A Privacy-Preserving Deep Learning Approach for Face Recognition with Edge Computing

Yunlong Mao[1], Shanhe Yi[2], Qun Li[2], Jinghao Feng[1], Fengyuan Xu[1], Sheng Zhong[1]

1. Nanjing University
2. College of William & Mary

USENIX Workshop on Hot Topics in Edge Computing 2018

Outline

1 Introduction

Face Recognition with Deep Learning

2 System Model

Edge Computing, Privacy Threat Model

3 Our DP-A Algorithm

Differentially Private Activation (DP-A)

4 Result

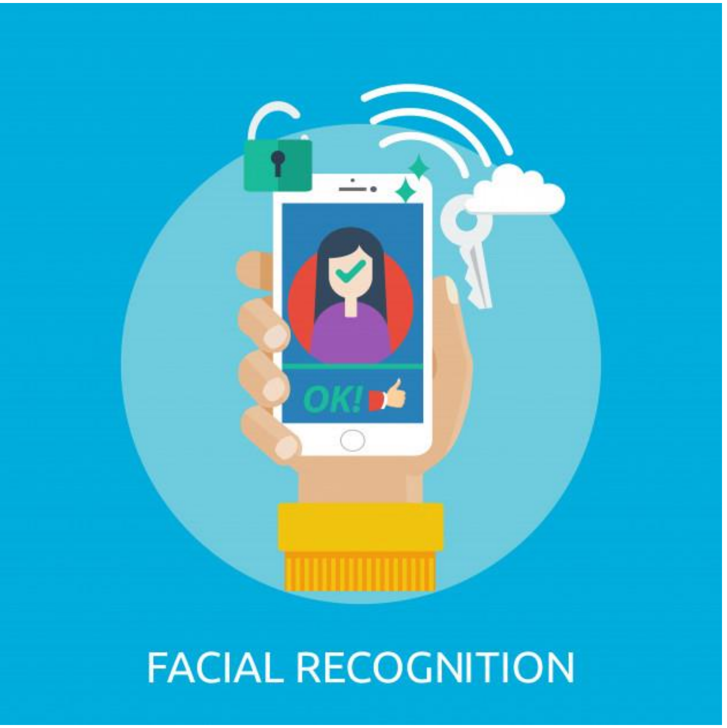
Training Accuracy and Mobile Performance.

01

Introduction

Face Recognition
with Deep Learning

01 Face Recognition



- Smart devices with multiple sensors have improved people’s daily life significantly, such as Smart Phone, Smart Glasses and Smart Watch.

- Deep neural networks are applied to various fields. Many successful deep neural networks are changing our life. Such as YOLO [1], VGGFace [2] and Cancer Detection [8].

- Plentiful sensor data has changed user’s role from data consumer to data producer.

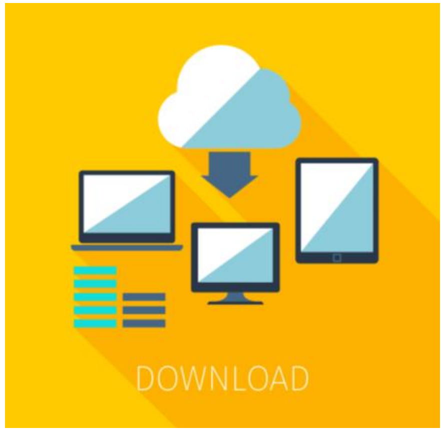
- How can smart device users benefit from deep learning based face recognition with their massive private data?

1. J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *2017 IEEE Conference on Computer Vision and Pattern Recognition*
2. Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman, "Deep Face Recognition," *BMVC*, 2015
3. Esteva, Andre, et al. "Dermatologist-level classification of skin cancer with deep neural networks." *Nature*, 2017

01 Face Recognition with Deep Learning

Solution #1

Any user who has interests in deep learning based face recognition should **upload all private data to a central server (or cluster)** which has enough computing power to train a deep neural network based classification model.



Solution #2

Any user who has interests in deep learning based face recognition should **perform collaborative training with a parameter server in charge of parameters aggregation.**

Many attentions have been attracted by this solution, e.g. [1-3].

