Adversarial attacks in practice

Gábor Rabatin

Budapest University of Technology and Economics
Fault Tolerant Systems Research Group
Agenda

- Background
- Attacking approaches
- Defending approaches
- Verification
- Demo
Adversarial attack

- **Adversarial example**: "a pair of inputs $x; x'$ is an adversarial example for a classifier, if a reasonable person would say they are of the same class but the classifier produces significantly different outputs."

- "they're like optical illusions for machines"
BACKGROUND
Fast gradient sign method (FGSM)

- Ian J. Goodfellow, Jonathon Shlens & Christian Szegedy: Explaining And Harnessing Adversarial Examples

\[ x^{adv} = x + \varepsilon \cdot \text{sign}(\nabla_x J(x, y_{true})) , \]

- Pixel-wide perturbation in the direction of gradient
- Computed in one step → very efficient
Targeted-FGSM

- Alexey Kurakin, Ian J. Goodfellow, Samy Bengio: Adversarial Examples In The Physical World

\[ x^{adv} = x - \varepsilon \cdot \text{sign}(\nabla_x J(x, y_{target})) , \]

- In the negative direction in respect to the target class
Iterative-FGSM

- Alexey Kurakin, Ian J. Goodfellow, Samy Bengio: Adversarial Machine Learning At Scale

\[ x_0^{adv} = x, \quad x_t^{adv} = x_t^{adv} + \alpha \cdot \text{sign} (\nabla_x J(x_t^{adv}, y)). \]

- Smaller steps
- Higher success rate in white box attacks
NIPS 2017 Competition

- “Adversarial Attacks and Defences” Kaggle competition in 2017 by Google Brain

- 3 categories:
  - targeted adversarial attack,
  - non-targeted adversarial attack
  - and defense against adversarial attacks
Momentum Iterative-FGSM

- Tsinghua University, Intel Labs China: Boosting Adversarial Attacks with Momentum
- Good transferability
- Performs well in black box attacks

---

Algorithm 1 MI-FGSM

**Input:** A classifier $f$ with loss function $J$; a real example $\mathbf{x}$ and ground-truth label $y$.

**Input:** The size of perturbation $\epsilon$; iterations $T$ and decay factor $\mu$.

**Output:** An adversarial example $\mathbf{x}^*$ with $\|\mathbf{x}^* - \mathbf{x}\|_\infty \leq \epsilon$.

1. $\alpha = \epsilon / T$;
2. $g_0 = 0$; $\mathbf{x}_0^* = \mathbf{x}$;
3. for $t = 0$ to $T-1$ do
4. Input $\mathbf{x}_t^*$ to $f$ and obtain the gradient $\nabla_\mathbf{x} J(\mathbf{x}_t^*, y)$;
5. Update $g_{t+1}$ by accumulating the velocity vector in the gradient direction as
   \[
   g_{t+1} = \mu \cdot g_t + \frac{\nabla_\mathbf{x} J(\mathbf{x}_t^*, y)}{\|\nabla_\mathbf{x} J(\mathbf{x}_t^*, y)\|_1};
   \]  
6. Update $\mathbf{x}_{t+1}^*$ by applying the sign gradient as
   \[
   \mathbf{x}_{t+1}^* = \mathbf{x}_t^* + \alpha \cdot \text{sign}(g_{t+1});
   \]  
7. end for
8. return $\mathbf{x}^* = \mathbf{x}_T^*$.  

---

9
What if there are more than models?

Algorithm 2 MI-FGSM for an ensemble of models

Input: The logits of $K$ classifiers $l_1, l_2, \ldots, l_K$; ensemble weights $w_1, w_2, \ldots, w_K$; a real example $x$ and ground-truth label $y$.

Input: The size of perturbation $\epsilon$; iterations $T$ and decay factor $\mu$.

Output: An adversarial example $x^*$ with $\|x^* - x\|_\infty \leq \epsilon$.

1. $\alpha = \frac{\epsilon}{T}$;
2. $g_0 = 0; x_0^* = x$;
3. for $t = 0$ to $T - 1$
4. Input $x_t^*$ and output $l_k(x_t^*)$ for $k = 1, 2, \ldots, K$;
5. Fuse the logits as $l(x_t^*) = \sum_{k=1}^{K} w_k l_k(x_t^*)$;
6. Get softmax cross-entropy loss $J(x_t^*, y)$ based on $l(x_t^*)$ and Eq. (9);
7. Obtain the gradient $\nabla_x J(x_t^*, y)$;
8. Update $g_{t+1}$ by Eq. (6);
9. Update $x_{t+1}^*$ by Eq. (7);
10. end for
11. return $x^* = x_T^*$.

