
Intriguing properties of neural networks

Christian Szegedy

Google Inc.

Wojciech Zaremba

New York University

Ilya Sutskever

Google Inc.

Joan Bruna

New York University

Dumitru Erhan

Google Inc.

Ian Goodfellow

University of Montreal

Rob Fergus

New York University

Facebook Inc.

Abstract

深度神经网络是一种高度表达的模型，并且在最近在语音和视觉识别任务上取得了最佳性能效果。尽管他们的表现力是他们成功的原因，但也造成他们可能学习到具有反直觉属性的难以解释的解决方法。在本文中，我们报告了两种这样的属性。

首先我们通过一系列的单元分析方法发现在各个高级单元和高级单元的随即线性组合之间没有区别。这说明，其实是空间而非是单个神经元保留了神经网络的高维层次结构的语义信息。

其次，我们发现深度神经网络学习到的输入-输出映射在很大程度上是不连续的。我们可以通过应用某些难以感知的扰动使网络对图像进行错误的分类，这些扰动使通过最大化网络的预测误差发现的。除此之外，这些扰动的特定性质不是学习的随机产物，相同的扰动会导致在数据集的不同子集上训练的不同网络对相同输入进行错误的分类。

1 引言

深度神经网络使功能强大的学习模型，可以在视觉和语音识别问题上表现出出色的性能 [9, 8]。神经网络之所以可以达到高性能是因为它们可以表达任意的计算，这些计算由适量的大量的并行非线性步骤组成。但是由于结果计算是通过监督学习的反向传播机制发现的，因此可能很难解释并且拥有反直觉的特性。

第一个属性与单个单元的语义有关。先前的工作 [6, 13, 7] 通过找到最大程度地激活给定单元的输入集来分析各种单元的语义含义。对单个单元的检查隐含了一个假设，即最后一个特征层的单元形成了一个可区分的基础，这对于提取语义信息特别有用。我们将在第 3 部分证明 $\phi(x)$ 的随机投影和 $\phi(x)$ 的坐标在语义上是不可区分的，这使人们对神经网络解开坐标间变化因素的猜想提出了质疑。一般来说，是整个激活的空间而不是单个的神经元包含了语义信息。Mikolov 等人 [12] 最近对单词的表示得出了一个相似但强的结论，他们发现单词的表征在矢量空间的各个方向上拥有令人惊讶的丰富的语义编码的关系和类比。同时，向量表示在空间旋转之前都是稳定的，因此矢量表示的各个单元不太可能包含语义信息。

第二个属性和神经网络对输入的微扰的稳定性相关。考虑一个性能优异的深度神经网络，它的泛化性能优秀并且在物体识别任务上取得良好性能。我们期望这样的神经网络能在输入由微小扰动的情况下依然具备鲁棒性，因为小的扰动不会图像的对象类别。但是我们发现将不可察觉的非随机扰动应用于测试图像后，就可以任意地干预到网络的预测结果（见图 5）。这些扰动是通过优化输入后最大化预测误差得到的，我们把这些干扰的样本称为“对抗性样本”。

我们很自然地期望最小的必要干扰的精确配置实在反向传播学习中不同的运行过程中出现的正常变异性的随机伪像。然而我们发现对抗性样本是相当鲁棒的，并且在具有不同层数、激活次数或在训练数据的子集上训练的神经网络上都能干扰。也就是说，如果我们使用一个神经网络生成一组对抗样本，则会发现这些样本对于另一个神经网络在统计学的角度上看依然被很难被正确分类，尽管其他的神经网络使用了不同的超参数、不同的样本数据训练。

这些结果表明，通过反向传播学习的神经网络具有非直觉特性和固有盲点，其结构与数据分布相关，只不过这种分布不能被显示地观察到。

2 Framework

定义 我们使用 $x \in \mathbb{R}^m$ 表示输入图像， $\phi(x)$ 表示某层的激活值。我们首先检查 $\phi(x)$ 的属性，然后搜索其盲点。

我们在不同的网络上对三个数据集进行了实验：

- 对于“MNIST”数据集，我们使用了如下的配置：[11]
 - 具有一个或多个隐藏层和 softmax 的分类器的简单的全连接网络。我们称其为“FC”。
 - 一个自动编码器上训练的分类器，我们称其为“AE”。
- ImageNet 数据集 [3].
 - 使用了 Krizhevsky[9] 的架构。我们称其为“AlexNet”。
- $\sim 10M$ 的 Youtube 图片样本（见 [10]）

