On the Machine Illusion

Proposal of Study on Adversarial Samples

Zhitao Gong

Auburn University

April 12, 2018

Outline

Neural Networks

It is a connectionist model.

1. Any state can be described as an *N*-dimensional vector of numeric activation values over neural units in a network.
2. Memory is created by modifying the strength of the connections between neural units.

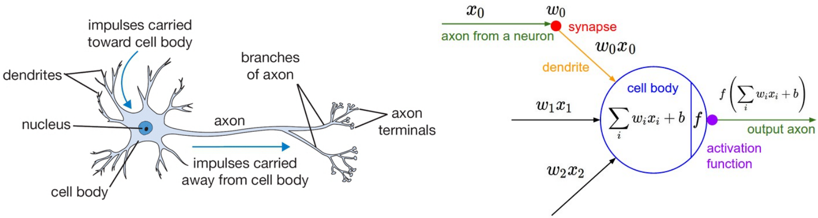


Figure: Biological neuron versus neuron model (credit: cs231n)

Architectures: Multi-Layer Perceptron (MLP)

MLP is one of the most simple feedfoward architectures.

1. Each neuron outputs to the neurons in the next layer.
2. Neurons in the same layer have no connections.

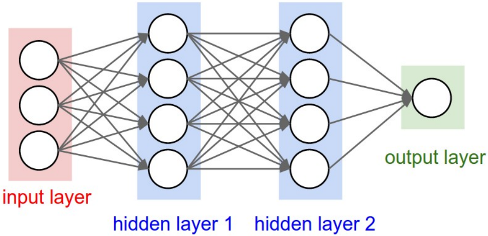


Figure: Multi-layer perceptron (credit: cs231n)

Architectures: Convolutional Neural Network (CNN)

CNN is inspired by eye structure, widely used in computer vision.

1. Each neuron receives inputs from a pool of neurons in previous layer, just like the convolution operation.
2. Neurons in the same layer have no connections

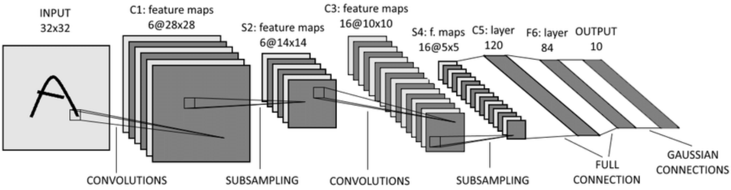


Figure: LetNet-5 [LeC+98]

Architectures: Recurrent Neural Network (RNN)

Some neurons get part of input from its output.

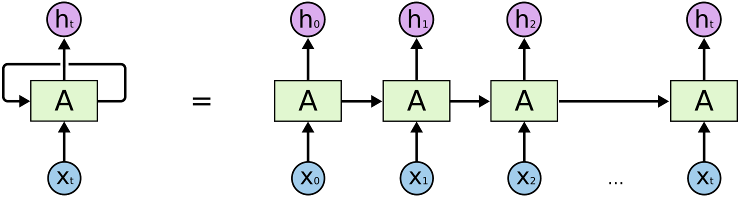


Figure: Dynamic unrolling of recurrent cells. (credit: colah's blog)

Architectures: Recurrent Neural Network (RNN)

Some neurons get part of input from its output.

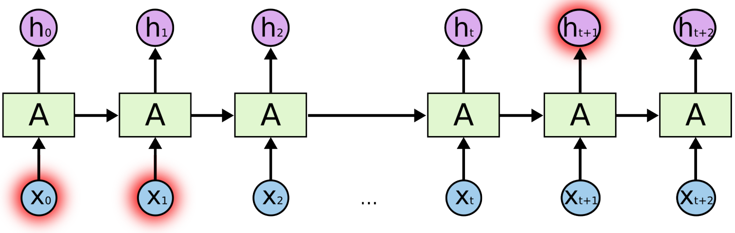


Figure: The double-edged sword: long term dependencies between outputs and inputs. (credit: colah's blog)

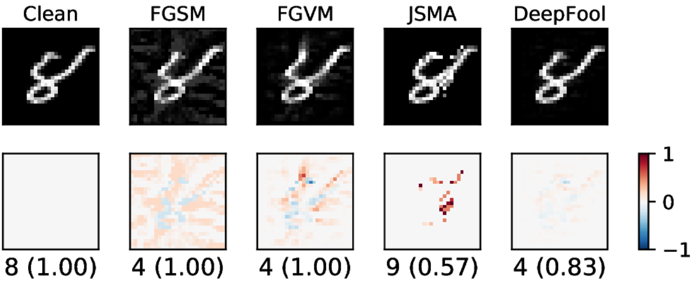
Notations

For clarity, we use the following notations in this slide.