\[
J(x, y) = -1_y \cdot \log(\text{softmax}(l(x))), \quad (9)
\]
Types of Adversarial Attack

- **White-box attack**
  - Attacker has access to the model’s parameters

- **Black-box attack**
  - No access to parameters, gradients
  - Uses a different model or no model
  - With hope that the examples will transfer to the target model
Examples of Adversarial Attack

- One Pixel Attack
- Physical Adversarial Examples
- Adversarial Patch
- Examples Fool Both Human and Computer
- Unrecognizable examples
- Adversarial Attack in Reinforcement Learning
- Robust Adversarial Examples
One Pixel Attack for Fooling Deep Neural Networks

- Limited scenario: only one pixel is modified

![Image of Planetarium with a pixel highlighted]

- HORSE
- DOG (88.0%)
- SHIP
- AIRPLANE (62.7%)
- CAT
- DOG (78.2%)

**Planetarium**

Mosque (7.81%)
Adversarial Examples in the Physical World

- Alexey Kurakin, Ian J. Goodfellow, Samy Bengio
- Attacks also work in real life
Adversarial Examples in the Physical World
Adversarial Patch
Adversarial Examples that Fool Both Human and Computer
High Confidence Predictions for Unrecognizable Images

- Unrecognizable for humans, but „easily recognized” by DNNs
- Evolutionary algorithms are used
Adversarial Attack in Reinforcement Learning

- Widely used deep reinforcement learning algorithms are vulnerable too.
Are they robust?
Scale-Invariant Adversarial Examples

Zoom: 1.000000x
Transformation-Invariant Adversarial Examples
Examples of Defenses

- Adversarial Training
- Defensive Distillation
- Gradient Masking
- Denoiser
Adversarial Training

- **Algorithm:**
  - Generate a lot of adversarial examples
  - Retrain the model not to be fooled by them
  - Do this iteratively

- **Danger of overfitting**

- **Less effective against black-box attacks**
Defensive Distillation

- Train a new model with a pretrained model’s output probabilities
- Inspired by Geoffrey Hinton’s knowledge compressing paper

Diagram:

1. Training Data X
2. DNN F trained at temperature T
3. Probability Vector Predictions F(X)
4. Training Labels F(X)
5. DNN F^d(X) trained at temperature T
6. Probability Vector Predictions F^d(X)
Gradient Masking – a failed defense

- Deny the attacker’s access to a useful gradient
- „Most likely class” output mode, a smooth change in input doesn’t change the output
- However, the model is not more robust, just fewer clues to finding the holes

![Diagram](image-url)
High-Level Representation Guided Denoiser

- Feature guided denoiser
  - Denoising U-Net (denoising autoencoder with lateral connections)
  - Learning objective: adversarial noise
- NIPS 1st place!
- More robust to white-box and black-box attacks
- Can be trained on small subset of the images
- Can be transferred to defend other models
The problems with defending

- It requires models to produce good outputs for every possible input
- Techniques are not adaptive

- But there are some tools...
Tools

- **Cleverhans**
  - *Ian J. Goodfellow and Nicolas Papernot*
  - Tool for developing more robust models
  - Attacking and defending techniques implemented

- **Darkon**
  - Helps understanding the decision of DNNs
  - Filters bad training examples
  - Grad-CAM

![Image of a dog and a cat with heat maps]
- LIME
- Helps interpretability
VERIFICATION
Verification

- Formal verification analyzes if the formal model satisfies the specification (properties)

How to formalize the working of AI?

How to formalize properties?

What kind of algorithms to use?

Verification

Real-life system

Ok

Counterexample
DeepXplore

- Differential testing approach
  - Running more versions of the same program (in our case: DNN)
- No difference found
  - Adversarial example generation
  - Image transformations on the input
- Two objectives
  - Modify the output of the target model, while keeping the original output of the other models
  - Increase the **neuron coverage** of the neural network
Searching for misclassified images

Diagram:
- Input image
- \( DNN_1 \) classifier
- \( DNN_2 \) classifier
- \( DNN_k \) classifier
- Compare outputs
- Difference
- No difference
- Adversarial Attack
- Misclassified input
Adversarial attack (DeepXplore workflow)

DNNs under test

Input images

DNN_1 classifier
DNN_2 classifier
\ldots
DNN_k classifier

Gradients of outputs and neurons

Maximize: difference and neuron coverage
Joint optimization with gradient ascent

Domain specific constraints

Difference inducing inputs
Evaluation

- Retraining with the generated samples
- Critical situation and counter-examples can be found
Future Work?

- Active research area
  - Join us!
- Demonstrator development
  - MoDeS3 intelligent control
  - Industrial partners
- Project laboratory, Student scientific report (TDK)
- International collaboration
THANK YOU
FOR YOUR ATTENTION!