Adversarial Robustness of Quantized Embedded Neural Networks

Rémi BERNHARD (CEA Tech)
Pierre-Alain MOELLIC (CEA Tech)
Jean-Max DUTERTRE (MSE)

Laboratoire de Sécurité des Architectures et des Systèmes,
Centre CMP, Equipe Commune CEA-Tech Mines Saint-Etienne,
F-13541 Gardanne France

November 21, 2019, Rennes, France

Outline

Context

Overview – Context

- **Neural networks**: state-of-the art performances in various complex tasks
  → Classical requirements: tremendous computation power and storage limitations
- **Major trend**: Massive efforts for models deployment and embedded ML-systems
  → Mobile phones, Internet of things, . . .
- **Major constraints**: Energy/Memory/Precision depending on the platform
  → From typical microcontroller to complex SoC




- Important threats against the **Confidentiality** / **Integrity** and **Accessibility** of Machine Learning systems.
  → Significant body of works in the ML community focused on these topics.

- **Adversarial examples**: threaten networks' integrity
  → Malicious perturbations which aim at fooling a model
  - Szegedy et al., *Intriguing properties of Neural Networks*, 2013
  - Goodfellow et al., *Explaining and harnessing adversarial examples*, 2015

What is the impact of quantization on adversarial examples ?

Adversarial Machine Learning

Adversarial Examples

Adversarial Examples: Attacking Integrity (at inference time)

Principle: Craft maliciously modified examples to fool a model.

Adversarial example = Clean example + Adversarial perturbation

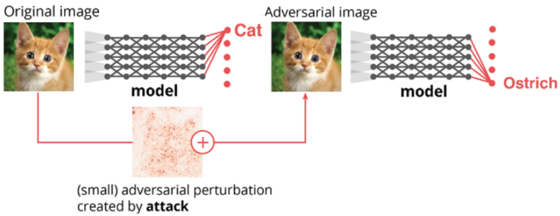
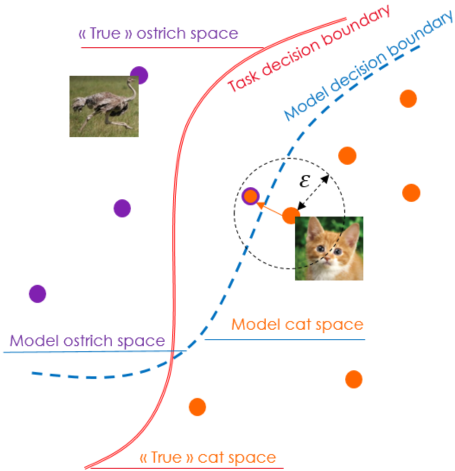


Figure: NIPS 2018 Adversarial Vision Challenge

- Classification errors
- Serious threat for critical decision systems

Adversarial Examples



"Stop" "Speed Limit 45"

Figure: Eykholt et al., *Robust Physical-World Attacks on Deep Learning Visual Classification*, 2018

C: number of labels
M_w: target classifier
$(x, y) \in \mathbb{R}^d \times \{1, ..., C\}$: observation with ground-truth label
$M(x) \in \{1, ..., C\}$: predicted label of x by M
$F(x) \in \mathbb{R}^C$: output probabilities (softmax) for x
$f(x) \in \mathbb{R}^C$: pre-softmax (logits) for x
$L(w, x, y) \in \mathbb{R}$: loss function of M

Pipeline:

$$\underbrace{M_w}_{model} : \underbrace{x}_{input} \rightarrow \underbrace{f(x)}_{logits} \rightarrow \underbrace{F(x)}_{softmax} \rightarrow \underbrace{M(x)}_{\substack{\text{predicted} \\ label}}$$

Adversarial goal: Fool a model at inference time

From $(x, M(x))$ with $M(x) = y$ (true label), craft $(x', M(x'))$ with

- $M(x') \neq M(x)$: ***untargeted* attack**
- $M(x') = t$: ***targeted* attack** towards label t

Adversarial capabilities: How much can the adversary alter x ?

$$x' = x + \alpha \text{ (α: adversarial perturbation)}$$

l_p norm-bounded adversarial examples: $\|\alpha\|_p \leq \epsilon$
$\rightarrow$ Classical attacks: l_2 or l_∞ (some l_0 attacks)

Adversarial Examples: **Threat Model**

Adversarial knowledge: What does the adversary know about the target model M?

