机器学习纳米学位-开题报告
2018 年 03 月 26 日 23:05:13

I. 问题的定义

项目背景

本项目来源于 kaggle 竞赛项目，最终目的是训练一个机器学习模型，输入一张图片来分辨图像中的猫和狗，也可是猫或狗的面部坐标甚至是身体的一部分。是一个典型的图像二分类问题。

本项目使用的卷积神经网络（CNN），相比于其他算法，卷积神经网络在图像识别领域具有更低的错误率，通过卷积神经网络达到的错误率非常接近人工标注的错误率，甚至机器比人更低的错误率。

本项目不仅仅需要训练基于 CNN 的机器学习模型，还需要有应用的场景，因此做一个简单的网页，游客可以上传图片，应用来识别图像并返回结果，告诉游客结果是猫和狗。

输入数据

项目的数据集来源于 kaggle 竞赛的数据，该数据集搜集了 25000 张猫和狗的图片，训练集中包含 12500 张被标记为猫和 12500 张被标记成狗的图片，测试集包含 12500 张未标记的图片。

经过训练后模型需要预测出猫或狗的概率(1=狗，0=猫)，如果是狗概率越接近 1，否则概率接近 0。

判断依据如果概率大于 0.5 判定为狗，概率小于 0.5 判定为猫，是个典型的二分类问题。

问题描述

1. 训练数据量比较大的问题

由于训练集的图片数量很庞大，如果将图片一次加载到内存中，一方面会导致加载时间过长，另一方面容易导致 OOM 内存溢出。因此采用 batch 进行小批量的加载数据解决该问题，这也是对训练精度与训练时间、资源的折中考虑。

2. 对模型的泛化的问题

虽然训练集的图片数量很庞大，但是仍然收集到所有猫狗的图片，因此需要提高模型的泛化性。否则，很有可能在训练集的效果很理想，但是在测试集效果一般的现象。这里可以利用 OpenCV 对图像进行翻转变换、镜像、拉伸等处理，提高训练集的数量进一步提高训练模型的复杂度。另一方面，也可以利用用户上传带有标记的图片，进一步扩充训练集的数量。从而达到提高模型泛化能力的目的。

3. 识别猫狗脸部的坐标或者身体的 mask

利用 OpenCV 或者 tf.image.draw_bounding_boxes 函数勾画出猫狗脸部的坐标以及身体的 mask,这里比较难的是识别出身体的 mask，比如有的猫藏在某个位置，只露出很小一部分，或者和背景颜色很接近，或者图片模糊，甚至混进狗的图像掩盖了猫。那么想要识别出猫是非常困难的，且容易误判。

辨别困难的图像样本

4.光线强弱等环境对识别效果的影响

和图像翻转类似，调整图像的亮度、对比图、饱和度和色相在很多图像识别的应用中都不会影响识别的结果。所以在训练神经网络模型，可以随机调整图像的这些属性，从而使训练得到的模型尽可能地受无关因素的影响。

高亮度和对比度图像

5.图像编码处理

一张 RGB 色彩模式的图像可以看成一个三维矩阵，矩阵中的每个数代表图像的不同位置，不同颜色的亮度。但是图像在存储时并不是直接记录这些矩阵中的数字，而是记录经过压缩编码后的结果。所以在做识别时候需要将一张图像还原成一个三维矩阵，需要解码的过程。TensorFlow 提供了对 jpeg 和 png 格式图像编码/解码的方式。对猫狗 jpg 格式图像进行编码/解码。

6.图像大小调整

数据集中的图像大小是不固定的，但是神经网络输入节点的个数是固定的。所以在将图像的像素作为输入之前，需要将图像的大小统计。有两种方式解决这个问题，一种方式是通过算法得到新的图像尽量保存原始图像上的所有信息，可以利用 TensorFlow 的 tf.image.resize_image 函数。另一种方式是对原始图像进行裁剪或者填充，利用 tf.image.crop_to_bounding_box 和 tf.image.pad_to_bounding_box 函数。当图像大于目标图像时进行裁剪，当图像小于目标函数时，对图像进行填充。从而达到统一图像的目的。

7.异常数据的处理

训练集中绝大部分数据都是准确的，但是仍然对个别图像进行人工标记。比如下图中，玩具猫混在训练集中。因此需要对其进行剔除。

异常图像

评价指标

对数损失（Log loss）亦被称为逻辑回归损失（Logistic regression loss）或交叉熵损失（Cross-entropy loss）。交叉熵是常用的评价方式之一，它实际上刻画的是两个概率分布之间的距离，是分类问题中使用广泛的一种损失函数。

