Introduction to Part 5

- **So far**: Various machine learning models, focused on functionality and performance.
- **Part 5: Deployment Issues**
  - Security: Adversarial Attack
  - Trust: Explainable AI (XAI)

  - Privacy: Federated learning
  - Fairness
  - Domain shift
  - ...

- Deep neural networks have demonstrated phenomenal success (often beyond human capabilities) in solving complex problems.
- However, Szegedy *et al.* 2014 discovered that deep networks are surprisingly susceptible to adversarial attacks in the form of small perturbations to images that remain (almost) imperceptible to human vision system.
- Such attacks can cause a neural network classifier to completely change its prediction about the image, and with high confidence.

Adversarial Examples

- **Adversarial Examples**: Inputs formed by applying small but intentionally worst-case perturbations to examples from the dataset, such that the perturbed input results in the model outputting an incorrect answer with high confidence. [Goodfellow *at al.* 2015]

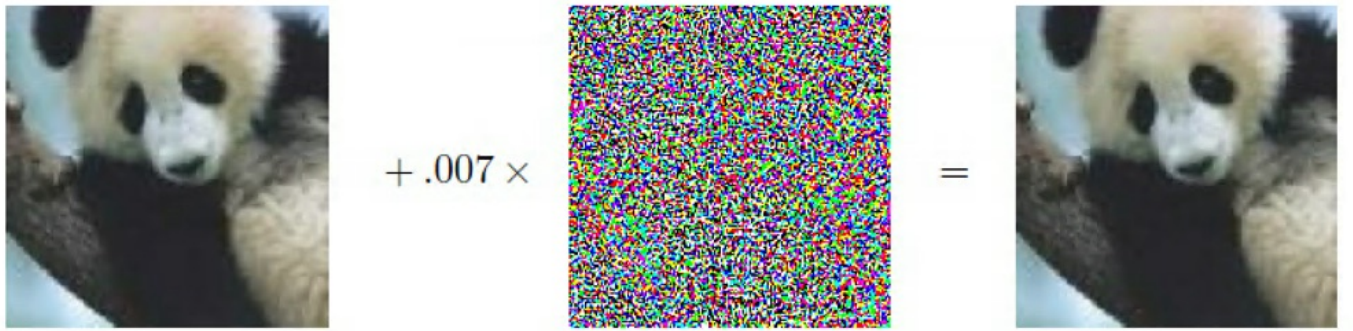


Figure 1: A demonstration of fast adversarial example generation applied to GoogLeNet (Szegedy et al., 2014a) on ImageNet. By adding an imperceptibly small vector whose elements are equal to the sign of the elements of the gradient of the cost function with respect to the input, we can change GoogLeNet's classification of the image. Here our ϵ of .007 corresponds to the magnitude of the smallest bit of an 8 bit image encoding after GoogLeNet's conversion to real numbers.

- Left: Graffiti on a Stop sign, something that most humans would not think is suspicious.
- Right: Adversarial sticker added to a Stop sign. We design our perturbations to mimic graffiti, and thus "hide in the human psyche."

Reese Witherspoon (left) wearing glasses (middle) impersonating Russel Crowe (Right) (IC: https://www.cs.cmu.edu/~sbhagava/papers/face-rec-ccs16.pdf)

Demos

Plan

- General principle
- Attack algorithms (FGSM, CW)
- Transferrability
- Defense
- Tutorial

Basic Concepts

Training vs Adversarial Example Generation

Fast Gradient Sign Method (FGSM) : Targeted

Fast Gradient Sign Method (FGSM) : Untargeted

Iterative FGSM

Targeted

$$\begin{aligned} \mathbf{x}'_0 &= \mathbf{x} \\ \mathbf{x}'_i &= clip_{\mathbf{x},\epsilon}[\mathbf{x}'_{i-1} - \alpha \text{sign}(\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}'_{i-1}, t))] \end{aligned}$$

- Add perturbations to $\mathbf{x}$ in multiple steps.
- At each step, a smaller step size α is used.
- $clip_{\mathbf{x},\epsilon}$ clips values at ϵ.

Untargeted

$$\begin{aligned} \mathbf{x}'_0 &= \mathbf{x} \\ \mathbf{x}'_i &= clip_{\mathbf{x},\epsilon}[\mathbf{x}'_{i-1} + \alpha \text{sign}(\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}'_{i-1}, y))] \end{aligned}$$

- Add perturbations to $\mathbf{x}$ in multiple steps.
- At each step, a smaller step size α is used.
- $clip_{\mathbf{x},\epsilon}$ clips values so that $\mathbf{x}'_i$ stays within the L_∞ ϵ-ball around $\mathbf{x}$.
- This is often called the **basic iterative method (BIM)**, and is equivalent to **projected gradient descent (PGD)** under L_∞, except the latter starts with a random point on the ϵ-ball around $\mathbf{x}$ instead of $\mathbf{x}$ itself.