- ***White-box*** setting: model's architecture and parameters
  $\rightarrow$ gradients available
- ***Black-box*** setting: model's outputs only
  $\rightarrow$ no knowledge of the gradients
  $\rightarrow$ can query M, with/without restriction
  $\rightarrow$ probability outputs, $F(x)$, or label output, $M(x)$

Adversarial Examples: **Transferability**

Principle:
An adversarial example crafted to fool classifier M_1 may fool another classifier M_2

$\rightarrow$ **For the adversary, a very powerful property**

Remarks:

- Inter and Intra-techniques transferability
- Need to train a substitute model

FGSM Attack

one-step, $\|\alpha\|_\infty \leq \epsilon$ as a constraint

BIM Attack

iterative version of FGSM

$\rightarrow$ **Principle:** Maximization of $L(\theta, x, y)$ with respect to x, s.t. $\|\alpha\|_\infty \leq \epsilon$

CWl2 Attack

iterative, minimization of $\|\alpha\|_2$ as an objective

$\rightarrow$ **Principle:** Minimization of $\|\alpha\|_2 + c\,K(x + \alpha, y)$ with respect to α

The adversary needs to be able to compute gradients.

ZOO Attack

iterative, minimization of $\|\alpha\|_2$ as an objective

$\rightarrow$ **Principle:** Same as for CWl2, discrete approximation of derivative is used

SPSA Attack

iterative, $\|\alpha\|_\infty \leq \epsilon$ as a constraint

$\rightarrow$ **Principle:** Minimization of $f_{M(x)}(x+\alpha) - \max_{j \neq M(x)} f_j(x')$ with respect to α, s.t. $\|\alpha\|_\infty \leq \epsilon$

The adversary approximates gradients.

Principle of Gradient Masking:
Make gradients useless to craft adversarial examples

Remarks:

- A false sense of security (Uesato, 2018)
- An adversary can use a substitute model to circumvent it.
- Gradient-free attacks, decision-based attacks, ...

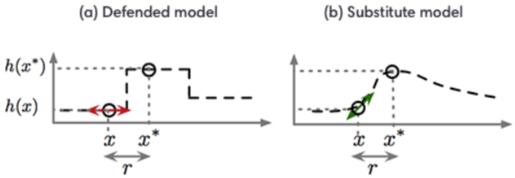


Figure: Goodfellow et al., *Attacking Machine Learning with Adversarial Examples*, openAI blog, 2017

Neural Networks Quantization

Quantization post-training

Principle: Quantize weights and/or activation values after the training phase.
Issues: Coarsely quantizing weights into – usually – no more than INT8.

Quantization-aware training

Principle: Learn a model with quantized weights and/or activation values during the training
Issues:

- Manage non-differentiability issues of quantization function during backward pass
- Training can be difficult

Binary Net (Courbariaux, Bengio et al. 2015 & 2016):

- Binarization: $w_b = sign(w)$, $a_b^k = sign(a^k)$
- Inference: only *bitcount* and *xnor* operations

Dorefa Net (Zhou et al. 2016):

- Quantization: n-bit width quantization of weights, activation and gradients
- Inference: bit convolution kernel

$\rightarrow$ Backward pass: usage of a *Straigth Through Estimator* (STE, Bengio et al., 2013)

Previous Work

Massive research efforts on the topic (both attacks and defenses) with associated benchmarks and competitions (*NIPS Adversarial Vision Challenge*) **but almost only on full-precision models**.

Existing works bridging quantization and adversarial robustness:

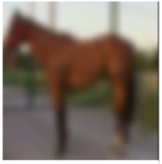
- Galloway, 2017 (*Attacking binarized neural networks*): claims natural robustness with binarization. But, MNIST only, stochastic quantization
- Khalil, 2018 (*Combinatorial attacks on binarized networks*): not scalable on big data sets
- Lin, 2019 (*Efficiency Meets Robustness*): FGSM attack only, white-box setting only (no transferability analysis)

Experiments: Robustness Evaluation

Experiments: **Setup**

Data sets:

- SVHN (73,257/26,032)
- CIFAR10 (50,000/10,000)



Models:

- One full-precision (32-bit float) model for each data set (same CNN architecture as in Courbariaux et al., 2016)
- Weight quantized models: 1,2,3,4 bits
- Weight and activation (*fully*) quantized models: 1,2,3,4 bits

Techniques: BinaryNet and DorefaNet

Experiments: **Training results**

	CIFAR10				SVHN			
Full-precision	0.89				0.96			
Bitwidth	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
Full quantization	0.79	0.87	0.88	0.88	0.89	0.95	0.95	0.95
Weight quantization	0.88	0.88	0.88	0.88	0.96	0.95	0.96	0.95

Table: Models accuracy on test set

During training, quantization acts as a:

