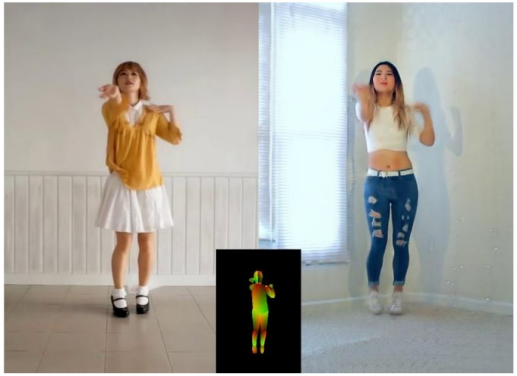
Adversarial attacks on DNNs

Ksenia Demskaya,
Research Engineer

Success of Deep Learning



Video-to-Video Synthesis [Ting-Chun Wang]

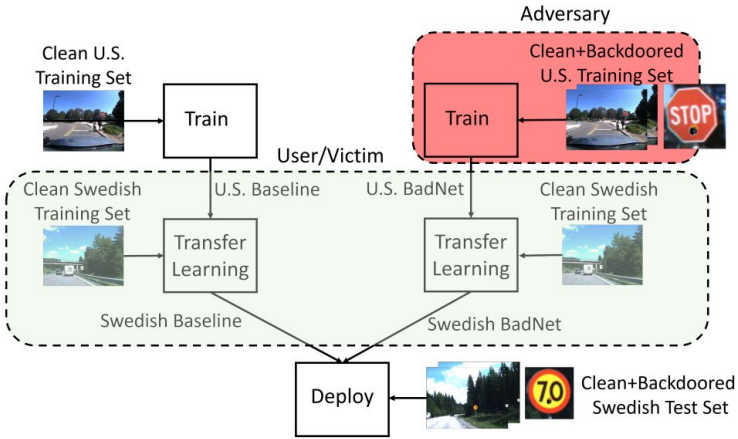
OpenAI Five Dota2 bots

Input sentence:	*Translation (PBMT):*	*Translation (GNMT):*	*Translation (human):*
李克強此行將啟動中加總理年度對話機制，與加拿大總理杜魯多舉行兩國總理首次年度對話。	Li Keqiang premier added this line to start the annual dialogue mechanism with the Canadian Prime Minister Trudeau two prime ministers held its first annual session.	Li Keqiang will start the annual dialogue mechanism with Prime Minister Trudeau of Canada and hold the first annual dialogue between the two premiers.	Li Keqiang will initiate the annual dialogue mechanism between premiers of China and Canada during this visit, and hold the first annual dialogue with Premier Trudeau of Canada.

Google machine translation

Can we rely on DL system?

- ***Data poisoning***: inject malicious points into the models' *training* sets
- ***Trojaning attack***: inserting malicious hardware Trojans in the implementation of a neural network classifier
- ***Adversarial attack***: at inference time



* BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain [Gu et al., 2017]

* Hardware Trojan Attacks on Neural Networks [Clements et al., 2018]

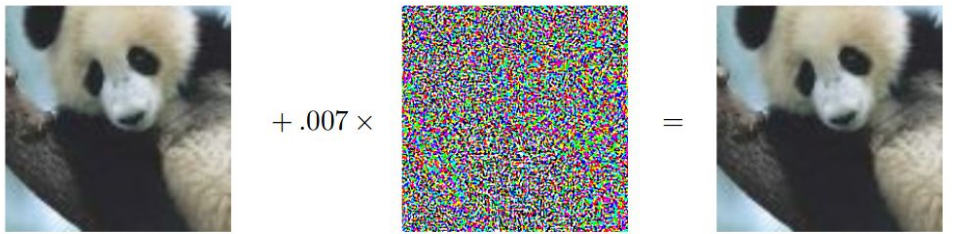
Structure of the talk

- Adversarial attacks:
  - *what* is this
  - *why* research this and what are the *areas* of research
  - *types* of attacks
  - *construction*
  - problems with adversarial attacks in real world

- Defense against adversarial attacks:
  - *types* of defense
  - *difficulties*
  - adversarial robustness and adversarial training

Adversarial attack

‘We can cause the network to misclassify an image by applying a certain hardly perceptible perturbation, which is found by maximizing the network’s prediction error.’



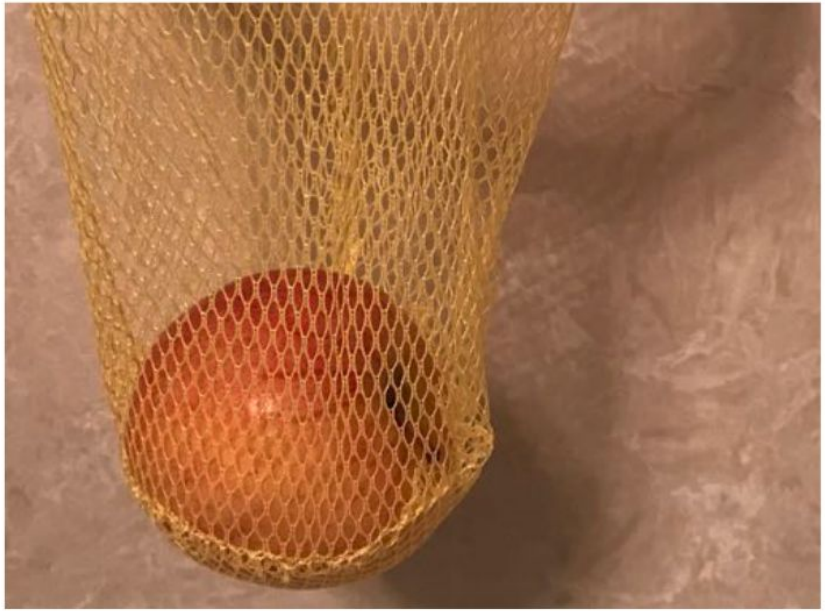
* Intriguing properties of neural networks (Szegedy, 2013)

* Explaining and harnessing adversarial examples (Goodfellow, 2014)

Definition of adversarial example

There is no standard community-accepted definition of the term “adversarial examples” and the usage has evolved over time.

