Adversarial Attacks

Vinay Bhaip

March 2019

1 Introduction

This year, we've talked about many types of models for solving a variety of tasks and how they work, but we haven't really talked about what happens when they don't work. In this lecture, we'll focus on adversarial attacks, which are targeted ways to make a network fail. Specifically, we'll focus on how they affect CNNs, because we can visually see what's happening.

The reason why adversarial attacks are important is because as machine learning algorithms become more prominent in important areas, it becomes even more crucial to protect against these attacks that can effectively compromise a system.

2 Concept

The main way an adversarial attack works is that it takes an image and adds some noise to get a wrong classification of the image.

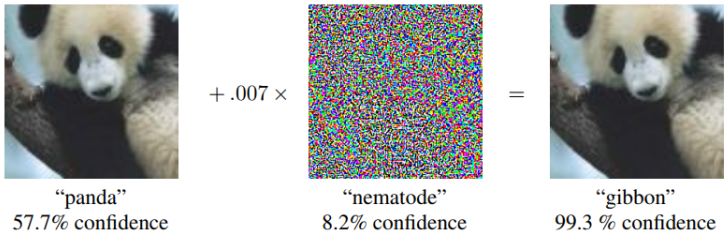

Figure 1: Adversarial Attack for the Misclassification of a Panda

As shown in the figure, When we add the noise to the original image, the model classifies it as a gibbon, although it clearly still looks like a panda. Even worse, the model has a 99.3% accuracy that it is a gibbon. The resulting modified image is known as an **adversarial image**. To quantify how much the original image has been changed, we use the l^∞-*norm*, which is just the greatest magnitude change in a pixel's value.

When conducting an adversarial attack, there's a couple of things to consider beforehand. First, you need to know whether there is access to the parameters of the model you're trying to trick. If these parameters are available, the attack is known as a **white box attack**. Otherwise, it is known as a **black box attack**.

Next, how you approach an adversarial attack depends on the type of problem. If you attempt to pass an image in and force a specific class out, then you want to conduct a **targeted attack**. This could be the case for tasks like impersonation, when someone is trying to make a model recognize an image of a face as a person it is not. The other type of attack is a **non-targeted attack**, in which the goal is just to get the image to be misclassified, but it doesn't matter what it is misclassified as. This could be used in cases such as forcing a system not to recognize license plates of cars.

Gradient-based attacks are the most common types of attacks that are currently out there. Neural networks through backpropagation try to minimize the error when outputting labels. Gradient-based attacks do the exact opposite.

3 One-shot Attacks

One-shot attacks take an image and modify it based off one step according to the gradient.

3.1 Fast Gradient Sign Method

The Fast Gradient Sign Method, or FGSM, finds the gradient of the error function with respect to the current class that is being outputted and takes a step towards it. The formula for this is

$$X_{new} = X + \epsilon * sign(\nabla E(x, y_{true})) \tag{1}$$

This looks familiar, because all it is doing is taking a step in the gradient away, rather than towards, the minima. ϵ here represents some hyper parameter for how large of a step we want to take. Notice with the FGSM, the goal is to just misclassify the image, and thus it is used in non-targeted attacks.

3.2 Targeted Fast Gradient Sign Method

This is very similar to the FGSM, but it allows us to focus on misclassifying the image to a specific target. The formula for this is

$$X_{new} = X - \epsilon * sign(\nabla E(x, y_{target})) \tag{2}$$

The key differences to note is that we are finding the error with respect to the target image we are trying to trick the model into classifying the image as. The T-FGSM takes one step into the negative gradient to try to approach the targeted image, and is accordingly used in targeted attacks.

4 Iterative Attacks

While one-shot attacks only take one step according to the gradient, iterative attacks take multiple.

4.1 Iterative Fast Gradient Sign Method

This is essentially the same thing as FGSM, except its modified so it can take multiple steps. The formula is

$$X_{t+1} = X_t + \alpha * sign(\nabla E(x, y_{true})) \tag{3}$$

where $\alpha = \frac{\epsilon}{t}$. The I-FGSM works tends to work better than the one-shot attack methods in white box attacks, whereas the one-shot attack methods do better in black box attacks. A possible explanation is that iterative methods can overfit models.

5 Black Box Attacks

A question that might arise is that when doing a black box attack, how are we able to find the gradient? There's a couple of methods that have been suggested for this.

Let's assume that with this black box model, all it outputs is the label of the image. We could pass in a picture of a car, for example, and the model should classify it. Then, we add some noise to the image, and see if we can get a different classification out of the model. We keep adding noise until we find a spot where the model misclassifies the original image. The goal is to find an adversarial image that we modified with minimum perturbations that gives out an incorrect class. We can use a binary-search to pinpoint the adversarial image with the smallest l^∞-*norm*. If the model outputs an accuracy as well, the job becomes even easier as we can maximize the error.

But while this is great, it seems kind of inefficient. Another method proposed is to create another model that mimics the one that we do not have parameters to. If the model we create can mirror the outputs of the actual model we want to fool, then the idea is that they will most likely suffer from the same downfalls. After training our own model, we can treat the problem like a white-box attack, as we have the model parameters, and can accordingly calculate the gradients. We then apply the noise that tricks our network to the original model.

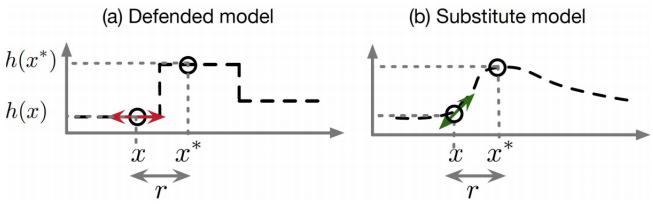

Figure 2: Using a Substitute Model for Black Box Attacks

6 Adversarial Defenses

Since networks are shown to be more and more fragile due to adversarial attacks, researchers have been looking at how to protect networks from these attacks. The most obvious way to protect against attacks is to just train on the adversarial samples that tricked the original model. However, this doesn't seem completely solid as there will most likely be some shortcoming in the model that an adversarial attack can still exploit. This lecture isn't going to go into all the specifics of adversarial defenses, but I encourage you to check them out if you're interested.

7 Acknowledgements

I didn't create any of the figures used in this lecture. All credit goes to their respective owners:

- Nicolas Papernot's Gradient Masking Lecture at Stanford
- Explaining and Harnessing Adversarial Examples Paper
- Emil Mikhailov and Roman Trusov's Adversarial Attack Explanation
- Joao Gomes' Summary on Adversarial Attacks and Defenses