机器学习纳米学位

毕业项目开题报告

刘佳业
2018年04月30日

摘要：

使用计算机算法，实现对猫、狗的图片进行识别并准确分类。

背景：

当前计算机领域，图像识别已经成为科学研究的重要领域，目前，一些领域已经在广泛应用图像识别，如：字符识别（名片扫描加载、车牌号码、手写字符、邮政编码、编码识别等），人脸识别（人脸解锁、安防等），指纹识别，路况判断（无人驾驶），越来越多的场景会应用到图像的识别。在此，我先以kaggle的竞赛项目“猫狗大战”作为起点，使用机器学习训练一个算法模型，用于对猫或狗的图片进行高精度的分类。该分类看起来使用意义不大，但是有趣。

一开始，猫狗大战是kaggle在13年的竞赛项目，那个时候机器学习都还没有很热，甚至连TensorFlow都还没有发布（TensorFlow于2015年11月9日在Apache 2.0开源许可证下发布），但时过境迁，随着人工智能这几年的快速发展，对图像识别的算法有了长足进步，精确度也有了较大的提高。

问题描述

使用算法对图片进行分类，指出图片里面的动物是猫或者是狗，这是一个二分类问题，同时，算法需要指明归类的自信度（0-1），根据算法在测试集的归类的准确率，以及由自信度数据计算得出的损失函数高低，由此损失函数我们能够知道算法的优劣。同时，因为我们的测试集有12500张各式各样的猫，狗图片，也就是数量较多，且这些图片的场景、光线、分辨率、动物肤色及纹理、数量、姿态等，各不相同，是有代表性的，如果算法能在测试集表现良好，我有理由相信它也能在其他场合对猫狗的判断表现良好。

数据和输入

输入数据被分为两个集合，一个作为训练数据集，用来改进算法，一个作为测试数据集，用来测试算法的性能。其中训练数据集有25000张图片，其中猫、狗的图片各占一半：12500张。测试数据集共有12500张，都未明确标注是猫或者是狗。

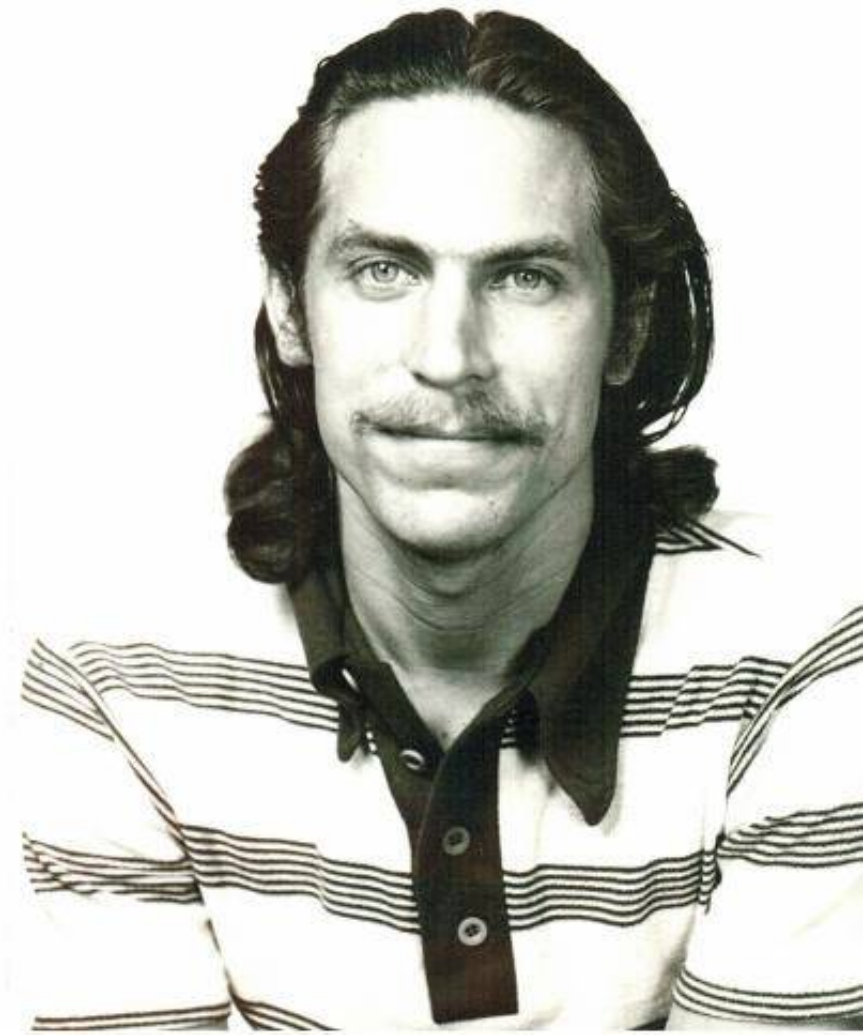
不论是训练数据集还是测试数据集，图片中的场景、光线、分辨率、动物肤色及纹理、数量、姿态等，各不相同。比如里面的猫有：花猫、白猫、黑猫、灰猫，各色各样；数量有1只、2只、1群等；光线有白天、黑夜，甚至有类似聚光灯的照射；背景也是户内户外皆有，但是户内的场景稍多。这些差异各色各样，这样可以提高算法在实际场合中的适用性，防止算法对训练集和测试集过拟合，导致对现实生活中的猫狗分辨能力变差。

