# 数据扩充|当数据有限时如何使用深度学习 - 第2部分

这篇文章是一个全面的回顾，数据增强技术为深入学习，特别是图像。这是在数据有限的情况下如何使用深度学习的第2部分。在这里查看第1部分。

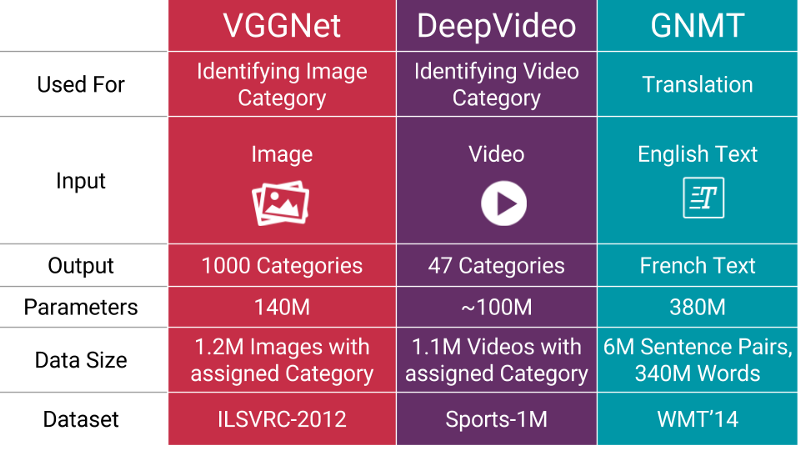


我们都去过那里。你有一个恒星概念，可以用机器学习模型来实现。热情洋溢地打开网页浏览器，搜索相关数据。很有可能，你会发现一个数据集有大约几百个图像。

您还记得，大多数流行的数据集都有数万张（或更多）的图像。您还记得有人提到拥有一个大型数据集对于良好的性能至关重要。感到失望，你想知道，我的“最先进的”神经网络能在我仅有的少量数据下表现良好吗？

答案是，是的！但是，在我们进入使之发生的魔力之前，我们需要思考一些基本的问题。

#### 为什么需要大量数据？



当你训练一个机器学习模型时，你真正要做的是调整它的参数，这样它就可以把一个特定的输入（比如，一个图像）映射到一些输出（一个标签）。我们的优化目标是在我们的模型损失很低的地方寻找最佳点，当您的参数以正确的方式调整时就会发生这种情况。

最先进的神经网络通常有数百万的参数！

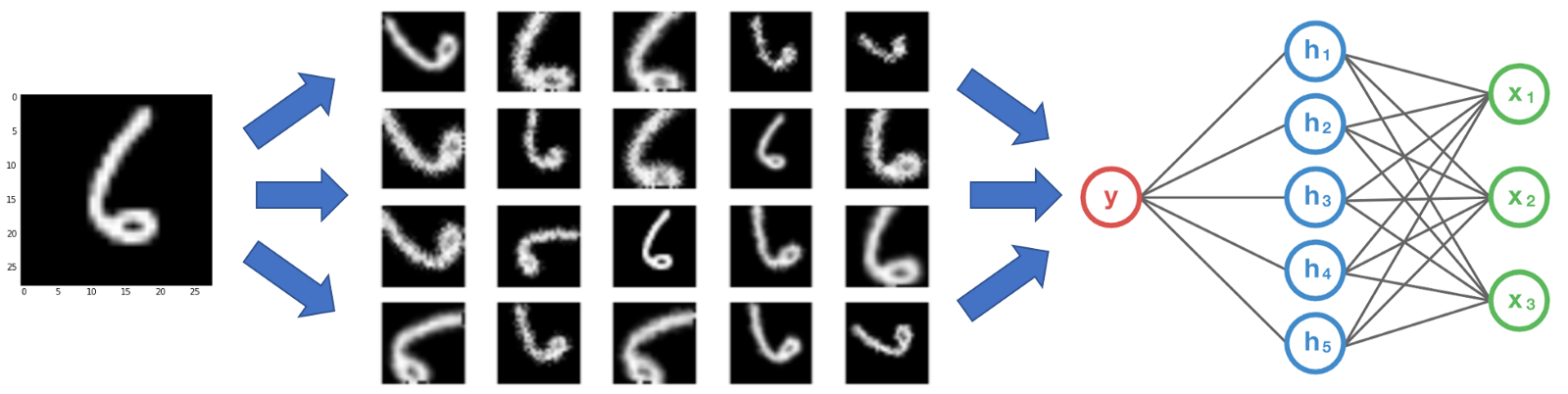
当然，如果你有很多参数，你需要向你的机器学习模型展示一定数量的例子，以获得良好的性能。而且，所需参数的数量与模型必须执行的任务的复杂程度成正比。

