



结构之法 算法之道

面试、算法、机器学习在线课程：julyedu.com

目录视图

摘要视图

RSS 订阅

个人资料



v_JULY_v



访问：12665980次
积分：49542
等级：BLOG 8
排名：第24名

原创：158篇 转载：0篇
译文：6篇 评论：14823条

博主简介

July，于2010年10月11日开始在CSDN上写博（搜索：“结构之法”，进入本博客），博客专注面试、算法、机器学习。2015年正式创业，七月在线创始人兼CEO，公司官网：七月在线（<https://www.julyedu.com/>），微博@研究者July。新书《编程之法》15年10月14日起正式上市。JulyEdu c/c++/算法Q群：123531805。July，2016/5月。

July和他朋友们的创业平台



我的微博

评论送书 | 7月书讯：众多畅销书升级！ 征文 | 你会为 AI 转型么？ 赠书 | AI专场（AI圣经！《深度学习》中文版）

Kaggle—So Easy!百行代码实现排名Top 5%的图像分类比赛

2017-05-10 22:56 6143人阅读 评论(0) 收藏 举报

分类： 30.Machine L & Deep Learning (19)

Kaggle—So Easy!百行代码实现排名Top 5%的图像分类比赛

作者：七月在线彭老师
说明：本文最初由彭老师授权翟惠良发布在公众号“七月在线实验室”上，现再由July重新编辑发布到本blog上。
Github: https://github.com/pengpaiSH/Kaggle_NCFM

前言

根据我个人的经验，学好AI，有五个必修：数学、数据结构、Python数据分析、ML、DL，必修之外，有五个选修可供选择：NLP、CV、DM、量化、Spark，然后配套七月在线的这些必修和选修课程刷leetcode、kaggle，最后做做相关开源实验。

今天，咱们就来看一看：如何用百行代码实现Kaggle排名Top 5%的图像分类比赛。

1. NCFM图像分类任务简介

为了保护和监控海洋环境及生态平衡，大自然保护协会（The Nature Conservancy）邀请Kaggle[1]社区的参赛者们开发能够出机器学习算法，自动分类和识别远洋捕捞船上的摄像头拍摄到的图片中鱼类的品种，例如不同种类的吞拿鱼和鲨鱼。大自然保护协会一共提供了3777张标注的图片作为训练集，这些图片被分为了8类，其中7类是不同种类的海鱼，剩余1类则是不含有鱼图片，每张图片只属于8类中的某一类别。

图1给出了数据集中的几张图片样例，可以看到，有些图片中待识别的海鱼所占整张图片的一小部分，这就给识别带来了很大的挑战性。此外，为了衡量算法的有效性，还提供了额外的1000张图片作为测试集，参赛者们需要设计出一种图像识别的算法，尽可能地识别出这1000张测试图片属于8类中的哪一类别。Kaggle平台为每一个竞赛都提供了一个榜单（Leaderboard），识别的准确率越高的竞赛者在榜单上的排名越靠前。



图1. NCFM图像分类比赛

2. 问题分析与求解思路

2.1卷积神经网络（ConvNets）



研究者July

加关注

不明觉厉 牛逼哄哄很屌的样子



今天 16:26 转发(1) | 评论

July新书《编程之法》上市



京东 当当 天猫 Amazon 异步社区 互动出版网

文章分类

03.Algorithms (实现) (9)

01.Algorithms (研究) (27)

02.Algorithms (后续) (22)

04.Algorithms (讨论) (1)

05.MS 100' original (7)

06.MS 100' answers (13)

07.MS 100' classify (4)

08.MS 100' one Keys (6)

09.MS 100' follow-up (4)

10.MS 100' comments (4)

11.TAOPP (编程艺术) (34)

12.TAOPP string (8)

13.TAOPP array (14)

14.TAOPP list (2)

15.stack/heap/queue (0)

16.TAOPP tree (2)

17.TAOPP c/c++ (2)

18.TAOPP function (2)

19.TAOPP algorithms (8)

20.number operations (1)

21.Essays (7)

22.Big Data Processing (5)

23.Redis/MongoDB (0)

24.data structures (12)

25.Red-black tree (7)

26.Image Processing (3)

27.Architecture design (4)