Adversarial examples are inputs with small perturbations to machine learning models that an attacker has intentionally designed to cause the model to make a mistake. [Attacking Machine Learning with Adversarial Examples, 2017]



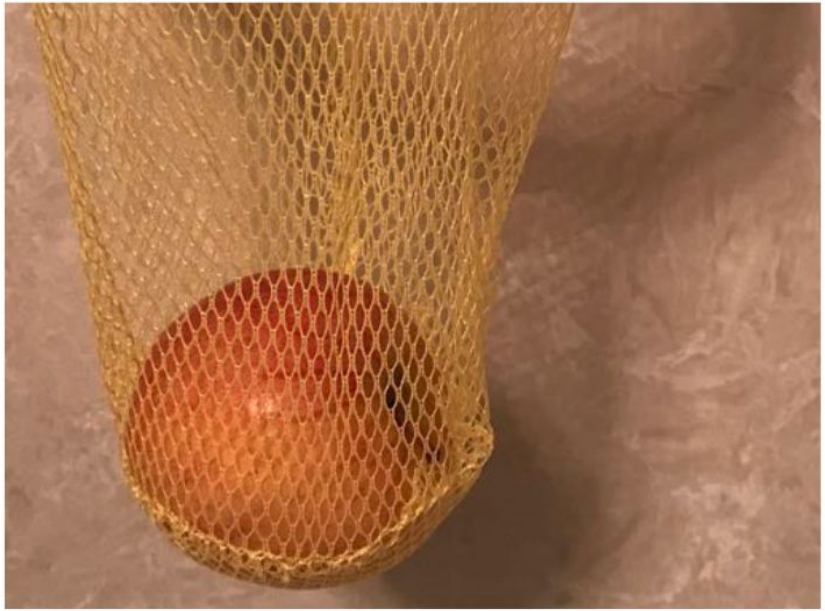
* Defense Against the Dark Arts [Goodfellow, 2018]

Definition of adversarial example

There is no standard community-accepted definition of the term “adversarial examples” and the usage has evolved over time.

*Adversarial examples are inputs **with small perturbations** to machine learning models that an attacker has intentionally designed to cause the model to make a mistake.* [Attacking Machine Learning with Adversarial Examples, 2017]

small with respect to what?

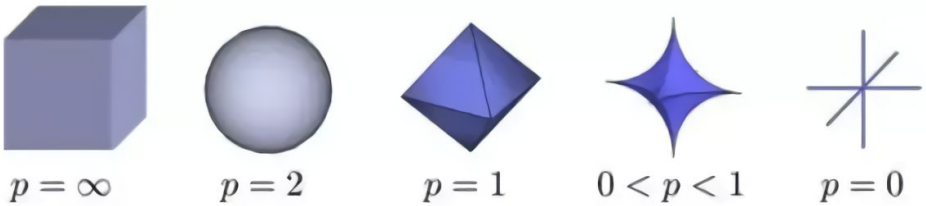


* Defense Against the Dark Arts [Goodfellow, 2018]

Perturbation measurement and imperceptibility

Usually perturbation considered to be small wrt l_p-norm

$$\|x\|_p = \left(\sum_{i=1}^{n} \|x_i\|^p \right)^{\frac{1}{p}}$$



3D case

l_∞ is the most common choice

* Image

Which Lp norm to choose?

	Pair	Diff	$L0$	$L1$	$L2$	$L\infty$
Nearest $L0$			63	35.0	4.86	1.0
Nearest $L1$			91	19.9	3.21	.996
Nearest $L2$			110	21.7	2.83	1.0
Nearest $L\infty$			121	34.0	3.82	.76
Clipped Random uniform			784	116.0	4.8	.3

(Goodfellow 2018)

Toy game, real attacks may be difficult to characterize in terms of *lp*-norm

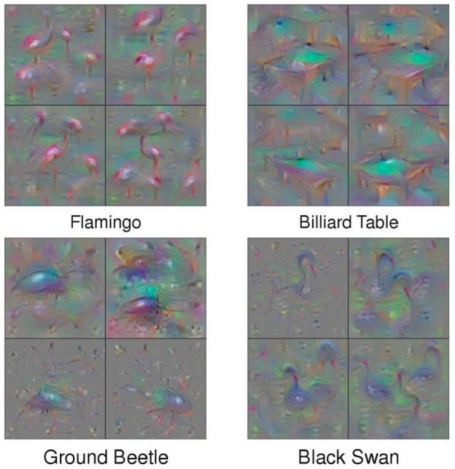
* Defense Against the Dark Arts [Goodfellow, 2018]

Why is this research important

- ***security:*** classic examples: face recognition, self-driving cars

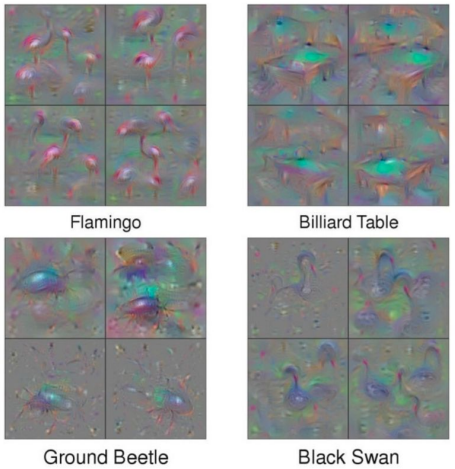
Why is this research important

- ***security:*** classic examples: face recognition, self-driving cars
- ***understanding classifiers:*** the study of adversarial perturbations help us better understand how classifiers generate features to represent the input and make decisions



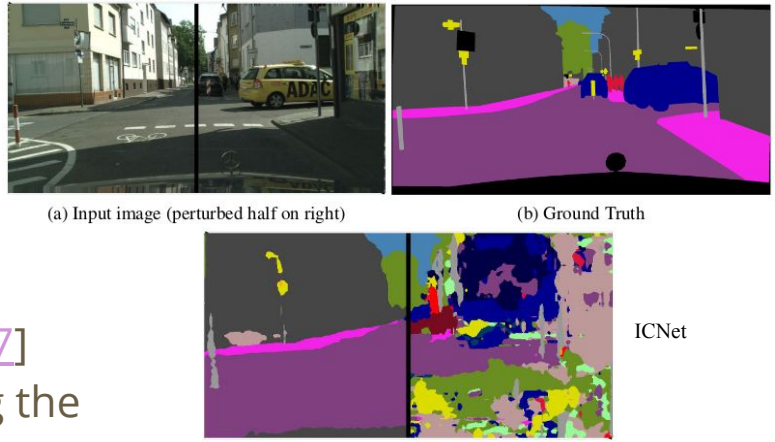
Why is this research important

- ***security:*** classic examples: face recognition, self-driving cars
- ***understanding classifiers:*** the study of adversarial perturbations help us better understand how classifiers generate features to represent the input and make decisions
- ***robustness:*** the adversarial approach allows us to go beyond the standard evaluation protocol of running a trained classifier on a carefully curated (and usually static) test set.