- f denotes the neural nets model, θ the model's parameters, and sometimes f_θ for brevity.
- x is the input, y the model's output, such that $y = f(x)$ or $y = f(x; \theta)$ to emphasize the parameters.
- z is the un-normalized logits, i.e., $y = \text{sigmoid}(z)$ or $y = \text{softmax}(z)$.
- L denotes the loss function, e.g., cross-entropy, mean-squared error. For simplicity, we use L_x to denote the loss value when x is the input.
- x^* denotes the adversarial sample crafted based on x.
- In a targeted method, y_t denotes the target class value, y_o the other class values. For example, $y = [0.2, 0.5, 0.3]$ and $t = 0$, then $y_t = 0.2$ and $y_o \in \{0.5, 0.3\}$. Same for z.

Outline

Adversarial Samples I



1. Visually very close, noises are very subtle.
2. Trick machines into wrong predictions with high confidence.

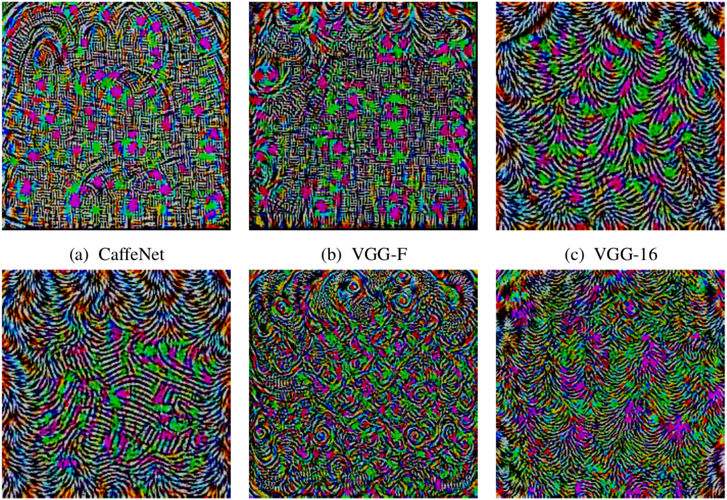
Adversarial Samples II

Clean Text	Label	WMD (n/L)	Adversarial Text
Quick summary of the book : **[...]** The book was n't bad , but was sooooooo cliché < br / > < br / > Now about the movie **[...]** (IMDB)	0→1	0.0317 (0.0050)	Quick summary of the book : **[...]** The book was n't bad , but was sooooooo TahitiNut < br / > < br / > Now about the movie **[...]**
zulchzulu < SM > TO OFFER SPECIAL DIVIDEND Southmark Corp said it will issue its shareholders a special dividend right **[...]** (REUTERS-2)	1→0	0.0817 (0.0125)	zulchzulu < SM > TO OFFER OFFERS SHARES Southmark Corp said it will issue its shareh olders a special dividend right **[...]**
U . K . MONEY MARKET GIVEN FURTHER 68 MLN STG HELP The Bank of England said it provided the market with a further **[...]** (REUTERS-5)	3→2	0.0556 (0.0077)	U . K . MONEY MARKET GIVEN FURTHER 68 ARL STG HELP The Bank of England said it provided the market with a further **[...]**

Figure: Adversarial texts by our framework.

The highlighted words are changed. The n/L is the number of words changed divided by the total number of words.

Adversarial Patterns for Machines



(a) CaffeNet (b) VGG-F (c) VGG-16

(d) VGG-19 (e) GoogLeNet (f) ResNet-152

Figure: Adversarial patterns for different neural nets [Moo+16].