28.Source analysis (3)

29.Recommend&Search (4)

30.Machine L & Deep Learning (20)

博客专栏

从问题的描述我们可以发现，NCFM竞赛是一个典型的“单标签图像分类”问题，即给定一张图片，系统需要预测出图像属于预先定义类别中的哪一类。在计算机视觉领域，目前解决这类问题的核心技术框架是深度学习（Deep Learning），特别地，针对图像类型的数据，是深度学习中的卷积神经网络（Convolutional Neural Networks, ConvNets）架构（关于卷积神经网络的介绍和算法，这里有个视频教程可以看下：[CNN之卷积计算层](#)，本博客也写过：[CNN笔记](#)）。

总的来说，卷积神经网络是一种特殊的神经网络结构，即通过卷积操作可以实现对图像特征的自动学习，选取那些有用的视觉特征以最大化图像分类的准确率。

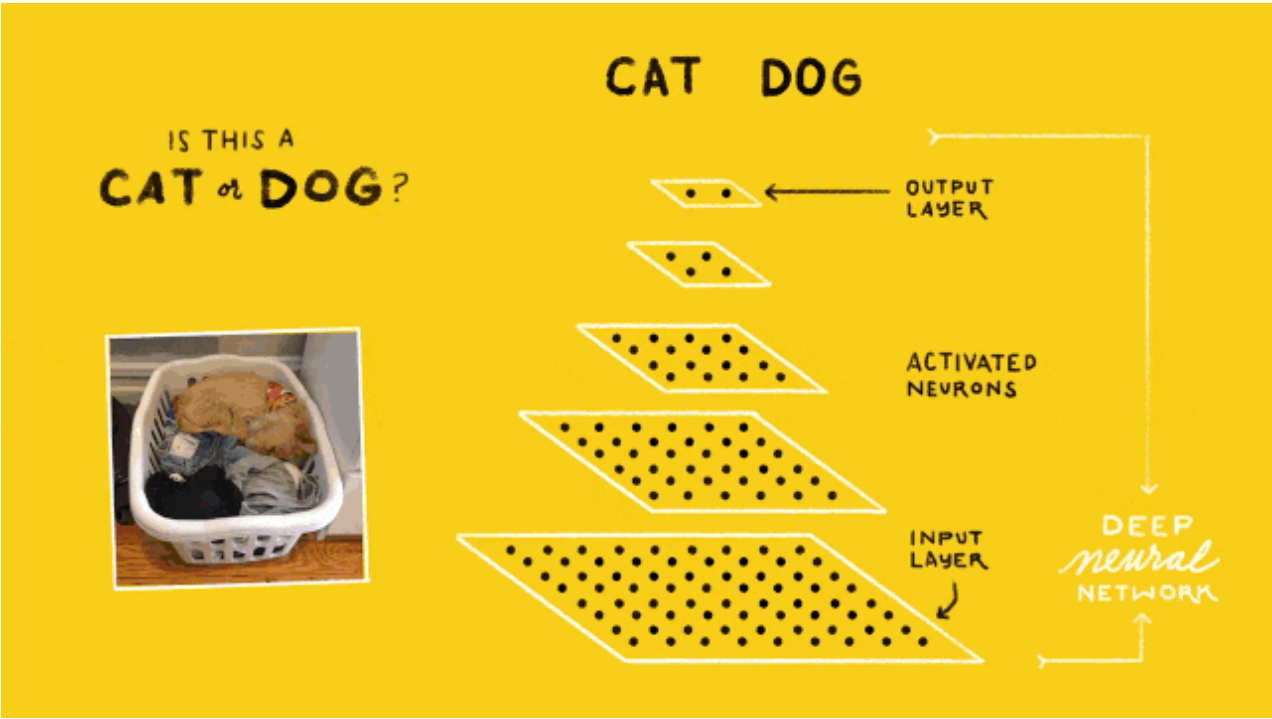


图2. 卷积神经网络架构

图2给出了一个简单的猫狗识别的卷积神经网络结构，在最底下（同时也是最大的）的点块表示的是网络的输入层（Input Layer），通常这一层作用是读入图像作为网络的数据输入。在最上面的点块是网络的输出层（Output Layer），其作用是预测并输出读入图像的类别，在这里由于只需要区分猫和狗，因此输出层只有2个神经计算单元。而位于输入和输出层的，都称之为隐含层（Hidden Layer），图中有3个隐含层，正如前文提到的，图像分类的隐含层都是由卷积操作完成的，因此这样的隐含层也成为卷积层（Convolutional Layer）。

