Privacy Preservation in Distributed Deep Learning Applications

Akash Govind Kuttikad
University of Illinois at Urbana-Champaign
NetID: agk4
agk4@illinois.edu
Project Files: Akash's GitHub

Abstract

This project aims to develop a method for privacy preservation of images in distributed deep learning applications. By modifying the loss function in a Split-NN based classification problem, we are able to study the trade-off we need to achieve to protect the privacy of input data versus the accuracy of the classifier. We demonstrate an instance of the above mentioned setting for MNIST data, where the attackers attempts to reconstruct the input data using the leaked-latent representation. We observe that the accuracy drops from 0.98 to 0.41 as the value of the weighting (given to privacy preservation) is increased from 0 to 0.2.

1 Introduction

Over the last few years, machine learning has tremendously eased our day-to-day lives with models that provide movie recommendations, language translations, healthcare predictions, face recognition, and many more applications. A Forbes 2022 report predicts that AI has achieved an inflection point and is poised to transform every industry - including healthcare, consumer experience, foreign policy and climate crisis [1]. The key advantage of such systems is its capability to identify, extract and predict crucial inferences from a data-set with minimal or no human intervention.

To train these models, large amounts of data is collected (with or without our awareness) from the users, since ML systems are only as good as the quality of the data that informs the training of ML models. Data required is more than what a single individual or organization can contribute. For instance, Google Ngram (an online search engine for text) uses 486 billion data records, and Google translate roughly uses 1 trillion data samples for training their models [2]. This grants these models and servers access to sensitive personal information - including pictures, financial records, and medical reports. However, with rising concerns about privacy regulations like the EU General Data Protection Regulation (GDPR) [3], businesses must be cognizant of security and privacy considerations associated with leveraging machine learning. According to Forbes, data privacy will be the most important issue in the next decade [4]. Shokri et al. [5] tested attacks on commercial platforms and were able to achieve 74% accuracy against Amazon's machine learning as a service and 94% accuracy against Google's machine learning as a service, substantiating fairly high risk for a machine learning task leaking private information.

Moreover, malicious attacks by hackers pose huge threats to the data being used by these models. The Facebook–Cambridge Analytica data scandal is an example of one of the most infamous privacy breaches in history, in which personal data belonging to millions of Facebook users was collected without their consent, and predominantly used for political advertising. Deep learning and AI have become an integral, ubiquitous component of several systems, and hence security and privacy of the data used in such models are issues of growing importance. Corporates and organizations that use ML models have the pressing need to preserve the confidentiality and privacy of data utilized and earn and maintain the trust of the people who use their products.

Traditionally, machine learning models require data to be transferred to a central server which could potentially tamper with the privacy of the data being used. Distributed learning enables machine learning systems to be decentralised limiting the information exposed from the contributor's dataset and thus controlling the risk of compromising the privacy of the data. However, such methods to overcome privacy attacks through encryption, anonymization, or distributed learning only partially solves the problem - prevents exposure of data to the server or any hacker that attempts to acquire it. Recent research has demonstrated that reverse-engineering input data through such intermediate representation is not a huge challenge.

In literature, Song et. al. [6] have shown that attacks on popular sentence embeddings (encoded representations in NLP) not only recover between 50%–70% of the input words, but also might reveal sensitive attributes inherent in inputs. Hence, employing a distributed deep learning technique without sharing raw input data does not guarantee privacy preservation.

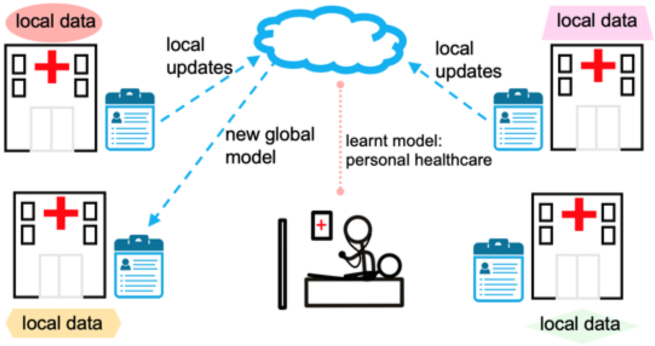


Figure 1: Example application of distributed deep learning

Vertical split learning was first introduced by Gupta & Raskar [7] in 2018, and has been studied extensively in literature [8, 9, 10] as a baseline architecture for several privacy preserving models. In SplitNN (Figure 2), each client computes a fixed portion of the computation and propagates it to the split layer. The outputs at the split layer are sent to another entity (server/another client) which completes the rest of the training without looking at raw data. During back-propagation, the gradients are passed back in a similar fashion through the split layer back to the clients who can perform their respective back-propagation.