Adversarial Patterns For Humans

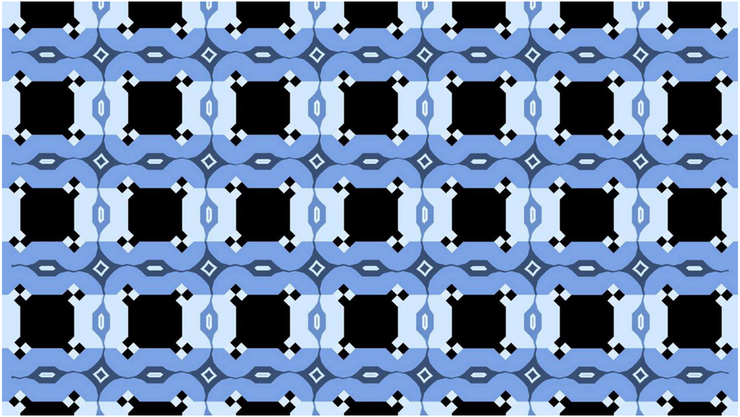


Figure: The blue lines are parallel. This illusion is possibly caused by the fringed edges [KPB04].

More examples: http://www.psy.ritsumei.ac.jp/~akitaoka.

Why Study Adversarial Samples

This phenomenon is interesting both in practice and in theory.

1. It undermines the models' reliability.
2. Hard to ignore due to it being transferable and universal.
3. It provides new insights into neural networks:
   - Local generalization does not seem to hold.
   - Data distribution: they appear in dense regions.
   - Trade-off between robustness and generalization.
   - ...

Outline

Overview

Intuitions behind the adversarial methods

1. Move the data points
   - towards the decision boundary [MFF15; Moo+16],
   - in the direction where loss increases for the clean samples [GSS14; KGB16], or decreases for the for the adversarial decreases [Sze+13], or
   - increase the probability for the correct label and/or decrease the others [Pap+15; CW16].
2. Map between clean and adversarial data points [ZDS17; BF17; Xia+18].

Intuition

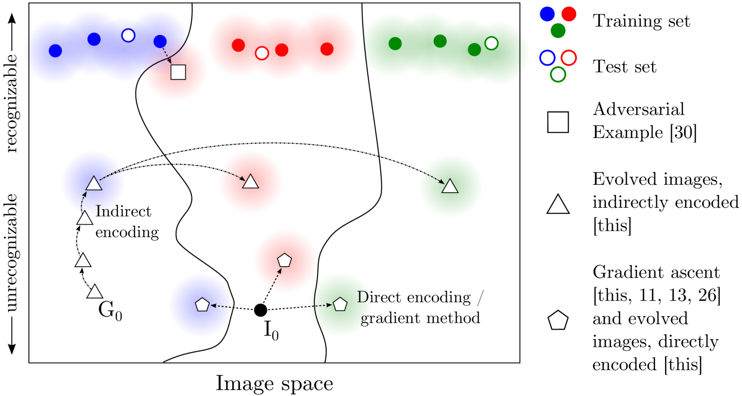


Figure: Data space hypothesis [NYC14]

Outline

Text Embedding Layer

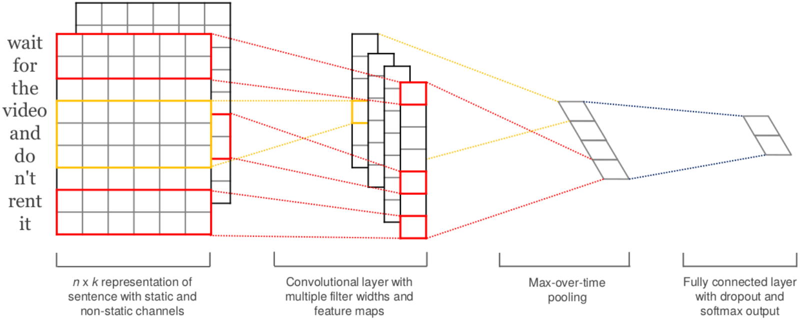


Figure: Architecture for sentence classification with CNN [Kim14]

Text Embedding Example

"wait for the video" $\xrightarrow{\text{tokenize}}$ ["wait", "for", "the", "video"] $\xrightarrow{\text{indexer}}$ [2, 20, 34, 8] $\xrightarrow{\text{embedding}}$ $\mathbb{R}^{4\times D}$, where D is the embedding size.

- Each sentence with be converted to $\mathbb{R}^{L\times D}$ before being fed into the convolution layer, where L is the sentence length.
- We usually truncate/pad sentences to the same length so that we could do *batch training*.
- Embedding may also be on the character-level.