因此，输入层、卷积层、输出层的结构及其对应的参数就构成了一个典型的卷积神经网络。当然，我们在实际中使用的卷积神经网络要比这个示例的结构更加复杂，自2012年的ImageNet比赛起，几乎每一年都会有新的网络结构诞生，已经被大家认可的常见网络有AlexNet[5], VGG-Net[6], GoogLeNet[7], Inception V2-V4[8, 9], ResNet[10]等等。

2.2 一种有效的网络训练技巧—微调（Fine-tune）

我们没有必要从头开始一个一个的参数去试验来构造一个深度网络，因为已经有很多公开发表的论文已经帮我们做了这些验证，我们只需要站在前人的肩膀上，去选择一个合适的网络结构就好了。且选择已经公认的网络结构另一个重要的原因是，这些网络几乎都提供了在大规模数据集ImageNet[11]上预先训练好的参数权重（Pre-trained Weights）。这一点非常重要！因为我们只有数千张训练样本，而深度网络的参数非常多，这就意味着训练图片的数量要远远小于参数搜索的空间，因此，如果只是随机初始化深度网络然后用这数千张图片进行训练，非常容易产生“过拟合”（Overfitting）的现象。

所谓过拟合，就是深度网络只看过了少量的样本，因而“坐井观天”，导致只能识别这小部分的图片，丧失了“泛化”（Generalization）能力，不能够识别其它没见过、但是也是相似的图片。为了解决这样的问题，我们一般都会使用那些已经在数百万甚至上千万上训练好的网络参数作为初始化参数，可以想象这样一组参数的网络已经“看过”了大量的图片，因此泛化能力大大提高了，提取出来的视觉特征也更加鲁棒和有效。

接下来我们就可以使用已经标注的三千多张海鱼图片接着进行训练，注意为了防止错过了最优解，此时的训练节奏（其实应该称为“学习速率”）应该比较缓慢，因此这样的训练策略我们称为“微调技术”（Fine-tune）。

当我们使用自己的标注数据微调某个预先训练的网络时候，有一些经验值得借鉴。以总图3为例，假设我们的网络结构是类似AlexNet这样的7层结构，其中前5层是卷积层，后2层是全连接层。



数据挖掘十大算法系列
文章：21篇
阅读：2454451



微软面试100题系列
文章：18篇
阅读：3180860



程序员编程艺术
文章：32篇
阅读：2437084



经典算法研究
文章：31篇
阅读：3271996

文章搜索

阅读排行

- 支持向量机通俗导论（理
程序员面试、算法研究、
教你如何迅速秒杀掉：99
从B树、B+树、B*树谈到
十道海量数据处理面试题
九月十月百度人搜，阿里
从头到尾彻底理解KMP（
十一、从头到尾解析Has
教你初步了解红黑树
横空出世，席卷互联网--
(617502)
(537269)
(498503)
(436251)
(313820)
(284246)
(274943)
(259327)
(247956)
(225101)

