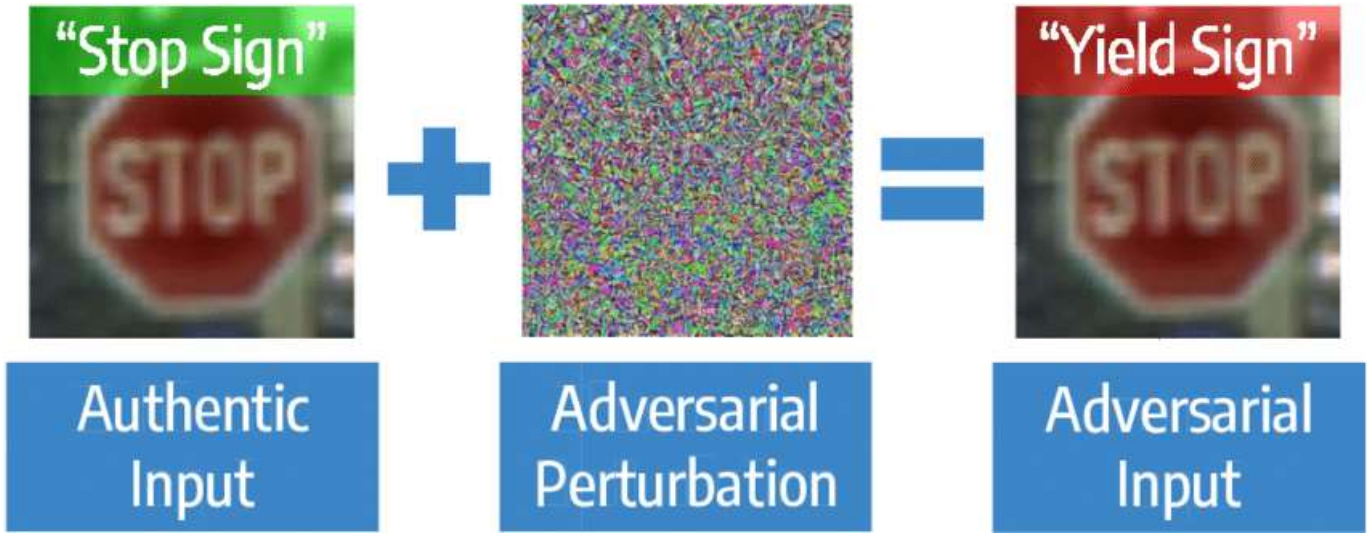

A Review of Adversarial Attacks on Machine Learning Algorithms

Sachin Saxena Mar 8, 2020 · 7 min read

In the recent past, machine learning has been proven to be susceptible to carefully crafted adversarial examples. Here is an overview of the most common adversarial attacks in the white and black box setting.

The generation of adversarial example can be formulated as an optimization
problem as follows: **Find a point within a small neighborhood of the original input to minimize the cost function which can be a suitable distance metric, such that the machine learning algorithm changes its decision on the perturbed input but the decision of a human observer remains the same as it was for the original input**. The attacks can be broadly classified based on the amount of information available to the attacker: black box or white box attacks. White box attacks are those in which the attackers have full information about the model's architecture, weights, and the examples it has trained. Black box attacks refer to those attacks in which only the final output of the model is accessible to the attacker. Black box attacks can be further classified into 3 types based. The first type involves those attacks in which the probability scores to the outputs are accessible to the attacker referred to as the 'score-based black-box attacks', the second type of attack involves the case where information of the training data is known to the attacker, the third attack type is the one in which only.

White Box Attacks

White box attacks involve the classifier f being exposed to the attackers. For neural networks, backpropagation can be conducted on the target model to formulate an attack, since the gradients are known to the attacker.
Examples are Fast Gradient Sign Method (FGSM) by (Goodfellow, Shlens, & Szegedy, 2014), DeepFool by(Moosavi-Dezfooli, Fawzi, & Frossard, 2016), Basic Iterative Method by (Kurakin, Goodfellow, & Bengio, 2016), Jacobian Saliency Map Approach by (Papernot et al., 2016), Carlini and Wagner Attack by (Carlini & Wagner, 2017)

- ***Fast Gradient Sign Method (FGSM)***: (Goodfellow et al., 2014) described the FGSM method in 2014. Given an input image x with a given
  label y, the adversarial image is generated by the following equation
  $x = x + \epsilon \cdot \text{sign}(\nabla_x J(\theta, x, y))$
- ***Basic Iterative Method***: Developed by (Kurakin et al., 2016) .FGSM is applied multiple times with a small step size, iteratively.

- ***Jacobian Saliency Map Approach***: Papernot et al developed the JSMA method for targeted misclassification in 2015. Given an input x with a class t which we need to misclassify as class t'. Firstly, a jacobian matrix is made using forward derivatives for a given input x i.e derivative of one output neuron F_j wrt one input feature (pixel), and the pixel pairs are found in each iteration such that increasing those pixel values leads to an increase in the value of the target class (t') neuron and decrease the neurons corresponding to all the other classes.

