Stochastic Substitute Training: A Gray-box Approach to Craft Adversarial Examples Against Gradient Obfuscation Defenses

Mohammad Hashemi, Greg Cusack,* Eric Keller
10/19/18

*Presenter
Neural Networks are everywhere

Vulnerable to attacks!

Response: Design robust neural networks that block or obfuscate gradients

Even these are vulnerable in a white box setting! [Athalye et al. ICML '18]

But are they vulnerable in a more realistic setting?
Stochastic Substitute Training

A general, gray-box attack for breaking defenses that obfuscate gradients
Crafting Adversarial Examples

Optimization Problem

$$\arg\min_{\delta} \|\delta\|_p \text{ s.t. } (x + \delta) \in [0, 1]^m \text{ and } F(x + \delta) = y_{\text{target}}$$

But neural networks are not convex!
Carlini & Wagner (C&W) Attack

Modified Objective Function

\[
\begin{align*}
\text{minimize} & \quad c \cdot \|\delta\|_p + f(x + \delta) \\
\text{subject to} & \quad x + \delta \in [0,1]^m
\end{align*}
\]

\[
f(x') = \max(\max_{i \neq t}(Z(x')_i - Z(x')_t), -k)
\]

- \(Z(x')_i\) -> Logits of non-target classes
- \(Z(x')_t\) -> Logit of target class
- \(k\) -> Determines classification confidence

[Carlini et al. IEEE S&P '17]
Black Box Attack (Transferability)

Adversarial Example

Target NN #1
Architecture: a1
Parameters: p1
Target Class
Successful Attack

Target NN #2
Architecture: a2
Parameters: p2
Target Class
Successful Attack
Leverage Transferability of Adversarial Examples

[Biggio et al. ECML/PKDD '13], [Papernot et al. Asia CCS '17]
Stochastic Substitute Training (Threat Model)

- Fortifying Defenses
  - Classifier predicts adversarial examples as their correct class
  - Threat Model
    - Send inputs and see logits

- Detecting Defenses
  - Identify when adversarial examples are fed into the classifier
  - Threat Model
    - Send inputs, see logits, and the output of the detector
Stochastic Substitute Training

\[
\text{Loss}_{\text{SST}} = \frac{1}{N} \sum_{i=0}^{N} \sum_{j=0}^{K} \frac{1}{K} \left( Z_j^{\text{robust}}(x_i + r_i) - Z_j^{\text{sub}}(x_i + r_i) \right)^2
\]
Crafting Adversarial Examples

For each $\kappa$...

For each $\kappa$...

minimize $||\delta||_p + c.f(x + \delta)$ s.t. $x + \delta \in [0, 1]^m$

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

Iteration
Noisy Data Augmentation

- Substitute model more closely approximates decision boundaries of target model
- Helps substitute model learn how the robust model’s class probabilities change in the neighborhood of each sample
- Multiple copies of training models created with varying levels of random noise
  - Each substitute model approximates the decision boundaries for some specific images better than others
Random Feature Nullification

Feature Vector

Randomly Generated Feature Mask

Randomly Nullified Feature Vector

Target Model

[Wang et al. SIGKDD '17]
Random Feature Nullification Attack

- Trained target model
- Nullified 50% of features
- Trained substitute model on multiple replications of MNIST test set
- Augmented each set with various levels of random noise
- Define three success metrics
  - RFN-50, RFN-70, RFN-90

[Wang et al. SIGKDD ’17]
Random Feature Nullification Attack

Accuracy vs. L2 Norm for Various Success Metrics

- **Accuracy (%)**
  - 98
  - 96.5
  - 95
  - 93.5
  - 92

- **L2 Norm**
  - 3
  - 2.25
  - 1.5
  - 0.75
  - 0

**Threshold for Success**

- **RFN-50**
- **RFN-70**
- **RFN-90**

**Accuracy for Various Failure Metrics**

- **RFN-50**
- **RFN-70**
- **RFN-90**

**Original Image**

**Adversarial Example (RFN-50)**
Thermometer Encoding

Idea: Discretize features to mask gradients

\[ \tau(p)_j = \begin{cases} 
1, & \text{if } p \geq j/k \\
0, & \text{otherwise}
\end{cases} \]

\( p \): pixel value
\( k \): encoded vector size
\( j \): index of resulting vector

Image Pixel Values

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.12</td>
<td>0.57</td>
<td>...</td>
<td>0.91</td>
</tr>
<tr>
<td>0.87</td>
<td>0.64</td>
<td>...</td>
<td>0.32</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>0.69</td>
</tr>
<tr>
<td>0.56</td>
<td>0.10</td>
<td>0.33</td>
<td>0.82</td>
</tr>
</tbody>
</table>

for \( k = 10 \):

\[
\begin{array}{cccccccccccc}
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

[Backman et al. ICLR '18]
Thermometer Encoding Attack

- Used pre-trained model fortified with thermometer encoding and adversarial training as target model.
- Trained four identical substitute models on CIFAR-10 test set with different levels of random noise.
- Crafted adversarial examples for the first 100 CIFAR-10 images in the test set.

[Buckman et al. ICLR '18], [Athalye et al. ICML '18]
Thermometer Encoding Attack

L2 Norm of Substitute Models
Success Rate: 100%

L2 Norm

<table>
<thead>
<tr>
<th>Substitute Model(s)</th>
<th>A, B, C, D</th>
<th>A, B</th>
<th>C, D</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>L2 Norm</td>
<td>2.2</td>
<td>3.3</td>
<td>3.3</td>
<td>4.4</td>
<td>4.4</td>
<td>4.4</td>
<td>4.4</td>
</tr>
</tbody>
</table>

Original Image
Adversarial Example
SafetyNet

Classifier

Quantizer

SVM w/ RBF

Output Class

Malicious Input?

OR

[Lu et al. ICCV '17]
SafetyNet Attack

- Trained
  - Two substitute models for the original classifier
  - Substitute model for the detector

- Metrics for Success
  - Does the adversarial example fool the classifier?
  - Is the confidence ratio less than 25%?
  - Did the detector predict the adversarial example as a legitimate sample?
For threshold $\theta$:

If $\| G(z^*) - x \|_2 > \theta$, then $x$ is an adversarial example
Defense-GAN Attack

- Trained substitute model with random noise with noise in range [-0.95 - 0.95]
- Used cross entropy loss function for training
- 100% success in fooling classifier and detector
- More powerful than the first approach as this is a true black box attack
Jacobian-Based Data Augmentation

Black Box Attack vs. SST (RFN)

Success Rate (%)

Threshold for Success

SST
JBDA

Adversarial Example Comparison against Thermometer Encoding

Original Image

SST
JBDA Black Box

[Papernot et al. Asia CCS ’17]
Conclusion

• Craft ways to attack deep neural network models that obfuscate gradients in attempt to protect themselves against adversarial examples

• Leveraged our approach against fortifying and detecting defenses

• We can design attacks with no knowledge of the type of defense, the defense and model parameters, and the training data.

• Black box attack evaluations
Questions?

Mohammad Hashemi
mohammad.hashemi@colorado.edu

Gregory Cusack
gregory.cusack@colorado.edu

Eric Keller
eric.keller@colorado.edu