- constraint
- regularizer

Experiments: **Attacks and metrics**

	FGSM	BIM	CWL2	SPSA	ZOO
Gradient-based	✓	✓	✓		
Gradient-free				✓	✓
one-step	✓				
iterative		✓	✓	✓	✓
l_∞	✓	✓		✓	
l_2			✓		✓

Adversarial accuracy: accuracy of the model on adversarial examples

l_p adversarial **distortion**:

$$\|x' - x\|_p = \left(\sum_{i=1}^{m} |x'_i - x_i|^p\right)^{\frac{1}{p}}$$

Experiments: **Fully quantized models**

	CIFAR10						SVHN					
	Float model *(32-bit)*			Binarized models *(1-bit)*			Float model *(32-bit)*			Binarized models *(1-bit)*		
	acc	l_2	l_∞	acc	l_2	l_∞	acc	l_2	l_∞	acc	l_2	l_∞
FGSM	0.12	1.65	0.03	**0.66**	1.65	0.03	0.29	1.66	0.03	**0.78**	1.64	0.03
BIM	0.07	1.17	0.03	**0.66**	1.01	0.03	0.05	1.16	0.03	**0.79**	1.0	0.03
CWl2	0.03	0.58	0.04	**0.11**	0.78	0.08	0.02	0.64	0.66	**0.06**	1.02	0.1

1) Fully binarized neural networks:

- Apparent robustness against FGSM and BIM attacks
- No robustness increase against CWl2 attack

$\rightarrow$ No additional robustness against gradient based attacks

Experiments: **Gradient masking**

	CIFAR10						SVHN					
	Float model *(32-bit)*			Quantized models *(1,2,3,4-bit)*			Float model *(32-bit)*			Quantized models *(1,2,3,4-bit)*		
	acc	l_2	l_∞	acc	l_2	l_∞	acc	l_2	l_∞	acc	l_2	l_∞
BIM	0.07	1.17	0.03	**0.66**	1.01	0.03	0.05	1.16	0.03	**0.79**	1.0	0.03
				0.06	1.14	0.03				**0.11**	1.13	0.03
				0.11	1.17	0.03				**0.11**	1.13	0.03
				0.06	1.14	0.03				**0.1**	1.13	0.03
SPSA	0.0	1.37	0.03	**0.16**	1.31	0.03	0.01	1.38	0.03	**0.4**	1.32	0.03
				0.0	1.34	0.03				**0.14**	1.34	0.03
				0.0	1.36	0.03				**0.07**	1.35	0.03
				0.0	1.36	0.03				**0.04**	1.37	0.03

2) Fully quantized neural networks:
BIM (gradient-based, l_∞) less efficient than SPSA (gradient-free, l_∞)
$\rightarrow$ Gradient masking

Experiments: **Gradient masking**

	CIFAR10						SVHN					
	Float model *(32-bit)*			Quantized models *(1,2,3,4-bit)*			Float model *(32-bit)*			Quantized models *(1,2,3,4-bit)*		
	acc	l_2	l_∞	acc	l_2	l_∞	acc	l_2	l_∞	acc	l_2	l_∞
CWl2	0.03	0.58	0.04	**0.11** **0.06** **0.09** **0.05**	**0.78** **0.6** **0.55** **0.6**	0.08 0.04 0.04 0.04	0.02	0.64	0.06	**0.06** **0.03** **0.02** **0.02**	**1.02** **0.67** **0.66** **0.68**	0.1 0.07 0.07 0.07
ZOO	0.0	0.72	0.09	**0.56** **0.83** **0.76** **0.73**	**0.1** **0.13** **0.24** **1.09**	0.05 0.06 0.07 0.14	0.0	0.91	0.11	**0.82** **0.93** **0.94** **0.93**	**0.07** **0.1** **0.11** **0.38**	0.05 0.06 0.05 0.1

3) Fully quantized neural networks:

- Quantization alters ZOO objective function ($\simeq 0$ or $>> 1$)
  $\rightarrow$ ZOO fails, CWl2 succeeds (thanks to STE)
- No effect from quantization
  $\rightarrow$ ZOO performs better (l_2 distortion)

$\rightarrow$ Gradient masking

Poor transferability capacities

	float	w_1a_1	w_1a_{32}	w_2a_2	w_2a_{32}	w_3a_3	w_3a_{32}	w_4a_4	w_4a_{32}
float	0.12	0.58	0.38	0.41	0.42	0.40	0.40	0.40	0.39
w_1a_1	0.82	0.66	0.82	0.81	0.81	0.72	0.81	0.80	0.80
w_1a_{32}	0.33	0.61	0.11	0.37	0.38	0.42	0.37	0.36	0.36
w_3a_3	0.49	0.62	0.51	0.43	0.44	0.17	0.44	0.43	0.43
w_3a_{32}	0.46	0.61	0.47	0.38	0.39	0.45	0.18	0.38	0.38

