CS 194-129 Deep Neural Networks
Spring 2018 Li Lecture 18

1 Reading

Adversarial Examples

We can fool ML models by applying worst-case perturbations to inputs. When it comes to neural networks, Goodfellow et al. argue that vulnerability to such **adversarial examples** arises from linear behavior (dot products) in high-dimensional spaces. Linearity is the issue, not nonlinearity.

On the plus side, adversarial training (in which adversarial examples are added to the training data) serves as regularization that works on top of dropout.

Transferability / Black-Box Attacks

Here, Liu et al. conduct a study of transferable non-targeted and targeted adversarial examples over large models and datasets. While adversarial examples will often transfer across models, it is rare for *targeted* examples (at least generated using current methods) to transfer with their target labels. Accordingly, the authors introduce a new ensemble-based method for generating transferable targeted examples.

Note: since the examples transfer, they can be used to attack black-box models. For example, we can fool black-box image classification systems with fake viaduct images.

Physical Perturbation

Evtimov et al. propose Robust Physical Perturbations (RP_2), a method for producing targeted adversarial examples that are robust to changing physical conditions. They also propose an evaluation methodology for evaluating physically realizable adversarial perturbations.

2 Lecture

Building secure ML systems is one of the most important research directions in the modern era. As it stands, we exist in environments conducive to adversarial agendas, and accordingly the increasing spread of "intelligent" systems has raised many security and privacy concerns.

We would like to make our learning systems more robust to attack.

In traditional ML systems, we usually make the assumption that the training and testing data are both drawn from a similar stationary distribution. However, this is no longer true if there are adversaries in the picture. Either the training or the testing distribution might be manipulated!

Examples of Adversarial Attacks

One example of an adversarial attack is an **FGSM-based** (fast gradient sign method) attack. It involves adding a very small perturbation along the loss gradient direction. This can be used for either a targeted or non-targeted attack.

Another example is **optimization-based attack**, which solves a constrained optimization problem in order to find the desired adversarial perturbation. On a high level, it tries to minimize the distance between x (the original) and $x + \delta$ (a generated instance; δ is noise) s.t. $x + \delta$ is misclassified as target t. This method is able to produce adversarial perturbations of minimal magnitude.

Generating Physical Adversarial Examples

We would like to synthesize physical adversarial examples (i.e. adversarial examples that look like the physical world). There are four main challenges: (1) *varying physical conditions*, (2) *physical limits on imperceptibility*, (3) *fabrication and perception error*, e.g. color reproduction, and (4) *background modifications.*

We can proceed via optimization. We will find

$$\arg\min_{\delta} \lambda\|\delta\|_p + J(f_\theta(x+\delta), y^*)$$

i.e. a perturbation δ of minimal magnitude that causes the input to be misclassified as y^*. J is a loss function between the classification result $f_\theta(x+\delta)$ and the desired adversarial target label y^*.

To handle varying physical conditions (1), we can generate a "universal perturbation" by using a bunch of examples from different physical conditions:

$$\arg\min_{\delta} \lambda\|\delta\|_p + \frac{1}{k}\sum_{i=1}^{k} J(f_\theta(x_i+\delta), y^*)$$

To handle limits on imperceptibility (2), we can add a mask M_x which introduces a spatial constraint on the perturbation.

$$\arg\min_{\delta} \lambda\|M_x \cdot \delta\|_p + \frac{1}{k}\sum_{i=1}^{k} J(f_\theta(x_i+M_x\cdot\delta), y^*)$$

Then the perturbation will be within a certain shape that can "hide in the human psyche" – it'll be a shape we're used to, presumably, and mimic vandalism. Then the perturbation can, in a way, exceed typical physical limits within this region.

To handle fabrication and perception error (3), we can add an NPS loss function:

$$\arg\min_{\delta} \lambda\|M_x \cdot \delta\|_p + \frac{1}{k}\sum_{i=1}^{k} J(f_\theta(x_i+M_x\cdot\delta), y^*) + \text{NPS}(M_x\cdot\delta)$$

where

$$\text{NPS}(\delta) = \sum_{\hat{p}\in\delta}\prod_{p'\in P}|\hat{p}-p'|$$

for P a set of printable RGB triplets. In other words, we minimize the distance between the generated colors and the printable colors, making sure we can reproduce physically realistic colors.

By solving this optimization problem, we can successfully generate physically realistic adversarial examples. Hence *adversarial examples can exist in the physical world*, and they can even be robust to different conditions and viewpoints (including variations in distance and viewing angle).

Black-Box Attacks

So far, all adversarial approaches have required knowledge of the ML model f_θ. But often people will not have access to this, which is why black-box attacks become relevant.

There are a few categories of black-box attacks:

- **Zero-query attacks**, in which we don't need to query the black-box model. This is mainly based on the idea of *transferability*, where adversarial examples transfer across different models.
- **Query-based attacks**, in which we do need to query the black-box model (and get more information about it before performing an attack). This is more systematic and provides more of a guarantee.

To make a zero-query black-box attack *targeted*, we can ensemble-attack different white-box (known) models in a targeted fashion. Thanks to the ensemble, the target will often transfer to even a previously unseen black-box system.

In a query-based fashion, we can perform *finite difference gradient estimation*, which makes $2d$ queries for an instance $x \in \mathbb{R}^d$ in order to estimate the gradient. [Note: we can reduce dimensionality (/the number of queries) by using PCA or random grouping.]

Generating Adversarial Examples with GANs

Instead of using an L2-norm as a distance metric between "adversarial" and "benign," we can use a discriminator (in a GAN) to tell us if an adversarial example looks good or not. Then the GAN will generate realistic adversarial examples.

Spatially Transformed Adversarial Examples

We can also move the pixels around (appearance flow) instead of changing pixels' values. For each pixel, we find a flow to a different position (training a spatial transformer network to optimize this transformation).

Adversarial Examples for Other ML Models

VAE-GAN: here, the attacker's goal is to get the VAE decoder to reconstruct a different image than the one that the encoder sees. In this setup, the adversary is between the encoder and decoder (and can "intercept" the encoding). Adversarial attacks have succeeded in this domain, meaning VAE-like generative models are also vulnerable to adversarial examples.

VQA models and deep RL algorithms can also be fooled by adversarial perturbation. To attack an RL system, we don't even need to perturb every frame! (We can perturb every 10th frame or something.)

Defense

To detect adversarial examples, we can take a batch of instances and use an arbitrary attack method to generate adversarial instances (such that we have one batch of benign data and one batch of malicious data). Then we can calculate LID (local intrinsic dimensionality) for each instance based on each layer (s.t. we end up with a k-dimensional feature vector for each instance, where k is the number of layers). Finally we can train a logistic regression model based on these feature vectors.

When a new instance comes in, we can again calculate the LID based on different layers and give it to our model, which will tell us whether it's adversarial or benign.

Note: there are two types of people in the defense community. One believes that if a model has higher capacity, it's more robust to adversarial attack. The other believes that a ***lower*** *capacity means more robustness.*

References

[1] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy. Explaining and Harnessing Adversarial Examples. *arXiv preprint arXiv:1412.6572* (2014).

[2] Yanpei Liu, Xinyun Chen, Chang Liu, Dawn Song. Delving into Transferable Adversarial Examples and Black-box Attacks. *arXiv preprint arXiv:1611.02770* (2016).

[3] Ivan Evtimov, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, Dawn Song. Robust Physical-World Attacks on Deep Learning Models. *arXiv preprint arXiv:1707.08945* (2017).