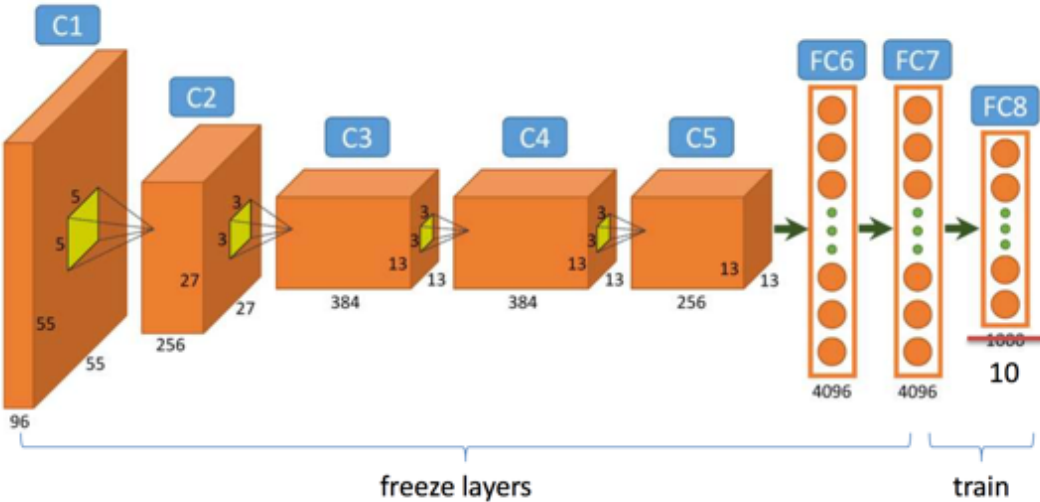
评论排行

- 支持向量机通俗导论（理
程序员面试、算法研究、
九月十月百度人搜，阿里
从头到尾彻底理解KMP（
从B树、B+树、B*树谈到
九月腾讯，创新工场，淘
当今世界最为经典的十大
横空出世，席卷互联网--
教你如何迅速秒杀掉：99
教你初步了解红黑树
(573)
(513)
(391)
(382)
(351)
(351)
(327)
(292)
(284)
(269)

最新评论

- 教你初步了解红黑树
u010235142: @cdnight:10分钟
看完这篇就能明了红黑树的插入
删除。。哪天找个实现代码撸撸
更推荐这篇。
教你初步了解红黑树
u010235142: @cdnight:有一说
一，http://www.cnblogs.com
/longdoughzt/ar...
几篇总点击率过10万的帖子，以
sinat_37131918: 发现大神博
客，希望两年的学习能帮我进过

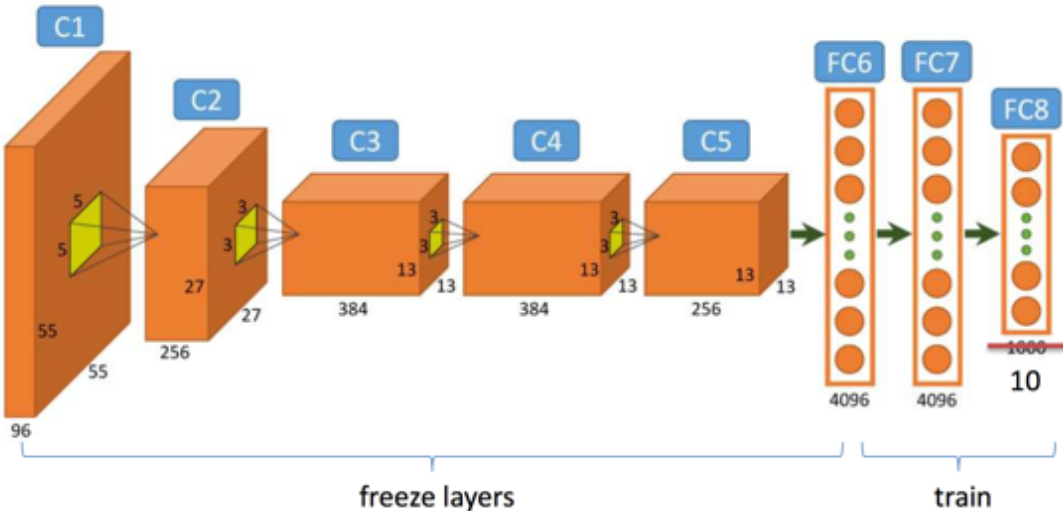
Fine-tuning Pretrained Network



(1)

（1）我们首先微调最后一层Softmax分类器，假设原来的网络是用来分类1000类物体的（例如ImageNet的目标），而现在我们的数据只有10个类别标签，因此我们最后一层输出层（FC8）的神经元个数变为10。我们使用很小的学习率来学习层FC7与FC8之间的权重矩阵而固定这之前所有层的权重；

