Assessing the Resistance of Biologically-Inspired Neural Networks to Adversarial Attack

Ted Moskovitz
Security and Robustness of ML Systems

May 8, 2018

1 Introduction

It is well known that many machine learning algorithms, and in particular deep neural networks, are highly susceptible to adversarial inputs. Such attacks, often consisting of subtle and carefully chosen perturbations to observed data, induce a normally high-performing target network to either misclassify inputs or otherwise behave erratically. An intriguing property of these adversarial examples is that to humans, they are often almost imperceptibly different from unmodified inputs. This raises the question: Why are these examples so devastating to artificial neural networks, and so useless against their biological progenitors? Another setting easily accounted for by biological systems, but in which artificial networks struggle, is that of natural image transformations, such as brightening or rotation. Employing data augmentation during the training process often ameliorates the issue, but it is also worth considering whether there is some fundamental feature of biological networks that instills invariance to these transformations. While deep learning models are fundamentally inspired by cortical networks and basic neurobiology, there are still substantial differences that separate the two. I explore these differences in the context of security, applying adversarial inputs to networks with biologically-inspired adaptations with the goal of testing if such implementations are more resilient against attack.

In particular, I explore the performance of two very different techniques, a top-down goal-directed attention mechanism, and a biologically feasible learning algorithm.

2 Related Work

Recent work has shown that despite the rapid improvement in computer vision through the embrace of deep learning, neural networks can be fooled. Many such methods take advantage of the observation by Szegedy et al. 2014 [14] that the high-dimensional nonlinearity embodied by a deep network is not necessarily locally smooth. That is, small, targeted modifications to an input can often induce a radically different output from the top layer of the network. Accordingly, a number of methods have been developed that attempt to subtly modify input images so as to shift top layer activations along the output manifold to produce an incorrect classification [14, 11, 2]. One of the most common and effective attacks, developed by Goodfellow et al. [2], creates a perturbation by taking the sign of the gradient with respect to the input, scaling it, and adding it to previously uncontaminated input (more detail in Section 3.3). The low computational cost required to generate these examples lends this technique its name, the Fast Gradient Sign Method (FGSM).

There are a number of deep learning techniques that are designed to take these architectures even closer to their biological counterparts. Apart from the visual cortex itself, one of the core neural functions affecting the way we parse visual inputs and identify objects

is attention [4]. In recognition of the significant effect attention has on localizing salient information in the environment, training neural networks to process a only subset of visual input at a given time has a long history [6, 1, 9]. While there have been recent attempts to induce Hebbian learning in deep networks [13], they have not had much success in a broader context.

One of the more biologically faithful formulations of attention is that of Mnih et al. 2014 [10]. In this approach, visual classification is formulated as a sequential decision process, wherein a recurrent neural network (RNN) controller is allowed a limited number of 'glimpses' of an input image, after which it is forced to make a classification decision. Each glimpse consists of a limited spatial window of high-resolution processing, with rapidly decaying resolution further from the center. For more details, see Section 3.1.

Another recent approach to the development of biologically plausible neural networks has focused on the study of the so-called *weight transport problem*, seen as the primary reason why backpropagation cannot be the learning algorithm implemented by the brain [16]. The problem observes that in performing gradient descent, backpropagation's reuse of the forward weight matrices implies a symmetry in synaptic connectivity that has not been observed in the brain. However, recent work [8, 12] has shown that replacing the forward weight matrices with fixed random matrices in the backward pass does not necessarily impede learning. This technique was termed *feedback alignment*. Such randomized gradient methods can match the classification performance of standard backpropagation for fully-connected networks trained on the MNIST handwritten digit dataset, and in some cases may even converge faster. It has also been shown that these methods can successfully be applied to deep convolutional networks [7], but not without significant modification. However, upcoming work by the author demonstrates that randomized gradient methods can meet and even surpass backpropagation when the Euclidean norm of the feedback alignment signal is standardized with respect to the true gradient.