However, as discussed previously, such a distributed network resolves data governance and ownership problems, but does not guarantee the privacy of training data unless coupled with other methods. This is because neural networks represent a form of unintentional memory mechanism of training data within the weights involved in the network. Such model inversion attacks [11, 12] are capable of reconstructing images from the algorithm weights with commendable accuracy, and falters the security of the input data involved.

$$Objective = L_{CrossEntropy}(X, Y) + \alpha D_{similarity}(X, Z) \tag{1}$$

This project aims to resist reconstruction attacks by using the modifying the objective function in classification, i.e the cross entropy loss ($L_{CrossEntropy}(X, Y)$) involved in the classification task. By introducing a distance correlation metric (which will be referred to as image similarity, denoted by $D_{similarity}(X, Z)$) between the raw input (X) and intermediate representations (Z), the newly proposed objective function will jointly optimize the cross entropy and the image similarity (Figure 2). By defining the objective function as a weighted sum (by modifying weight α, given in Eq. 1) of the image similarity and cross entropy, the model privacy can be preserved without a significant effect on the model accuracy.

1.1 Related works

Reconstruction attacks attempt to recreate one or more training samples partially or fully, where the adversary makes use of the knowledge of the intermediate vectors to achieve its goal. It has

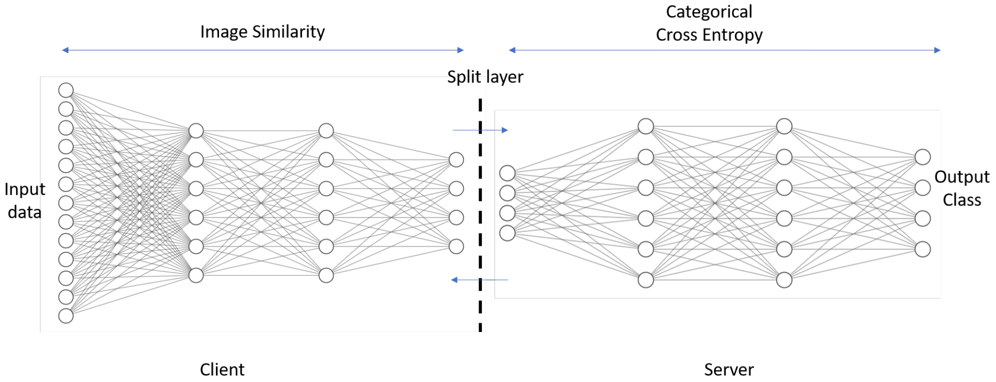


Figure 2: Proposed model

also been referred to as attribute inference or model inversion in popular literature, where sensitive features or the full data sample is recovered by the adversary. Raynal et al. [13] introduce a method for quantifying visual privacy introduced in images through obfuscation. By employing a variety of obfuscation techniques like adding noise, pixel shuffling, pixel grafting, blurring and mixing, they succeed in quantifying and validating their privacy metrics with a state-of-the-art computer vision. Ziegler et al. integrated differential privacy with a Gaussian noise mechanism into the federated learning to avoid privacy leakages from chest X-ray data [14].

2 Theory and Modelling

2.1 Information theory and loss function

Let $X \in R^{nxd}$ be the input matrix and $Z \in R^{nxp}$ be the intermediate representation with p neurons with the Markov chain $X \rightarrow Z \rightarrow \hat{Y}$. From an information theory perspective, the objective of the loss function then becomes [15]

$$\underset{\theta}{min} - I(Z,Y) + \lambda I(X,Z)$$

The first term encourages the learned encoding to be maximally informative about the label Y and measures the prediction performance of the model. The second term reduces the dependency between X and Z which improves the robustness against the adversarial attacks.