- 无监督的训练网络，大约有 10 亿可学习参数。我们称其为“QuocNet”。

对于“MNIST”数据集上的实验，我们使用了权重衰减为 λ 的正则化方法。此外，在一些实验中，我们将‘MNIST’上训练的数据集分为两个子集 P_1 和 P_2 ，每一个子集都有 30,000 个样本。

3 $\phi(x)$ 的单元

传统的计算机视觉系统依赖于特征提取：通常单个特征很容易解释，比如说颜色的直方图或量化的局部导数。这允许研究者去检查特征空间中的每一个体的坐标，并将它们与语义丰富的原始图像进行连接。先前的工作 [6, 13, 7, 4] 在分析应用神经网络解决计算机视觉问题时使用了相似的机理。这些工作将一个激活的隐藏层单元作为一个有意义的特征，他们寻找一种一种使该单一特征激活值最大化的输入图形。

前面提及的这种技术可以正式表述为图像 x' 的视觉检查，这些图像满足（或接近最大可到达值）：

$$x' = \arg \max_{x \in \mathcal{I}} \langle \phi(x), e_i \rangle$$

其中， \mathcal{I} 为未接受网络训练的数据分布的图像集， e_i 为第 i 个隐藏单元关联的自然基向量。我们的实验表明，任何随机方向 $v \in \mathbb{R}^n$ 都会产生相似的可解释性的语义属性。更正式地说，我们发现图像 x' 在语义上彼此相关，对于许多 x' ：

$$x' = \arg \max_{x \in \mathcal{I}} \langle \phi(x), v \rangle$$

这说明对于检查 $\phi(x)$ 的属性而言，自然基向量并不会比随机的基向量效果更好。这使人们对神经网络解开坐标间变化因素的猜想提出了质疑。

首先，我们在“MNIST”数据集上训练卷积神经网络以评估我们上述的想法。我们使用 MNIST 的测试集作为 \mathcal{I} 。图1展示了自然基础上的最大化的激活图像，图2展示了在随机方向上的最大化激活图像。在这两种情况下，两种图像都有许多高层的相似性。

接着我们在“AlexNet”上重复了我们的实验，我们使用了验证集作为 \mathcal{I} 。图3和图4对比了训练网络上的随机和自然基向量。对于单个单元以及单元的组合，每行所展示的语义似乎都是有意义的。

尽管这种分析提供了关于 ϕ 在输入分布的特定子集上生成不变性的能力的见解，但它并未解释其其余域的行为。在下一节中，我们将看到 ϕ 在几乎每个点形式的数据分布附近都具有违反直觉的特性。

4 神经网络的盲点

到目前为止，除了确认有关由深度神经网络学习的表示的复杂性的某些直觉之外，单元级检查方法的实用性相对较小 [6, 13, 7, 4]。全局的网络级别检查方法在解释模型做出的分类决策时可能很有用 [1]，并且可以用于例如识别导致对给定视觉输入实例进行正

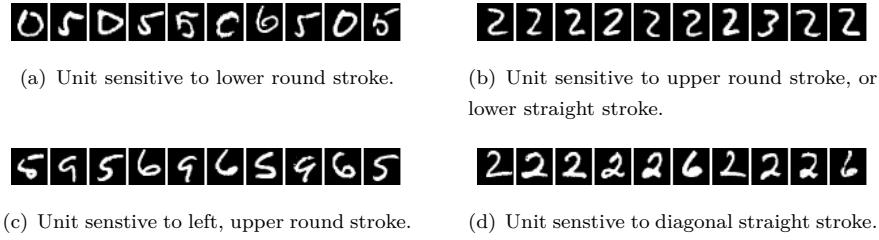


图 1: An MNIST experiment. The figure shows images that maximize the activation of various units (maximum stimulation in the natural basis direction). Images within each row share semantic properties.

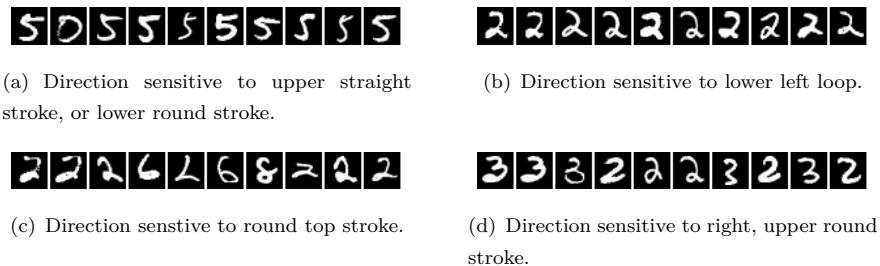


图 2: An MNIST experiment. The figure shows images that maximize the activations in a random direction (maximum stimulation in a random basis). Images within each row share semantic properties.

