UNDERSTANDING ADVERSARIAL ATTACKS

YANG Rongfeng
Big Data Technology
Hong Kong University of Science and Technology
Clear Water Bay, Hong Kong
ryangag@connect.ust.hk

ABSTRACT

This article tents to make an overview of the Adversarial Attacks in image recognition area. To help you understand the topic, we do explanations rather than covering mathematical formulae. Firstly, questions are asked to make you think the differences between humans and neural network models when learning knowledges. Then we dive further into the Adversarial Attacks technologies and discuss the nature of them. At last, we'll show that attacks in real world and how to defend against them.

As the development of theories of deep learning, scholars are very excited at the remarkable performance that neural networks achieve. In ILSVRC (ImageNet Large Scale Visual Recognition Challenge) competition from 2010 to 2015, the network went deeper and deeper and finally the ResNet achieved 3.57% error rate which is smaller than human's 5.1%. However, do these accomplishments mean the win of deep learning? Do the neural networks really learn knowledges like humans?

Computer scientists borrowed the concept of neural network in biology but strictly speaking we need to be very careful with the brain analogies. Actually, London and Hausser [Dendritic Computation, 1910] summarized that biological neurons have many different types and dentrites can perform complex non-linear computations which are much more complicated than the activation functions we have proposed nowadays. They also proposed that synapses are not a single weight but a complex non-linear dynamical system. Therefore, many researchers dislike the term of neural network because it is imprecise and not scientific. One saying is *Any artificial intelligence is the bad imitation of human's brain*, so could the model really understand the concept of things like humans?

1 BEGIN WITH A STORY OF CLEVER HANS

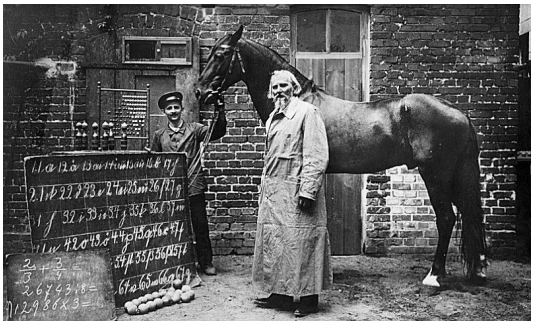

Figure 1: Clever Hans and Wilhelm von Osten

In the late 1800s, a German high school mathematics instructor and amateur horse trainer, Wilhelm von Osten, had a horse that he claimed was able to perform arithmetic and other intellectual tasks. Clever Hans, an Orlov Trotter, was said to have been taught to add, subtract, multiply, divide, work with fractions, tell time, spell, read, and understand German. Questions were asked orally and in writing, and the horse answered by tapping his foot.

It's sound wonderful, isn't it? Just A horse can do such complex things. When the New York Times reported on the horse's amazing abilities in 1904, many researchers began to study the phenomenon. They found that Hans could answer correctly only when the questioners knew the answers and Hans could see them. If Hans couldn't see the person who asked the question, his accuracy dropped from 89% to 6%.

Imagine that we went back to the past and had opportunity to ask questions to Hans. When we asked that "What is three plus five, Hans?". When he knocked his foot to the right answer - eight times, we may have a tense facial expression or some kinds of delight emotion or whatever. These were captured by Clever Hans so he knew that he got the correct answer. However, if asking something that we didn't know such as "How many countries in the Middle East?", there would no any clues for Hans to follow so he could not answer it correctly.

Therefore, Hans actually had no idea of the meanings of questions. He got the right anwsers by observing humans' small involuntary postural and expression changes.

The neural network in deep learning is like Clever Hans. It focuses on the wrong clues without "learning" something. Although, the networks defeated humans at image classificition area in ILSVRC, they were not truely understanding the pictures contents. With **visualizaition technology**, the stacked conventional layers, pooling layers, outputs layers, activation function, normalization and dropout may be explained, but it's still not convincing. Besides visualizaition, **adversarial attacks** go from another perspective to study the neural network models by fooling the networks.