本文实际上是二分类问题，因此可以采用 logloss 损失函数作为评价指标，计算公式如下：

$$\text{LogLoss} = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y_i}) + (1-y_i)\log(1-\hat{y_i})]$$

采用交叉熵作为损失函数可以有效的解决梯度消失和梯度爆炸的问题。

为了进一步的对模型进行优化，可以采用交叉熵和 L2 正则化损失和，作为损失函数。

L2 正则化公式如下所示：

$$\mathrm{R}(w) = \|w\|_2^2 = \sum_i \left|w_i^2\right|$$

完整的损失函数公式为：

$$\mathrm{loss} = \mathrm{LogLoss} + \mathrm{R(w)}$$

II. 分析

基准模型

基准模型选择 VGG16，是 ImageNet2014 年非常流行的模型。

VGG 是一种由 K. Simonyan 和 A. Zisserman 提出的卷积神经网络模型，出自牛津大学的论文“非常深度的卷积网络用于大规模图像识别”[1]。模型在 ImageNet 图像分类测试集中包含 1000 个分类 1400 多万张图片，其中每张图片属于且只属于一个分类。该模型达到了 92.7%的 top-5 正确率。

VGG16 结构如下所示：

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Table 2: **Number of parameters** (in millions).

Network	A,A-LRN	B	C	D	E
Number of parameters	133	133	134	138	144

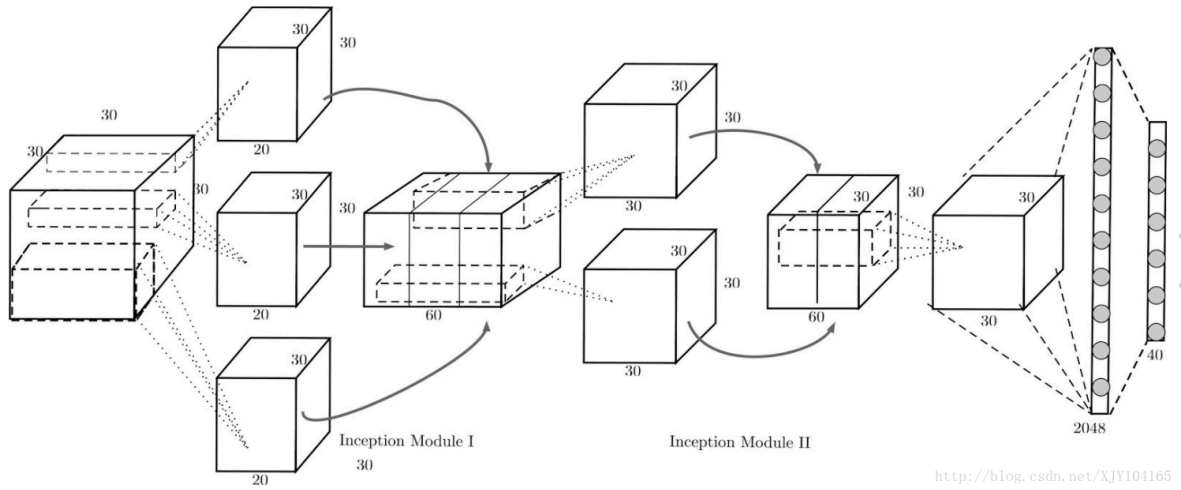
VGG16 结构图解

本文基准阈值参考 kaggle 排行榜 10%，也就是 logloss 要低于 0.06114。

设计大纲

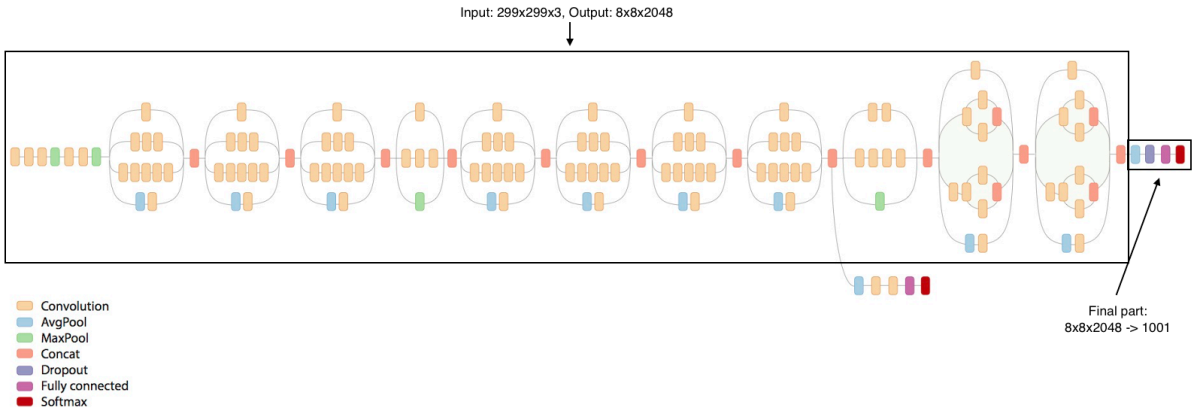
本文采用迁移学习，所谓的迁移学习，就是讲一个问题上训练好的模型通过简单的调整使其适用于一个新的问题。目标模型采用 Inception-v3 模型.

Inception (GoogLeNet)是 Google 2014 年发布的 Deep Convolutional Neural Network。Inception-v3 是 2015 年提出的模型，跟前者比卷积神经网络模型的层数和复杂度都发生了巨大的变化。



Inception 模块示意图

上图就是 2 个 Inception 模块，这种结构是一种和 LeNext-5 结构完全不同的卷积神经网络。在 LeNext-5 模型中，不同卷积层通过串联的方式连接在一起，而 Inception-v3 模型的 Inception 结构将不同的卷积层通过并联的方式结合起来。



Inception-v3 模型架构图

Inception-v3 模型总共有 46 层，由 11 个 Inception 模块组成。在 Inception-v3 模型中有 96 个卷积层[2]。

在 Inception-v3 的实际测试结果中，Top-1 的错误率为 4.2%，Top-5 的错误率为 18.77%，与其他的几种卷积神经网络来说错误率低了不少。

Network	Models Evaluated	Crops Evaluated	Top-1 Error	Top-5 Error