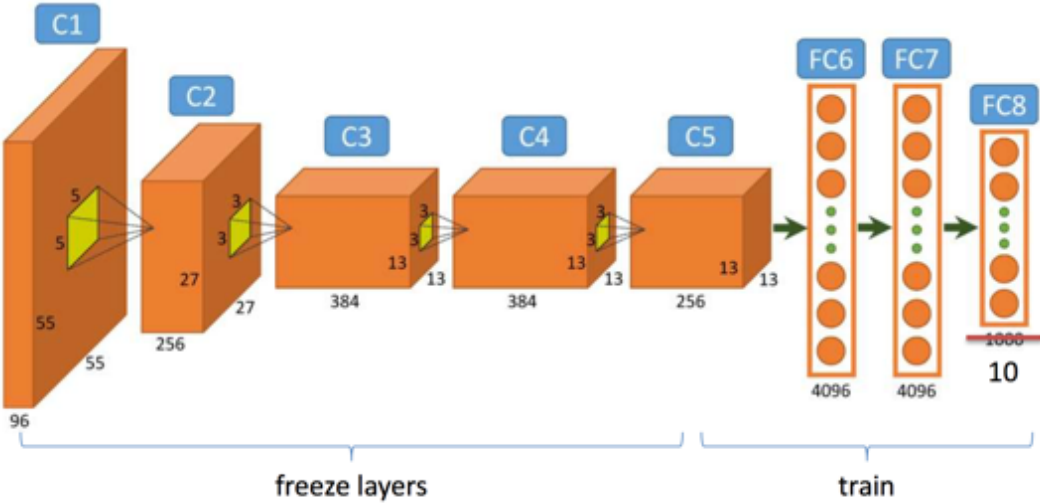
Fine-tuning Pretrained Network



(2)

（2）一旦网络趋于收敛，我们进一步扩大微调的范围，这时微调两个全连接层，即FC6与FC7，以及FC7与FC8之间的权重，与此同时固定FC6之前的所有卷积层权重不变；

Fine-tuning Pretrained Network



(3)

（3）我们将微调的范围扩大至倒数第一个卷积层C5；

微软面试

Adaboost 算法的原理与推导
wokaonagekao: 第三步的求和项呢？ $\frac{1}{N} \sum_{i=1}^N e^{-y_i f(x_i)}$...

十三个经典算法研究与总结、目：
c_wangbin: 前辈,谢谢

从B树、B+树、B*树谈到R 树
qq_35203425: 请问有源码吗

没GPU也能玩梵高作画：Ubuntu
fuxuemingzhu: 想在服务器上跑着试试，可是1G的内存带不动，直接报错MemoryError..

从头到尾彻底理解KMP（2014年
fanmula: 写得很棒！

通俗理解LDA主题模型
niebing1: 你好，我感觉你的PLSA的EM算法存在问题，在E步之后，并没有使用Jenson不等式来求似然函数的下...