1. Bonawitz, Keith, et al. "Practical secure aggregation for privacy-preserving machine learning." Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2017.
2. Shokri, Reza, and Vitaly Shmatikov. "Privacy-preserving deep learning." Proceedings of 2015 ACM SIGSAC conference on computer and communications security. ACM, 2015.
3. Hitaj, Briland, Giuseppe Ateniese, and Fernando Perez-Cruz. "Deep models under the GAN: information leakage from collaborative deep learning." Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2017

01 Face Recognition with Edge Computing

Motivation:

- Existing solutions have serious privacy and resource issues. User’s private data will be violated seriously if the central server is untrusted.
- Smart device users are holding plentiful data which is very valuable for deep learning models.
- Users should outsource heavy computing task to an edge server to avoid intensive computing.
- However, the edge server could be untrusted, user’s privacy may be violated.

02

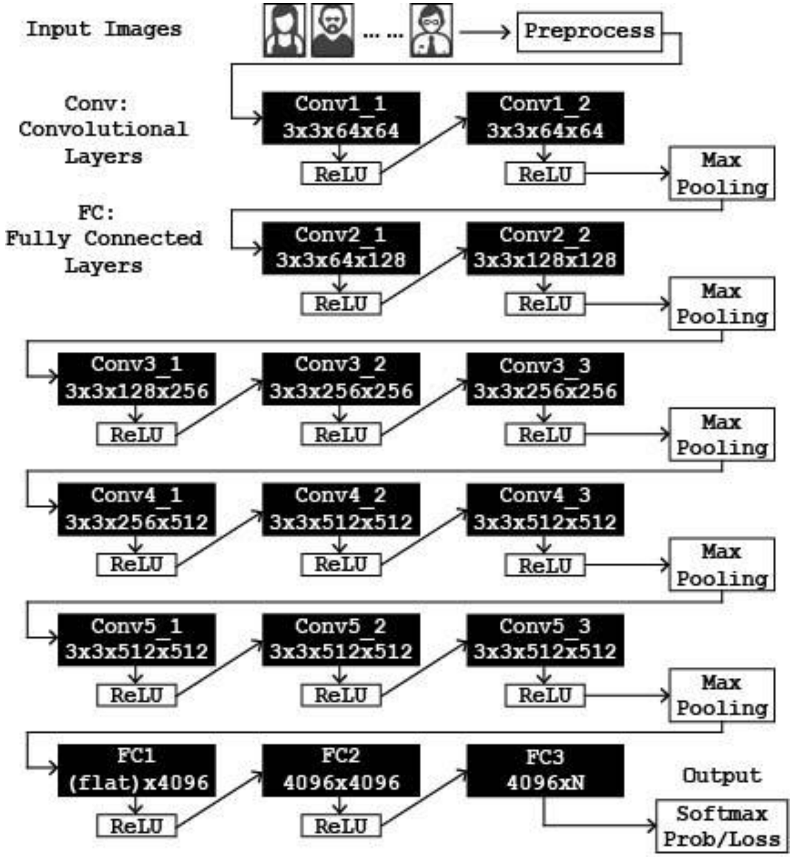
System Model

Edge Computing

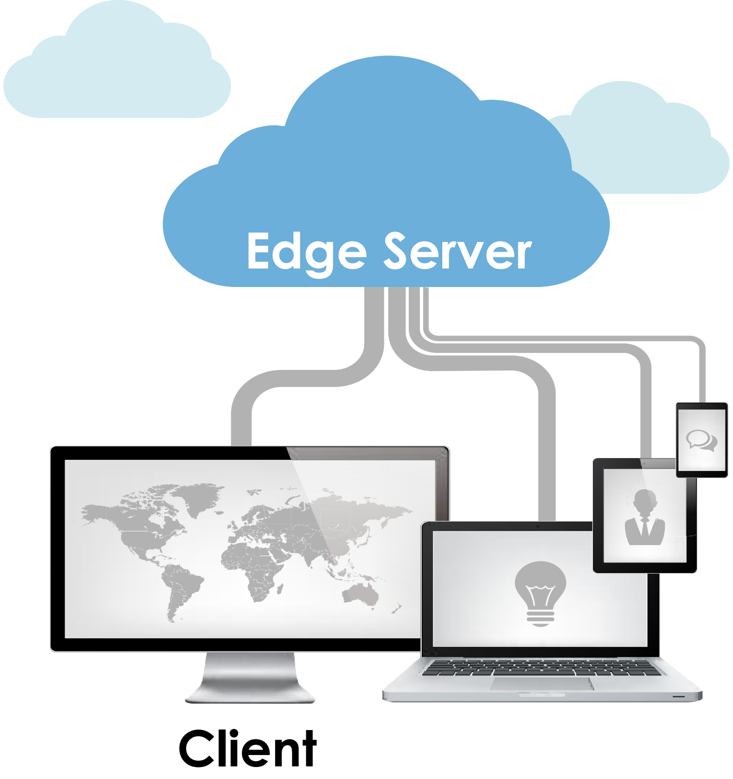
Privacy Threat Model

02 System Model

VGG-Face Network [1]