### 如果我没有“更多的数据”，如何获得更多的数据？

你不需要寻找新的图像，可以添加到你的数据集。为什么？因为，神经网络一开始并不聪明。例如，训练不好的神经网络会认为下面显示的这三个网球是不同的、独特的图像。



因此，为了获得更多的数据，我们只需要对现有的数据集做一些小的改动。微小的变化，如翻转、平移或旋转。我们的神经网络无论如何都会认为这些是不同的图像。



一种卷积神经网络，即使它被放置在不同的方向上，也能对物体进行强有力的分类，称之为方差。更具体地说，CNN可以对平移、视点、大小或照明（或以上两者的组合）保持不变。

这本质上是数据扩充的前提。在现实世界中，我们可能有一个在有限条件下拍摄的图像的数据集。但是，我们的目标应用可能存在于不同的条件下，例如不同的方向、位置、尺度、亮度等。我们用附加的综合修正数据训练神经网络来解释这些情况。

#### 即使我有很多数据，增强功能也有帮助吗？

对。它有助于增加数据集中相关数据的数量。这与神经网络的学习方式有关。让我举个例子来说明。



假设您有一个数据集，由两个品牌的汽车组成，如上图所示。假设A品牌的所有汽车都与左边的图片完全一致（即所有汽车都面向左边）。同样，B品牌的所有汽车都与右边的图片完全一致（即面朝右边）。现在，你把这个数据集输入到你的“最先进的”神经网络中，你希望一旦它被训练出来，就能得到令人印象深刻的结果。



我们假设它已经完成了训练，并且你提供了上面的图像，这是一辆品牌汽车。但是你的神经网络输出它是B品牌的车！你很困惑。你不是用你的“最先进的”神经网络在数据集上获得了95%的准确率吗？我没有夸张，

为什么会这样？这是因为大多数机器学习算法都是这样工作的。它发现了区分一个类和另一个类的最明显的特性。这里的特点是A品牌的所有车都朝左，B品牌的所有车都朝右。

你的神经网络只和你提供的数据一样好。

我们如何防止这种情况发生？我们必须减少数据集中不相关特征的数量。对于我们上面的汽车模型分类器，一个简单的解决方案是添加两个类的汽车图片，面向原始数据集的另一个方向。更好的是，你可以在现有的数据集中水平地翻转图像，这样它们就面向另一边！现在，在这个新的数据集上训练神经网络，你就能得到你想要的性能。

通过执行增强，可以防止神经网络学习不相关的模式，本质上提高了整体性能。

### 入门

在我们深入研究各种增强技术之前，有一个问题我们必须事先考虑。

#### 我们应该在哪里增加ML管道中的数据？

答案似乎很明显；我们在将数据输入到模型之前进行了扩充，对吧？是的，但你有两个选择。一种选择是预先执行所有必要的转换，实际上是增加数据集的大小。另一个选择是在一个小批量上执行这些转换，就在将其输入到机器学习模型之前。