However, in order to implement this practically, we require a metric that captures/approximates the essence of the loss function that preserves privacy. Hence, in addition to the cross entropy loss, we introduce a term $D_{similarity}(X,Z)$ which is the correlation score. The distance correlation between any two random variables X and Y is given by [16]:

$$dCorr^2(X,Y) = \frac{dCov^2(X,Y)}{\sqrt{dVar^2(X).dVar^2(Y)}}$$

In this project, we modify the above correlation score for matrices as follows:

$$D^2_{corr}(X,Z) = \mathbf{1}^T.diag^{-1}(\bar{X}^T\bar{X}).(\bar{X}^T\bar{Z})^2.diag^{-1}(\bar{Z}^T\bar{Z}).\mathbf{1}$$

where $\bar{Z} \in R^{nxd}$ and $\bar{X} \in R^{nxp}$ are mean-centered versions of Z and X respectively. $\mathbf{1}$ and $\mathbf{1}^T$ are vectors of 1s of appropriate length multiplied to get the sum of the matrix elements.

2.2 Modelling

The following NN architectures have been used for the SplitNN (Figure 3) and Attacker model (Figure 4) respectively. The SplitNN model has been trained on the MNIST dataset for 10 epochs for various weights α in $\{0, 0.01, 0.05, 0.1, 0.15, 0.2\}$. The attacker is trained for 5 epochs on 5000

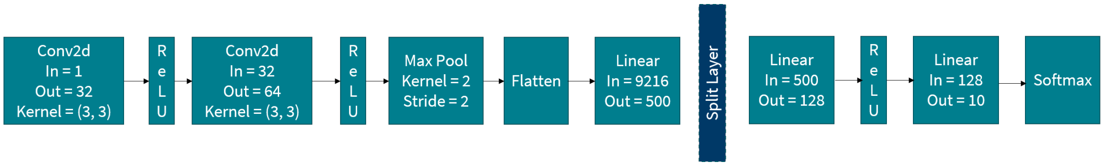


Figure 3: SplitNN model

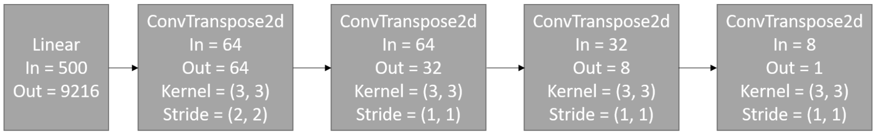


Figure 4: Attacker model

images from the EMNIST dataset; this is done to replicate a real-world attack scenario, where the attacker would not have access to the real training data.

3 Results and Discussion

Figure 5 shows the test-train curves obtained for various values of alpha in $\{0, 0.01, 0.05, 0.1, 0.15, 0.2\}$. As expected, we observe that the accuracy of the classifier drops as the value of alpha increases. This is attributed to the relative weighting and importance given to the privacy preservation part of the loss function, i.e. the distance correlation term. The model with $\alpha = 0$ denotes the vanilla model, without any privacy preserving constraints.