- ***Carlini and Wagner Attack***: Given a neural network F, this attack minimizes an objective function that consists of the p norm of the perturbation δ made to the original input x and a loss function that quantifies how close $F(x+\delta)$ is to the target class T. MINIMIZE $||\delta||_p + c \cdot F(x+\delta)$ such that, $x+\delta \in [0,1]^n$

- ***Forward gradient-based approach***: Kolosnjaji in 2018 used a gradient-based approach to attack Malconv and generated Adversarial Malware Binaries. Given an initial binary byte sequence xo, they modify almost 'q' padding bytes at the end of the file thereby assuring that the functionality of the malware binary remains intact. Assuming that we have a binary classifier, and we wish to change the decision of the classifier value from 'Malware' i.e 1 to 'Not Malware" i.e 0. Optimization problem becomes $\mathbf{\min_x f(x)}$ **such that,** $\mathbf{d(x,xo) \leq q}$, *where d(x,xo)*

counts the number of bytes altered in xo to get x. The forward (negative) gradient of the classifier f wrt the j^{th} padding byte x_j is found out and a line is defined in the direction of this gradient passing through x_j. The padding byte is then replaced with that byte that is closest to this line given its projection on the line in the direction of the negative gradient. Thereby, each byte replacement decreases f

- ***A* search in Transformation graph using heuristics***: Kulynych in 2018 proposed a generalized method for white-box adversarial attacks in the discrete domain as an A* graph search problem in the transformation graph. The transformation graph is a weighted directed graph, where each edge is a transformation, its weight is the transformation cost and its children nodes are the transformed examples. The authors used Taylor's expression and the holder's inequality to come up with a heuristic for the graph search. Let us start with an initial example $x \in R^n$ to which we do a perturbation δ in R^n which puts the perturbed example, $x+\delta$, on the decision boundary. Assuming that the decision boundary is $\theta = 0$, we end up with $f(x+\delta) = 0$. Let us write the first-order Taylor approximation of $f(x+\delta)$ at the point x:

Note: If the edge costs are induced by **L_p** norm, then **q** is the Holder's conjugate of **p** given by:

$$\frac{1}{p} + \frac{1}{q} = 1$$

$$f(x+\delta) = f(x) + \nabla f \cdot (x + \delta - x)$$

$$0 = f(x) + \nabla f \cdot \delta$$

$$\delta = \frac{|f(x)|}{|\nabla_x f(x)|^q}$$

Black Box Attacks

Black box attacks can be of the following 3 types:

• ***Score-Based attacks***: Attackers can query the softmax layer output in addition to the final classification result.
-*Scores Feedback* (Guo, Gardner, You, Wilson, & Weinberger, 2019) constructed black-box adversarial images using Confidence scores from the classifier as feedback, and taking random directions(among a pre-specified set of orthogonal directions) to first increase the pixel values (say) ,if confidence increase ,then pixel values are decreased so that confidence would decrease, leading to misclassification.

-*GenAttack*: (Alzantot et al., 2019) proposed a genetic algorithm based approach for gradient-free optimization to generate adversarially images. The fitness function uses the output scores for different classes , maximizing the log scores of the target class and minimizing the log scores of all other classes.

• ***Transfer based attack***: Instead of attacking the original model f, attackers try to construct a substitute model fo to mimic f and then attack fo using white-box attack methods ((Papernot et al., 2017),(Liu, Chen, Liu, & Song, 2016))

• ***Decision-based attack***- Only the final class decision for a given input x is accessible to the attacker

- *Evolutionary Algorithms based approach* (Cresci, Petrocchi, Spognardi, & Tognazzi, 2019) considered the DNA-like representation of the lifetime of each of the Twitter accounts. The LCS curve was taken as the behavioral similarity among a group of users. In each iteration of the genetic algorithm, a group of spambot account DNAs were evolved and the KL divergence between the LCS curves of legitimate accounts and evolved spambots minimized.Although, the evolved spambots after a set of iterations have been shown to evade state pf the art classifiers but, the paper does not talk about the average number of changes made to the spambot DNA in order to evade classification, as it has a dollar-cost linked to and is a critical parameter linked to adversarial example generation.

-*Random Walk on the Boundary* (Brendel, Rauber, & Bethge, 2017) described a random walk on boundary approach to finding adversarial images. The algorithm starts from an adversarial image and a random walk is performed along the boundary between adversarial and non-adversarial regions such that it stays in the adversarial region and the distance from the original image is reduced after each iteration. They used the approach to find adversarial images in, both, targeted and untargeted cases.

-*Optimization-based approach*-cheng et al in 2018 formulated the hard label black box attack as an optimization problem and solved it using a zeroth-order optimization technique known as Randomized Gradient Free Method (Nesterov and Spokoiny, 2017). The loss function in the case, where only the final decision is accessible to the attacker, is discontinuous with discrete outputs. Optimizing the function requires a combinatorial optimization technique which difficult because of the high dimensionality of the input. The formulation of the problem is as follows:

Given a hard label black box model f: $R^d \rightarrow \{1,2....M\}$ and a given example input xo with label yo

$$g(\theta) = \min_{\lambda > 0} \lambda$$

$$f(x + \lambda\theta/|\theta|) \neq y_0$$

The optimization problem then becomes:

$$\min_{\theta} g(\theta)$$

• **TextDecepter** is an algorithm for Hard-Label black-box attacks on Text classification. The code can be found on the Github Link and the paper can be found at Arxiv.

Adversarial Example Adversarial Attack Deep Learning Machine Learning