An alternative to federated learning reducing the risk of GAN attacks and membership inference attacks

Cameron Hochberg

Thesis

Submitted in Partial Fulfillment of the Requirements for the Degree of

Master in Computer Science

Thesis Advisor: Erman Ayday

Department of Computer and Data Science

Case Western Reserve University

May, 2022

An alternative to federated learning reducing the risk of GAN attacks and membership inference attacks

Case Western Reserve University
Case School of Graduate Studies

We hereby approve the thesis[1] of

Cameron Hochberg

for the degree of

Master in Computer Science

Erman Ayday

Committee Chair, Erman Ayday Date
Department of Computer and Data Science

Soumya Ray

Committee Member Date
Department of Computer and Data Science

Xusheng Xiao

Committee Member Date
Department of Computer and Data Science

Stanley Omeike

Committee Member Date
Department of Computer and Data Science

[1] We certify that written approval has been obtained for any proprietary material contained therein.

Table of Contents

List of Figures

Acknowledgements

I would like to thank Professor Erman Ayday for the opportunity work on such a project. I would also like to thank the members of this jury to take time to review this project.

ABSTRACT

An alternative to federated learning reducing the risk of GAN attacks and membership inference attacks

Cameron Hochberg

Previous research has shown that federated learning is vulnerable to multiple attacks, in particular whitebox attacks. Even limiting the number of parameters shared can still lead to the training data of the victim being leaked. This is problematic especially when considering laws like GDPR and HIPAA that demand that sensitive data be protected. Our alternative proposes to only share the output labels themselves of the training phase to a server who would then choose the correct label and send it back to all participants. This ensures that only blackbox attacks could be performed on the system. Our preliminary results seem to show that this would make it much harder to run membership inference attacks on this system.

1 Introduction

Recent developments in data gathering have led to the development of new techniques in machine learning. With this development, laws have also started to evolve in order to catch up with the advances in technology. The EU just a few years ago added new regulations to data gathering and data processing in order to protect consumers and prevent abuse from companies[1]. Because of such laws, it can be hard to share data with other agents in order to create big enough datasets for machine learning to be possible. With all the anonymization and security requirements to share data, the quality of the data gathered is going to be affected. To solve this, there exists methods like federated learning so that research can be done while protecting the data gathered and preserving its quality[2].

Federated Learning is a technique that allows multiple agents to train a model in parallel by sharing the model updates with a central server. In this scenario, the data would only be stored on each edge device and the server wouldn't have access to that data. The server would only receive the model updates of each local model, compute the average and send that back to each participant so that they can update their model. This would make each of them converge towards a central model while respecting all the laws on data privacy.

However, researchers recently showed that federated learning was vulnerable to multiple form of attacks that could recover the local training data[3]. Because federated learning shared the model updates themselves, a lot more data about the training itself is shared and each update helps attacker refine their search. While they did show that using techniques such as differential privacy, federated learning could reduce the risk of model inversion attacks greatly, GAN attacks still had great success.

GDPR has various clauses that would then make federated learning unusable in Europe. And as discussed earlier, techniques to reduce the risk of attacks proposed under these guidelines would only hurt the quality of the data and thus the quality of the model trained. This is why we wanted to produce an alternative to federated learning that would allow researchers to continue with collaborative learning techniques.

2 Experimental Methods

2.1 AIMHI algorithm

In order to test the algorithm's efficacy, I created a python script that would be able to emulate the client-server relationship we see in federated learning. As we can see in figure 2.1, our experiment is essentially divided into 5 big parts:

(1) Each participant trains their local model with their local dataset
(2) The server sends global unlabeled data to each local model
(3) Each participant classifies the data and sends the results to the server
(4) The server uses a consensus function to determine the correct label and sends the result back to each participant
(5) Each local participant updates their local data

This is repeated until there the training is complete or there is no more training data available.

In order to test the efficacy of the algorithm, I used the CIFAR-10 dataset and trained 5 participants on local datasets of varying sizes. I also varied the size of the global dataset as well as the number of training epochs and measured the accuracy of each model on a validation set of 10K images given in the CIFAR-10 dataset after each epoch. Finally we also varied the consensus function used by the server. First using simple majority voting then using unanimous voting.