第一个选项称为离线增强。对于相对大小的数据集，这种方法是首选的，因为您最终会将数据集的大小增加一个等于您执行的转换次数的因子（例如，通过翻转所有图像，我会将数据集的大小增加一个因子2）。

第二种方法称为在线增强（online augmentation），即动态增强（oraugmentation on The fly），这种方法对于较大的数据集是首选的，因为您无法承受数据量的爆炸性增长。相反，您将对要提供给模型的小批量执行转换。一些机器学习框架支持在线增强，这可以在GPU上加速。

### 流行的增强技术

在本节中，我们将介绍一些常用的基本但功能强大的增强技术。在我们探索这些技术之前，为了简单起见，让我们做一个假设。假设是，我们不需要考虑图像边界之外的内容。我们将使用以下技巧，使我们的假设是有效的。

如果我们使用一种迫使我们猜测图像边界之外的东西的技术，会发生什么？在这种情况下，我们需要插入一些信息。我们将在讨论增强类型之后详细讨论这个问题。

对于这些技术中的每一种，我们还指定了增加数据集大小的因素（aka）。数据增强因子）。

#### 一。轻弹

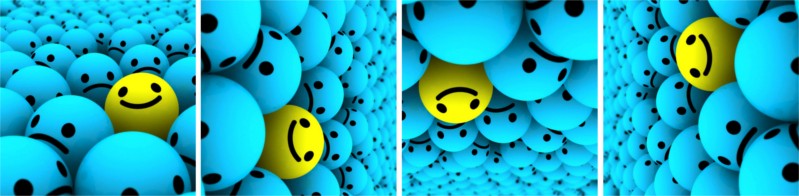
您可以水平和垂直翻转图像。有些框架不提供垂直翻转功能。但是，垂直翻转相当于将图像旋转180度，然后执行水平翻转。下面是翻转图像的示例。



您可以使用您最喜爱的软件包中的以下任意命令执行翻转。数据增强因子=2到4倍

#### 2。旋转

关于此操作，需要注意的一点是，旋转后图像尺寸可能不会保留。如果您的图像是正方形，则以直角旋转它将保留图像大小。如果它是一个矩形，旋转180度可以保持它的大小。以更精细的角度旋转图像也会更改最终图像大小。我们将在下一节讨论如何处理这个问题。下面是以直角旋转的正方形图像的示例。



您可以使用来自您最喜爱的软件包的以下任何命令来执行旋转。数据增强因子=2到4倍

#### 三。比例尺

图像可以向外或向内缩放。向外缩放时，最终图像大小将大于原始图像大小。大多数图像框架从新图像中剪切出一个部分，其大小与原始图像相同。在下一节中，我们将处理向内缩放的问题，因为它减少了图像的大小，迫使我们对超出边界的部分进行假设。下面是正在缩放的示例或图像。



使用scikit映像，可以使用以下命令执行缩放。数据增强因子=任意。

#### 四。作物

与缩放不同，我们只是从原始图像中随机抽取一个部分。然后我们将此部分调整为原始图像大小。这种方法通常被称为随机种植。下面是随机种植的例子。如果你仔细观察，你会发现这个方法和缩放之间的区别。



通过对TensorFlow使用以下任何命令，可以执行随机裁剪。数据增强因子=任意。

#### 5个。翻译

平移只涉及沿X或Y方向（或两者）移动图像。在下面的例子中，我们假设图像有一个超出其边界的黑色背景，并且被适当地转换。这种增强方法非常有用，因为大多数物体几乎可以位于图像的任何位置。这迫使你的卷积神经网络到处寻找。



可以使用以下命令在TensorFlow中执行翻译。数据增强因子=任意。

#### 6。高斯噪声

过度拟合通常发生在你的神经网络试图学习高频特征（频繁出现的模式）时，这些特征可能并不有用。高斯噪声具有零均值，在所有频率上都有数据点，有效地扭曲了高频特征。这也意味着低频成分（通常是你想要的数据）也被扭曲了，但是你的神经网络可以学会忽略它。加入适量的噪声可以提高学习能力。