Problem Overview

Difficulties we face:

1. The text space is discrete. Moving the data points in small steps following a certain direction does not work, directly.
2. Text quality is hard to measure. *Much to learn, you still have* (the Yoda-style) v.s. *You still have much to learn* (the mundane-style)

General directions:

1. Three basic operations are available, *replacement*, *insertion*, and *deletion*.
2. They may work at character, word or sentence level.

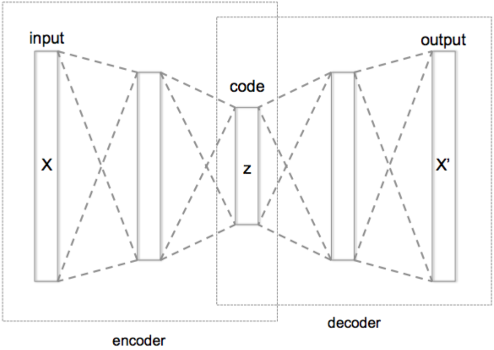
Methods in Text Space

This class of methods need to solve two problems:

1. what to change, e.g., random [Ano18], ∇L [Lia+17], manually picking [SM17].
2. change to what, e.g., random, synonyms [SM17] or nearest neighbors in embedding space [Ano18], or forged facts [JL17; Lia+17].

Methods in Transformed Space

Autoencoder [HS06] is used to map between texts and a continuous space [ZDS17]. The embedded space is smooth.



Adversarial Text Framework

We propose another method in the embedding space.

```
GENERATE-ADVERSARIAL-TEXTS(f, x)
for i = 1 to x.length
        z_i = EMBEDDING(x_i)
z' = ADV(f, z)
for i = 1 to z'.length
        x'_i = NEAREST-EMBEDDING(z'_i)
        s_i = REVERSE-EMBEDDING(x'_i)
return s
```

Assumptions:

1. The text embedding space preserve the semantic relations.
2. Important features get more noise.

Result: https://github.com/gongzhitaao/adversarial-text

Next Step

1. Find appropriate quality measurement for texts, e.g., language model scores, Word Mover's Distance (WMD).
2. Find a way to control the quality of generated adversarial texts.
3. Test the transferability of adversarial texts.

Outline

Enhance Model

Basic ideas: incorporate adversarial samples during training process, and/or improve architectures.

Given a training set $\mathcal{X}$, instead of minimizing

$$\theta^* = \arg\min_\theta \mathbb{E}_{x\in\mathcal{X}} L(x; f_\theta)$$

we expand each data point a bit

$$\theta^* = \arg\min_\theta \mathbb{E}_{x\in\mathcal{X}} \left[\max_{\delta\in[-\epsilon,\epsilon]^N} L(x+\delta; f_\theta) \right]$$

[GSS14; Mad+17] solve the inner maximization problem by mixing dynamically generated adversarial samples into training data.

Preprocess Inputs

Without re-training the models, this direction focuses on the inputs.

1. Transform inputs to (hopefully) recover the bad samples.
2. Filter out bad samples by image statistics.

Binary Classifier as A Defense

Taking advantage of the observation that the adversarial noise follows a specific direction [GSS14]. We build a simple classifier to separate adversarial from clean data [GWK17].

Table: FGSM ϵ sensitivity on CIFAR10

ϵ	$f_2\big\vert_{\epsilon=0.03}$ X_{test}	$X_{test}^{adv(f_1)}$
0.3	0.9996	1.0000
0.1	0.9996	1.0000
0.03	0.9996	0.9997
0.01	0.9996	**0.0030**

Limitation: different hyperparameters, different adversarial algorithms may elude the binary classifier or adversarial training.
Results: https://github.com/gongzhitaao/adversarial-classifier

Next Step

1. Closely investigate the limitation of binary classifier approach.
2. Detect and/or recover adversarial texts

GENERATION IS CHEAP,
DEFENSE IS DIFFICULT.

Outline

Adversarial Samples

1. All classification models are affected.
2. Seems to exist in dense regions.
3. Distribute along only certain directions.
4. Transfer to different models or techniques.
5. ...

ALL EMPIRICAL AND HYPOTHESIS SO FAR

Outline

[Ano18] Anonymous. “Adversarial Examples for Natural Language Classification Problems”. In: *International Conference on Learning Representations* (2018). URL: https://openreview.net/forum?id=r1QZ3zbAZ.

