DeepReDuce: ReLU Reduction for Fast Private Inference

Nandan Kumar Jha, Zahra Ghodsi,

Siddharth Garg, Brandon Reagen

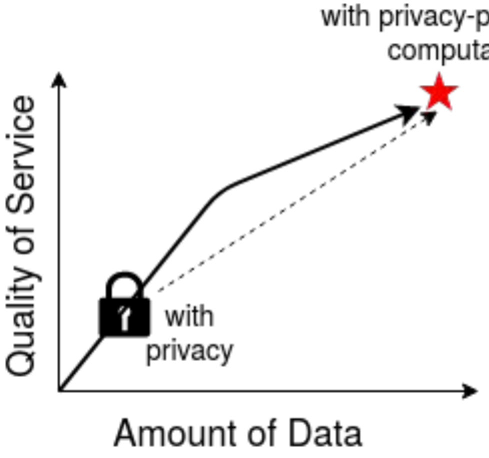
New York University

ICML'21

The Need for Privacy-Preserving Machine Learning

Privacy concerns are growing

Privacy-preserving computation **breaks** the privacy-utility tradeoff.



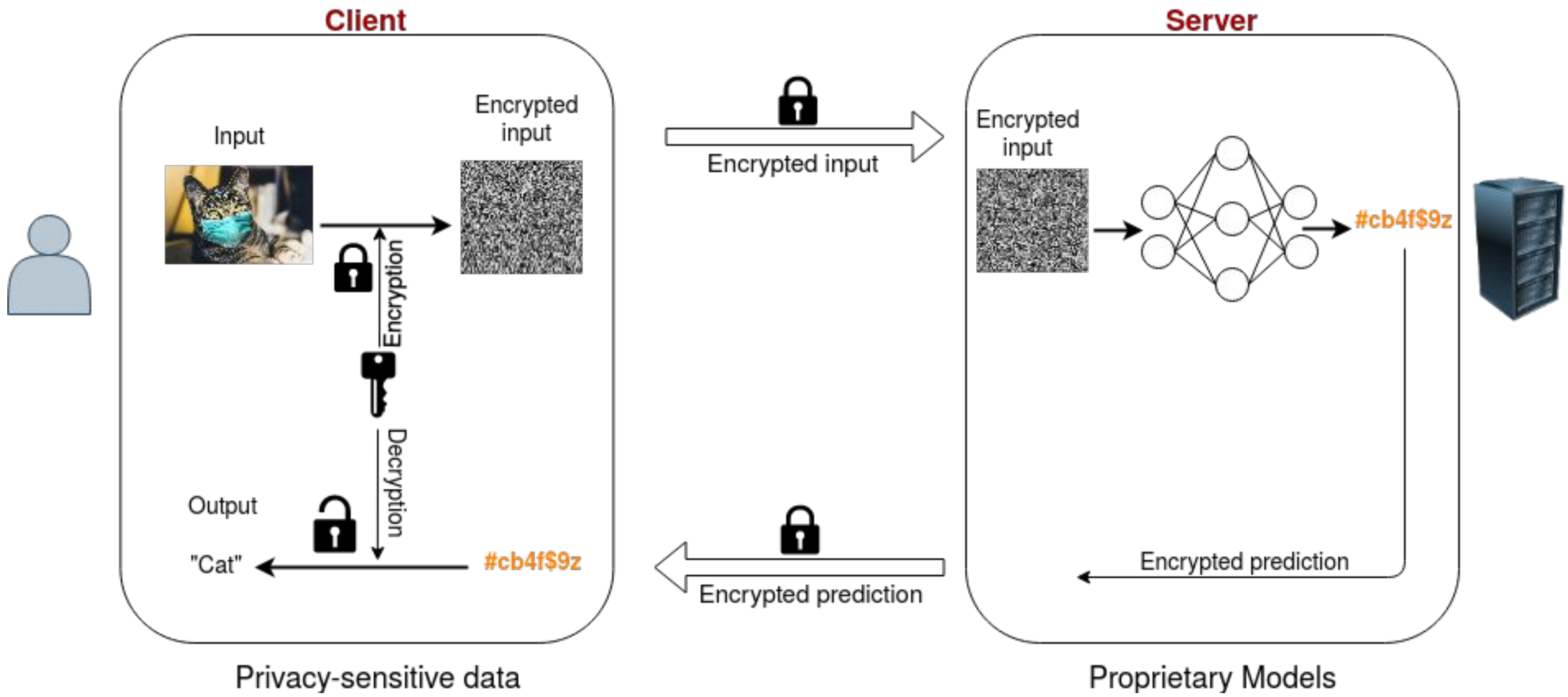
88% companies spent >**$1M** for compliance with GDPR in 2020[1].