通俗理解LDA主题模型
niebing1: 你好

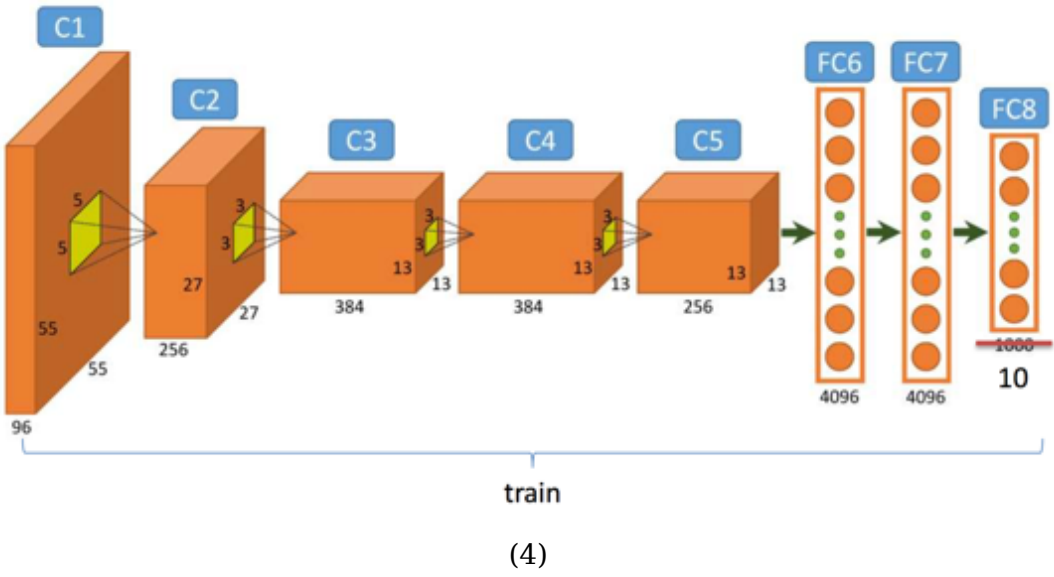
我的驻点

- 00、我的新浪微博
- 01、我的Github主页
- 02、七月在线
- 03、寒
- 04、冯
- 05、zxy
- 06、廖雪峰
- 07、左耳朵耗子
- 07、zyx
- 08、阮一峰
- 09、唐巧的技术博客
- 10、sumnous
- 11、caopengcs
- 12、joycewyj的机器学习笔记
- 13、Harry
- 14、酷勤网
- 15、IT面试论坛
- 16、北大朋友的挖掘乐园
- 17、跟Rachel-Zhang一起读硕士
- 18、51nod
- 19、韩寒
- 20、曾经的叛逆与年少
- 21、code4app:iOS代码示例
- 22、Memory Model与并发编程

文章存档

- 2017年05月 (1)
- 2017年03月 (1)
- 2017年01月 (1)
- 2016年11月 (1)
- 2016年10月 (2)
- 2016年09月 (2)
- 2016年07月 (1)
- 2015年10月 (1)
- 2015年08月 (1)
- 2014年11月 (4)
- 2014年10月 (1)
- 2014年02月 (1)
- 2014年01月 (2)
- 2013年12月 (2)
- 2013年09月 (2)
- 2013年08月 (2)

Fine-tuning Pretrained Network



（4）我们将微调的范围扩大至更多的卷积层。不过事实上，我们会认为位置相对靠前的卷积层提取出来的特征更加的底层和具有通用性，而位置相对靠后的卷积层以及全连接层更加与数据集的相关性大一些，因此有时候我们并不会微调前几个卷积层。

总图3. 网络Finetune的基本步骤

3. 算法实现和分析

在NCFM这个比赛的论坛里已经有开源的实现供大家参考(<https://www.kaggle.com/c/the-nature-conservancy-fisheries-monitoring/discussion/26202>)，

在这里分析一下模型训练文件train.py的逻辑结构。

ü Import相关的模块以及参数的设置——图4；

ü 构建Inception_V3深度卷及网络，使用在ImageNet大规模图片数据集上已经训练好的参数作为初始化，定义回调函数保存训练中在验证集合上最好的模型——图5；

ü 使用数据扩增（Data Augmentation）技术加载训练图片，数据扩增技术是控制过拟合现象的一种常见的技巧，其思想很简单，同样是一张图片，如果把它水平翻转一下，或者边角裁剪一下，或者色调再调暗淡或者明亮一些，都不会改变这张图片的类别——图6；

ü Inception_V3网络模型训练；

```
from inception_v3 import InceptionV3, preprocess_input
import os
from keras.layers import Flatten, Dense, AveragePooling2D
from keras.models import Model
from keras.optimizers import RMSprop, SGD
from keras.callbacks import ModelCheckpoint
from keras.preprocessing.image import ImageDataGenerator

learning_rate = 0.0001
img_width = 299
img_height = 299
nbr_train_samples = 3019
nbr_validation_samples = 758
nbr_epochs = 50
batch_size = 32
```