Areas of adversarial attacks research

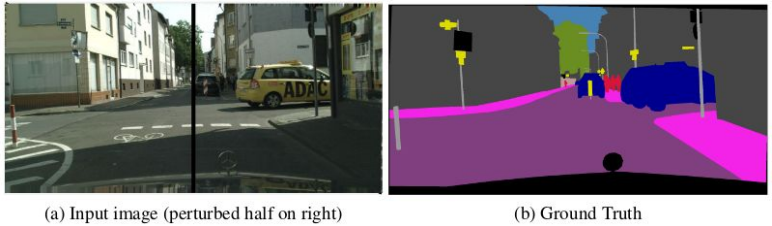
- Adversarial attacks on images:
  - image classification [Szegedy, 2013]
  - object detection [Eykholt, 2018]
  - segmentation [Arnab, 2018]
  - generative models on images [Kos, 2017]
  - reinforcement learning by manipulating the images the RL agent sees [Huang, 2017]



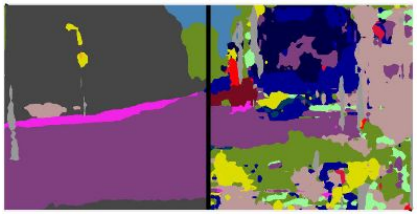
* On the Robustness of Semantic Segmentation Models to Adversarial Attacks (Arnab, 2018)

Areas of adversarial attacks research

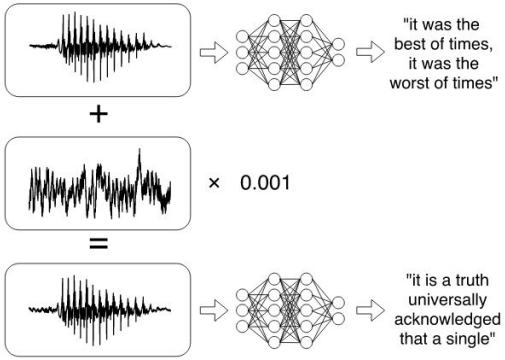
- Adversarial attacks on images:
  - image classification [Szegedy, 2013]
  - object detection [Eykholt, 2018]
  - segmentation [Arnab, 2018]
  - generative models on images [Kos, 2017]
  - reinforcement learning by manipulating the images the RL agent sees [Huang, 2017]
- Audio (Speech-To-Text) [Carlini, 2018]



(a) Input image (perturbed half on right) (b) Ground Truth



ICNet

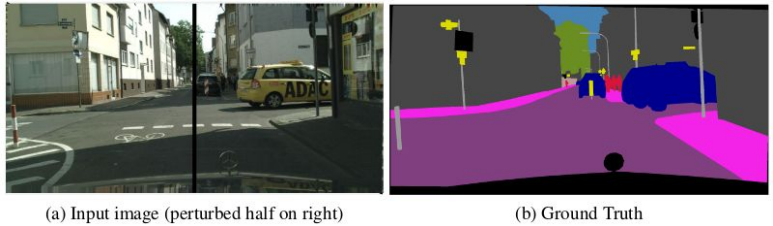


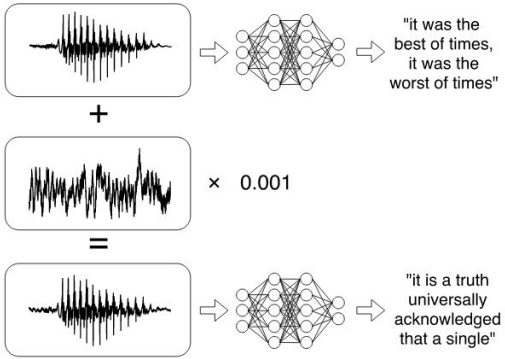
* On the Robustness of Semantic Segmentation Models to Adversarial Attacks (Arnab, 2018)

* Audio Adversarial Examples: Targeted Attacks on Speech-to-Text (Carlini, 2018)

Areas of adversarial attacks research

- Adversarial attacks on images:
  - image classification [Szegedy, 2013]
  - object detection [Eykholt, 2018]
  - segmentation [Arnab, 2018]
  - generative models on images [Kos, 2017]
  - reinforcement learning by manipulating the images the RL agent sees [Huang, 2017]
- Audio (Speech-To-Text) [Carlini, 2018]
- Text classification (reading comprehension systems) [Jia, 2017]



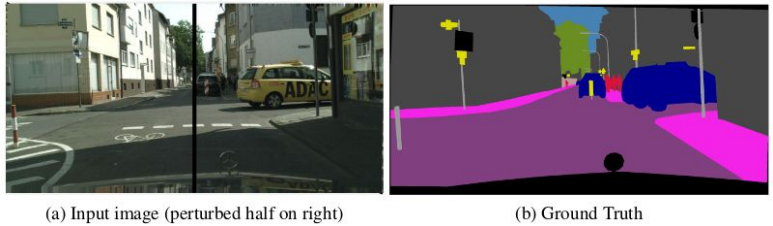


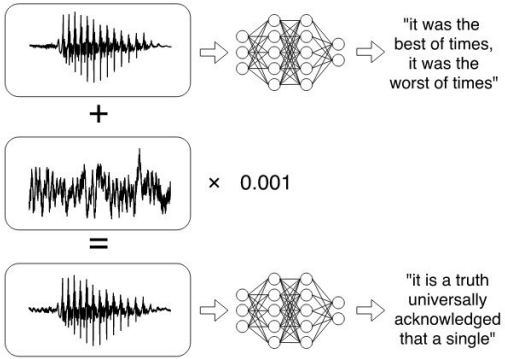
* On the Robustness of Semantic Segmentation Models to Adversarial Attacks (Arnab, 2018)

* Audio Adversarial Examples: Targeted Attacks on Speech-to-Text (Carlini, 2018)

Areas of adversarial attacks research

- Adversarial attacks on images:
  - **image classification** [Szegedy, 2013]
  - object detection [Eykholt, 2018]
  - segmentation [Arnab, 2018]
  - generative models on images [Kos, 2017]
  - reinforcement learning by manipulating the images the RL agent sees [Huang, 2017]
- Audio (Speech-To-Text) [Carlini, 2018]
- Text classification (reading comprehension systems) [Jia, 2017]