1. https://www.itgovernance.eu/blog/en/how-much-does-gdpr-compliance-cost-in-2020

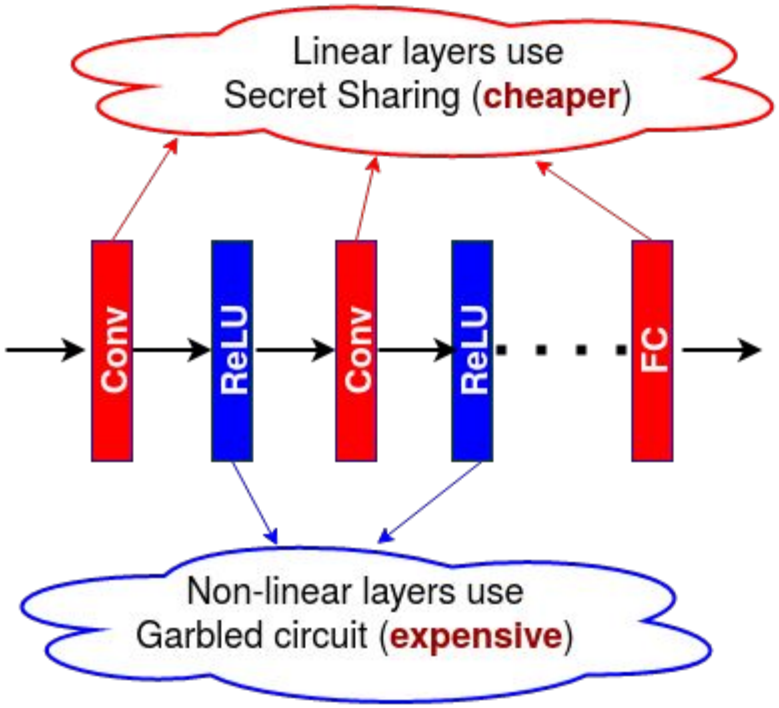
Private Inference

In Private Inference

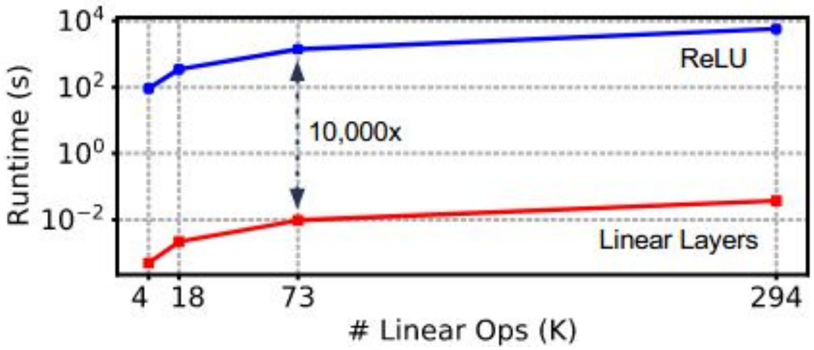
- Client *learns nothing* about Server’s model
- Server *learns nothing* about Client's data.