FGSM

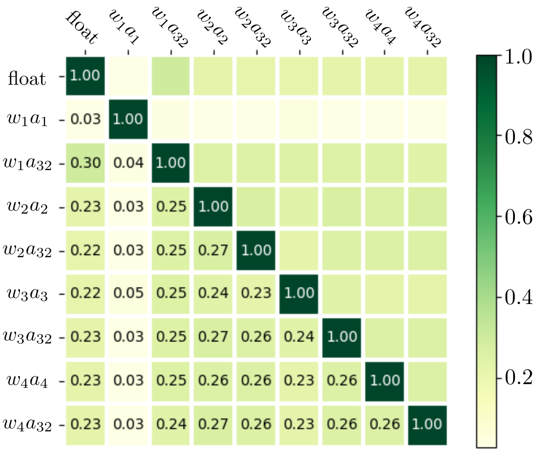
Quantization Shift Phenomenon: Quantization ruins the adversarial effect

- activation shift:
  Two activation values mapped to the same quantization bucket
- weight shift:
  Weight quantization can cancel adversarial effect

Gradient misalignment:

Cosinus similarity values near 0
$\rightarrow$ near orthogonal gradients

Hard to transfer from/to fully binarized networks



Experiments: Ensemble Defense

Ensemble Defense: **Motivation**

Observations:

Fully quantized (1, 2, 3 and 4 bits) models:

- **More likely to disagree on successful adversarial examples**
- **More likely to agree on unsucessful adversarial examples**

Idea:

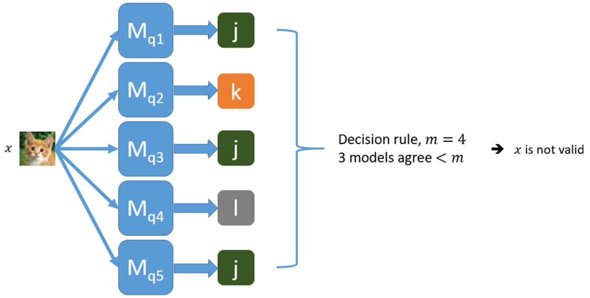
Ensemble-based defense to take advantage of this sieve phenomenon

Realization:

Define a proper prediction criterion considering the trade-off test set accuracy / adversarial accuracy

$\rightarrow$ perform prediction for the most <u>well-classified</u> examples and the fewest <u>adversarial examples</u>

An input is said *valid* if more than m models agree.



m regulates the adjustment of the clean/adversarial accuracy trade-off.

$valid_{m,\mathcal{M}}(X)$ is the ensemble of valid inputs from X.
Then, the *Prediction Rate* (**PR**) is

$$PR_{m,\mathcal{M}}(X) = \frac{|valid_{m,\mathcal{M}}(X)|}{|X|}$$

For CIFAR10 ($m = 4$) and SVHN ($m = 5$), the prediction is performed for 87% of the clean test set:

	CIFAR10		SVHN	
	PR	accuracy	PR	accuracy
Test set	0.87	0.90	0.87	0.98

Figure: Ensemble test set accuracy

Ensemble Defense: **Results**

When evaluating on the adversarial test set X':
Defense Accuracy (**d_acc**): proportion of adversarial examples filtered out or unsuccessful.

Main results and observations:

- Better results for SVHN than CIFAR10
- Ensemble of quantized models shows better robustness to transferred adversarial examples than all single models, *if* the adversarial examples are not crafted on a fully binarized model
- Interesting results for the powerful CWl2 attack:
  $d\_acc^{CIFAR10} = 0.53$ and $d\_acc^{SVHN} = 0.8$.

Conclusion

Conclusion

Complete study of quantized models vulnerabilities against adversarial examples, under various threat models.

Take-away:

- Quantization is not a robust *"natural"* defense when facing advanced attacks
  $\rightarrow$ Detection of some gradient masking issues
- But, interestingly, gradient misalignment and *quantization shift phenomenon* cause poor transferability
- This enables to build a defense based on an ensemble of quantized models

Thank you for your attention

Contact

Secure Architectures and Softwares, *SAS*

Centre de Microélectronique Provence, Gardanne (13)

- Remi Bernhard: *remi.bernhard@cea.fr*
- Pierre-Alain Moellic: *pierre-alain.moellic@cea.fr*
- Jean-Max Dutertre: *dutertre@emse.fr*