In this work, I test the recurrent attention model developed by Mnih et al. [10] and the feedback alignment method developed by Lillicrap et al. [8] on adversarial inputs. To the best of my knowledge, such techniques have not previously been applied in the context of security.

3 Methods

3.1 Recurrent Attention

Unlike most deep learning approaches to visual classification, which rely on many-layered convolutional neural networks (CNNs) to process the entire input image at once, this approach uses an RNN to classify each image over a series of time steps, only observing a subset, or 'glimpse,' of the visual field at a time. In this way, classification is formulated as a multi-step decision process, wherein the network must learn how best to deploy each glimpse, receiving a reward signal based on its output after a pre-selected number of time steps. Overall, the model can be broken down into three distinct components.

The first component, the glimpse sensor (Figure 1a), extracts a retina-like representation of the input image x_t, centered at location l_{t-1}. The center of the representation is sampled at a high resolution, with a progressively lower resolution applied further from the center. This representation is denoted by $\rho(x_t, l_t)$. The second component, the glimpse network (Figure 1b), is a two-layer feedforward network that takes as input the retinal representation $\rho(x_t, l_{t-1})$ and the location l_{t-1} and produces an embedded glimpse representation g_t. This representation is then fed, along with the previous hidden state h_{t-1}, to the central component of the model, the RNN controller (Figure 1c). At each time step, this controller outputs the next glimpse location l_t and optionally an action a_t. In a

classification setting, a_t is a class label, though it may be used to represent any action in a dynamic environment.

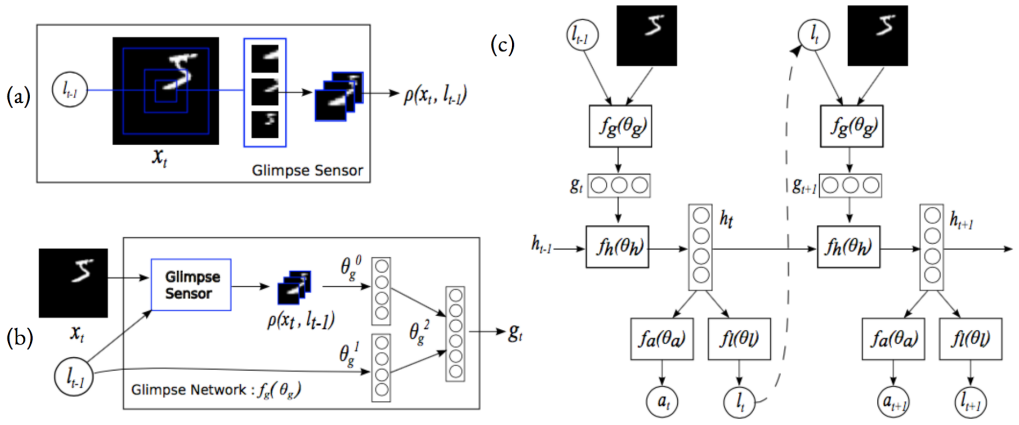

Figure 1: RNN attention model architecture. (a) The glimpse sensor, which extracts a retina representation ρ centered at l_{t-1} on image x_t. (b) The feedforward glimpse network, which combines the retina representation ρ and glimpse location l_{t-1} to produce the glimpse embedding g_t. (c) The RNN controller, which accepts the glimpse g_t and previous hidden state h_{t-1} to produce a classification a_t and the next time step's glimpse location l_t. Figure adapted from [10].

