Attacking and Protecting Data Privacy in Edge–Cloud Collaborative Inference Systems

inference data privacy in edge–cloud collaborative systems

Edge-Cloud Inference System Model:

a DNN is partitioned into two parts:

Where the edge device hosts first part of model as it collects inference data from environment and create an intermediate value which are sent to cloud. The cloud is the hosts second part of model as it receives the intermediate value from edge device and calculates the final output then finally return this output to edge device.

There are 3 factors to be considered to partition the DNN model, 3 of these should be balanced:

1. Latency (inference time/inference speed/transmission time)
2. Power (energy consumption of edge device)
3. Memory Size

一張含有 文字, 螢幕擷取畫面, 圖表, 行 的圖片

自動產生的描述

With these factors, DNN partitioning can become an optimization problem.

Edge–cloud system can achieve lower latency and energy than a cloud-only or an edge-only system: by offloading some DNN layers to the cloud, the processing time and energy consumed on the device is less than the edge-only system.

In real-world scenarios, due to power and resource limitations, edge devices typically only compute a few convolutional layers for feature extraction. Most layers, including all fully connected layers, are usually offloaded to the cloud. This practice potentially provides an opportunity for untrustworthy cloud providers to illicitly access sensitive inference input.

Threat model:

Cloud is untrusted, attempting to steal the input. Cloud cannot compromise the inference process conducted by Edge device (does not know the input x), except the intermediate values .

We consider 3 different adversaries capabilities:

White Box: the Cloud has the knowledge of the model at the edge device (network structure and parameters)

Black Box: the Cloud does not have any knowledge of edge device , but can query the model . It does not need to know the training data and can still collect the same type of samples as the training data, e.g. face samples for face recognition model.

Query Free: the Cloud does not have any knowledge of , or the permission to query the model . However, it can still collect the same type of samples as the training data.

Experimental Configurations:

Standard dataset split into training and testing samples for MNIST CIFAR10.

Set learning rate of 10^-3 and choose ADAM optimizer.

All attack method, defense solutions are implemented with Pytorch.

We are going to run these experiments on Colab.

To measure the effectiveness of attacks and defense, peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) will be used. Larger values of these metrics means that the recovered input x is higher quality.

Attack method:

1. White-Box Attack

Propose using regularized maximum likelihood estimation

(rMLE) to recover the input from the model parameters

and intermediate value

一張含有 文字, 字型, 行, 螢幕擷取畫面 的圖片

自動產生的描述

1. Black-Box Attack

propose the inverse-network attack to identify the reverse

mapping from the intermediate outputs to inputs without the

knowledge of model information

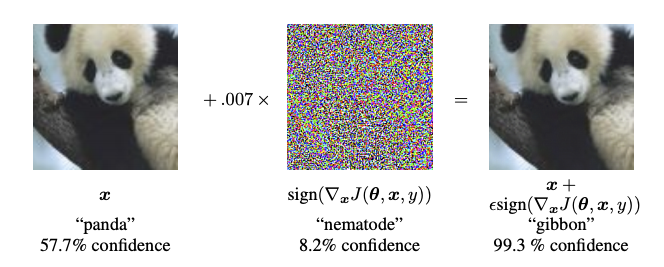
1. Query-Free Attacks

consider the most limited adversarial capability where the cloud has no knowledge of the target model and is not allowed to query the model. Conducting privacy attacks under this setting is extremely difficult, and this threat model is rarely considered in past work. For these query-free attacks, we introduce a new method of shadow model reconstruction to achieve this attack.

More Details/Code:

white-box and black-box. A white-box attack assumes the attacker has full knowledge and access to the model, including architecture, inputs, outputs, and weights. A black-box attack assumes the attacker only has access to the inputs and outputs of the model, and knows nothing about the underlying architecture or weights. There are also several types of goals, including misclassification and source/target misclassification. A goal of misclassification means the adversary only wants the output classification to be wrong but does not care what the new classification is. A source/target misclassification means the adversary wants to alter an image that is originally of a specific source class so that it is classified as a specific target class.

We are going to propose the fast gradient sign method (FGSM) attack which is a white-box attack whose goal is to ensure misclassification.



For example, here is an image of the panda with the class “panda”. The attacks start to adds small perturbation(distortions) to the original image. Then it results in the model mislabeling this image as other class (e.g. “gibbon”) with high confidence.

Process of FGSM:

It works by using the gradient of neural network to create an adversarial example.

It creates a new image (adversarial image) that maximizes the loss using the gradients of the loss with respect to the input image.

How? Find how much each pixel in image contributes to the loss value and then add a perturbation accordingly. The efficiency of this process is attributed to the straightforward method of determining the influence of each input pixel on the loss. This is achieved by applying the chain rule and calculating the necessary gradients, which are then taken with respect to the image itself. Furthermore, as the model is not in training mode, the gradients are not computed with respect to the trainable variables, that is, the parameters of the model. Consequently, these parameters remain unchanged. The sole objective of this process is to deceive a model that has already been trained.

Reference:

FGSM: [[1412.6572] Explaining and Harnessing Adversarial Examples (arxiv.org)](https://arxiv.org/abs/1412.6572)

Defense solution:

1. Obfuscation with random noise (Gaussian noise to defend against a white-box, Laplacian noise to defend against a white-box)
2. Dropout Defense

by deactivating random neurons during the inference, the adversary is not able to precisely generate the original images from the intermediate values

1. Privacy-Aware DNN Partitioning

evaluate different factors that can affect the attack results and propose some guidelines to partition the DL models for better privacy