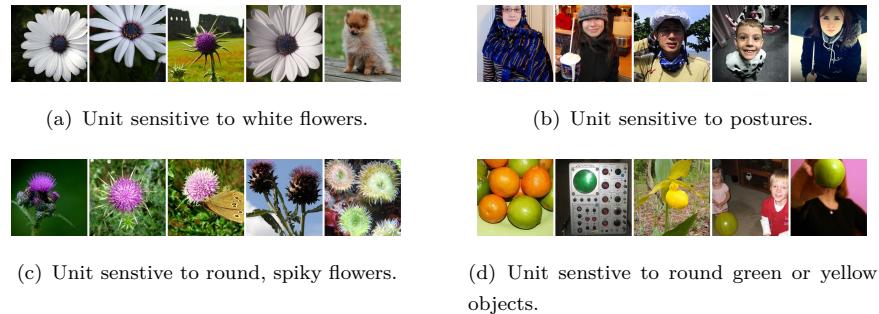


图 3: Experiment performed on ImageNet. Images stimulating single unit most (maximum stimulation in natural basis direction). Images within each row share many semantic properties.

确分类的输入部分。换句话说，可以使用经过训练的模型进行弱监督定位。这样的全局分析很有用，因为它们可以使我们更好地理解受过训练的网络所代表的输入到输出映射。

一般来说，神经网络的输出层单元是其输入的高度非线性函数。当使用交叉熵损失对其进行训练时（用 Softmax 激活函数），它表示给定输入（以及到目前为止提供的训练集）的标签的条件分布。有人认为，在神经网络的输入和输出单元之间的非线性层的深层堆栈是模型在输入空间上编码非局部泛化先验的一种方法 [2]。换句话说，假设输出单元可以为输入空间的各个区域分配非重要（可能是非 ε ）概率，而未训练的样本就在

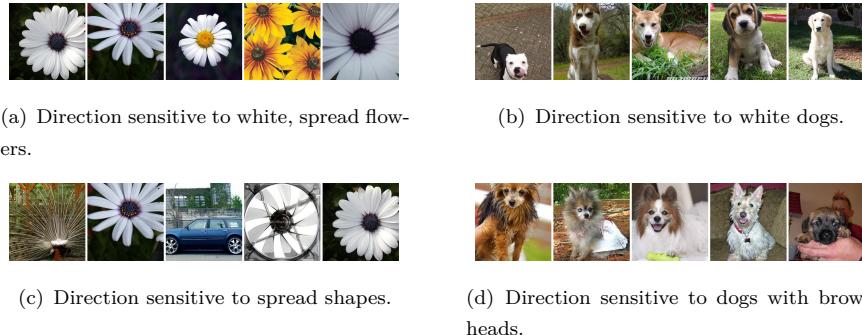


图 4: Experiment performed on ImageNet. Images giving rise to maximum activations in a random direction (maximum stimulation in a random basis). Images within each row share many semantic properties.

这个邻域中。例如，这些区域是可以从不同的角度对相同的对象及逆行表示，这些对象在像素空间中相对较远，但是以他们共享原始输入的标签和统计特性。

如上的想法意味着局部的泛化在训练过程中可以如期地工作。尤其是在给定训练输入 x 时添加的一个足够小的半径 $\varepsilon > 0$ ，满足 $\|r\| < \varepsilon$ 的样本 $x + r$ 将会被模型输出到一个较高的正确类别的概率。这种平滑的先验通常对计算机视觉的问题有效，因为通常来说，给定图像细微的扰动不会改变其基础的类别。

我们的主要结果是，对于深度的神经网络，许多核方法在平滑假设阶段就不成立了。具体而言，我们通过使用简单的优化方法找到了对抗样本，这些样本时通过对正确分类的输入图像进行不明显的微小扰动获得的，所以将不再被正确分类。

从某种意义上说，我们所描述的是一种以有效方式（通过优化）遍历网络所表示的流形并在输入空间中找到对抗样本的方法。对抗样本代表的就是流形中的低概率（但是高维的）的“口袋”，他们很难仅通过围绕给定的样本随机地对输入进行采样得到。目前，各种最新的计算机视觉模型都在训练过程中采用了输入变换，以提高模型的鲁棒性和收敛速度 [9, 13]。但是对于给定的样本，这些变换在统计上效率低下：因为他们高度相关，并且在整个模型训练过程中都是从相同的分布中得出的。我们提出了一种使该过程具有自适应性的方案，该方案利用了模型及其在训练数据的局部空间建模中的缺陷。