图4. Import和参数设置

2013年06月 (1)
2013年03月 (1)
2012年12月 (1)
2012年11月 (1)
2012年09月 (1)
2012年06月 (1)
2012年05月 (2)
2012年04月 (1)
2012年03月 (2)
2012年02月 (2)
2012年01月 (2)
2011年12月 (5)
2011年11月 (1)
2011年10月 (7)
2011年09月 (2)
2011年08月 (11)
2011年07月 (3)
2011年06月 (7)
2011年05月 (10)
2011年04月 (7)
2011年03月 (17)
2011年02月 (11)
2011年01月 (18)
2010年12月 (14)
2010年11月 (6)
2010年10月 (5)

```
print('Loading InceptionV3 Weights ...')
InceptionV3_notop = InceptionV3(include_top=False, weights='imagenet',
                                input_tensor=None, input_shape=(299, 299, 3))
# Note that the preprocessing of InceptionV3 is:
# (x / 255 - 0.5) x 2

print('Adding Average Pooling Layer and Softmax Output Layer ...')
output = InceptionV3_notop.get_layer(index = -1).output # Shape: (8, 8, 2048)
output = AveragePooling2D((8, 8), strides=(8, 8), name='avg_pool')(output)
output = Flatten(name='flatten')(output)
output = Dense(8, activation='softmax', name='predictions')(output)

InceptionV3_model = Model(InceptionV3_notop.input, output)
#InceptionV3_model.summary()

optimizer = SGD(lr = learning_rate, momentum = 0.9, decay = 0.0, nesterov = True)
InceptionV3_model.compile(loss='categorical_crossentropy', optimizer = optimizer, metrics = ['accuracy'])

# autosave best Model
best_model_file = "./weights.h5"
best_model = ModelCheckpoint(best_model_file, monitor='val_acc', verbose = 1, save_best_only = True)
```

图5. 构建Inception_V3网络并加载预训练参数

```
# this is the augmentation configuration we will use for training
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.1,
    zoom_range=0.1,
    rotation_range=10.,
    width_shift_range=0.1,
    height_shift_range=0.1,
    horizontal_flip=True)

# this is the augmentation configuration we will use for validation:
# only rescaling
val_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size = (img_width, img_height),
    batch_size = batch_size,
    shuffle = True,
    classes = FishNames,
    class_mode = 'categorical')

validation_generator = val_datagen.flow_from_directory(
    val_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    shuffle = True,
    classes = FishNames,
    class_mode = 'categorical')
```

图6. 使用数据扩增技术加载训练和验证图片集


```
InceptionV3_model.fit_generator(  
    train_generator,  
    samples_per_epoch = nbr_train_samples,  
    nb_epoch = nbr_epochs,  
    validation_data = validation_generator,  
    nb_val_samples = nbr_validation_samples,  
    callbacks = [best_model])
```

图7. 模型训练

4. 提升排名的若干技巧

一旦我们训练好了模型，我们就用这个模型预测那些测试图片的类别了，论坛中predict.py中的代码就是预测鱼类的并且生成提交文件。这里我们给大家分享一下在机器学习和图像识别类竞赛中常见的两个技巧，简单而有效。他们的思想都是基于平均和投票思想。其背后的原理用一句话总结就是：群众的眼睛是雪亮的！

技巧1：同一个模型，平均多个测试样例

这个技巧指的是，当我们训练好某个模型后，对于某张测试图片，我们可以使用类似数据扩增的技巧生成与改张图片相类似的多张图片，并把这些图片送进我们训练好的网络中去预测，我们取那些投票数最高的类别为最终的结果。Github仓库中的predict_average_augmentation.py实现的就是这个想法，其效果也非常明显。

技巧2：交叉验证训练多个模型

还记得我们之前说到要把三千多张图片分为训练集和验证集吗？这种划分其实有很多种。一种常见的划分是打乱图片的顺序，把所有的图片平均分为K份，那么我们就可以有K种<训练集，验证集>的组合，即每次取1份作为验证集，剩余的K-1份作为训练集。因此，我们总共可以训练K个模型，那么对于每张测试图片，我们就可以把它送入K个模型中去预测，最后选投票数最高的类别作为预测的最终结果。我们把这种方式成为“K折交叉验证”（K-Fold Cross-Validation）。图9表示的就是一种5折交叉验证的数据划分方式。