ReLU is the Source of Slowdown in Private Inference



Inverted operator latency in Private Inference



ReLU dominates the network's private inference time[1]

1. Ghodsi et al., CryptoNAS: Private Inference on a ReLU Budget, NeurIPS'20

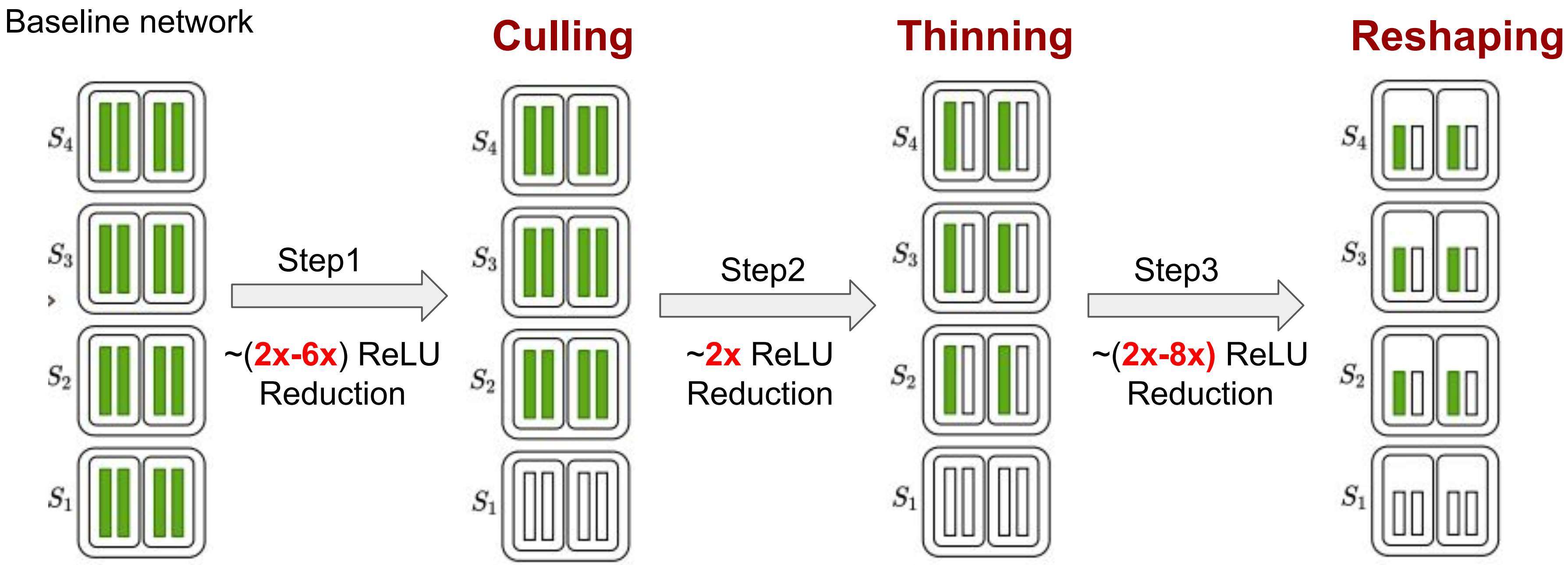
DeepReDuce: ReLU Dropping for Fast Private Inference

If ReLUs are so problematic, can we simply remove them?

Yes, in DeepReDuce we exploit the ReLUs' **heterogeneity** and drop/remove the **less-critical** ReLUs while preserving the **most-critical** ReLUs with negligible impact on accuracy.

We achieve **4.9x** and **5.7x** ReLU reduction on CIFAR-100 and TinyImageNet (respectively) for ResNet18 without losing accuracy.