另外，需要处理图片分辨率。鉴于原始图片的分辨率不一致，比如500*374(cat.24.jpg)，499*375(cat.17.jpg)，154*290(cat.154.jpg)，也就是输入数据格式不一致，需要预先处理，可以考虑直接拉伸为同一分辨率作为输入数据。

在对数据进行探索时发现，有几个异常值，将会在作为输入前直接删除，如：1、cat.4085.jpg，标注为猫，但其实是一只狗；2、cat.7377.jpg，标注为猫，但其实是一个人。如下：

1、cat.4085.jpg：

2、cat.7377.jpg



在算法的训练阶段，应该从训练数据集切割一部分数据作为训练过程中的验证数据集，而不能将测试数据集作为验证。防止算法对测试数据集进行了过度适配，增加过拟合的风险。这样，就会有三部分数据集：训练数据集，验证数据集和测试数据集。

解决方法描述

使用预训练的网络模型（参考：TensorFlow-Slim image classification model library），如：Inception，VGG 19，ResNet等，也可以是几个模型的组合，对训练的图片提取特征向量（bottleneck特征），然后再对导出的特征（bottleneck）应用全连接层、池化层、dropout、输出层等。

这样的话，就不对现有模型（如Inception的网络权重、参数）进行微调，而是直接使用，训练时，只是调节最后添加的全连接层，池化层和输出层等。

使用已有的模型有几方面好处：1、可以不用从头开始训练一个复杂的网络，节约时间；2、这些算法模型是经过大量训练得出的，效果经得起考验，而如果我们从头开始训练一个复杂的神经网络，25000张训练集图片不一定足够。

另外，需要对训练集、验证集、测试集图片采用一样的预处理方法，包括处理图片的分辨率大小一致，及预处理模型参数一致，这样才能保证训练出来的算法模型有效的应用到测试集上。

同时，为了模型的可复现，训练时，需要使用确定的随机数种子。

基准模型

以Kaggle的Public LeaderBoard排名10%作为基准，也就是在测试集上的logloss=0.06127（这是当前Public LeaderBoard上排名为131/1314的分数）。

我在此将会沿用该logloss算法评估我的分类器性能，期望是使算法模型能在kaggle的Public LeaderBoard排名10%以内，即logloss<0.06127

评估标准

这是一个二分类问题，竞赛使用的是log Loss（即对数损失函数）来评估算法的性能，我在此将会沿用该算法评估我的分类器性能：

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

其中：

n是指测试集图片的数量；

$\hat{y}_i$是预测图片是狗的概率（自信度）

y_i是指图片为狗或者猫（1代表狗，0代表猫）

$\log()$是以e为底的自然对数

可以发现：

当实际图片为狗时，有y_i=1，1-y_i=0，因此$-(1-y_i)\log(1-\hat{y}_i)$=0，主要看$-y_i \log(\hat{y}_i) = -\log(\hat{y}_i)$，那么当$\hat{y}_i$越大（即算法预测为狗的自信度越高），$-\log(\hat{y}_i)$越小，损失函数越小；

当实际图片为猫时，有y_i=0，1-y_i=1，因此$-y_i \log(\hat{y}_i)$=0，主要看$-(1-y_i)\log(1-\hat{y}_i)$=$-\log(1-\hat{y}_i)$，那么当$\hat{y}_i$越小（即算法预测为狗的自信度越低，预测为猫的自信度越高），$-\log(1-\hat{y}_i)$越小，损失函数越小。

项目设计

1、载入一个预先训练的模型Xception，用于图片特征提取（如有需要，可以对图片进行预处理），再构建Flatten、Dense、Dropout以及输出层等。

2、对图片（训练集和验证集，测试集）进行特征的提取，这些特征用于输入到后面构建的几层神经网络，并对这几层网络进行训练及优化。

3、若需要再提升准确率：a)、可使用多个预训练模型，包括Xception、ResNet以及Inception v3提取特征，将这些特征组合起来之后再传输给后面构建的网络进行训练；b)、对训练数据集进行数据增强；

引用：

[1] Francois Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. arXiv:1610.02357

[2] Kaiming He. Xiangyu Zhang. Shaoqing Ren. Jian Sun. Deep Residual Learning for Image Recognition. arXiv:1512.03385

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. arXiv:1406.4729v4

[4] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. arXiv:1512.00567