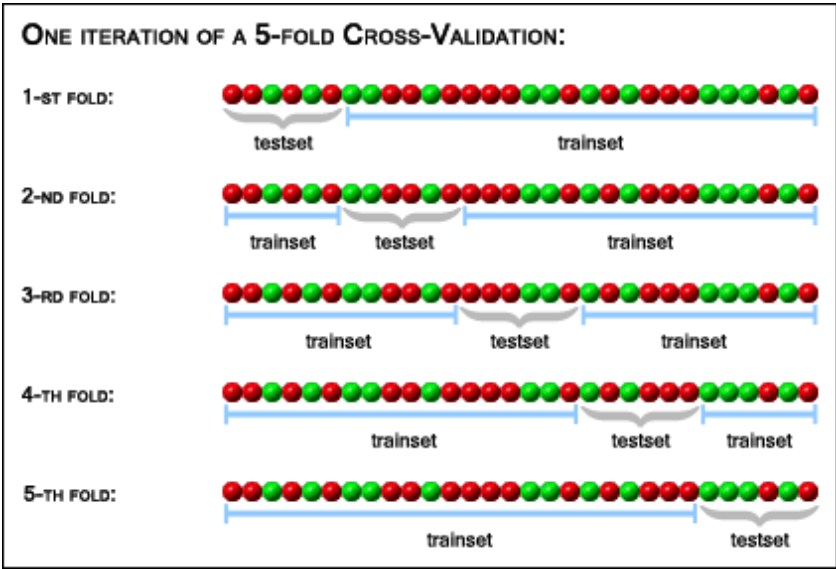


图9. 五折交叉验证

当然，技巧1和2也可以联合在一起使用。假设我们做了5折交叉验证，并且对于每一张测试图片都用5次数据扩增，那么不难计算，每一张测试图片的投票数目就是25个。采用这种方式，我们的排名可以更进一步。

5. 后记

我们回顾了深度学习中的深度卷积网络的典型结构和特点，并且知道了如何使用梯度下降算法来训练一个深度网络。我们展示了如何用微调技术，使用Inception_V3网络来解决Kaggle的NCFM海鱼分类比，并且通过两个简单而有效的小技巧，使得我们的排名能够进入Top 5%。

如果读者对该比赛有兴趣，想进一步提升名次，那么一种值得尝试的方法是：物体检测（Object Detection）技

术。试想一下，其实我们只要区分海鱼的品种，由于摄像头远近等关系，图片中海鱼的区域其实只占据一小部分像素点，更多的区域都是船体、桅杆或是海洋等噪音。如果有一种算法能够帮我们把海鱼从照片中“扣”（检测）出来，那么可以想象，深度网络的准确率就能够进一步提升了，这部分的工作就留给有兴趣的同学自己做进一步研究了。

七月在线彭老师、二零一七年五月十日。

参考文献

1. [1] <https://www.kaggle.com/>
2. [2] <http://cs231n.github.io/neural-networks-3/>
3. [3] <https://github.com/tensorflow/tensorflow>
4. [4] <https://github.com/fchollet/keras>
5. [5] Image Classification with Deep Convolutional Neural Networks. NIPS 2012.
6. [6] Very Deep Convolutional Networks for Large-Scale Image Recognition. ICLR 2015.
7. [7] Going Deep with Convolutions. CVPR 2015.
8. [8] Rethinking the Inception Architecture for Computer Vision. CVPR 2016.
9. [9] Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. ICLR 2016.
10. [10] Deep Residual Learning for Image Recognition. CVPR 2016.
11. [11] <http://www.image-net.org/>
12. 七月在线《[kaggle案例实战班](#)》
13. kaggle实战公开课《[模型分析与模型融合](#)》

顶

2

踩

0

上一篇 GAN之父在NIPS 2016上做的报告：两个竞争网络的对抗（含译文下载）

相关文章推荐

- 数据挖掘领域十大经典算法初探
 - 从Hadoop框架与MapReduce模式中谈海量数据处...
 - 教你从头到尾利用DQN自动玩flappy bird（全程命...
 - 程序员面试、算法研究、编程艺术、红黑树、数据...
 - 教你如何迅速秒杀掉：99%的海量数据处理面试题
- 从决策树学习谈到贝叶斯分类算法、EM、HMM
 - 最简单的基于FFmpeg的解码器-纯净版（不包含li...
 - 没GPU也能玩梵高作画：Ubuntu tensorflow CPU版
 - 教你从头到尾利用DL学梵高作画：GTX 1070 cud...
 - July 博客中的机器学习系列

猜你在找



查看评论

暂无评论

发表评论

用户名：

haijunz

评论内容：

提交

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

公司简介 | 招贤纳士 | 广告服务 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |

江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved