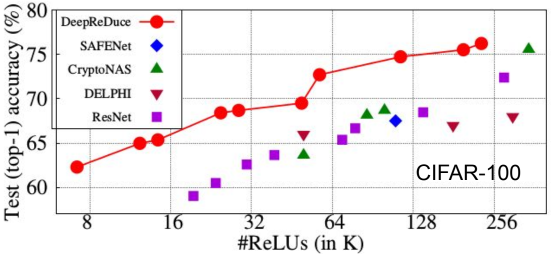
ReLU Optimization in DeepReDuce



Green bars = Layers with ReLUs
White bars = Layers without ReLUs

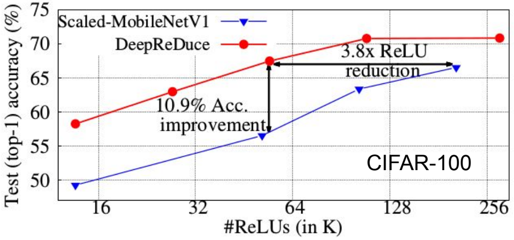
Experimental Results

Comparison with SOTA



3.5% accuracy gain (iso-ReLU), **3.5x** ReLU saving (iso-accuracy)

DeepReDuce on MNetV1



DeepReDuce **generalize** beyond ResNet

Comparison with ch. pruning[1]

	Method	Baseline Acc.(%)	Pruned Acc.(%)	Acc. ↓(%)	FLOPs	ReLUs
C10	Channel pruning	93.59	93.34	-0.25	59.1M	311.7K
	DeepReDuce	93.48	94.07	+0.59	87.7M	221.2K
			93.16	**-0.32**	**66.5M**	**147.5K**
C100	Channel pruning	71.41	70.83	-0.58	60.8M	311.7K
	DeepReDuce	70.93	73.66	+2.57	87.7M	221.2K
			71.68	**+0.59**	**66.5M**	**147.5K**

2x more ReLU savings with similar FLOPs and accuracy

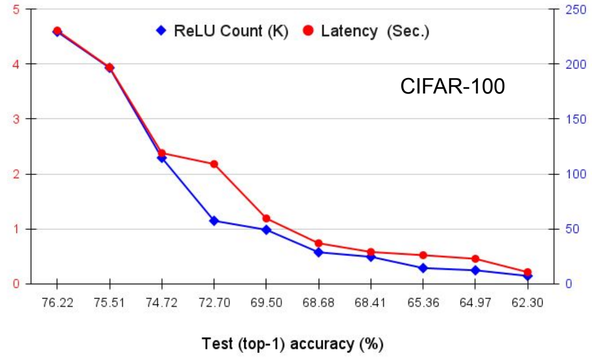
1. He et al., Learning Filter Pruning Criteria for Deep Convolutional Neural Networks Acceleration, CVPR 2020

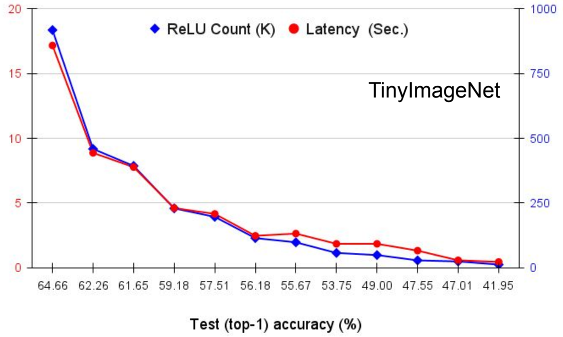
Takeaways from DeepReDuce

1. DeepReDuce strategically drops ReLUs upto **4.9x** with *no loss in accuracy* and achieves **3.5x** ReLU saving over SOTA.

2. The **key insight** is ReLUs *do not equally* contribute to accuracy and less-critical ReLUs can be dropped with negligible accuracy loss.

3. Existing techniques for FLOPs/parameter optimization **are not optimized** for ReLU reduction.





450mS latency (65% accuracy)

4.6S latency (60% accuracy)