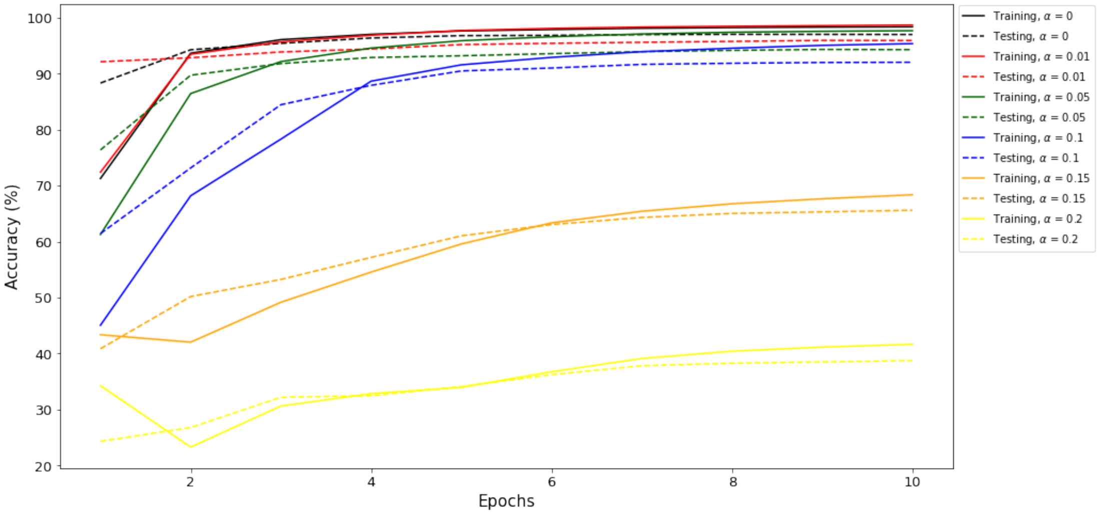
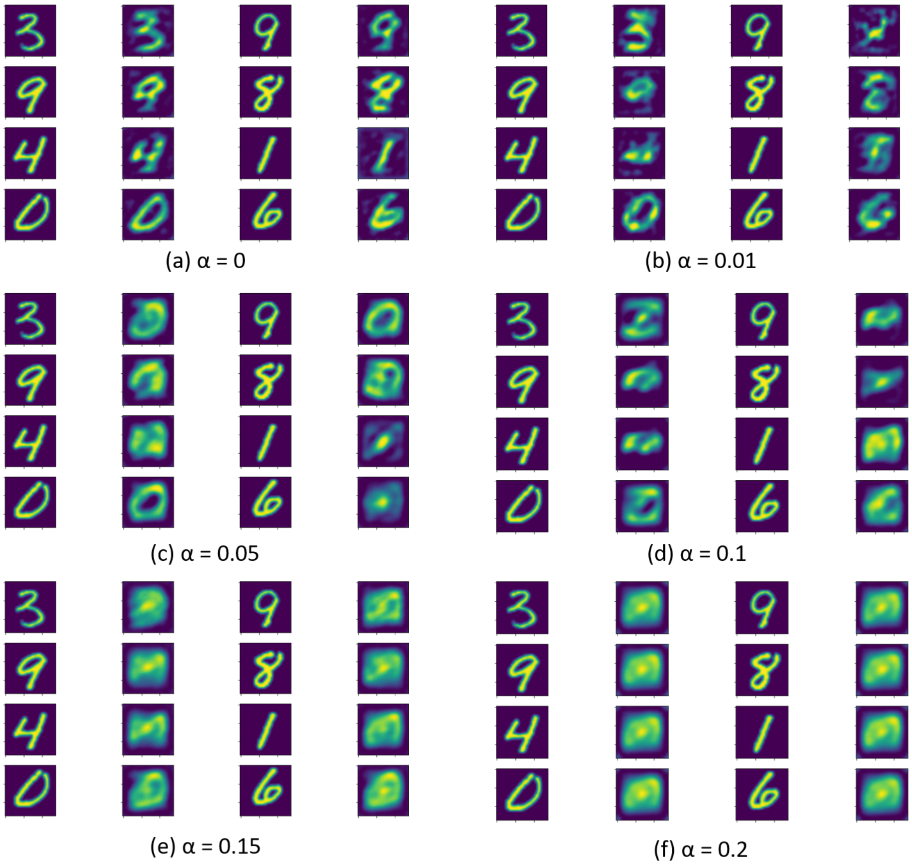


Figure 5: Training and test curves for different α

Figure 6 shows the reconstruction attacks on 8 digits that have been implemented by training the attacker on the EMNIST data set. The first column is the original image and the second columns denotes the reconstructed image from the intermediate split layer. For $\alpha = 0$, we can see that most the digits are successfully reconstructed by the attacker, and privacy of the input images is at stake. For high-stake application that contain sensitive information (medical records, personal images), such an attack could be detrimental.

However, as we increase the value of α, we can see that the attacker struggles to reconstruct the original digit. This becomes almost impossible for values of $\alpha \geq 0.15$. However, the models in that range of α have very poor accuracy (< 70%). Hence, there is a clear trade-off that needs to be achieved in case of an ideal-privacy preserving classifier. In Figure 6(d), we observe that such an optimal point is achieved for $\alpha = 0.1$, where majority of the images is not reconstructed without having a significant effect on accuracy (90%). We observe that the accuracy drops from 0.98 to 0.41 as the value of the weighting (given to privacy preservation) is increased from 0 to 0.2.