我们在本质上很接近难负样本挖掘，这与它密切相关：在计算机视觉中，难负样本挖掘包括识别训练集示例（或其中的一部分），这些示例被模型赋予了较低的概率，但是应该相反，为高概率 [5]。然后更改训练集分布，以强调这种难负样本，并执行下一轮模型训练。如将要描述的那样，这项工作中提出的最优化问题也可以以建设性的方式使用，类似于难负样本挖掘原理。



(a)

(b)

图 5: Adversarial examples generated for AlexNet [9].(Left) is a correctly predicted sample, (center) difference between correct image, and image predicted incorrectly magnified by 10x (values shifted by 128 and clamped), (right) adversarial example. All images in the right column are predicted to be an “ostrich, *Struthio camelus*”. Average distortion based on 64 examples is 0.006508. Please refer to <http://goo.gl/huaGPb> for full resolution images. The examples are strictly randomly chosen. There is not any postselection involved.

4.1 规范表述

我们用 $f : \mathbb{R}^M \rightarrow \{1 \dots k\}$ 表示一个将图像像素值向量到离散标签集的分类器。我们假设 f 有一个相关的连续损失函数 $loss_f : \mathbb{R}^m \times \{1 \dots k\} \rightarrow \mathbb{R}^+$ 。对于一个给定的 $x \in \mathbb{R}^m$ 的图像和标签 $l \in \{1 \dots k\}$, 我们旨在解决如下盒约束的优化问题:

- 最小化 $\|r\|_2$, 使得

1. $f(x + r) = l$
2. $x + r \in [0, 1]^m$

最小化器 r 可能不是唯一的, 但是我们将其定义为从 $D(x, l)$ 随机抽取的一个并将其用于表示 $x + r$ 。不正式地说, $x + r$ 就是一张能被 f 分类为 l 的最接近的图片。显然, $D(x, f(x)) = f(x)$, 因此这个任务只有当 $f(x) \neq l$ 的时候才有意义。通常来说, $D(x, l)$ 的精准计算是一个困难的问题, 因此我们使用盒约束的 L-BFGS 方法进行近似。具体来说就是, 我们通过线性搜索找到一个最小的 $c > 0$ 来找到 $D(x, l)$ 的近似。对于这个最小值, 下面问题的 r 需要满足 $f(x + r) = l$:

- 最小化 $c|r| + loss_f(x + r, l)$ 同时满足约束 $x + r \in [0, 1]^m$

在凸损失的情况下, 这种惩罚函数方法将给出 $D(x, l)$ 的精确解, 但是神经网络通常是非凸的, 因此这种情况下我们一般只能得到一个近似值。



图 6: Adversarial examples for QuocNet [10]. A binary car classifier was trained on top of the last layer features without fine-tuning. The randomly chosen examples on the left are recognized correctly as cars, while the images in the middle are not recognized. The rightmost column is the magnified absolute value of the difference between the two images.

4.2 实验结果

我们的最小失真函数 D 具有如下有趣的属性，我们将在本节通过非正式的证据和定量实验支撑我们的观点：

1. 对于我们研究的所有网络 ((MNIST, QuocNet [10], AlexNet [9]), 对每个样本，我们都能生成非常接近、在视觉上难以区分的对抗样本，这些样本被上述网络错误分类。(见图 5 和 <http://goo.gl/huaGPb>).
2. 对抗样本具有跨模型的泛化能力：在 A 模型上产生的对抗样本，有很大一部分在 B 模型（和 A 模型结构相同，超参数不同）上也有效（也能是 B 模型错误分类）；
3. 跨训练数据的泛化能力：相当大的一部分样本就会被网络错误地输出分类，这些网络都是从给定训练集的子集中训练得到的。