一个调低版本是盐和胡椒噪声，它呈现为随机的黑白像素在图像中传播。这类似于将高斯噪声添加到图像中所产生的效果，但可能具有较低的信息失真级别。



可以使用TensorFlow上的以下命令将高斯噪声添加到图像中。数据增强因子=2倍。

### 先进的增强技术

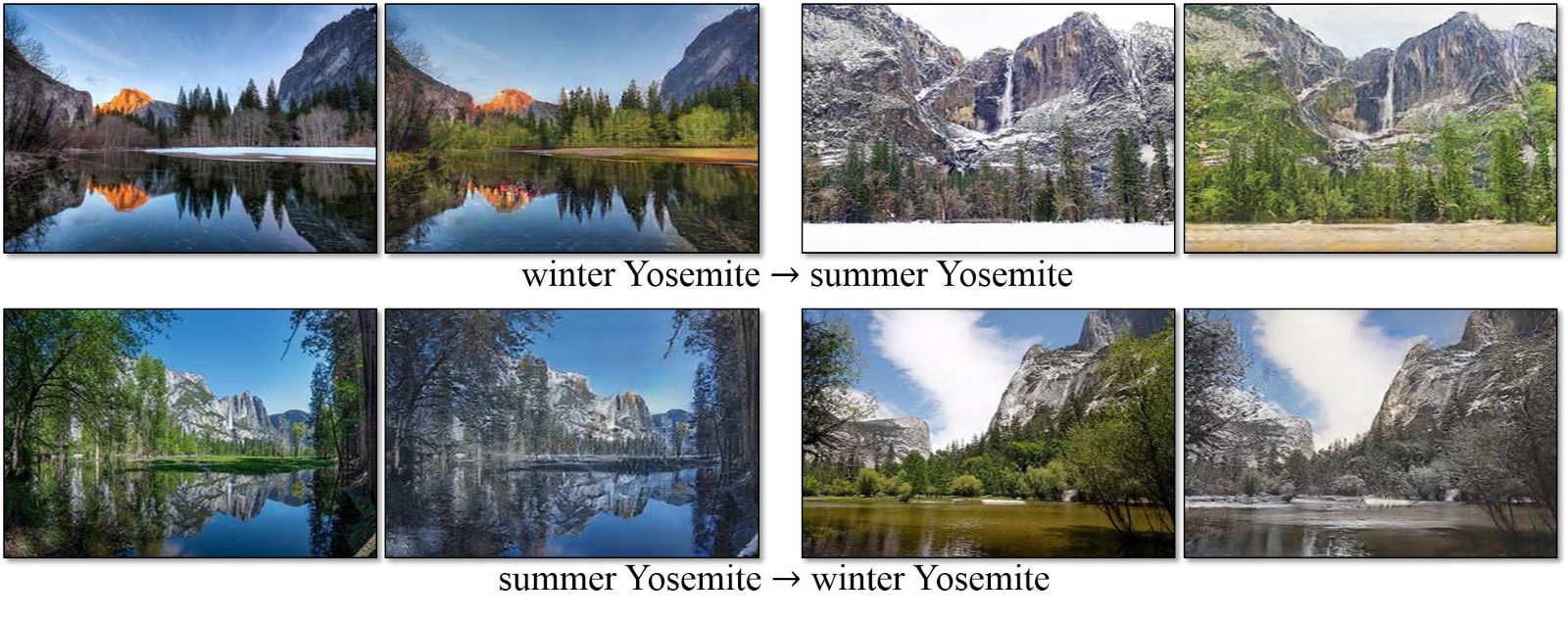
现实世界中，自然数据仍然可以存在于上述各种简单方法无法解释的条件中。例如，让我们来完成识别照片中的风景的任务。风景可以是任何东西：冻原，草原，森林等等。听起来像是一个非常直接的分类任务，对吧？你是对的，除了一件事。我们正在俯瞰照片中的一个关键特征，它将影响拍摄季节的表现。