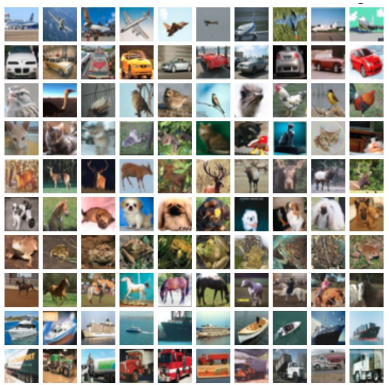
* On the Robustness of Semantic Segmentation Models to Adversarial Attacks (Arnab, 2018)

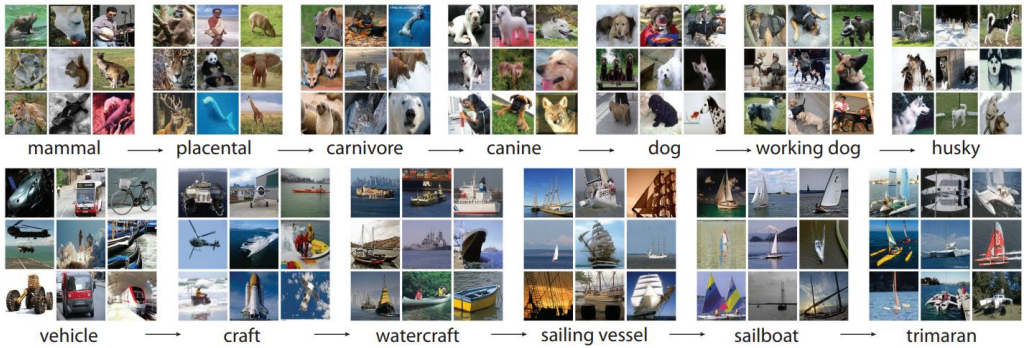
* Audio Adversarial Examples: Targeted Attacks on Speech-to-Text (Carlini, 2018)

Popular in AEs research datasets

Adversaries show the performance of their adversarial attacks based on different datasets and victim models.

- MNIST
- CIFAR10
- ImageNet





Adversarial attacks classification

- Targeted vs non-targeted
- White box vs black box
- Digital vs physical
- Individual vs universal

Attack scenarios: non-targeted vs targeted

Non-targeted attack: In this the case adversary's goal is to cause the classifier to predict any incorrect label. The specific incorrect label does not matter.

Targeted attack: In this case the adversary aims to change the classifier's prediction to some specific target class.

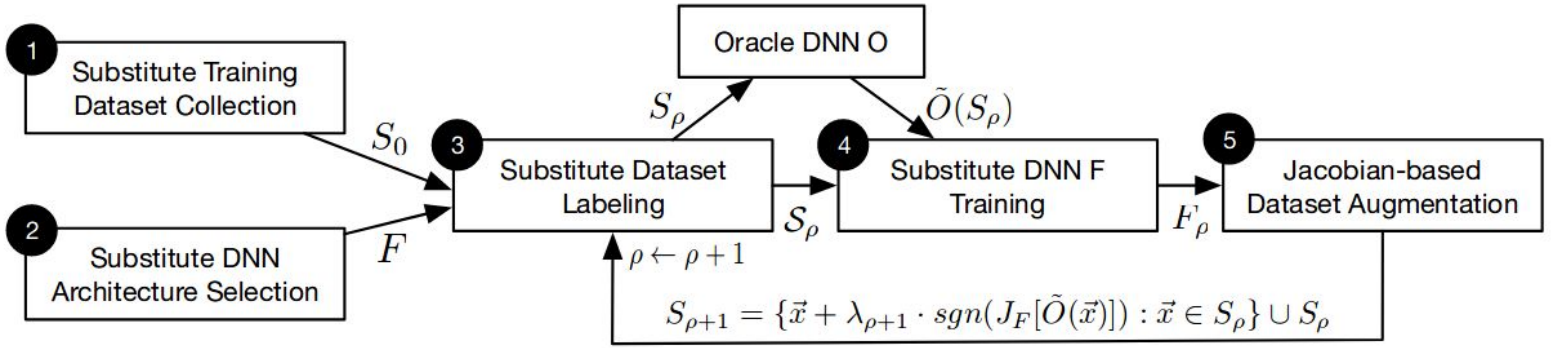

```
UNMODIFIED IMAGE (left)
        Predicted class: espresso
        Predicted prob: 94.47%
        True class:      espresso
NONTARGETED ADVERSARIAL IMAGE (center)
        Predicted class: tray
        Predicted prob: 99.98%
        Prob of true class: 0.00%
TARGETED ADVERSARIAL IMAGE (right)
        Predicted class: nail
        Predicted prob: 99.66%
        Prob of true class: 0.00%
        Target class:    nail
```

Adversarial images are generated using CleverHans library (PGD method)

Attack scenarios: white box vs black box

White box: the adversary has full knowledge of the model including model type, model architecture and values of all parameters and trainable weights.

Black box: the adversary has limited or no knowledge about the model under attack



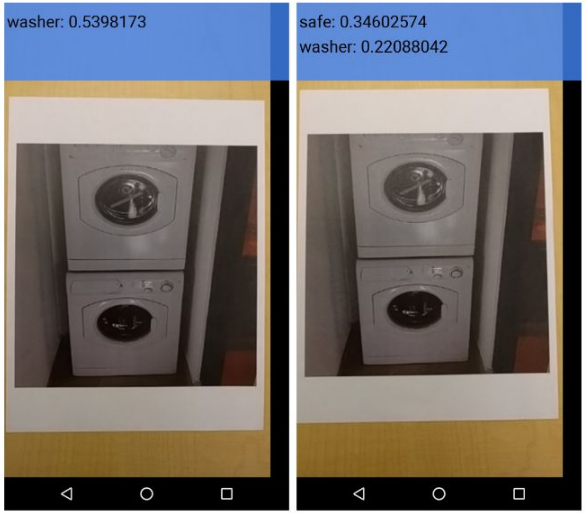
*Black box attack usually rely on **transferability** property of adversarial examples*

* Practical Black-Box Attacks against Machine Learning [Papernot, 2016]

Attack scenarios: digital vs physical

Digital attack: the adversary can choose specific numerical values as input for the model. In a real world setting, this might occur when an attacker uploads a PNG file to a web service, and intentionally designs the file to be read incorrectly.

Physical attack: the adversary does not have direct access to the digital representation of provided to the model. Instead, the model is fed input obtained by sensors such as a camera or microphone. The adversary is able to place objects in the physical environment seen by the camera.