2 Adversarial Attacks Technologies

The adversarial attacks experiments prove that the neural networks are not like our humans, learning and understanding the concept of objects. In figure 2, we add small but intentionally worst-case perturbations to the input image and then the model completely changes its original prediction of pandas into gibbon with high confidence (99.3%). This is what adversarial attacks do - try to change a picture a bit (humans still can recognize) but dramatically change the outcome of the model.

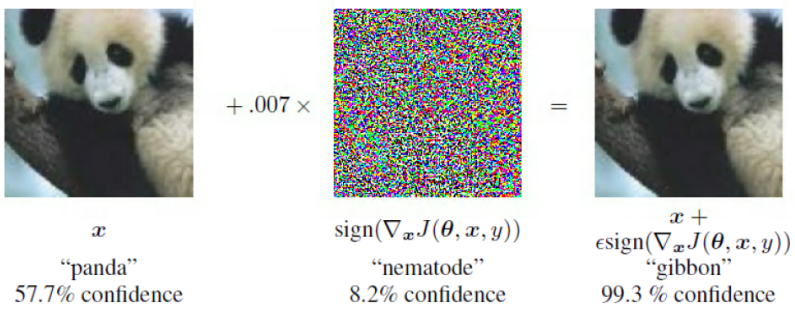

Figure 2: An adversarial attack demonstration: pandas to gibbon

According to whether the attacker has the knowledges of the targeted model, adversarial attacks can be divided into white-box attacks and black-box attacks. The former assumes the complete knowledge of the targeted model, including its parameter values, architecture, training method, and in some cases its training data as well. The latter feeds a targeted model with the adversarial examples that are generated without the knowledge of that model.

Based on the effect that adversarial attacks have on the model, they can be divided into untargeted attacks and targeted attacks. If the adversarial examples force the model to go far away from the ture class, it is called the untargeted attack. However, fooling the model into classifying an input to a particular target class is the targeted attack.

There are many researches doing on adversarial attacks these years. Box-constrained L-BFGS [Szegedy at al. 2014] , C & W attack (Carlini and Wagner 2017) are targeted attacks. C & W attack is considered strong attack but computationally expensive. Fast gradient sign method (FGSM) [Goodfellow et al. 2015] is popular and it has several variations such as fast, basic iterative method

(BIM) [Kurakin et al. 2016], least likely class. FGSM also can be used to produce targeted adversarial examples. Jacobian-based Saliency Map Attack (JSMA) [Papernot et al. 2016] takes advantages of the saliency map [Simonyan et al. 2014] to find the key pixels in images and modify them to increase the likelihood of a target class. Moreover, DeepFool [Moosavi-Dezfooli et al. 2016] is more efficient than L-BFGS. It's interesting that Universal Adversarial Perturbations [Moosavi-Dezfooli et al. 2017] designed a method to find a universal adversarial vector that can cause misclassification of multiple images.

3 Nature of Adversarial Attacks

The story of Clever Hans explains the nature of adversarial attacks in some extent. The model follows the wrong clues and doesn't **"understand"** the knowledges like humans. In another word, the model learns an incomplete, low-level form of knowledges.

From other aspects, linearity hypothesis [Goodfellow et al. 2015] and boundary tilting perspective [Tanay and Griffin 2016] also can explain the nature of adversarial attacks. Linearity hypothesis considers that even if the perturbation is tiny, the change to the output can be very large with large dimensionality of weight $\boldsymbol{w}$ or input $\boldsymbol{x}$.

Boundary tilting perspective (figure 3) is more understandable. The data sampled in the training and test sets only extends in a submanifold of the image space which represents by grey hyperplane. The red hyperplane, the class boundary, separates the two classes well by intersecting the submanifold but it'll extend beyond it. If the boundary is lying very close to the data, at boundary nearby, small perturbations directed towards the boundary might cross it causing class changing.

