

****

**通信与信息技术开放实验**

基于TensorFlow的卷积神经

网络（CNN）图像分类

作者：杨福廷 学号：ZY1702123

2018年6月15日

**一、实验目的**

1、了解卷积神经网络（CNN）的原理和特性；

2、熟悉并掌握Python编程和TensorFlow框架；

3、加深对神经网络前向传播和反向传播的理解；

4、使用卷积神经网络（CNN）对CIFAR10图像数据集分类；

**二、实验背景**

随着AlphaGo打败李世石，到2016年年底Master取得60胜，再到AlphaGo Zero诞生。人工智能这一概念被推上了历史的新高度。

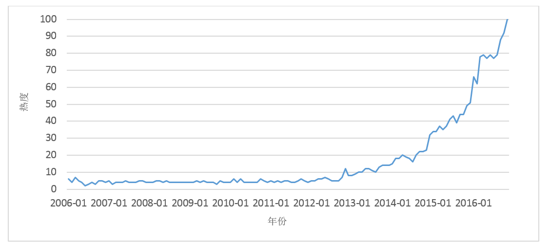


图1 “deep learning”十年内在谷歌搜索的热度趋势

深度学习是通过算法，使得机器能从大量历史数据中学习规律，从而对新的样本做智能识别或对未来做预测。从1980年代末期以来，深度学习的发展大致经历了两次浪潮：浅层学习（Shallow Learning）和深层学习（Deeping Learning）。

* 第一次浪潮：浅层学习

反向传播算法（BP）的发明，掀起了基于统计模型的机器学习热潮。但是后来，因为理论分析的难度，巨大的计算量和优化求解难度，神经网络慢慢淡出了科研领域的主流方向。

神经网络在层次深的情况下性能变得很不理想（传播时容易出现所谓的梯度弥散Gradient Diffusion或称之为梯度消失，即随着梯度的逐层不断消散导致其对网络权重调整的作用越来越小），所以只能转而处理浅层结构（小于等于3），从而限制了性能。

* 第二次浪潮：深度学习

2006年，加拿大多伦多大学教授、机器学习领域泰斗——Geoffrey Hinton和他的学生Ruslan Salakhutdinov在顶尖学术刊物《科学》上发表了一篇文章，开启了深度学习在学术界和工业界的浪潮：

很多隐层的人工神经网络具有优异的特征学习能力，学习得到的特征对数据有更本质的刻画，从而有利于可视化或分类；

深度神经网络在训练上的难度，可以通过“逐层初始化”（Layer-wise Pre-training）来有效克服，在这篇文章中，逐层初始化是通过无监督学习实现的。

**三、实验原理**

**1、卷积神经网络的基本概念和特性**

* 卷积：CNN的一个重要操作卷积是CNN的核心思想，就是这个卷积有效的提取了图像特征用于后面的图像识别。下图就是一个2维卷积的示意图，这里因为是离散的卷积，所以可以直接把卷积理解为矩阵相乘，即两个矩阵相乘，一个是输入矩阵，一个是卷积核矩阵。输入矩阵一般都表示二维的输入图像，而卷积核其实可以理解为图像处理里面的算子，比如这些算子可以实现一些边缘检测或者高斯模糊的效果，那么其实卷积操作可以理解为对图像进行一些特征处理。

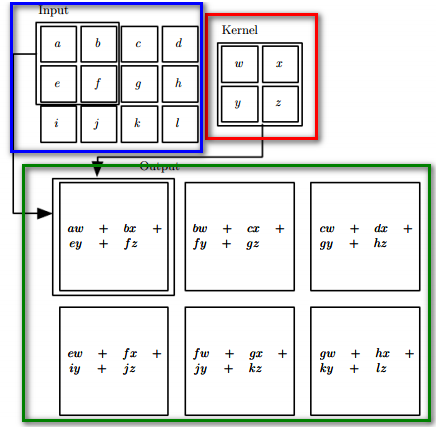


图2 卷积示意图

* 稀疏性（Sparse Connectivity）

正是由于卷积核的存在，而且卷积核的大小比整幅图像小，所以才产生了稀疏连接这样的思想。我们可以把卷积核就想成连接权，蓝框卷积核和图像大小相同，相当于全连接，所以输出层中的𝑠\_3收到输入层所有神经元的影响。