Modeling this as a reinforcement learning problem also results in an unusual training paradigm for classification. The goal of the model is maximize the reward signal $R = \sum_{t=1}^{T} r_t$, where it receives a sparse reward signal r_T of 1 if the image is classified correctly after T glimpses, and 0 otherwise. This constitutes a partially observable Markov decision process (POMDP), because the RNN does not have access to the full input, and it must learn a policy $\pi(u_t|s_{1:t}, \theta)$, where $u_t = (l_t, a_t)$, $s_{1:t}$ represents the states and actions at previous time steps from 1 to t (encoded in this case by the RNN hidden units h_t), and θ are the parameters of the model. The agent, in this case, the RNN, seeks to maximize its expected reward, $J(\theta) = \mathbb{E}[R]$. The model is then trained via gradient ascent on $J(\theta)$, with the approximation of the gradient given by Williams [15] for the REINFORCE algorithm:

$$\nabla_\theta J(\theta) = \sum_{t=1}^{T} \mathbb{E}[\nabla_\theta \log \pi(u_t|s_{1:t}; \theta) R] \approx \frac{1}{M} \sum_{i=1}^{M} \sum_{t=1}^{T} \nabla_\theta \log \pi(u_t^i|s_{1:t}^i; \theta) R^i, \tag{1}$$

where the $s_{1:t}^i$'s are the states obtained after running the current policy π_θ for M episodes. For more model and training details, see Mnih et al. 2014 [10].

3.2 Feedback Alignment

Below is mostly a review of the work presented in [8] and [12]. Assume a simple fully-connected feedforward neural network of the form

$$\begin{aligned} u_1 &= W_1 x_0 + b_1,\ x_1 = f(u_1), \\ u_2 &= W_2 x_1 + b_2,\ x_2 = f(u_2), \\ &\dots \\ u_L &= W_L x_{L-1} + b_L,\ x_L = f(u_L), \end{aligned} \tag{2}$$

where f is a nonlinear activation function. Given a loss $J(y, x_L)$, denote $\frac{\partial J}{\partial u_l}$ by δ_l. In standard backpropagation, the gradient for the lth hidden layer is then

$$\delta_l = (W_{l+1}^T \delta_{l+1}) \odot f'(u_l), \tag{3}$$

where $\odot$ denotes the element-wise Hadamard product. In feedback alignment (FA) [8], however, the gradient is replaced by

$$\delta_l = (B_{l+1}\delta_{l+1}) \odot f'(u_l), \tag{4}$$

where B is a randomized, fixed matrix of the same shape as W^T. The rest of the training algorithm is unchanged.

3.3 Resistance to Gradient-Based Attacks

Goodfellow et al., 2015 [2] argued that despite the nonlinear nature of deep neural networks, the fact that common activation functions such as ReLU or the sigmoid function are either piecewise linear or are tuned to spend most of training in linear regimes makes them susceptible to linear attacks. Given a linear perturbation η of a non-adversarial input, $\tilde{x} = x + \epsilon\eta$, the output of a linear model parametrized by the weight vector w can be written as

$$w^T\tilde{x} = w^T x + \epsilon w^T \eta, \tag{5}$$

where ϵ is a constant smaller than the precision of the stored image format. As they showed, the model output is altered most significantly when $\eta = \text{sign}(w)$, given the max norm constraint $||\eta_\infty|| < \epsilon$. In the case where $w \in \mathbb{R}^{n\times 1}$ and $x \in \mathbb{R}^{n\times 1}$, with the mean $\mu_w = m$, then for an adversarial example with $\eta = \text{sign}(w)$, the difference d in the model output is $\epsilon w^T\eta = \epsilon nm$. They extend this framework to deep neural networks by setting

$$\eta = \epsilon\text{sign}(\nabla_x J(\theta, x, y)). \tag{6}$$

Given a single layer neural network $\mathcal{F}$ parametrized by $\theta = \{W, b\}$, nonlinearity f, and a squared loss function J, we can write the forward pass as

$$u = Wx_0 + b,\ x_1 = f(u), \tag{7}$$

with

$$J(y, \mathcal{F}(x_0)) = \frac{1}{2}(y - x_1)^2, \tag{8}$$

. Through the chain rule, we can see that the gradient with respect to the input x_0 is then

$$\nabla_x J(y, \mathcal{F}(x)) = \frac{\partial J}{\partial x_1}\frac{\partial x_1}{\partial u}\frac{\partial u}{\partial x_0} = \delta W^T \tag{9}$$