Client-server Model



1. Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman, "Deep Face Recognition," *BMVC*, 2015

02 System Model

Privacy Leakage Threat

- The adversary we deal against will be untrusted edge server.
- The edge server is assumed to be honest-but-curious.
- The edge server can peek at client’s training data.
- The adversary can launch membership inference attack based on model’s parameters [1-2].

1. Shokri, Reza, et al. "Membership inference attacks against machine learning models." *Security and Privacy (SP), 2017 IEEE Symposium on*. IEEE, 2017.
2. Abadi, Martin, et al. "Deep learning with differential privacy." *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016.

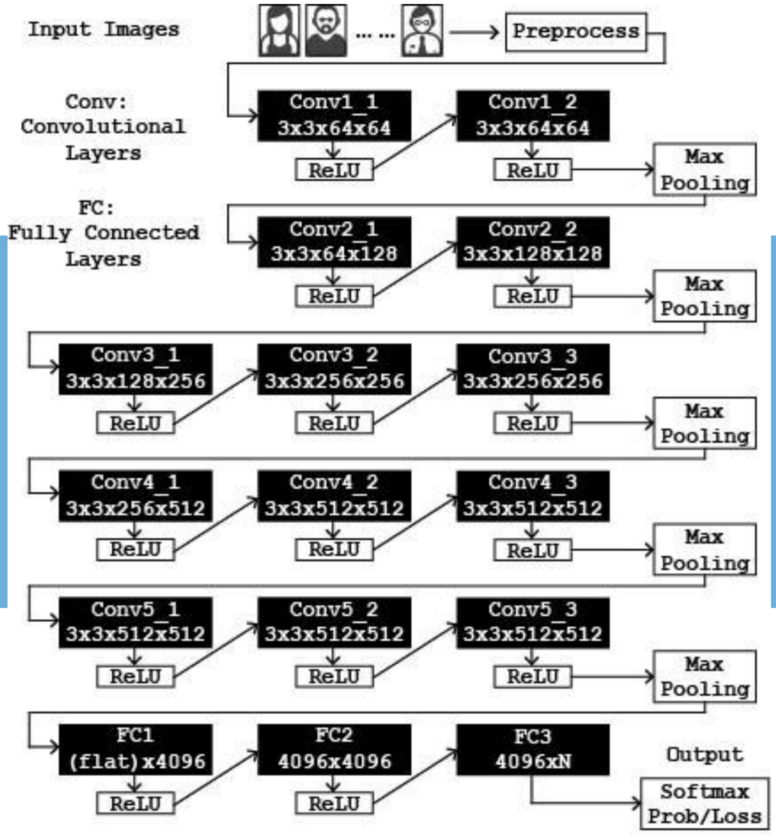
03

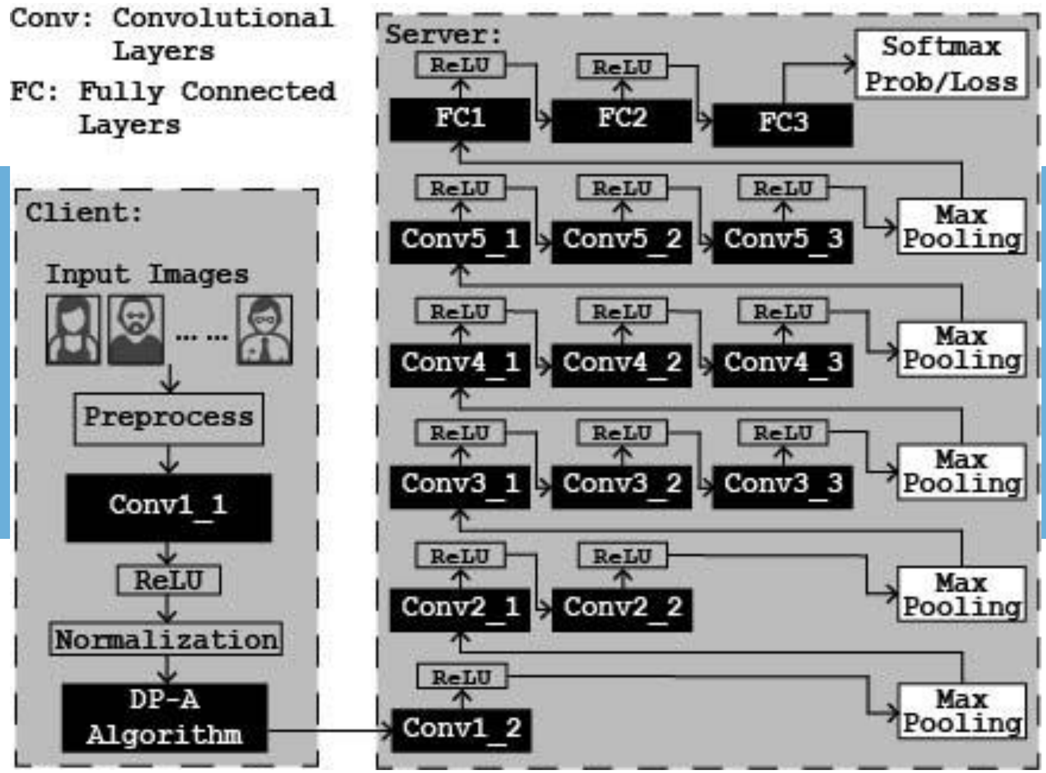
Our DP-A Algorithm

Differentially Private Activations

