the ground-truth input data especially for MNIST. However, the prediction score of the reconstructed images, remarkably for MNIST somehow surprise us where the deep model is able to infer a correct prediction with almost perfect confidence level on the heavily distorted reconstructed images in spite of the loss of important features in the reconstructed images.

When the batch size is increased to 4, it further strengthen this hypothesis. From Fig. 6, the number of correct classified reconstructed data for batch size of 4 is comparatively higher than batch size of 2, specifically when ResNet-18 model is employed. For both MNIST and CIFAR-10 using ResNet-18, the correct classified rate shown in Fig. 6(a) increases approx-

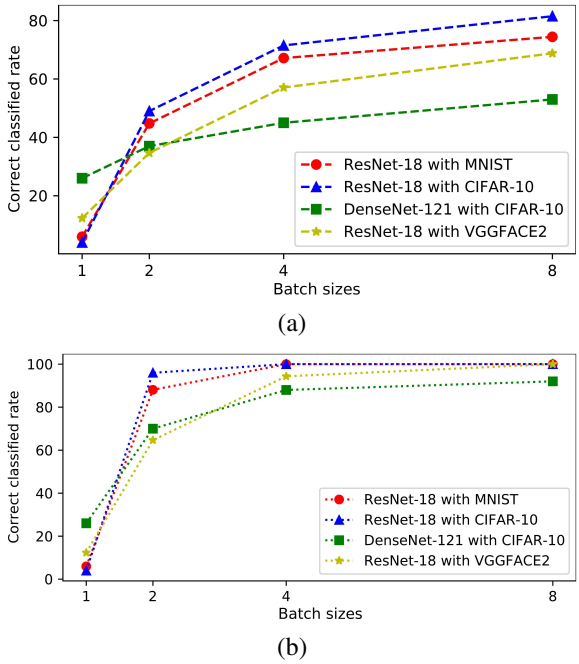

Fig. 6: (a) shows the correct classified rate of number of reconstructed images , (b) shows the correct classified rate of ground-truth images with at least one correctly classified reconstructed images for each datasets and batch sizes. Both graphs exhibit a positive relationship between the correct classified rate and batch size.

imately 20% when the batch size increase from 2 to 4 while for Fig. 6(b), both have achieved 100% when using batch size of 4.

For both MNIST examples shown in Fig. 1, the reconstructed images using batch size of 2 and 4 are correctly predicted, the confidence level of the predictions made increase with the batch size where the effect of batch sizes towards the distortions and correctly classified rate is clearly shown. For the facial recognition dataset VGGFACE2, the ground-truth id_label of the woman in (third row) Fig. 1 is *1908*, the ground-truth image and the reconstructed image without using batch are initially wrongly predicted as *1186* with confidence level 95.89% and 99.95% respectively. When the input data is stacked into batch size of 2 for reconstruction, the confidence level of the reconstructed image decreases to 66.45% simultaneously with the quality of the reconstructed image. When batch size = 4, the reconstructed image is fully distorted and unrecognizable yet it is correctly classified as ground-truth id_label, *1908* with a high confidence level of 92.72%. This scenario also applies to the last row of Fig. 1. From here, we have proved that the relationship between the batch sizes, level of distortions and correctly classified rate is positively correlated.

All datasets employed ResNet-18 model achieved 100%

Fig. 7: Examples of failure case for VGGFACE2 with ResNet-18 using batch size 4 where the reconstructed outputs cannot be correctly predicted.

correct classified rate as shown in Fig. 6(b) when the batch size increases to 8 except DenseNet-121 with CIFAR-10 because a different model architecture is employed. Yet, our hypothesis still applies to DenseNet-121 where correct classified rate exhibited in both Fig. 6(a) and Fig. 6(b) increase with the batch sizes even though the increment is not that dramatics if compared to ResNet-18 model.

4.4. Different Network Architectures

The VGGFACE2 data reconstructed from gradients are further predicted using different networks architectures trained on VGGFACE2 data, the models chosen are FaceNet [14], ResNet-50 [12] and SE-ResNet-50 [15]. Only 5% of reconstructed images can be correctly predicted by FaceNet; 0% of reconstructed outputs are correctly classified by ResNet-50 and SE-ResNet-50. Hence, this shows that the inference only works on the target model used for reconstruction.

4.5. Failure Analysis

There are some failure cases by which the reconstructed images are wrongly predicted by the target model despite the distortions in the reconstructed images for ResNet-18 with VGGFACE2 using batch size of 4 as shown in Fig. 7.

5. CONCLUSION

Inspired by the proposed inverting gradient algorithm, we demonstrated the potential threat in federated learning from another perspective. From our experiment, manipulating hyperparameters of the algorithm and batch size of input data are able to generate distorted images to fool the classification model which could lead other potential threats. Luckily, there are some existing works [16–18] which suggest different encryption methods in federated learning as a defence. We also questioned the reliability of DNN model in performing classifications task in computer vision and federated learning environment specifically for images without human recognizable features.

6. REFERENCES

[1] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–38, 2019.

[2] Reza Shokri and Vitaly Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, 2015, pp. 1310–1321.

[3] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[4] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 1273–1282.

[5] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, H Brendan McMahan, et al., "Towards federated learning at scale: System design," *arXiv preprint arXiv:1902.01046*, 2019.

[6] Ligeng Zhu, Zhijian Liu, and Song Han, "Deep leakage from gradients," in *Advances in Neural Information Processing Systems*, 2019, pp. 14774–14784.

[7] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller, "Inverting gradients–how easy is it to break privacy in federated learning?," *arXiv preprint arXiv:2003.14053*, 2020.

[8] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen, "idlg: Improved deep leakage from gradients," *arXiv preprint arXiv:2001.02610*, 2020.

[9] Li Deng, "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.

[10] Alex Krizhevsky, Geoffrey Hinton, et al., "Learning multiple layers of features from tiny images," 2009.

[11] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*. IEEE, 2018, pp. 67–74.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[13] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on CVPR*, 2017, pp. 4700–4708.

[14] Florian Schroff, Dmitry Kalenichenko, and James Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[15] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[16] Runhua Xu, Nathalie Baracaldo, Yi Zhou, Ali Anwar, and Heiko Ludwig, "Hybridalpha: An efficient approach for privacy-preserving federated learning," in *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, 2019, pp. 13–23.

[17] Lixin Fan, Kam Woh Ng, Ce Ju, Tianyu Zhang, and Chee Seng Chan, "Deep polarized network for supervised learning of accurate binary hashing codes," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 2020, pp. 825–831.

[18] Qiushi Li, Wenwu Zhu, Chao Wu, Xinglin Pan, Fan Yang, Yuezhi Zhou, and Yaoxue Zhang, "Invisiblefl: Federated learning over non-informative intermediate updates against multimedia privacy leakages," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 753–762.