However, when applying feedback alignment, Equation 9 changes to

$$\nabla_x J(y, \mathcal{F}(x)) = \delta B \tag{10}$$

When attempting to disrupt network performance using FGSM, then,

$$\eta_{FA} = \epsilon\text{sign}(\delta B) \neq \epsilon\text{sign}(\delta W^T) = \eta_{BP}, \tag{11}$$

and therefore

$$w^T\tilde{x}_{FA} = w^T x + w^T\eta_{FA} < w^T x + w^T\eta_{BP} = w^T\tilde{x}_{BP}, \tag{12}$$

and the effectiveness of the attack is diminished.

Specifically, consider again a simple linear model with output determined by $w^T x$, where $w \in \mathbb{R}^{n\times 1}$ and $x \in \mathbb{R}^{n\times 1}$, with the mean $\mu_w = m$. Then in feedback alignment, $\eta_{FA} = \text{sign}(B)$. Assume that B is drawn from the same distribution as w, such that the mean $\mu_B = m$ as well. However, for $1 \leq i \leq n$, $\text{sign}(B_i) = \text{sign}(w_i)$ with probability $\frac{1}{2}$, and therefore the difference d in the ith element of the perturbation $\epsilon w^T\eta_{FA}$ is

$$d_i = \begin{cases} w_i & \text{with probability } \frac{1}{2} \\ -w_i & \text{with probability } \frac{1}{2} \end{cases} \tag{13}$$

Therefore, the expected total perturbation for feedback alignment is

$$\mathbb{E}[d_{FA}] = \mathbb{E}\left[\sum_{i=1}^{n} d_i\right] = \sum_{i=1}^{n} \mathbb{E}[d_i] = 0, \tag{14}$$

and the effect of the attack is completely nullified.

4 Experiments

4.1 Implementation Details

The fast gradient sign method (FGSM) was applied as both a white box and black box attack to both the RNN attention model and several CNNs trained using feedback alignment. For the attention model, a single layer long short term memory (LSTM) [3] network with 256 hidden units was constructed as the RNN controller, and trained with stochastic gradient descent and a momentum of 0.9. The glimpse network used layers with 128 hidden units. The attention model was trained on the MNIST handwritten digit dataset for 100,000 iterations, a mini-batch size of 32, and with 6 glimpses allowed per image.

Feedback alignment was applied to CNNs trained on both the MNIST dataset and the CIFAR-10 natural image dataset. A two-layer convolutional network with one fully-connected layer was trained on MNIST, with a five-layer network trained on CIFAR-10, for 20,000 and 60,000 iterations, respectively. For more architecture details, see Table 1. Both models were trained using the Adam optimizer [5] and a batch size of 128.

Layer	MNIST	CIFAR10
Input	$28 \times 28 \times 1$	$24 \times 24 \times 3$*
1	5×5 conv. 32 ReLU	5×5 conv. 64 ReLU
2	2×2 max-pool stride 2	2×2 max-pool stride 2
3	5×5 conv. 64 ReLU	5×5 conv. 64 ReLU
4	2×2 max-pool stride 2	2×2 max-pool stride 2
5	1024 dense ReLU	384 Dense ReLU
6	10-way softmax	192 Dense ReLU
7	-	10-way softmax

Table 1: Model architectures. *CIFAR10 images were cropped as part of data augmentation to increase the size of the training set.

Both methods were also tested on brightened and 90°-rotated images from the MNIST dataset as a means of measuring their inherent resistance to common natural image transformations. The results were compared to a standard two-layer CNN. No data augmentation-enhanced training was used, and all other experimental details were identical to those listed above.