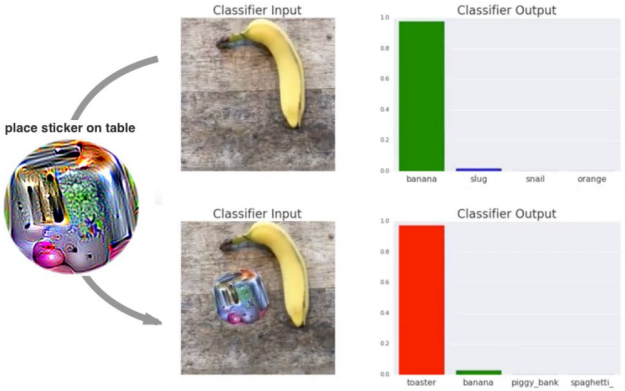
* Adversarial Examples in Physical World [Kurakin, 2016]

Physical world attacks

Attacker could control:

- the sensor to generate the desired stream of bytes
- communication channel between the sensor and the model to modify the data
- the stream of data values to the model by showing the sensor physical object:
    - printed patches
    - 3D objects

Synthesizing Robust Adversarial Examples [Athalye et al., 2017]

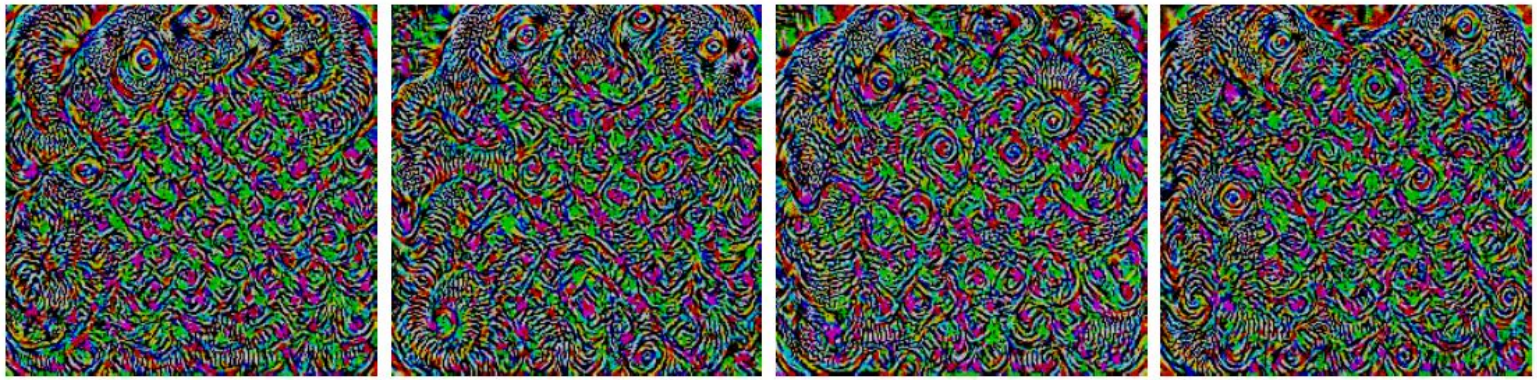


Adversarial Patch [Brown, 2018]

Individual vs universal attacks

Individual attack: generate different perturbations for each clean input

Universal attack: only create a universal perturbation for the whole dataset. Makes it easier to deploy adversary examples in the real world.



Diversity of universal perturbations for the GoogLeNet architecture. The five perturbations are generated using different random shufflings of the set X

* Universal adversarial perturbations [Seyed-Mohsen Moosavi-Dezfooli, 2016]

How to construct adversarial image (FGSM)

The idea is to perform the perturbation of the input in a way that maximally changes the loss function of the model:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{\eta}$$

$$\boldsymbol{\eta} = \epsilon \text{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y)\right)$$

Note that this method was first historical method to demonstrate the idea. In practice it turns out to be weak attack, but the idea of using gradient is used in more strong attacks

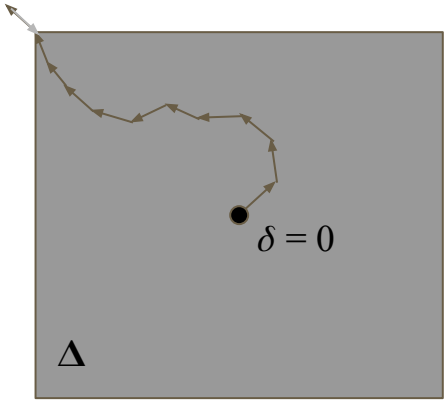
```
UNMODIFIED IMAGE (left)
        Predicted class: geyser
        Predicted prob: 99.89%
        True class:      geyser
NONTARGETED ADVERSARIAL IMAGE (center)
        Predicted class: loggerhead, loggerhead turtle
        Predicted prob: 13.58%
        Prob of true class: 0.02%
```

Iterative attacks: projected gradient descent

Compared with one-time attacks, iterative attacks usually perform better adversarial examples, but require more time to be generated

$$x_0^{adv} = \boldsymbol{X}, \quad x_{N+1}^{adv} = Clip_{X,\varepsilon}\left\{\boldsymbol{X}_N^{adv} + \alpha\,\mathrm{sign}\left(\nabla_X J(\boldsymbol{X}_N^{adv}, y_{true})\right)\right\}$$

Most attacks in practice are some variations of PGD



Schematic illustration: projected gradient descent applied to l_∞ ball

Adversarial attacks zoo

L-BFGS attack [Szegedy et al., 2013]

$$\min_{x'} \quad c\|\eta\| + J_\theta(x', l')$$
$$s.t. \quad x' \in [0, 1].$$

Model-based Ensembling Attack [Liu et al, 2017]

$$\arg\min_{x'} \quad -\log\left((\sum_{i=1}^{k} \alpha_i J_i(x', l'))\right) + \lambda\|x' - x\|$$

DeepFool [Moosavi-Dezfooli et al., 2015]