再看红色框里面的连接方式，可以把图像想象成二维的，这里只展示出了一维信息，而卷积核是3乘3大小的，所以就产生了连接权的稀疏性，即最终𝑠\_3只受到了三个神经元的影响。

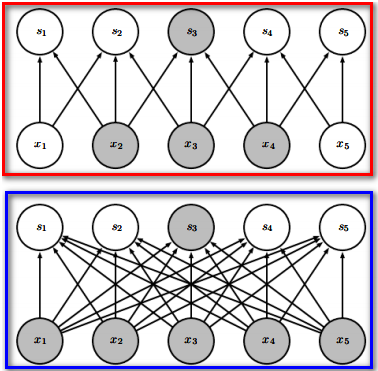


图3 上图表示稀疏性，下图表示非稀疏性

* 参数共享（Parameter Sharing）

上述提到的卷积核在整幅图像上滑动其实就是一种权值共享，因为这里的卷积核代表的其实就是权值，而这个权值用于了整幅图像从而产生了下一层网络，也就是说对于图像的每个局部位置，它们连接到下一层的权值都是一样的，所以权值共享还有另一种叫法，是𝑡𝑖𝑒𝑑 weights。这大大减少了权重的数量，降低了训练的难度和过拟合的风险。

2、**卷积神经网络的结构**

* 卷积层（Convolutional Layer）

我们现在了解了卷积操作和稀疏连接以及权值共享的思想，由这些其实就可以产生卷积层了。因为权值共享，即一个卷积核其实只能提取图像的一种特征，那想要提取图像的多种特征，就需要多个卷积核，每一个卷积核产生一个feature map，如LeNet-5，它的第一个卷积层含有6个feature map，每一个feature map对应一个卷积核，也就对应提取了图像的一种特征。这里注意最终的feature map并不是做完卷积后的结果，然后还要加一个非线性激活的操作，一般用ReLU函数，这个过程一般叫做detector stage。

* 池化层（Pooling）

pooling操作是指在生成卷积层以后，图像某块区域的值被这个区域内所有值得统计量所取代，例如max pooling操作就是把一个矩形局域内最大的输出当做这块区域的输出。当然还有其他pooling function，比如average pooling，weighted average pooling等。

pooling操作很大的用处就是，对于输入图像的局部较小的平移来说，pooling使得卷积操作后得到的特征可以保持对于平移的不变性。给定一幅图像，比如我们在做图像检测，我们只关心某个特征是否在这幅图像中出现了，而它的位置不那么重要，这时候局部平移不变性这个性质对于我们的工作就很有用。比如我们要做人脸识别，我们并不能精确的知道眼睛在图像的什么像素的位置，只是大概知道在左上角。

* 全连接层（Fully-connected Layer）

经过几次的卷积以及池化操作，网络得到了C5层，它由120个神经元组成，F6即为全连接层和FNN一样。注意这里F6层最后还要做一次sigmoid非线性变换，最后得到一个84维的二值向量。