上述观察结果表明，对抗样本在某种程度上是普遍的，而不仅仅是过度拟合特定模型或特定选择训练集的结果。这些结果还表明，将对抗样本用于训练可能会提高结果模型的通用性。我们的初步实验也为 MNIST 提供了积极的证据来支持这一假设：我们通过保留一组对抗样本作为随机子集，成功地训练了两层 100-100-10 非卷积神经网络，其测试误差低于 1.2% 其中不断被新生成的对抗样本所取代，并始终与原始训练集中混合。我们使用了权重衰减，但该网络没有 dropout。为了进行比较，如果仅通过权重衰减对其进行调整，则该大小的网络的误差为 1.6%，并且可以通过使用精心应用的 dropout 将其提高到 1.3% 左右。一个微妙但必不可少的细节是，我们仅通过为每层输出生成对抗样本来进行改进，这些示例用于训练以上所有层。该网络以交替的方式进行了训练，除了原始训练集之外，还分别维护和更新了每一层的对抗样本库。根据我们的初步观察，高层的对抗样本似乎比输入层或较低层的对抗样本有用得多。在未来的工作中，我们计划系统地比较这些影响。

出于空间考虑，我们只介绍我们执行的‘MNIST’实验的代表性子集（参见表 1）的结果，此处显示的结果与各种非卷积模型的结果一致。对于 MNIST，我们尚无卷积模型的结果，但我们与‘AlexNet’进行的首次定性实验使我们有理由相信卷积网络的行为也可能

Model Name	Description	Training error	Test error	Av. min. distortion
FC10(10^{-4})	Softmax with $\lambda = 10^{-4}$	6.7%	7.4%	0.062
FC10(10^{-2})	Softmax with $\lambda = 10^{-2}$	10%	9.4%	0.1
FC10(1)	Softmax with $\lambda = 1$	21.2%	20%	0.14
FC100-100-10	Sigmoid network $\lambda = 10^{-5}, 10^{-5}, 10^{-6}$	0%	1.64%	0.058
FC200-200-10	Sigmoid network $\lambda = 10^{-5}, 10^{-5}, 10^{-6}$	0%	1.54%	0.065
AE400-10	Autoencoder with Softmax $\lambda = 10^{-6}$	0.57%	1.9%	0.086

表 1: Tests of the generalization of adversarial instances on MNIST.

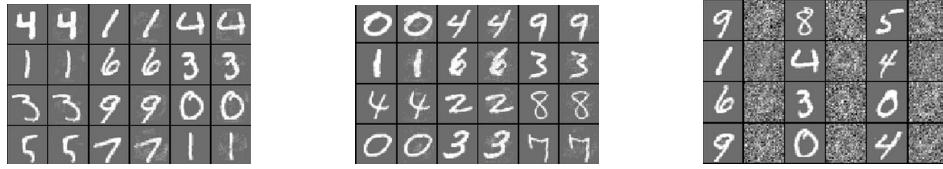
相似。我们的每个模型都经过 L-BFGS 训练，直到收敛为止。前三个模型是线性的分类器，他们工作在具有各种权重衰减参数 λ 上。

我们证明了深度神经网络在单个单元的语义含义和不连续性方面都具有违反直觉的属性。对抗性负面因素的存在似乎与该网络实现高泛化性能的能力相矛盾。的确，如果网络能够很好地推广，那么如何将这些对抗性的否定与常规的例子区分开来呢？可能的解释是，对抗性否定集的可能性极低，因此从未（或很少）在测试集中观察到，但是它很密集（很像有理数），因此几乎在每个测试用例中都可以找到它。但是，我们对对抗性否定出现的频率没有深刻的了解，因此应该在以后的研究中解决这个问题。我们所有的样本都使用了二次权重衰减：即 $loss_{decay} = \lambda \sum w_i^2 / k$ 将被添加到总的损失中，其中 k 是每层的单元数量。我们的三个模型都是简单的线性 (softmax) 的分类器，没有任何的隐藏单元 (FC10(λ))。其中的一个 FC10(1) 使用了很高的衰减，即 $\lambda = 1$ 的值训练以用于测试是否仍可能在极端的训练条件下产生对抗样本。另外两个模型是简单的 sigmoid 神经网络，由两个隐藏层和一个分类器组成。最后一个模型，AE400 – 10 由单层的稀疏编码器和 sigmoid 激活函数以及 400 个节点的 Softmax 分类器组成。这个网络已经被训练好，在第一层已经具备了高质量的过滤器，并且这层并没有微调。最后一列测量了训练集上达到 0% 准确率的最小平均像素失真情况。失真的测量使用了 $\sqrt{\frac{\sum(x'_i - x_i)^2}{n}}$ 以衡量原始的 x 和失真的 x' 程度，其中 $n = 784$ 代表的是图像像素的大小。像素值被缩放到了 $[0, 1]$ 区间。