Effectiveness of Attack

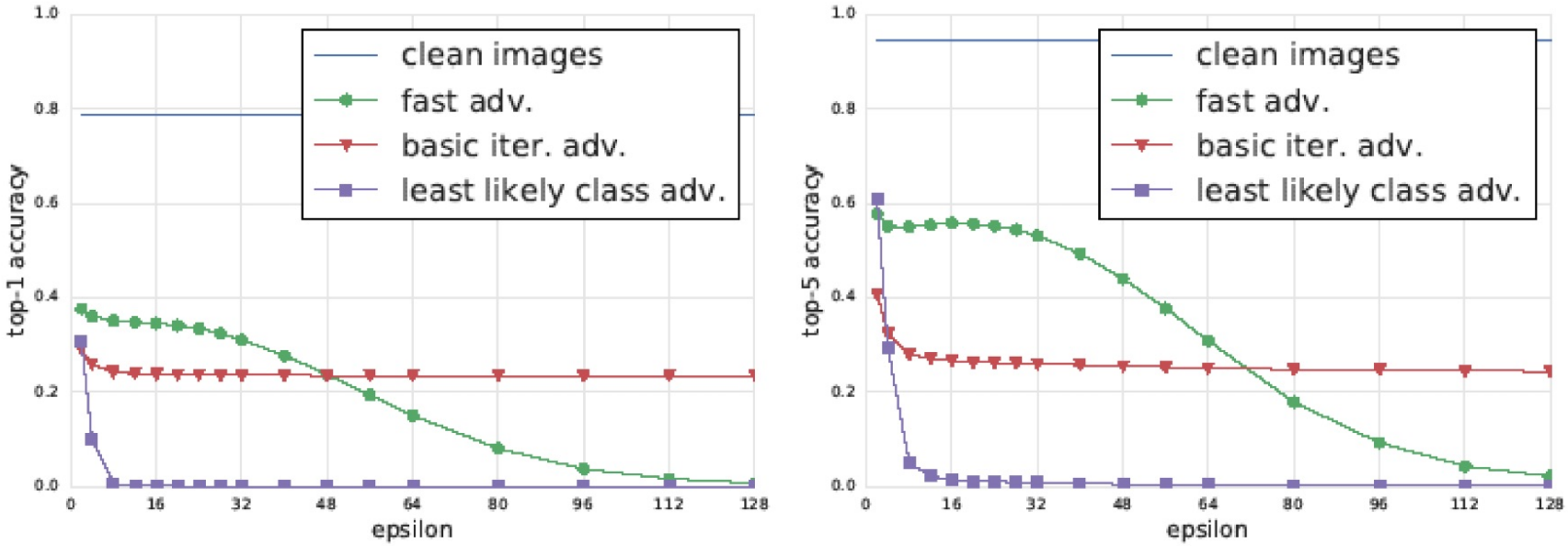


Figure 2: Top-1 and top-5 accuracy of Inception v3 under attack by different adversarial methods and different ϵ compared to "clean images" — unmodified images from the dataset. The accuracy was computed on all $50,000$ validation images from the ImageNet dataset. In these experiments ϵ varies from 2 to 128.

Tutorial

* CIFAR10 : Results

* Data Gradient : $\nabla_x L$

```
loss = torch.nn.functional.cross_entropy(pred_logits, original_lbl)
model.zero_grad()

loss.backward()

data_grad = original_img.grad.data
eps = 0.02
attacked_img = fgsm(original_img, eps, data_grad)
```

* Untargeted FGSM

```
def fgsm(img, eps, grad):
    sign_grad = grad.sign()   # Compute the sign of gradient

    attacked_img = img + eps * sign_grad   # Compute the perturbation

    attacked_img = torch.clamp(attacked_img, 0, 1)   # Clip the attac

    return attacked_img
```

Carlini and Wagner Margin-Based Attack [2017b] (Powerful)

Threat Models

Adversarial examples can be generated under different sittings:

So far:

- **White-box attacks** assume the complete knowledge of the targeted model, including its parameter values, architecture, training method, and in some cases its training data as well.

Next:

- **Black-box attacks** feed a targeted model with the adversarial examples (during testing) that are generated without the knowledge of that model:
  - In some instances, it is assumed that the adversary has a limited knowledge of the model (e.g. its training procedure and/or its architecture) but definitely does not know about the model parameters.