如果我们的神经网络不理解某些景观可以存在于各种条件下（雪、湿、亮等），它可能会将冰冻湖滨作为冰川或潮湿的田地标记为沼泽。

缓解这种情况的一个方法是添加更多的图片，这样我们就可以考虑到所有的季节变化。但这是一项艰巨的任务。扩展我们的数据增强概念，想象一下人工生成不同季节的效果有多酷？

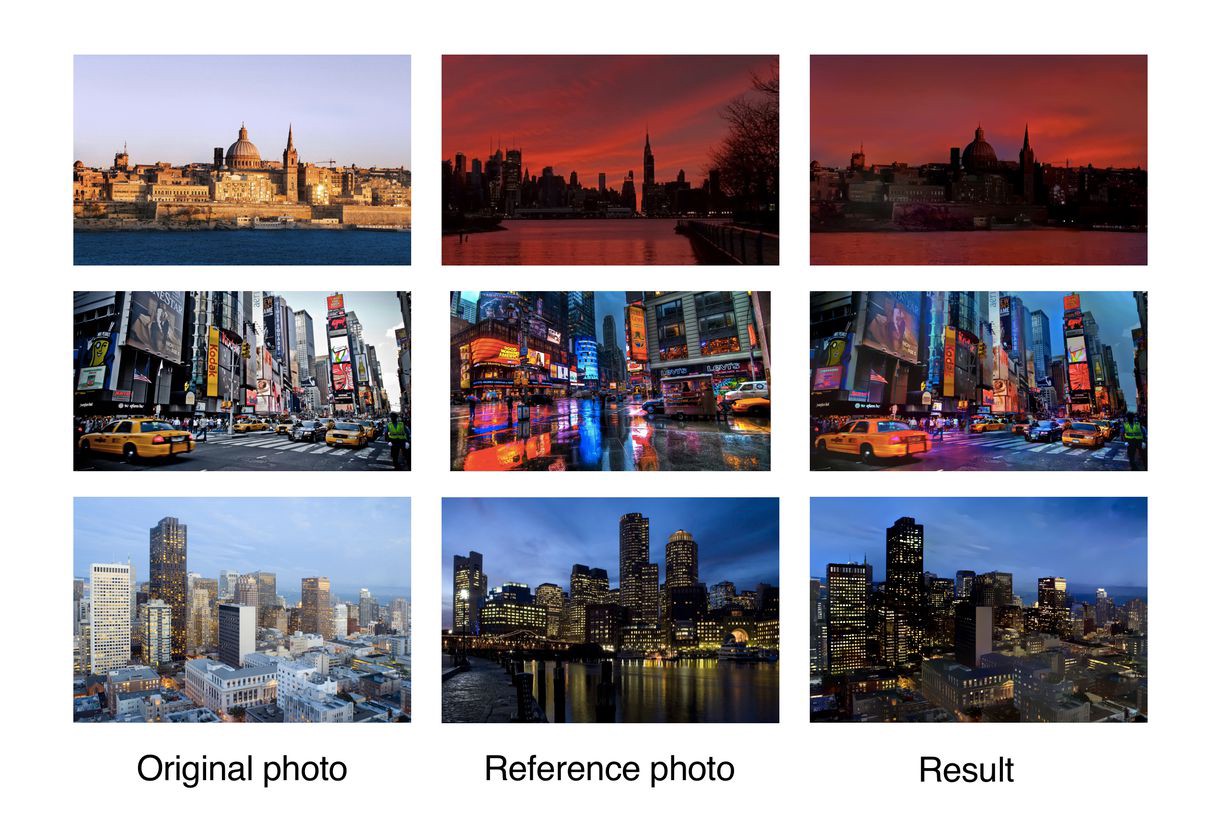
#### 有条件的甘斯去营救！

无需深入血淋淋的细节，条件GANs可以将图像从一个域转换为另一个域。如果你觉得这听起来太模糊，那就不是了；下面是一个有条件的GANs的例子，它用于将夏季风景照片转换为冬季风景。