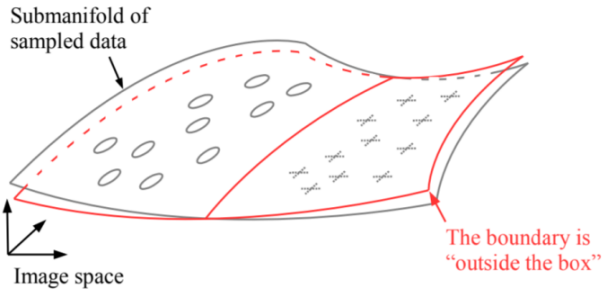

Figure 3: The boundary tilting perspective to explain adversarial examples

Szegedy et al. [2014] also found that the adversarial examples generated for one model often fool other models with different structures and trained on different datasets. This phenomenon was call transferability.

4 Attacks in Real World and How to Defend

In this section, let's see some attacks in real world. Do you know why self-driving cars are not widely used? One of the most important reasons is that the safety of cars needs to be further proved. Eyholt et al. put some stickers on a STOP sign making the recognition system of a self-driving car ignore it. When the car passed a pedestrian crossing without traffic lights, it didn't stop and nearly crashed a woman crossing the road. It's dangerous and may be utilized for bad purposes.

The facial identity recognition is widely used in our lives. We use our face to unlock our phones, pay the bill, withdraw money from the bank and etc. The accessory attack showed that we are able to hack a recognition system with some intentionally designed accessory like a pair of glasses. If we could impersonate any person in front of a facial recognition, many security facilities are in danger.

The 3D attack is the most valuable real-world application I think. Athalye et al. made a 3D-printed turtle adversarially perturbed to classify as a rifle at every viewpoint. Real world environment is complicated, some adversarial examples may fail when changing viewpoint. This experiment,

however, showed the "super" adversarial example did exist and it's urgent to do research on defences against adversarial attacks.

At last, we discuss how to protect the neural networks model from being hacked by adversarial examples. First we need to understand that the attack vulnerability is caused by the network itself. Therefore, currently, no defenses are considered successful. We could improve the robustness of model against adversarial examples but we cannot eliminate the vulnerability.

To defend against the attacks, adversarial training method comes at first. Since you produce adversarial examples to attack my model, I also feed the adversarial samples during training period. This is effective to attacks considered in training but not effective to those without training. Compared with the green curve, the adversarially trained model with light blue curve is more robust against adversarial examples (figure 4).

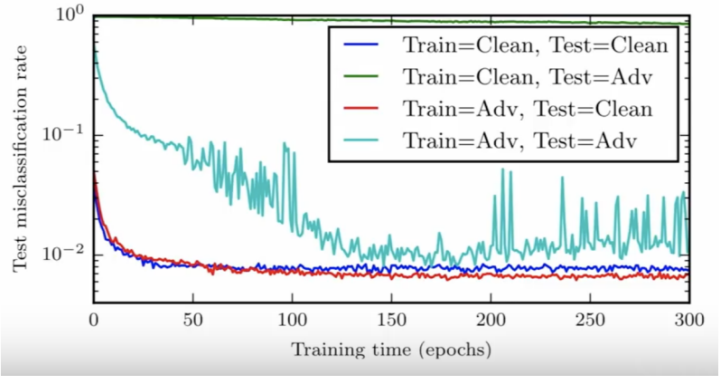

Figure 4: Adversarial training comparison

We also have the following methods to reinforce the model:

- Adversarial Loss [Madry et al. 2018]
- Distillation [Papernot et al 2016]
- Feature Squeezing [Xu et al]

The key point of *adversarial loss* is to define an adversarial loss function. When minimizing the loss, the effectiveness of adversarial inputs also be minimized and meanwhile we get an optimal anti-attack model. This is one of the strongest defenses against adversarial attacks but fails to scale up the ImageNet-scale tasks. *Distillation* creates a term of temperature. As temperature increases the distribution of the probability vector of class becomes more and more smooth, tending to the uniform distribution. It aimed to reduce the amplitude of the gradient i.e. gradient masking. The distillation make attackers hard to find the direction of gradient implying model vulnerability. It was proved to be effective in defending JSMA attacks and with minimum impact on accuracy, but was broken by C & W attack. *Feature squeezing* includes two core ideas: bit depth reduction and spatial smoothing. If prediction on original image differs from that on modified image, then the original image is classified as adversarial example. It was shown to be quite effective and can be combined with adversarial training.