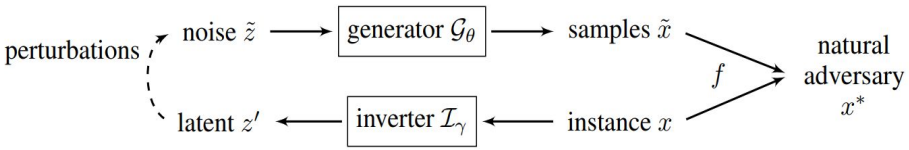
$$\boldsymbol{r}_*(\boldsymbol{x}_0) := \arg\min \|\boldsymbol{r}\|_2$$
$$\text{subject to } \text{sign}\,(f(\boldsymbol{x}_0 + \boldsymbol{r})) \neq \text{sign}(f(\boldsymbol{x}_0))$$
$$= -\frac{f(\boldsymbol{x}_0)}{\|\boldsymbol{w}\|_2^2}\boldsymbol{w}. \quad \mathscr{F} = \{\boldsymbol{x} : \boldsymbol{w}^T\boldsymbol{x} + b = 0\}$$

separating affine hyperplane

Jacobian saliency map [Papernot et al., 2015]

$$S(\mathbf{X}, t)[i] = \begin{cases} 0 \text{ if } \frac{\partial \mathbf{F}_t(\mathbf{X})}{\partial \mathbf{X}_i} < 0 \text{ or } \sum_{j \neq t} \frac{\partial \mathbf{F}_j(\mathbf{X})}{\partial \mathbf{X}_i} > 0 \\ \left(\frac{\partial \mathbf{F}_t(\mathbf{X})}{\partial \mathbf{X}_i}\right) \left|\sum_{j \neq t} \frac{\partial \mathbf{F}_j(\mathbf{X})}{\partial \mathbf{X}_i}\right| \text{ otherwise} \end{cases}$$

Using GAN [Zhao et al., 2017]



Carlini Wagner [Carlini et al., 2016]

$$\text{minimize} \quad \|\frac{1}{2}(\tanh(w) + 1) - x\|_2^2 + c \cdot f(\frac{1}{2}(\tanh(w) + 1)$$

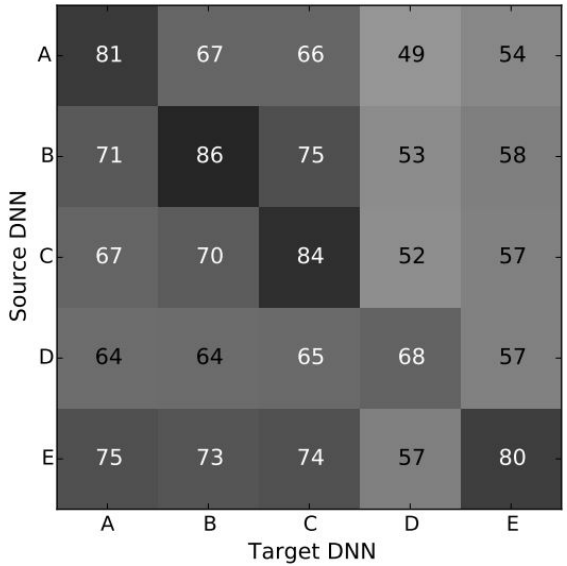
with f defined as

$$f(x') = \max(\max\{Z(x')_i : i \neq t\} - Z(x')_t, -\kappa).$$

and many other ...

Why is it hard to attack DL systems

White box scenario in not realistic. Black-box attacks rely on important property of adversarial examples: its transferability, which means that the same adversarial example fools more than one model.

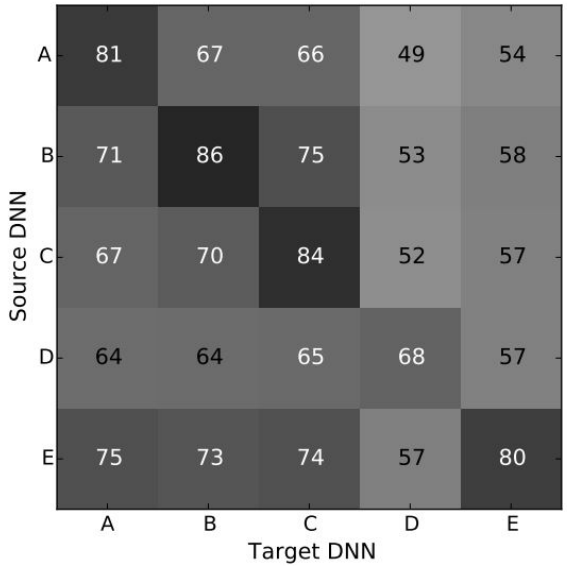


Cell (*i, j*) reports the intra-technique transferability between models *i* and *j, i*.e. the percentage of adversarial samples produced using model *i* misclassified by model *j*. (results for MNIST dataset)

Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples [Papernot, 2016]

Why is it hard to attack DL systems

White box scenario in not realistic. Black-box attacks rely on important property of adversarial examples: its transferability, which means that the same adversarial example fools more than one model.



Cell (*i, j*) reports the intra-technique transferability between models *i* and *j, i*.e. the percentage of adversarial samples produced using model *i* misclassified by model *j*. (results for MNIST dataset)

Ensembles could be a solution

Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples [Papernot, 2016]

Defense against adversarial attacks

Defence approaches: increasing robustness vs detection

Adversarial detecting:

- DNN based binary classifiers [Gong et al., 2017]
- adding an outlier class to the original DL model [Grosse et al., 2017]

Defence approaches: increasing robustness vs detection

Adversarial detecting:

- DNN based binary classifiers [Gong et al., 2017]
- adding an outlier class to the original DL model [Grosse et al., 2017]

Adversarial training: include adversarial examples in the training stage to make model more robust

Defence approaches: increasing robustness vs detection

Adversarial detecting:

- DNN based binary classifiers [Gong et al., 2017]
- adding an outlier class to the original DL model [Grosse et al., 2017]

Adversarial training: include adversarial examples in the training stage to make model more robust

Network verification: checks the properties of a neural network: whether an input violates or satisfies the property (promising approach, but hardly scalable)

Defence approaches: increasing robustness vs detection

Adversarial detecting:

- DNN based binary classifiers [Gong et al., 2017]
- adding an outlier class to the original DL model [Grosse et al., 2017]

Adversarial training: include adversarial examples in the training stage to make model more robust

Network verification: checks the properties of a neural network: whether an input violates or satisfies the property (promising approach, but hardly scalable)

Input reconstruction: transformation to clean data via reconstruction

Defence approaches: increasing robustness vs detection

Adversarial detecting:

- DNN based binary classifiers [Gong et al., 2017]
- adding an outlier class to the original DL model [Grosse et al., 2017]

Adversarial training: include adversarial examples in the training stage to make model more robust

Network verification: checks the properties of a neural network: whether an input violates or satisfies the property (promising approach, but hardly scalable)

Input reconstruction: transformation to clean data via reconstruction

Ensembling defenses: combining several approaches

Why is it hard to defend DL systems

- Usually ***defender goes first*** and should ideally consider all possible types of attacks.

Why is it hard to defend DL systems

- Usually ***defender goes first*** and should ideally consider all possible types of attacks.