VGGNet [18]	2	-	23.7%	6.8%
GoogLeNet [20]	7	144	-	6.67%
PReLU [6]	-	-	-	4.94%
BN-Inception [7]	6	144	20.1%	4.9%
Inception-v3	4	144	**17.2%**	**3.58%***

2012-2015 年 ILSVRC 综合比较结果

考虑到 Inception-v3 模型非常庞大，需要耗费大量时间和精力。如果自己重新去训练没有足够的分类数据，因此我们采用迁移学习，直接使用 ImageNet 图形训练后的 Inception-v3 作为模型，作为本文项目的识别算法。这样可以节约对复杂的卷积神经网络训练的时间，这里可能是几天甚至几周。同时也解决了收集大量数据集和标注数据集所耗费的时间。

根据论文 DeCAF:A Deep Convolutional Activation Feature for Generic Visual Recongition 中的结论，可以保存训练好的 Inception-v3 模型中所有卷积层的参数，只是替换最后一层全连接层。在最后一层全连接层之前的网络层称之为瓶颈层[3]。

将新的图形通过训练好的卷积神经网络直到瓶颈层的过程可以看成是对图像进行特征抽取的过程。在训练好的 Inception-v3 模型中，因为将瓶颈层的输出在通过一个单层全连接层神经网络可以很好的区分 1000 种类别的图像。由于本项目只需要识别猫和狗实际上是一个二分类问题，因此最后再加上一个 softmax 层或者 sigmoid 层。于是，在新的数据集上，可以直接利用训练好的神经网络对图像进行特征提取，然后再讲提取得到的特征向量作为输入来训练一个新的单层全连接神经网络处理新的分类问题。

一般情况下，在数据量足够的情况下，迁移学习的效果不如完全重新训练。但是迁移学习所需要的训练时间和训练样本数远远小于训练完整的模型。

III.参考文献

[1] Karen Simonyan and Andrew Zisserman. VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION. At ICLR,2015.

[2] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng and Trevor Darrell. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition.In ICML,2014.

[3] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In arXiv,2015.

1 Karen Simonyan and Andrew Zisserman. VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION. At ICLR,2015.
2 Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In arXiv,2015.

3 Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng and Trevor Darrell. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition.In ICML,2014.