In simple majority voting, the global dataset was divided into batches of fixes sizes and fed to each participants for a number of epochs. We also tried to see if using some form a linear feeding, that is feeding the same training data again for the

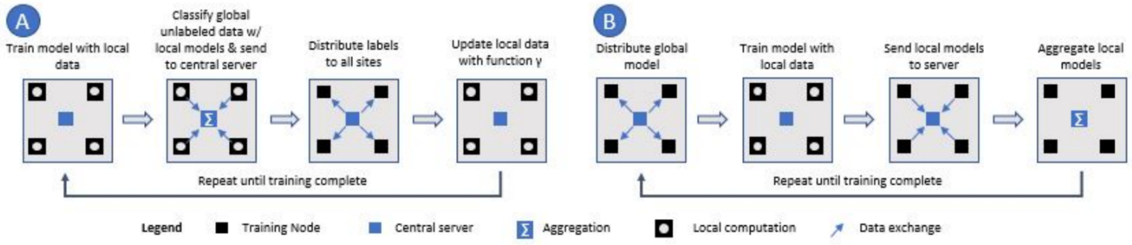

A
Train model with local data
Classify global unlabeled data w/ local models & send to central server
Σ
Distribute labels to all sites
Update local data with function γ
Repeat until training complete
B
Distribute global model
Train model with local data
Send local models to server
Aggregate local models
Σ
Repeat until training complete
Legend
Training Node
Central server
Σ Aggregation
Local computation
Data exchange

Figure 2.1. Process and data flow for a) AIM HI and b) Federated Learning

same number of epochs, would raise the accuracy of each participants after they gone through the global dataset a first time. In unanimous voting, we changed the way data was processed by each participants. In that case, we directly fed the entire global dataset to each participant and divided the data according to the consensus function. If all participants labeled an image the same, it was classified as labeled and participants would then train again with this additional set of images. In the case there was different labels, we separated these images and sent them back to each participant to be classified again after each model was trained. This was then repeated until either all the data was labeled or a certain number of epochs was reached.

2.2 Privacy Analysis

After showing some results of aimhi, I used the Machine Learning Privacy Meter tool[4] to perform membership inference attacks[5] on one of the trained models. The tool provides different parameters to fit the attack to the parameters of our experiment. In this case, we assume that the server was the attacker and thus had knowledge of the dataset prior to the attack and the attack model could be trained on it.

We limited the attacks to a realistic number of epochs based on our experiments. That is when the training period is over, we assume that the attacker can't continue his attack beyond that point.

We ran two type of membership inferences attacks: whitebox attacks for federated learning and blackbox attacks for aimhi[6]. We varied the number of epochs the attack was ran over as well as the amount of data the server had access to.

3 Experimental Results

3.1 AIMHI Algorithm

3.1.1 Majority Voting

Using majority voting, we saw an increase in model accuracy ranging from 10 to 20%. In figure 3.1, the experiment we ran also included feeding the training data to each model an additional time for the same number of epochs to see if the accuracy would go even higher. It was concluded that this didn't increase accuracy but rather increased the loss of each model even more. While majority voting couldn't get to the target accuracy of around 70% that federated learning could reach, we still showed that AIMHI could indeed raise the accuracy of participants using only the classification labels.

3.1.2 Unanimous Voting

In using this system, we could see that each model reached the 70% benchmark of federated learning in only 10-20 epochs 3.2. This result shows that AIMHI depends heavily on the consensus function chosen and can reach federated learning in accuracy and efficacy. It can also be argued that it is somewhat better than federated learning in that to achieve the same level of accuracy, federated learning requires on the order of 100 epochs.