03 Deep Neural Network Partitioning

- To prevent the adversary from peeking at training data, we partition original DNN into two parts.
- The first part contains a small portion of the network, which pre-processes raw input data.
- The second part containing the rest of the network will be deployed on the edge server.
- To defend against inference attack based on parameters, differential privacy will be applied.



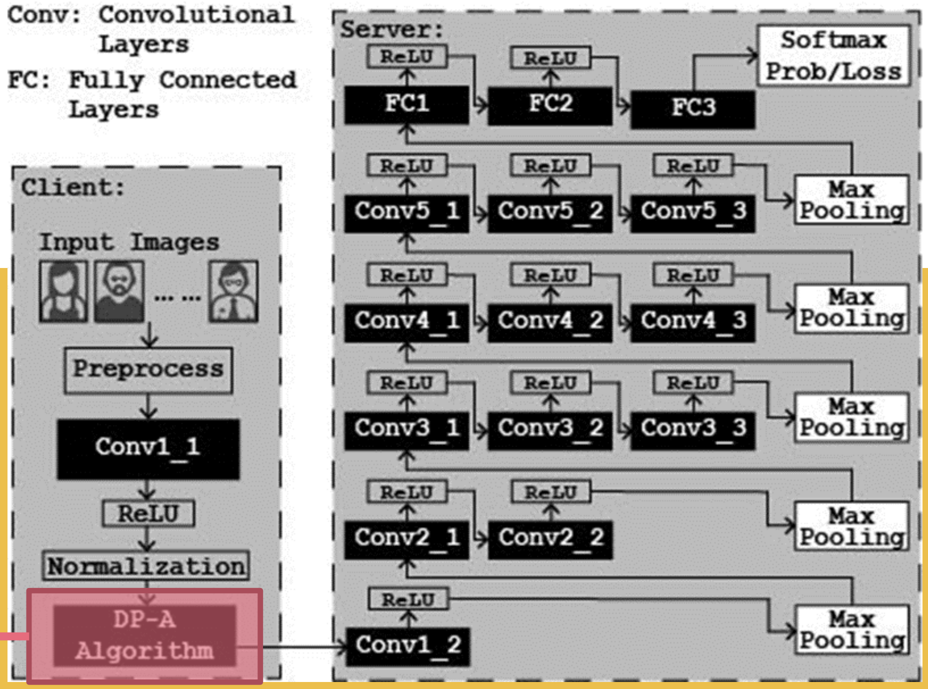


03 Differentially Private Activations Algorithm

DP-A Algorithm

1. Define the sensitivity function of convolutional layer output activations.
2. Create noise addition with regards to a given privacy loss budget.
3. Add artificial noise to the original activations to construct differentially private activations.
4. The edge server takes DP-Activations as its input.