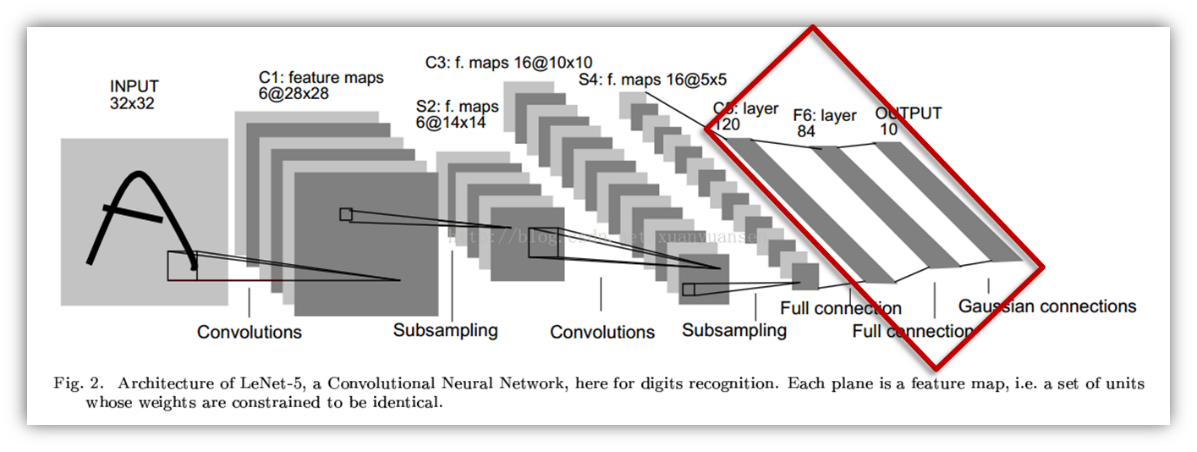


图4 LeNet-5模型

**三、设计步骤与实例**

**1、实现框架——TensorFlow**

TensorFlow是谷歌2015年开源的一个人工智能平台。就如命名一样，TensorFlow是张量从图的一端流动到另一端的计算过程。Tensorflow是将复杂的数据结构传输至人工智能神经网中进行分析和处理过程的系统。Tensorflow可被用于语音识别或图像识别等多项机器深度学习领域，它可在小到一部智能手机、大到数千台数据中心服务器的各种设备上运行。

TensorFlow 使用图来表示计算任务。图中的节点被称之为 op(operation的缩写)。一个op获得0个或多个Tensor，执行计算, 产生0个或多个 Tensor。每个 Tensor是一个类型化的多维数组。

一个TensorFlow图描述了计算的过程。为了进行计算, 图必须在会话里被启动。会话将图的op分发到诸如CPU或GPU之类的设备上, 同时提供执行op的方法。这些方法执行后, 将产生的 tensor 返回。

**2、数据集介绍**

CIFAR10数据集是一个影响力很大的图像分类数据集。CIFAR10数据集分为CIFAR10和CIFAR100两个问题，它们都是图像词典项目（Visual Dictionary）中800万张图片的一个子集。CIFAR数据集中的图片为32×32的彩色图片，这些图片是由Alex Krizhevsky教授、Vinod Nair博士和Geoffrey Hinton教授整理的。

鉴于实验硬件和实验时间的限制，我们决定使用CIFAR10来作为实验数据集。CIFAR10收集了来自10个不同种类的60000张图片——分别为供训练的50000张图片和供测试的测试集。

**3、训练（Training）**

训练即是对数据的拟合，通过最小化损失函数的方式来迭代参数。本实验采用下列方式来进行训练：

* 定义损失函数：本实验采用交叉熵作为损失函数，交叉熵刻画了两个概率分布之间的距离。



* 正则化：正则化是避免过拟合的一个很好地方式，即在损失函数中加入刻画模型复杂程度的指标，分为L1正则化和L2正则化，本实验采用L2正则化。



* 反向传播算法：通过反向传播算法可以调整神经网络中参数的取值。本实验采用批梯度下降算法进行降低损失函数。

通过反向传播算法计算损失函数对每一个参数的梯度，再根据梯度和学习率使用梯度下降算法更新每一个参数。需要注意的是，梯度下降算法并不能保证被优化的函数达到全局最优解。只有当损失函数为凸函数时，梯度下降算法才能保证达到全局最优解。