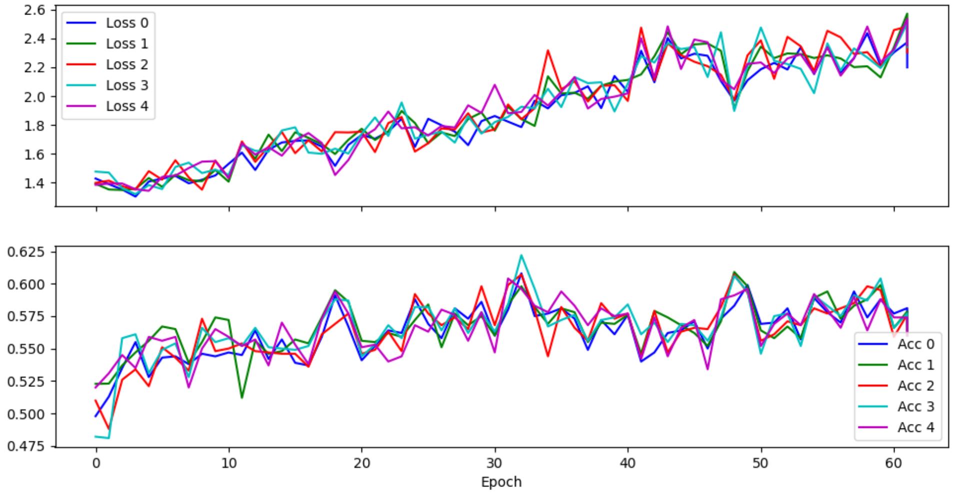


Figure 3.1. AIM HI majority voting over 30 epochs using linear feeding

3.2 Privacy Analysis

In this section we measure the success of each type of attack against a Federated Learning trained model versus an AIMHI trained model under each condition. We measure the success of an attack based on 2 factors: the value of the Area Under the Curve or AUC in our ROC graph and the present of 2 columns at both extremities of the privacy analysis graph. In addition, these 2 columns should correspond to the non members around 0.0 and training member around 1.0 respectively.

3.2.1 Whitebox attacks

In both the 100 and 30 epochs cases than we can see in figures 3.3 and 3.4, both correspond to very successful attacks. The attacker was able to classify over 80% of the training data members and keep a false positive rate below 10% at the worst in the case of 30 epochs.

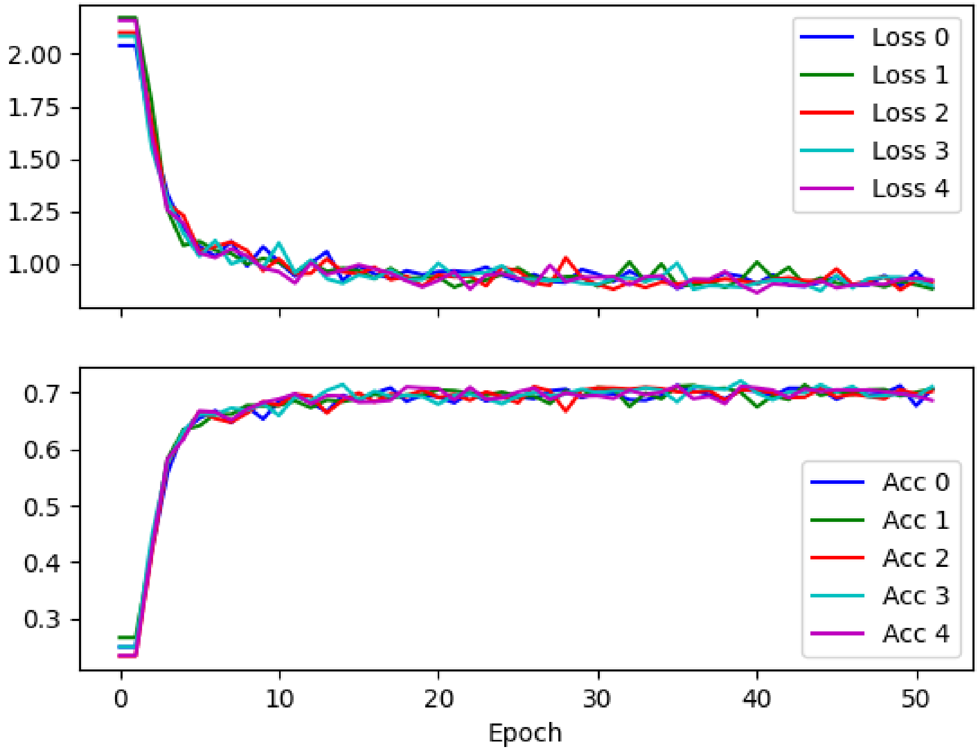
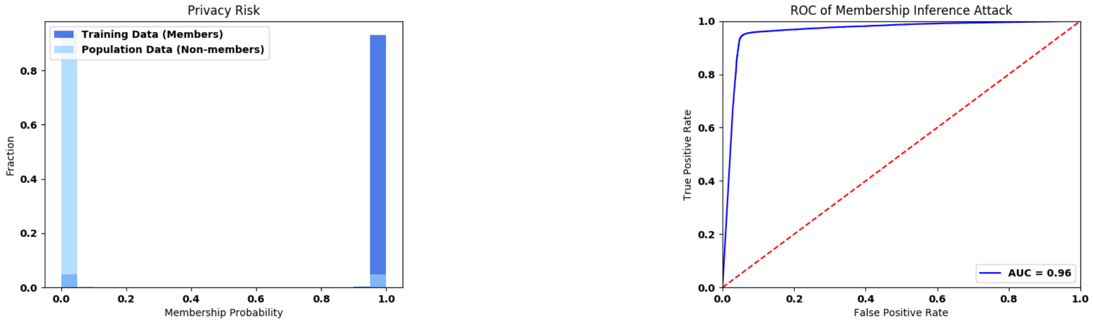