- Defence mechanism should not influence the performance of the model on benign input and (significantly) increase the inference time, whereas attacker often in not restricted with resources.

Why is it hard to defend DL systems

- Usually ***defender goes first*** and should ideally consider all possible types of attacks.

- Defence mechanism should not influence the performance of the model on benign input and (significantly) increase the inference time, whereas attacker often in not restricted with resources.

In paper *Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples* authors have circumvented 6 attacks completely and 1 partially out of 9 presented on ICLR 2018 defenses *in white box setting.*

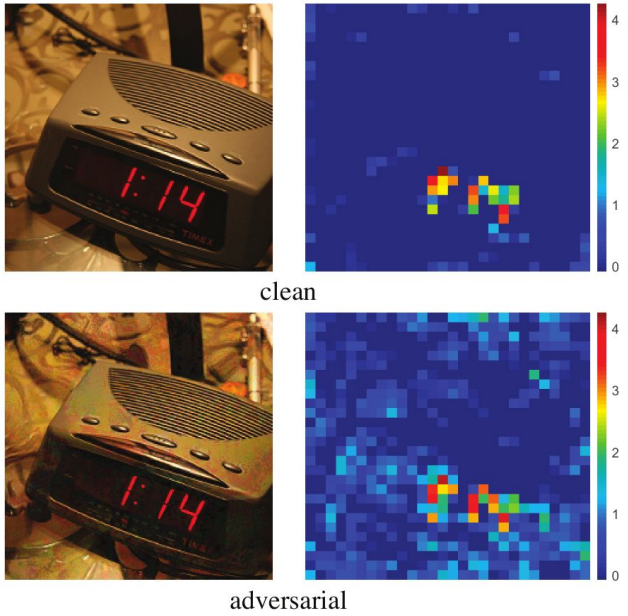
Example: Feature denoising

The model was ranked first in Competition on Adversarial Attacks and Defenses (CAAD) 2018

On ImageNet, under 10-iteration PGD white-box attacks where prior state-of-the-art has 27.9% accuracy, this method achieves 55.7%

***Resource consuming:** 128 Nvidia V100 GPUs is approximately 38 hours for the baseline ResNet-101 model, and approximately 52 hours for the baseline ResNet-152 model on ImageNet*

Drop in accuracy for 'clean' images:** the top-1 accuracy of an adversarially trained network is **65.30%** on clean images, whereas its cleanly trained counterpart obtains **79.08%



* Feature Denoising for Improving Adversarial Robustness [Xie, 2018]

Adversarial robustness and adversarial training

Standard robustness vs adversarial robustness

- Standard robustness: the goal is to train the model that have low expected loss

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\mathcal{L}(x, y; \theta)]$$

- Adversarial robustness: the goal is to train the model that is resistant to adversarial examples (low expected adversarial loss):

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{\delta\in\Delta}\mathcal{L}(x+\delta, y; \theta)\right]$$

Adversarial training

Adversarial training is currently the most successful approach to building adversarially robust models so far.

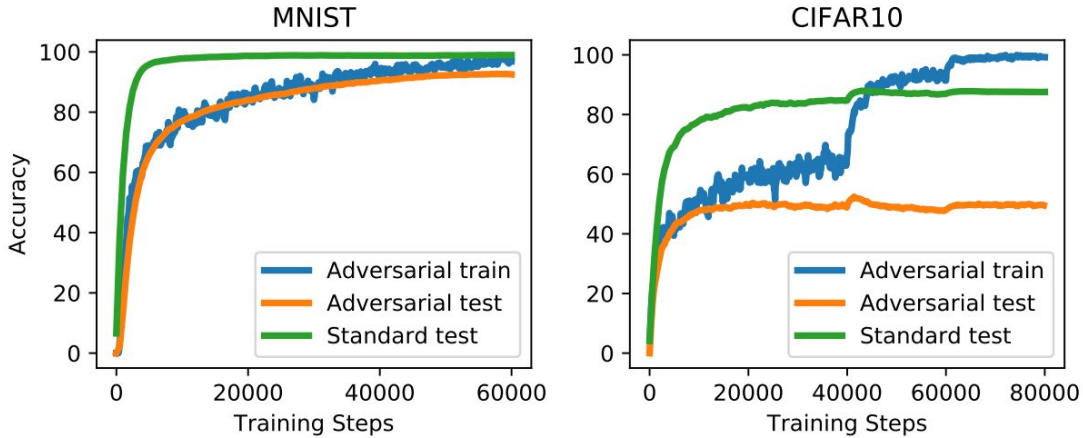
To get adversarial robust model we need to solve the corresponding (adversarial) empirical risk minimization problem:

$$\min_{\theta} \mathop{\mathbb{E}}_{(x,y)\sim\widehat{\mathcal{D}}} \left[\max_{\delta \in S} \mathcal{L}(x+\delta, y; \theta) \right]$$

In practice, the solution is tractable and can be found by repeatedly solving the inner maximization problem (finding the worst-case input perturbations), and then update the model parameters to reduce the loss on these perturbed inputs.

No free lunch in adversarial robustness

- computationally expensive training methods (more training time)
- the potential need for more training data
- there is an inherent trade-off between the standard accuracy and adversarially robust accuracy of a model.



* Adversarially Robust Generalization Requires More Data [Schmidt et al., 2018]

No free lunch in adversarial robustness

Basic intuition:

In *standard training*, all correlation is good correlation

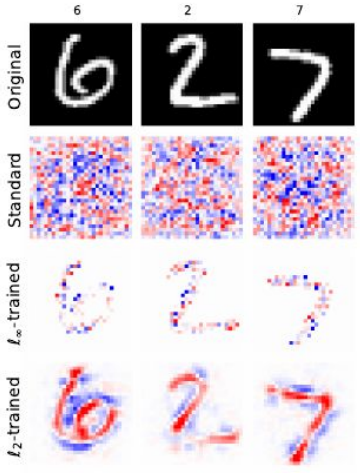
If we want robustness, must avoid weakly correlated features

Standard training: use all of features, maximize accuracy

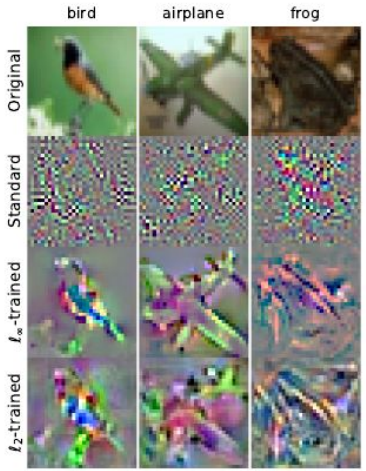
Adversarial training: use only single robust feature (at the expense of accuracy)