(a) α = 0

(b) α = 0.01

(c) α = 0.05

(d) α = 0.1

(e) α = 0.15

(f) α = 0.2

Figure 6: Reconstruction attacks

4 Conclusion

In this project, we were able to develop a method to protect the privacy of input data in a SplitNN model. By introducing a weighted (α) term to quantify the correlation between input and intermediate data, we are able to regulate the reconstruction effectiveness of the attacker. However, such privacy preservation comes at the cost of accuracy, and studying the accuracy-robustness trade-off helps us to determine the optimal weighting needed to be achieved. This definition of 'optimality' varies based on the application. In future, this work can be extended to study the effect of varying amounts of training data the attacker has access to, and how different models of attackers perform. I would like to take this opportunity to thanks Dr. Han Zhao for his continued support and guidance throughout this project.

The project files are available at Akash's GitHub.

References

[1] https://www.forbes.com/sites/forbesbusinesscouncil/2022/05/05/the-future-of-ai-5-things-to-expect-in-the-next-10-years/?sh=65476ba7422b

[2] https://developers.google.com/machine-learning/data-prep/construct/collect/data-size-quality

[3] Goldsteen, A., Ezov, G., Shmelkin, R. et al. Data minimization for GDPR compliance in machine learning models. AI Ethics 2, 477–491 (2022)

[4] https://www.forbes.com/sites/marymeehan/2019/11/26/data-privacy-will-be-the-most-important-issue-in-the-next-decade/?sh=3f5ec0918823

[5] https://www.infoq.com/articles/privacy-attacks-machine-learning-models/

[6] Congzheng Song and Ananth Raghunathan. 2020. Information Leakage in Embedding Models. In Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS '20). Association for Computing Machinery, New York, NY, USA, 377–390. https://doi.org/10.1145/3372297.3417270

[7] Gupta, Otrkist, & Raskar, Ramesh. 2018. Distributed learning of deep neural network over multiple agents. arXiv preprint arXiv:1810.06060

[8] Ceballos, Iker & Sharma, Vivek Mugica, Eduardo & Singh, Abhishek & Roman, Alberto & Vepakomma, Praneeth & Raskar, Ramesh. (2020). SplitNN-driven Vertical Partitioning.

[9] Romanini, D., Hall, A.J., Papadopoulos, P., Titcombe, T., Ismail, A., Cebere, T., Sandmann, R., Roehm, R., Hoeh, M.A. (2021). PyVertical: A Vertical Federated Learning Framework for Multi-headed SplitNN. ArXiv, abs/2104.00489.

[10] Vepakomma, Praneeth & Gupta, Otkrist & Dubey, Abhimanyu Raskar, Ramesh. (2019). Reducing leakage in distributed deep learning for sensitive health data.

[11] Zhang, Y. et al. Te secret revealer: generative model-inversion attacks against deep neural networks. Preprint at https://arxiv.org/abs/1911.07135 (2019).

[12] Hitaj, B., Ateniese, G. Perez-Cruz, F. Deep models under the GAN: information leakage from collaborative deep learning. In Proc. 2017 ACM SIGSAC Conf. Computer and Communications Security 603–618 (ACM, 2017).

[13] Raynal, Mathilde & Achanta, Radhakrishna Humbert, Mathias. (2020). Image Obfuscation for Privacy-Preserving Machine Learning.

[14] Ziegler, J.; Pfitzner, B.; Schulz, H.; Saalbach, A.; Arnrich, B. Defending against Reconstruction Attacks through Differentially Private Federated Learning for Classification of Heterogeneous Chest X-ray Data. Sensors 2022, 22, 5195.

[15] Wang, Tianhao, Yuheng Zhang and R. Jia. “Improving Robustness to Model Inversion Attacks via Mutual Information Regularization.” AAAI (2021).

[16] Wikipedia contributors. "Distance correlation." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 6 Sep. 2022. Web. 2 Dec. 2022.