Figure 3.2. AIM HI majority voting over 30 epochs using linear feeding



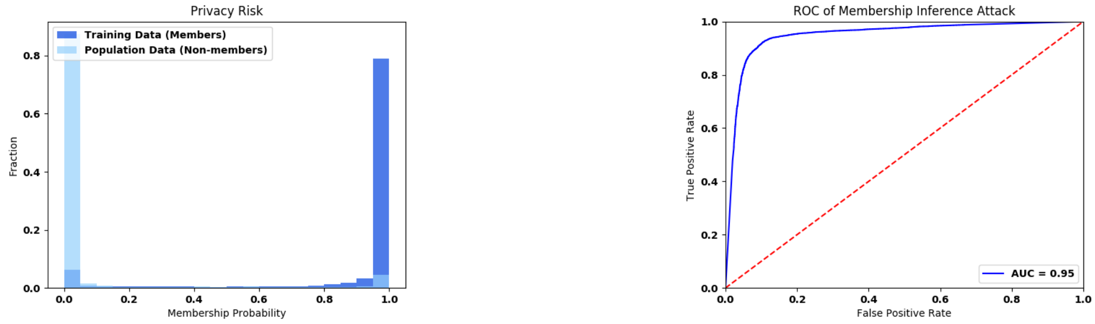
(a) Privacy Analysis

(b) ROC Graph

Figure 3.3. Privacy analysis of Federated Learning over 100 epochs

3.2.2 Blackbox attacks

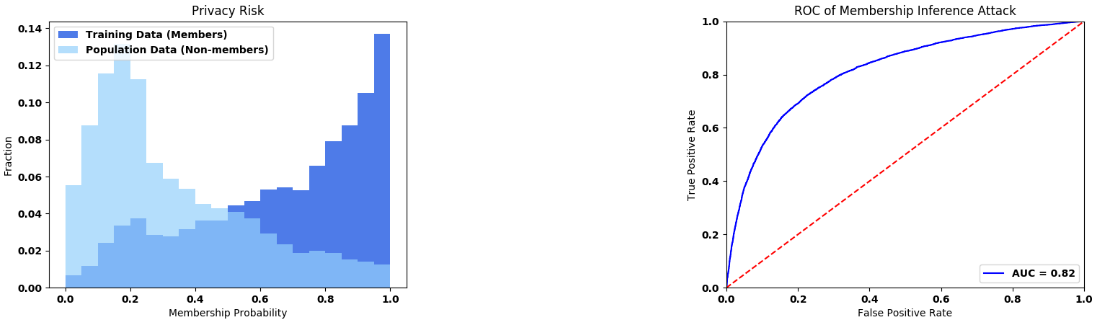
In the case of AIMHI, the results are more spread. In figure 3.5, we can see that the attack is somewhat successful or at least starting to be. 80% of the training data



