

Machine Learning Engineer Nanodegree

“猫狗大战”开题报告

叶硕杨

06月11号，2018

开题报告

Domain Background

“猫狗大战”的毕业项目是 kaggle 的一个比赛项目，要求参赛人员判别一张图片是猫还是狗。随着计算机硬件依照“摩尔定律”的高速发展，以及数据量的大幅提升，通过深度学习进行图形识别的准确率大幅度提高。“猫狗大战”这个毕业项目，使用卷积神经网络 (CNN) 进行建模，因为以往的结果表明，卷积神经网络对图形识别、图形分类上有更加良好的表现。目前，机器学习在无人汽车、美颜相机、图片应用和短视频应用等等领域有越来越多的应用，帮助提升产品的“智能”水平，提高服务水平、预测帮助等，比如一些图形识别的研究在进行，比如[AI App Identifies Plants and Animals In Seconds](#)等等一些报道。无论是现在很火热的无人汽车，还是现在无处不在的监控系统，或者机器人领域，等等需要“视觉”的系统，对于图形的识别都具有研究的前景。

Problem Statement

在给定的已经做了“标签”的猫和狗的图片库中进行训练，通过深度学习训练识别模型，从而达到：

输入：一张图片。

输出：是“狗”的概率，0~1之间。

结果：大于0.5判定为“狗”，反之则为“猫”。

Datasets and Inputs

数据集是kaggle的 Dogs vs. Cats Redux: Kernels Edition (<https://www.kaggle.com/c/dogs-vs-cats-redux-kernels-edition/data>) 网站上下载的数据集，分为两个文件夹，训练数据集train和测试数据集test。

train 里面有25000个已经标记cat/dog的图片，各一半；test 里面有12500个未标记的图片。

图片特点：

1. 图片大小不是固定，大小不一，如下图：

	cat.0.jpg	类型: JPG 文件 分辨率: 500 x 374	大小: 12.1 KB
	cat.1.jpg	类型: JPG 文件 分辨率: 300 x 280	大小: 16.4 KB
	cat.2.jpg	类型: JPG 文件 分辨率: 312 x 296	大小: 24.3 KB
	cat.3.jpg	类型: JPG 文件 分辨率: 500 x 414	大小: 37.0 KB
	cat.4.jpg	类型: JPG 文件 分辨率: 499 x 375	大小: 20.1 KB
	cat.5.jpg	类型: JPG 文件 分辨率: 175 x 144	大小: 5.25 KB
	cat.6.jpg	类型: JPG 文件 分辨率: 400 x 303	大小: 20.9 KB
	cat.7.jpg	类型: JPG 文件 分辨率: 495 x 499	大小: 36.0 KB
	cat.8.jpg	类型: JPG 文件 分辨率: 461 x 345	大小: 22.5 KB
	cat.9.jpg	类型: JPG 文件 分辨率: 320 x 425	大小: 15.8 KB

进行建模之前，需要进行预处理，使得图片大小相同。

2. 判别的“猫”、“狗”场景不唯一，有单个猫狗出现、有多个猫狗出现、有猫狗不是图片主体的情况、猫狗同时出现等等，如下图：



cat.16.jpg



cat.2236.jpg



dog.1486.jpg



cat.2159.jpg

Solution Statement

识别的图片只要求进行cat和dog的判断，是二元分类问题，可以使用决策树、支持向量机 SVM 等等算法进行。根据课程的学习，决定采用基于tensorflow的深度学习，运用卷积神经网络（CNN）对train里面的图片进行学习，然后使用学习的模型进行测试。

Benchmark Model

这个毕业设计的要求是达到 kaggle Public Leaderboard 前10%，也就是 logloss 要低于 0.06127，最后的训练结果需要小于这个数值。

Evaluation Metrics

kaggle网页上说明，本问题采用的 log loss 进行训练结果的评估：

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

其中：

- n 是test测试集的图片数量
- \hat{y}_i 是可能是dog狗的预测可能性
- y_i 如果图片是狗则为1，如果是猫则为0
- $\log()$ 是自然底数对数函数

Project Design

1. 在 kaggle 网站上下载训练图片 (train) 和测试图片 (test) 。
2. 对 train 图片进行分类处理，然后对图片尺寸统一化。^[1]
3. 使用 Keras 到处特征向量。^[5]
4. 载入特征向量并且合并成一条进行训练。
5. 使用 Keras 构建模型。
6. 训练模型，按照课程的惯例，按照8：2的比例划分为训练集和验证集。
7. 进行 test 图片的预测。

参考文献

[1] Keras中文文档. Application应用.

<http://keras-cn.readthedocs.io/en/latest/other/application/>

[2] 杜客. CS231n课程笔记翻译：反向传播笔记. <https://zhuanlan.zhihu.com/p/21407711>
.Published: 2016-08-21.

[3] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385, 2015.

[4] Francois Chollet Google, Inc. Xception: Deep Learning With Depthwise Separable Convolutions. arXiv:1610.02357v3, 4 Apr 2017.

[5] 优达学城 (Udacity) . 手把手教你如何在Kaggle猫狗大战冲到Top2%.
<https://zhuanlan.zhihu.com/p/25978105>