[BF17] Shumeet Baluja and Ian Fischer. “Adversarial Transformation Networks: Learning To Generate Adversarial Examples”. In: *CoRR* abs/1703.09387 (2017). URL: http://arxiv.org/abs/1703.09387.

[CW16] Nicholas Carlini and David Wagner. “Towards Evaluating the Robustness of Neural Networks”. In: *CoRR* abs/1608.04644 (2016). URL: http://arxiv.org/abs/1608.04644.

[GSS14] I. J. Goodfellow, J. Shlens, and C. Szegedy. “Explaining and Harnessing Adversarial Examples”. In: *ArXiv e-prints* (Dec. 2014). arXiv: 1412.6572 [stat.ML].

[GWK17] Zhitao Gong, Wenlu Wang, and Wei-Shinn Ku. “Adversarial and Clean Data Are Not Twins”. In: *CoRR* abs/1704.04960 (2017).

[HS06] G. E. Hinton and R. R. Salakhutdinov. “Reducing the Dimensionality of Data With Neural Networks”. In: *Science* 313.5786 (2006), pp. 504–507. DOI: 10.1126/science.1127647. eprint: http://www.sciencemag.org/content/313/5786/504.full.pdf. URL: http://www.sciencemag.org/content/313/5786/504.abstract.

[JL17] Robin Jia and Percy Liang. “Adversarial Examples for Evaluating Reading Comprehension Systems”. In: *arXiv preprint arXiv:1707.07328* (2017).

[KGB16] A. Kurakin, I. Goodfellow, and S. Bengio. “Adversarial Examples in the Physical world”. In: *ArXiv e-prints* (July 2016). arXiv: 1607.02533 [cs.CV].

[Kim14] Yoon Kim. “Convolutional Neural Networks for Sentence Classification”. In: *CoRR* abs/1408.5882 (2014).

[KPB04] Akiyoshi Kitaoka, Baingio Pinna, and Gavin Brelstaff. “Contrast Polarities Determine the Direction of Café Wall Tilts”. In: *Perception* 33.1 (2004), pp. 11–20.

[LeC+98] Yann LeCun et al. “Gradient-Based Learning Applied To Document Recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[Lia+17] Bin Liang et al. “Deep Text Classification Can Be Fooled”. In: *arXiv preprint arXiv:1704.08006* (2017).

[Mad+17] Aleksander Madry et al. "Towards Deep Learning Models Resistant To Adversarial Attacks". In: *arXiv preprint arXiv:1706.06083* (2017).

[MFF15] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. "Deepfool: a Simple and Accurate Method To Fool Deep Neural Networks". In: *CoRR* abs/1511.04599 (2015). arXiv: 1511.04599. URL: http://arxiv.org/abs/1511.04599.

[Moo+16] Seyed-Mohsen Moosavi-Dezfooli et al. "Universal Adversarial Perturbations". In: *arXiv preprint arXiv:1610.08401* (2016).

[NYC14] Anh Mai Nguyen, Jason Yosinski, and Jeff Clune. "Deep Neural Networks Are Easily Fooled: High Confidence Predictions for Unrecognizable Images". In: *CoRR* abs/1412.1897 (2014). URL: http://arxiv.org/abs/1412.1897.

[Pap+15] Nicolas Papernot et al. "The Limitations of Deep Learning in Adversarial Settings". In: *CoRR* abs/1511.07528 (2015). URL: http://arxiv.org/abs/1511.07528.

[SM17] Suranjana Samanta and Sameep Mehta. "Towards Crafting Text Adversarial Samples". In: *arXiv preprint arXiv:1707.02812* (2017).

[Sze+13] Christian Szegedy et al. "Intriguing Properties of Neural Networks". In: *CoRR* abs/1312.6199 (2013). URL: http://arxiv.org/abs/1312.6199.

[Xia+18] C. Xiao et al. "Generating Adversarial Examples With Adversarial Networks". In: *ArXiv e-prints* (Jan. 2018). arXiv: 1801.02610 [cs.CR].

[ZDS17] Z. Zhao, D. Dua, and S. Singh. "Generating Natural Adversarial Examples". In: *ArXiv e-prints* (Oct. 2017). arXiv: 1710.11342 [cs.LG].