Robust models give more tractable results

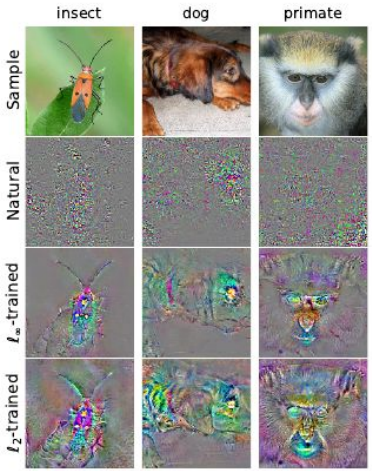
Robust models are more aligned with human vision than standard models



(a) MNIST



(b) CIFAR-10

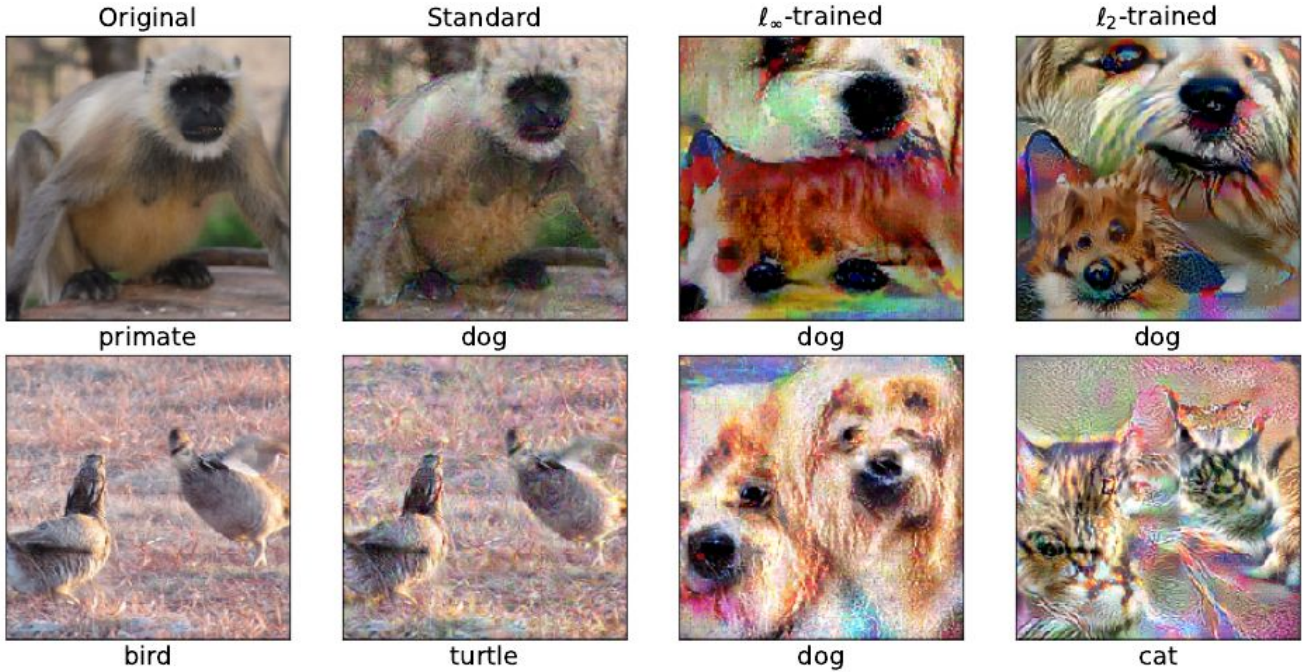


(c) Restricted ImageNet

Visualization of the loss gradient with respect to input pixels

* Robustness May Be at Odds with Accuracy (Tsipras, 2018)

Robust models give more tractable results

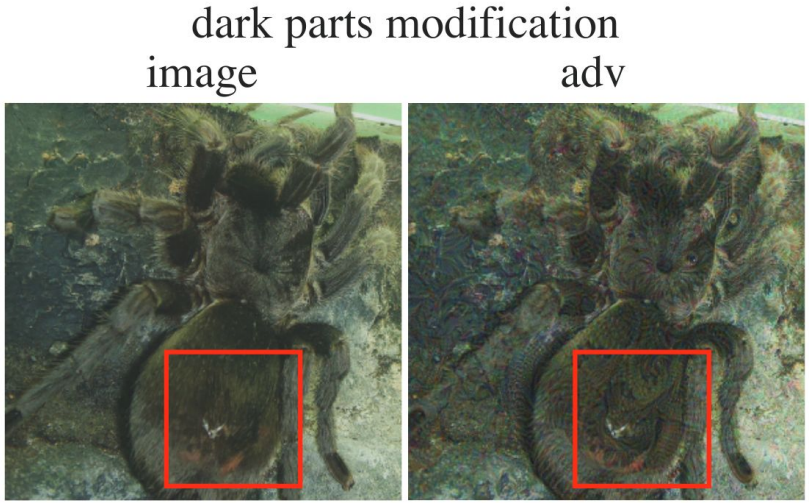


Visualization of large-epsilon adversarial examples for standard and robust models

* Robustness May Be at Odds with Accuracy (Tsipras, 2018)

Human vs CNN

Adversarial examples that strongly transfer across computer vision models influence the classifications made by time-limited human observers.



For time-limited human observer adversarially perturbed spider may look like snake

The perturbations made were small enough that they generally do not change the output class for a human who has no time limit

* Adversarial Examples that Fool both Computer Vision and Time-Limited Humans [Elsayed, 2018]

Conclusions

- Most of the papers shows their results on MNIST and CIFAR10 datasets (could be overfitting to these datasets). There is no guarantee that attack/defence strategies would work on other data.
- Almost all defenses are shown to be effective only for part of attacks. They tend not to be defensive for some strong and unseen attacks
- A lot of defence techniques are not really scaled to large problems
- Not all publish their code to confirm the results
- Adversarial robustness usually is not free - this might not be ready to be deployed it in real world
- It is really an open field of research

Thank you for your attention!

Resources

Libraries:

- CleverHans [Papernot et al., 2016]
- FoolBox [Rauber et al., 2017]
- Adversarial Robustness Toolbox [Nicolae, 2018]

Tutorials and other resources:

- Adversarial Robustness - Theory and Practice
- Attacking Machine Learning with Adversarial Examples