4.2 Results

The RNN attention model obtained a test performance of 99.34% accuracy on non-adversarial examples. On the white box attack, it manages to maintain performance despite increasing degrees of perturbation (determined by the settings of the FGSM ϵ parameter), while a standard CNN suffers from considerable and progressive drops in performance (Figure 2a). Performance declines more significantly on the black box attack (Figure 2b), but still remains significantly higher than the traditional CNN.

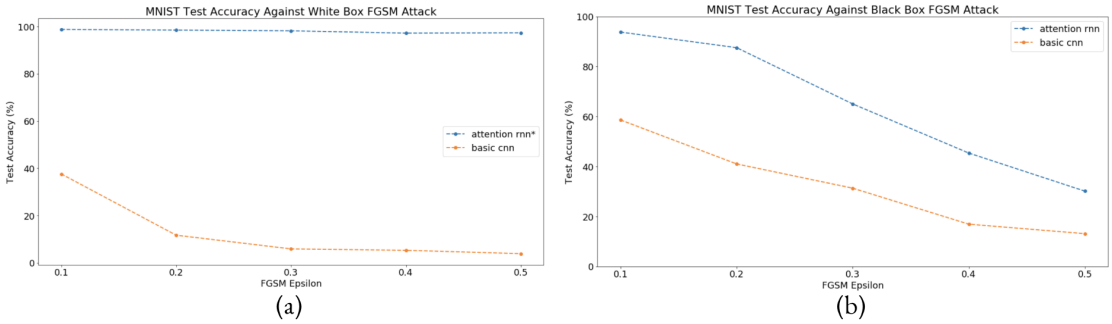

Figure 2: Adversarial results for attention model on white box (a) and black box (b) attacks.

The feedback alignment networks display a similar pattern of resistance to both white box (Figure 3a,b) and black box (Figure 3c,d) adversarial attack. On non-adversarial inputs, the networks achieve test performances of 99.02% accuracy and 82.81% accuracy on MNIST and CIFAR-10, respectively. For each network, white box results remain stable at approximately 7-10% below non-adversarial accuracy, compared to far larger drops for the networks trained with backpropagation (see Section 5 for discussion). Black box results follow the same trend observed in the attention RNN: worse than white box results but still superior to backpropagation test accuracies.

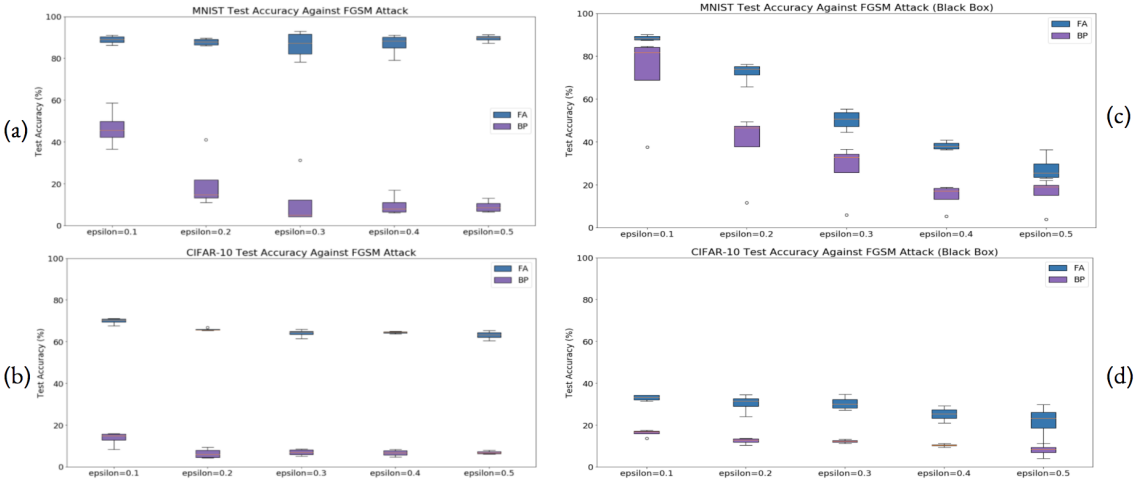

Figure 3: White box (a,b) and black box (c,d) feedback alignment (FA) adversarial results compared to backpropagation (BP) results, averaged over 5 training and testing runs.