在我们的第一个实验中，我们为给定的网络生成了一组对抗样本，并为每个其他网络提供了这些实例，以衡量误分类实例的比例。最后一列显示在整个训练集上达到 0% 准确度所需的平均最小失真。实验结果显示在表2中。图 7 展示了该实验中使用的两个网络生成的对抗样本的可视化。总的结论是，即使对于使用不同超参数训练的模型，对抗样本也趋于困难。尽管基于自动编码器的版本似乎可以克服对抗样本的攻击，但也不能完全免疫。值得注意的是，对于除其中一个模型以外的所有模型，即使标准差为 0.1 的噪声也大于我们对抗性噪声的标准差。

尽管如此，实验仍有问题没有解决，这些问题来自训练集。生成的样本的难度是否仅取决于我们从训练集中特定选择的样本，还是这种效果甚至可以泛化到在完全不同的训练集上训练的模型？

为了研究交叉训练集的泛化性能，我们将 60000 个 MNIST 训练图像划分为大小分别为 30000 的两个部分 P_1 和 P_2 ，并训练了三个具有 Sigmoid 型激活的非卷积网络：两



(a) Even columns: adversarial examples for a linear (FC) classifier (std-dev=0.06)

(b) Even columns: adversarial examples for a 200-200-10 sigmoid network (std-dev=0.063)

(c) Randomly distorted samples by Gaussian noise with stddev=1. Accuracy: 51%.

图 7: Adversarial examples for a randomly chosen subset of MNIST compared with randomly distorted examples. Odd columns correspond to original images, and even columns correspond to distorted counterparts. The adversarial examples generated for the specific model have accuracy 0% for the respective model. Note that while the randomly distorted examples are hardly readable, still they are classified correctly in half of the cases, while the adversarial examples are never classified correctly.

	FC10(10^{-4})	FC10(10^{-2})	FC10(1)	FC100-100-10	FC200-200-10	AE400-10	Av. distortion
FC10(10^{-4})	100%	11.7%	22.7%	2%	3.9%	2.7%	0.062
FC10(10^{-2})	87.1%	100%	35.2%	35.9%	27.3%	9.8%	0.1
FC10(1)	71.9%	76.2%	100%	48.1%	47%	34.4%	0.14
FC100-100-10	28.9%	13.7%	21.1%	100%	6.6%	2%	0.058
FC200-200-10	38.2%	14%	23.8%	20.3%	100%	2.7%	0.065
AE400-10	23.4%	16%	24.8%	9.4%	6.6%	100%	0.086
Gaussian noise, stddev=0.1	5.0%	10.1%	18.3%	0%	0%	0.8%	0.1
Gaussian noise, stddev=0.3	15.6%	11.3%	22.7%	5%	4.3%	3.1%	0.3

表 2: Cross-model generalization of adversarial examples. The columns of the Tables show the error induced by distorted examples fed to the given model. The last column shows average distortion wrt. original training set.

Model	Error on P_1	Error on P_2	Error on Test	Min Av. Distortion
FC100-100-10: 100-100-10 trained on P_1	0%	2.4%	2%	0.062
FC123-456-10: 123-456-10 trained on P_1	0%	2.5%	2.1%	0.059
FC100-100-10' trained on P_2	2.3%	0%	2.1%	0.058

表 3: Models trained to study cross-training-set generalization of the generated adversarial examples. Errors presented in Table correspond to original not-distorted data, to provide a baseline.

个网络 FC100-100-10 和 FC123-456-10 在 P_1 上训练以及一个网络 FC100-100-10 在 P_2 上训练。我们为 P_1 训练两个网络的原因是要研究同时更改超参数和训练集的累积效果。FC100-100-10 和 FC100-100-10 共享相同的超参数：它们都是 100-100-10 网络，而 FC123-456-10 具有不同数量的隐藏单元。在这个实验中，失真的对象是测试集的样本而非训练集的样本。表3总结了有关这些模型的基本情况。在为测试集生成具有 100% 错误率且失真最小的对抗样本之后，我们将这些样本输入给每个模型。表4上部的相应栏中显示每个模型对应的错误率。最后的实验中我们使用了失真程度为 $x + 0.1 \frac{x'-x}{\|x'-x\|_2}$ 的样本而不是单纯的 x' 。这种变化使得是真成都提高了 40%，标准差从 0.06 扩大到

	FC100-100-10	FC123-456-10	FC100-100-10'
Distorted for FC100-100-10 (av. stddev=0.062)	100%	26.2%	5.9%
Distorted for FC123-456-10 (av. stddev=0.059)	6.25%	100%	5.1%
Distorted for FC100-100-10' (av. stddev=0.058)	8.2%	8.2%	100%
Gaussian noise with stddev=0.06	2.2%	2.6%	2.4%
Distorted for FC100-100-10 amplified to stddev=0.1	100%	98%	43%
Distorted for FC123-456-10 amplified to stddev=0.1	96%	100%	22%
Distorted for FC100-100-10' amplified to stddev=0.1	27%	50%	100%
Gaussian noise with stddev=0.1	2.6%	2.8%	2.7%

表 4: Cross-training-set generalization error rate for the set of adversarial examples generated for different models. The error induced by a random distortion to the same examples is displayed in the last row.