We have proved that parameters will be differentially private if activations in each iteration received from the client are differentially private.

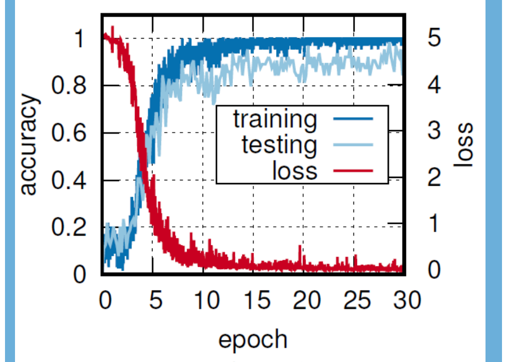


04

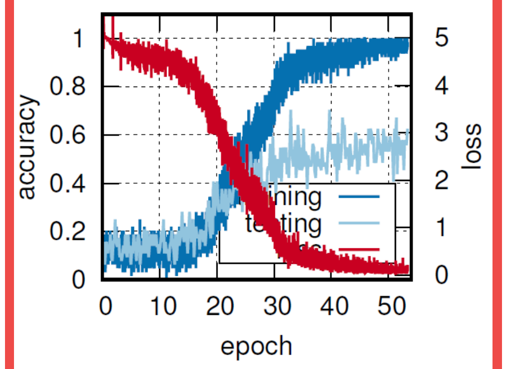
Evaluation Results

04 Evaluation Results

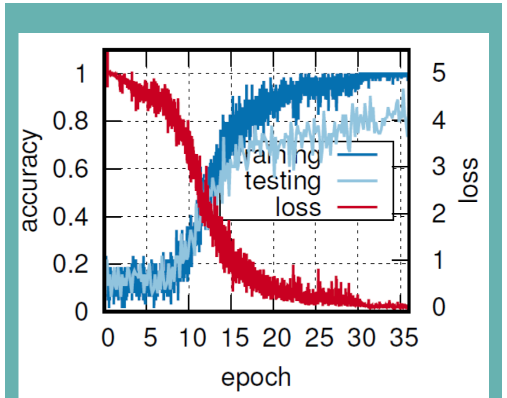
- We use VGG-Face network and Labeled Faces in the Wild (LFW) dataset.
  The network is partitioned at the first convolutional layer.
  Training results of accuracy and loss with regards to different privacy budget are shown in the figures.
  The smaller epsilon is, the more privacy will be preserved.



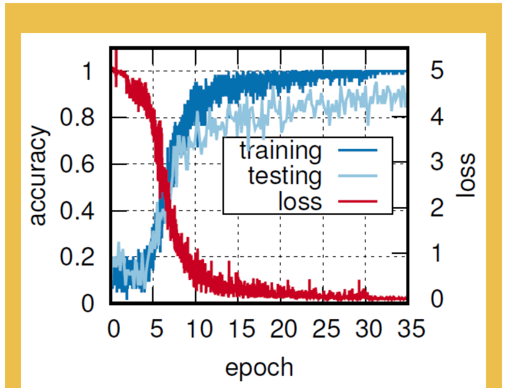
(a) baseline



(b) $\varepsilon = 1$



(c) $\varepsilon = 2$



(d) $\varepsilon = 5$

- Mobile client: Huawei Nexus 6P phone (2GHz Qualcomm Snapdragon 810 processor)
  Edge server: AWS based edge server.
  A batch of training samples from LFW can be loaded on the phone under 0.4s when batch size is 8.
  Forward pass for the batch will be done under 0.6s. Backward pass will cost less than 0.2s per sample.
  The allocated mobile memory usage will be under 500MB for processing a batch of samples. When the mobile phone is processing the first convolutional layer, battery will be consumed under 3.5mAh per minute for the batch.

05 Conclusion

i) what kind of feedback you are looking to receive

- Any helpful idea.
- Which part should be improved.

ii) the controversial points of the paper

- Resource consumption for device user is high.

iii) the open issues the paper does not address

- How to select the optimal partitioning layer.
- Whether our method is capable of training other deep neural networks.

iv) under what circumstances the whole idea might fall apart.

- Unknown for now.

THANK YOU