(a) Privacy Analysis

(b) ROC Graph

Figure 3.4. Privacy analysis of Federated Learning over 30 epochs

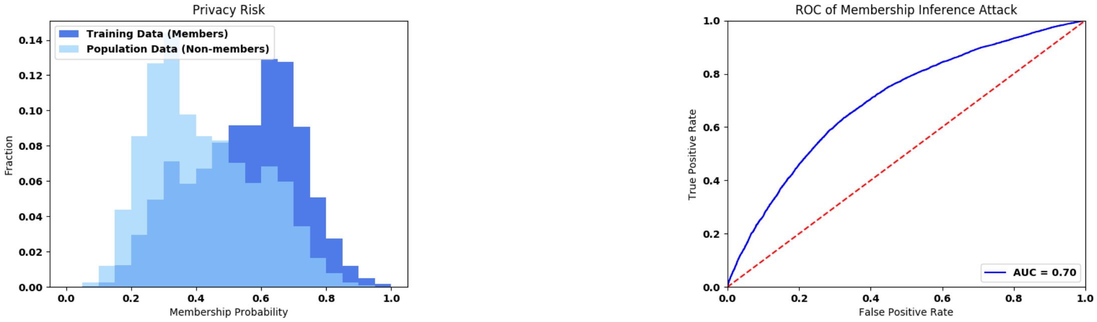


(a) Privacy Analysis

(b) ROC Graph

Figure 3.5. Privacy analysis of AIMHI over 100 epochs

members are identified with only a 20% false positive rate however, this also isn't necessarily done with great confidence either as we can see in the privacy analysis graph. On the other hand, in figure 3.6, the attack isn't successful at all. We can see the start of both columns but the confidence in either isn't great which is reflected in the ROC graph. When compared with federated learning in the same conditions, this is a great improvement in securing the training data.



(a) Privacy Analysis

(b) ROC Graph

Figure 3.6. Privacy analysis of AIMHI over 30 epochs

4 Discussion

From our results, we have confirmed that federated learning is indeed vulnerable to whitebox attacks. While we didn't implement any defense measures against membership inference attacks like differential privacy[3], we were testing both cases best possible conditions for an attacker. From our results, it is clear that using AIMHI, attacker would have a much more trouble in achieving a successful attack. I acknowledge that there is no way to prevent this sort of attack entirely but even when running a blackbox attack with a number of epochs similar to that of a federated learning training, the attack wasn't the great success it was for the whitebox attack on federated learning itself.

The next step in this research is to explore other areas that could affect the success of an attack other than the number of epochs such as the amount of labels collected per epochs as well as explore other more realistic cases where the attacker would be one of the participants for example instead of the server itself.

5 Conclusions

In conclusion, AIMHI is an alternative to federated learning that provides more data security as well as similar accuracy performance. The system isn't perfect and as shown in this worst case scenario, there is always the possibility of a successful attack but, as outlined in the GDPR rules[1], the goal is mostly to make it harder for an attacker to recover the training data rather than preventing the attack all together. While there is still a lot of tests to do on AIMHI itself to test all of its efficacy especially test in on a real dataset, this is still a very good proof of concept for a collaborative learning framework.

References

[1] Council of European Union. Council regulation (EU) no 2016/679, 2016. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN.

[2] Jose Corbacho. Federated learning. a machine learning adventure with... | by jose corbacho | proandroiddev, 2018.

[3] Briland Hitajm, Giuseppe Ateniese, and Fernando Perez-Curz. Deep models under the gan: Information leakage from collaborative deep learning. 2019.

[4] Aadyaa Maddi, Jiayuan Ye, Sasi Kumar Murakon, and Reza Shokri. Machine learning privacy meter: A tool to quantify the privacy risks of machine learning models with respect to inference attacks, notably membership inference attacks.

[5] Liwei Song, Reza Shokri, and Prateek Mittal. Membership inference attacks against adversarially robust deep learning models. *2019 IEEE Security and Privacy Workshops (SPW)*, 2019.

[6] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. *2019 IEEE Symposium on Security and Privacy (SP)*, 2019.

[7] Frank Ernst. Background of SAM atom-fraction profiles. *Materials Characterization*, 125:142–151, 2017.

[8] Salvatore Brischetto, Carlo Giovanni Ferro, Paolo Maggiore, and Roberto Torre. Compression Tests of ABS Specimens for UAV Components Produced via the FDM Technique. *Technologies*, 5(2):20, June 2017. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[9] Emiliano De Cristofaro. An overview of privacy in machine learning. *CoRR*, abs/2005.08679, 2020.

[10] Stacey Truex, Ling Liu, Mehmet Emre Gursoy, and Wenqi Wei Lei Yu. Demystifying membership inference attacks in machine learning as a service. *IEEE Transactions on Services Computing*, 2021.