0.1。这种失真的样本将反馈到每个模型，对应的错误率也显示在表4的下部。有趣的结论是，即便在不相交的训练集上训练的模型，对抗样本依然很难找到，但他们的有效性也大大降低。

4.3 不稳定性分析

前一节中我们展示了纯监督训练得到的深度神经网络的样例在某些微小的扰动下是不稳定的。对抗样本展示了存在一种微小的加性(欧几里得意义上的)输入干扰可以在网络的最后一层的输出中产生较大的扰动，这种特性独立于训练集和网络。本节将描述一种简单的方法，通过测量每一矫正后的层的测量以控制网络的额外稳定性。(注：不知道这个矫正是怎么矫正的，文中也没提及)

从数学的角度来讲，如果 $\phi(x)$ 表示拥有训练参数 W 的对输入为 x 时的 K 层网络的输出，我们将其写为：

$$\phi(x) = \phi_K(\phi_{K-1}(\dots \phi_1(x; W_1); W_2) \dots; W_K)$$

其中， ϕ_k 表示了第 $k-1$ 层到第 k 层的映射。 $\phi(x)$ 的稳定性可以通过检查每一层 $k = 1 \dots K$ [Lipschitz 常数上半部分进行表征，定义这个常数 $L_k > 0$ 为：

$$\forall x, r, \|\phi_k(x; W_k) - \phi_k(x + r; W_k)\| \leq L_k \|r\|$$

这样训练得到的网络将因此满足 $\|\phi(x) - \phi(x + r)\| \leq L \|r\|$ ，其中 $L = \prod_{k=1}^K L_k$ 。

一个半矫正后的层(无论是卷积层还是全连接层)都通过映射 $\phi_k(x; W_k, b_k) = \max(0, W_k x + b_k)$ 定义。令 $\|W\|$ 表示 W 的算子范数(比如说它的极大值)。由于非线性函数 $\rho(x) = \max(0, x)$ 是有界的，比如说它对于所有的 x, r 都满足 $\|\rho(x) - \rho(x + r)\| \leq |r|$ 。同时也遵循：

$$\|\phi_k(x; W_k) - \phi_k(x + r; W_k)\| = \|\max(0, W_k x + b_k) - \max(0, W_k(x + r) + b_k)\| \leq \|W_k r\| \leq \|W_k\| \|r\|$$

因此满足 $L_k \leq \|W_k\|$ 。另一方面，最大池化层的 ϕ_k 也是有界的：

$$\forall x, r, \|\phi_k(x) - \phi_k(x + r)\| \leq \|r\|$$

由于其雅各比矩阵就是输入坐标子集的一个投影，并因此不会扩张其梯度。最后，如果 ϕ_k 是一个对比归一化层

$$\phi_k(x) \frac{x}{(\varepsilon + \|x\|^2)^\gamma}$$

那么可以证明，对于任意的 $\gamma \in [0.5, 1]$ 均有：

$$\forall x, r, \|\phi_k(x) - \phi_k(x + r)\| \leq \varepsilon^{-\gamma} \|r\|$$

其中 γ 的取值可以对应到大部分的网络层处理方式。

下面我们就可以通过简单的计算每一全连接层和卷积层的算子范数得到一个测量网络不稳定性的保守测量。全连接层的计算很简单，因为其范数通过全连接矩阵的极大值直接给出，我们主要探究卷积层的情况。假设 W 是一个普通的 4 维张量，输入特征数为 C ，输出特征数为 D ，支持 $N \times N$ 大小以及空间步长为 Δ 的运算，有：

$$W_x = \left\{ \sum_{c=1}^C x_c * w_{c,d}(n_1\Delta, n_2\Delta); d = 1 \dots, D \right\}$$