The results for the image transformation tests are shown in Table 2. Though the size of the effects are generally smaller than that seen with the adversarial examples, there is still an increase in performance by both biologically-inspired models compared to the standard CNN.

Transformation	BP	FA	Attention RNN
Brightening	10.89	18.79	23.75
90° Rotation	10.48	12.82	13.73

Table 2: Test performance (%) of each method on transformed inputs, averaged over 3 runs.

5 Conclusions

In each case, the biologically-inspired models display greater resilience to both adversarial attack and natural image transformations. In particular, feedback alignment appears to be fundamentally resistant to gradient-based attacks. The reason for the moderate drop in test performance for feedback alignment—as opposed to no difference at all—is likely the visual noise introduced in the input image by the perturbation. There are likely different reasons for why the biologically-inspired models exhibit higher performance on white box as opposed to black box attacks (the opposite of what is observed with standard implementations). For the attention model, it's likely because the standard attack only affects the first glimpse, allowing the rest of the image to be processed without perturbation. For the feedback alignment models, it's likely because white box attacks use that model's gradient to construct the input, which maximizes the model's resistance. Black box attacks lack this obstacle, and therefore are more disruptive. In the future, it would be highly beneficial to test whether the resistive effects of these methods are additive: that is, whether the resistance of a model with multiple biological adaptations is greater than that of a model with one of them. It would also be helpful to test other biologically-inspired models, such as those with explicit memory, as well as a greater number of adversarial attacks and image transformations. More analysis is needed to explain the robustness of these methods, but these results support the idea that biological adaptations increase the resistance of deep neural networks to perturbation.

References

[1] Misha Denil, Loris Bazzani, Hugo Larochelle, and Nando de Freitas. Learning where to attend with deep architectures for image tracking. *CoRR*, abs/1109.3737, 2011.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and Harnessing Adversarial Examples. *ArXiv e-prints*, December 2014.

[3] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[4] James F. Juola, Don G. Bouwhuis, Eric E. Cooper, and C. Bruce Warner. Control of attention around the fovea. *Journal of Experimental Psychology: Human Perception and Performance*, 17(1):125–141, 1991.

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[6] Hugo Larochelle and Geoffrey E Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1243–1251. Curran Associates, Inc., 2010.

[7] Qianli Liao, Joel Z. Leibo, and Tomaso A. Poggio. How important is weight symmetry in backpropagation? *CoRR*, abs/1510.05067, 2015.

[8] Timothy P. Lillicrap, Daniel Cownden, Douglas B. Tweed, and Colin J. Akerman. Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*, 7:13276 EP –, 11 2016.

[9] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. *CoRR*, abs/1406.6247, 2014.

[10] Volodymyr Mnih, Nicolas Heess, Alex Graves, and koray kavukcuoglu. Recurrent models of visual attention. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2204–2212. Curran Associates, Inc., 2014.

[11] Nina Narodytska and Shiva Prasad Kasiviswanathan. Simple black-box adversarial perturbations for deep networks. *CoRR*, abs/1612.06299, 2016.

[12] Arild Nøkland. Direct feedback alignment provides learning in deep neural networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1037–1045. Curran Associates, Inc., 2016.

[13] H. Sebastian Seung and Jonathan Zung. A correlation game for unsupervised learning yields computational interpretations of hebbian excitation, anti-hebbian inhibition, and synapse elimination. *CoRR*, abs/1704.00646, 2017.

[14] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *CoRR*, abs/1312.6199, 2013.

[15] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, May 1992.

[16] D. Zipser and D.E. Rumelhart. pages 192–200. MIT Press, 1990.