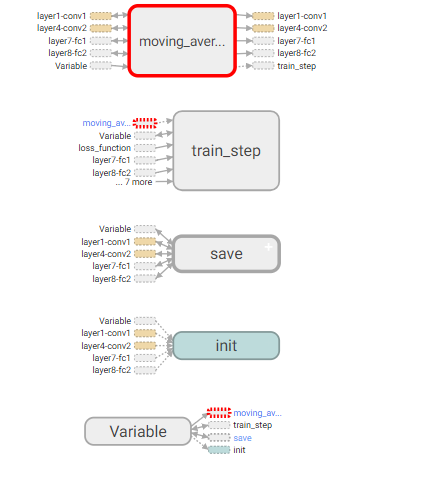
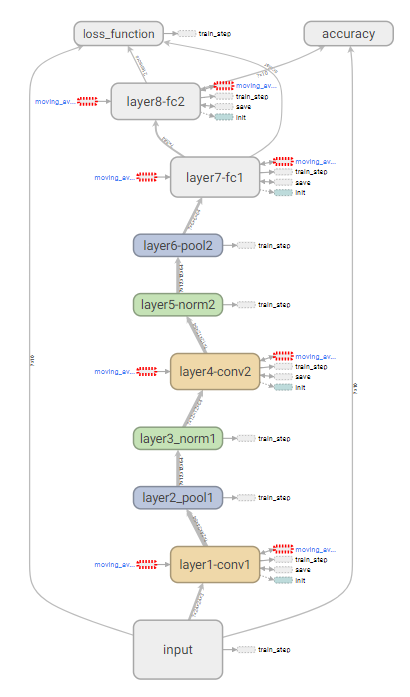
* 滑动平均模型：在采用随机梯度下降算法训练神经网络时，采用滑动平均模型的意义在于提高模型在测试数据上的健壮性。



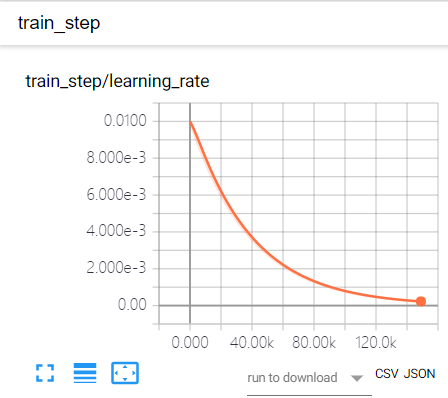
* Dropout：Dropout是指在模型训练时随机让网络某些隐含层节点的权重不工作，通过这样的方式可以避免过拟合。

**四、实验结果**

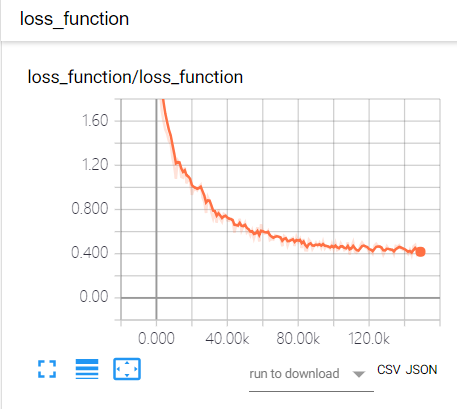
通过TensorBoard展示训练的神经网络图。

****

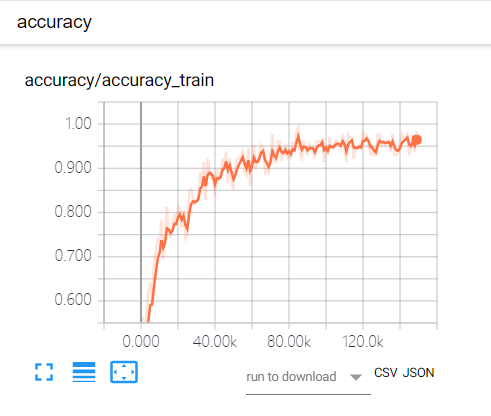
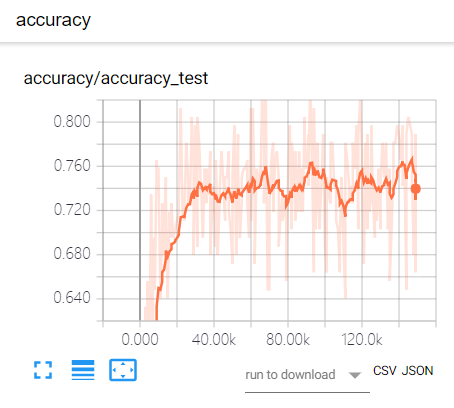
从图中可以看出，最后实现的模型与设计的模型是一致的。



上图是学习率的迭代情况。学习率初始值是0.01，每代衰减是0.99。



从上图可以看出，损失函数是在不断减小的，最后稳定在0.4左右。

****

可以看出模型对训练集的拟合为98%左右，基本稳定。测试集正确率在78%。