其中， x_c 表示第 c 个输入特征图像， $w_{c,d}$ 表示对应输入特征为 c 以及输出特征为 d 的空间卷积核，使用帕斯瓦尔公式我们可以得到他的操作算子为：

$$\|W\| = \sup_{\xi \in [0, N\Delta^{-1})^2} \|A(\xi)\| \quad (1)$$

其中， $A(\xi)$ 是一个 $D \times (C \cdot \Delta^2)$ 的矩阵，它的行表示为：

$$\forall d = 1 \dots D, A(\xi)_d = (\Delta^{-2} \widehat{w}_{c,d}(\xi + l \cdot N \cdot \Delta^{-1}); c = 1 \dots C, l = (0 \dots \Delta - 1)^2)$$

其中， $\widehat{w}_{c,d}$ 为 $w_{c,d}$ 二维傅里叶变换：

$$\widehat{w}_{c,d}(\xi) = \sum_{u \in [0, N)^2} w_{c,d}(u) e^{-2\pi i (u \cdot \xi) / N^2}$$

表 5 展示了从“ImageNet”训练的深度神经网络（即“AlexNet”）使用式计算的 Lipschitz 常数的上边界。它表明不稳定性可以在很快得再第一层卷积层就出现了。

这些结果与上一节中构造的盲点的出现是一致的，但是它们并没有试图解释为什么这些对抗样本会针对不同的超参数或训练集有泛化效果。我们强调我们计算上限：大的界限并不会是对抗性示例存在的原因；但是，小的边界范围可保证不会出现此类对抗性样本。这表明可以对参数进行简单的正则化，包括对每个利普希茨上界进行惩罚，这可能有助于改善网络的泛化误差。

Table 5 shows the upper Lipschitz bounds computed from the ImageNet deep convolutional network of [9], using (1). It shows that instabilities can appear as soon as in the first convolutional layer.

These results are consistent with the existence of blind spots constructed in the previous section, but they don't attempt to explain why these examples generalize across different

Layer	Size	Stride	Upper bound
Conv. 1	$3 \times 11 \times 11 \times 96$	4	2.75
Conv. 2	$96 \times 5 \times 5 \times 256$	1	10
Conv. 3	$256 \times 3 \times 3 \times 384$	1	7
Conv. 4	$384 \times 3 \times 3 \times 384$	1	7.5
Conv. 5	$384 \times 3 \times 3 \times 256$	1	11
FC. 1	9216×4096	N/A	3.12
FC. 2	4096×4096	N/A	4
FC. 3	4096×1000	N/A	4

表 5: Frame Bounds of each rectified layer of the network from [9].

hyperparameters or training sets. We emphasize that we compute upper bounds: large bounds do not automatically translate into existence of adversarial examples; however, small bounds guarantee that no such examples can appear. This suggests a simple regularization of the parameters, consisting in penalizing each upper Lipschitz bound, which might help improve the generalisation error of the networks.

5 讨论

我们证明了深度神经网络在单个单元的语义含义和不连续性方面都具有违反直觉的属性。对抗性负面因素的存在似乎与该网络实现高泛化性能的能力相矛盾。的确，如果网络能够很好地推广，那么如何将这些对抗性的否定与常规的例子区分开来呢？可能的解释是，对抗性否定集的可能性极低，因此从未（或很少）在测试集中观察到，但是它很密集（很像有理数），因此几乎在每个测试用例中都可以找到它。但是，我们对对抗性否定出现的频率没有深刻的了解，因此应该在以后的研究中解决这个问题。

参考文献

- [1] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *The Journal of Machine Learning Research*, 99:1803–1831, 2010.
- [2] Yoshua Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.
- [3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.
- [4] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. Technical Report 1341, University of Montreal, June 2009. Also presented at the ICML 2009 Workshop on Learning Feature Hierarchies, Montréal, Canada.
- [5] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

- [6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv preprint arXiv:1311.2524*, 2013.
- [7] Ian Goodfellow, Quoc Le, Andrew Saxe, Honglak Lee, and Andrew Y Ng. Measuring invariances in deep networks. *Advances in neural information processing systems*, 22:646–654, 2009.
- [8] Geoffrey E. Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process. Mag.*, 29(6):82–97, 2012.
- [9] Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1106–1114, 2012.
- [10] Quoc V Le, Marc’Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg S Corrado, Jeff Dean, and Andrew Y Ng. Building high-level features using large scale unsupervised learning. *arXiv preprint arXiv:1112.6209*, 2011.
- [11] Yann LeCun and Corinna Cortes. The mnist database of handwritten digits, 1998.
- [12] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [13] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional neural networks. *arXiv preprint arXiv:1311.2901*, 2013.