上述方法是稳健的，但计算密集。另一种更便宜的方法是神经型转移。它获取一个图像（又称“样式”）的纹理/环境/外观，并将其与另一个图像的内容混合。使用这种强大的技术，我们产生了与条件GAN类似的效果（事实上，这种方法是在cgan发明之前引入的！）.

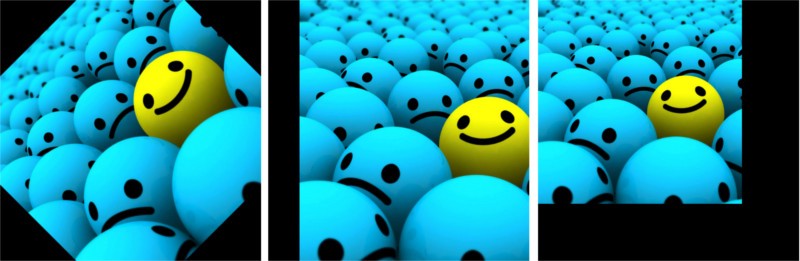
这种方法唯一的缺点是，输出往往看起来更艺术，而不是现实。然而，有一些进步，如深照片风格转移，如下所示，有令人印象深刻的结果。



我们没有对这些技术进行深入的探索，因为我们不关心它们的内部工作。我们可以使用现有的训练模型，以及转移学习的魔力，用它来扩充。

### 关于插值的一点注记

如果你想翻译一个没有黑色背景的图像呢？如果你想向内扩展呢？或者以更精细的角度旋转？执行这些转换后，我们需要保留原始图像大小。由于我们的图像没有任何关于它的边界之外的东西的信息，我们需要做一些假设。通常，图像边界以外的空间假设为每个点的常数0。因此，当您执行这些转换时，会得到一个未定义图像的黑色区域。



但这是正确的假设吗？在现实世界中，这基本上是不可能的。图像处理和ML框架有一些标准的方法，可以用来决定如何填充未知的空间。它们的定义如下。



#### 一。常数

最简单的插值方法是用一个常数来填充未知区域。这可能不适用于自然图像，但适用于在单色背景下拍摄的图像

#### 2。边缘

图像的边值在边界之后扩展。这种方法可以用于温和的翻译。

#### 三。反映

图像像素值沿图像边界反射。此方法对于包含树木、山脉等的连续或自然背景非常有用。

#### 四。对称的

该方法与反射法相似，只是在反射边界处复制边缘像素。通常，反射和对称可以互换使用，但在处理非常小的图像或图案时会看到差异。

#### 5个。包裹

图像只是在其边界之外重复，就好像它被平铺一样。这种方法不像其他方法那样被广泛使用，因为它在很多场景中都没有意义。

除此之外，您还可以设计自己的方法来处理未定义的空间，但通常这些方法对大多数分类问题都很好。

### 所以，如果我使用所有这些技术，我的ML算法将是健壮的，对吧？

如果你用对了，那就用对了！你问得对吗？嗯，有时候并不是所有的增强技术对数据集都有意义。再看看我们的汽车例子。下面是一些可以修改图像的方法。



当然，它们是同一辆车的图片，但是您的目标应用程序可能永远不会看到这些方向的车。

例如，如果你只是要对道路上的随机车辆进行分类，那么只有第二幅图像在数据集上才有意义。但是，如果你拥有一家处理车祸的保险公司，而且你还想识别倒挂、破损的汽车模型，那么第三种形象是有意义的。最后一张图片可能对上述两种情况都没有意义。