Black-Box Attack

Transferability Discovered by Szegedy *et al.* [2014], results from Liu *et al.* [2017]

- Adversarial examples generated for one model often fool other models with different structures and trained on different datasets.
- Results (Accuracy):

Models used to test adversarial examples

Models used to generate adversarial examples

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	22.83	0%	13%	18%	19%	11%
ResNet-101	23.81	19%	0%	21%	21%	12%
ResNet-50	22.86	23%	20%	0%	21%	18%
VGG-16	22.51	22%	17%	17%	0%	5%
GoogLeNet	22.58	39%	38%	34%	19%	0%

Panel A: Optimization-based approach

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	23.45	4%	13%	13%	20%	12%
ResNet-101	23.49	19%	4%	11%	23%	13%
ResNet-50	23.49	25%	19%	5%	25%	14%
VGG-16	23.73	20%	16%	15%	1%	7%
GoogLeNet	23.45	25%	25%	17%	19%	1%

Panel B: Fast gradient approach

Table 1: Transferability of non-targeted adversarial images generated between pairs of models. The first column indicates the average RMSD of all adversarial images generated for the model in the corresponding row. The cell (i, j) indicates the accuracy of the adversarial images generated for model i (row) evaluated over model j (column). Results of top-5 accuracy can be found in our online technical report: Liu et al. (2016).

Transferability

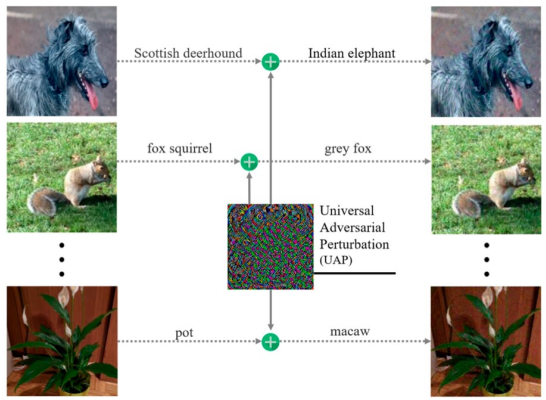
- Ensemble-based approach: Generate adversarial examples for an ensemble of models. Such adversarial examples transfer better.

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
-ResNet-152	17.17	0%	0%	0%	0%	0%
-ResNet-101	17.25	0%	1%	0%	0%	0%
-ResNet-50	17.25	0%	0%	2%	0%	0%
-VGG-16	17.80	0%	0%	0%	6%	0%
-GoogLeNet	17.41	0%	0%	0%	0%	5%

Table 4: Accuracy of non-targeted adversarial images generated using the optimization-based approach. The first column indicates the average RMSD of the generated adversarial images. Cell (i, j) corresponds to the accuracy of the attack generated using four models except model i (row) when evaluated over model j (column). In each row, the minus sign "–" indicates that the model of the row is not used when generating the attacks. Results of top-5 accuracy can be found in our online technical report: Liu et al. (2016).

Universal Adversarial Perturbations

- There exist universal (image-agnostic) and very small perturbation vector that causes multiple images to be misclassified with high probability. [Moosavi-Dezfooli *et al.* 2017]

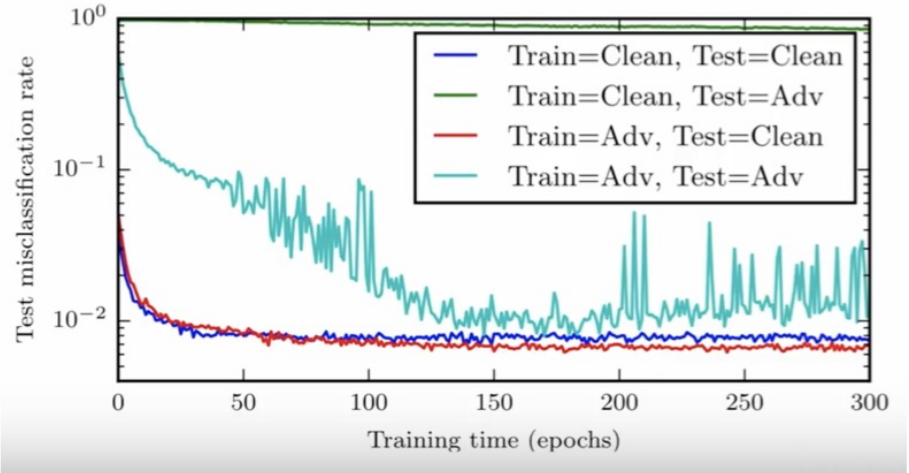


Adversarial Training

* One of the most effective ways to defend against adversarial attacks, BUT far from being perfect

Adversarial Training [Szegedy *et al.* 2014]

- Add adversarial examples to data and re-train model (Patch up weak spots)



- Blue: Original model (trained on clean data) has low error rate on test data.
- Green:Original model misclassified most adversarial examples
- Light Blue: Adversarially trained model is more robust again adversarial examples. However, it is not effective to attacks not considered in training.
- Red: Adversarial training has regularization effects.