关键是，在使用增强技术时，我们必须确保不增加不相关的数据。

### 真的值得这么做吗？

你可能在期待一些结果来激励你多走一英里。很公平，我也把它盖上了。让我用一个玩具例子来证明增强确实有效。你可以复制这个实验来验证。

让我们创建两个神经网络，将数据分为四类：猫、狮子、老虎或豹。关键是，一个不使用数据扩充，而另一个使用数据扩充。你可以从这里下载数据集。

如果您已经签出了数据集，您将注意到每个类只有50个用于培训和测试的图像。显然，我们不能对其中一个分类器使用增强。为了使赔率更公平，我们使用在数据量稀少的情况下给模型一个更好的机会。

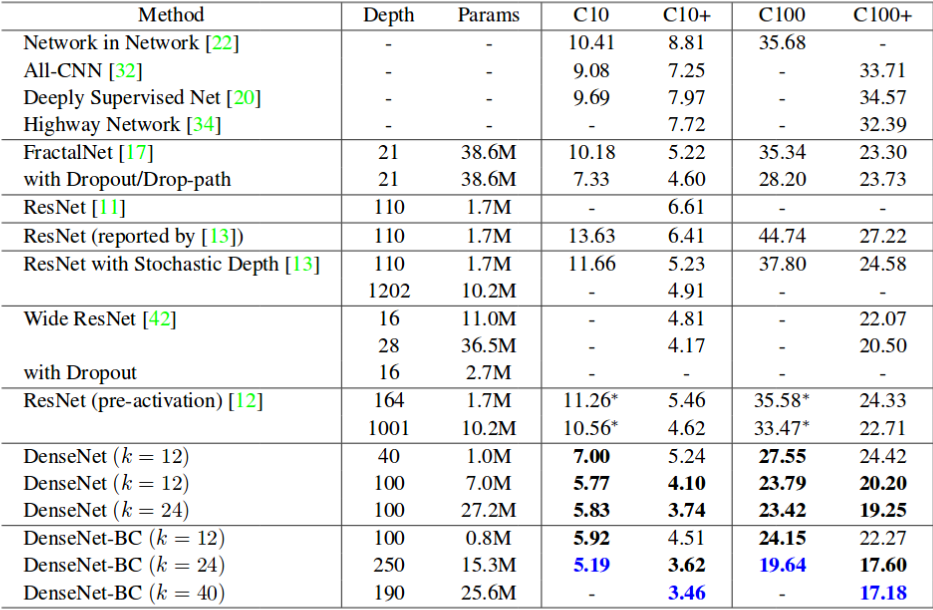


对于没有增强的，我们使用VGG19网络。我已经编写了一个TensorFlow实现，它基于。克隆后，可以从中获取数据集，并从中获取vgg19.npy（用于传输学习）现在可以运行模型来验证性能。

不过，我同意，为数据扩充编写额外的代码确实有点费劲。所以，为了建立我们的第二个模型，我求助于。他们在内部使用转移学习和数据扩充，以使用最少的数据提供最佳结果。你所需要做的就是把数据上传到他们的服务器上，然后等到他们的服务器接受了训练（通常大约30分钟）。你知道吗，这对我们的比较实验来说是完美的。

完成培训后，您可以请求调用它们的API来计算测试精度。签出我的repo以获取示例代码片段（不要忘记在代码片段中插入模型的ID）。

令人印象深刻的不是吗。事实上，大多数模型在处理更多数据时表现良好。为了提供一个具体的证据，我提到了下表。给出了Cifar 10（C10）和Cifar 100（C100）数据集上常用神经网络的错误率。C10+和C100+列是数据增加时的错误率。



谢谢你读这篇文章！如果是的话就按那个拍子！希望能对数据增长有所启发。如果你有任何问题，你可以打电话给我或给我发电子邮件（bharathrajn98@gmail.com）。

About Nanonets: Nanonets is building APIs to simplify deep learning for developers. Visit us at https://www.nanonets.com for more)