

# Machine Learning Engineer

## Nanodegree

### Capstone Proposal 毕业项目开题报告

---

蒋似尧

December, 2017

### 侦测走神司机

---

#### 背景

这是一个采用深度学习解决计算机视觉领域的问题。神经网络的概念起源于上世纪 50 年代，在 80,90 年代产生了很多重要的算法突破。本世纪含有多层神经网络的模型，深度神经网络再次兴起。2007 年李飞飞创立 ImageNet 数据集。2012 年谷歌将深度学习用于猫脸识别和语音识别。2016 年 AlphaGo 战胜李世石。现今深度学习已经成为了机器学习领域最火热的方向之一。

每年都有很多交通事故源于司机走神。Statefarm，美国的一家交通保险公司，

在 [kaggle](#) 上悬赏机器学习算法，用来自动识别开车时走神的司机。通过摄像机拍摄到的司机画面，分辨司机的不同状态，包括安全驾驶，打电话，和其他乘客说话等。这有助于为司机提供更好的保险服务，而且可以用于自动报警，提示司机安全驾驶。

#### 问题描述

算法是通过捕捉到的司机驾驶时的画面，通过深度学习来进行图像分类，辨识司机的状态。司机的状态大致分为 10 类，是一个多分类问题。我们需要构建一个分类模型，输入一张司机的图片，计算出图片中司机处于每种状态的概率。最后

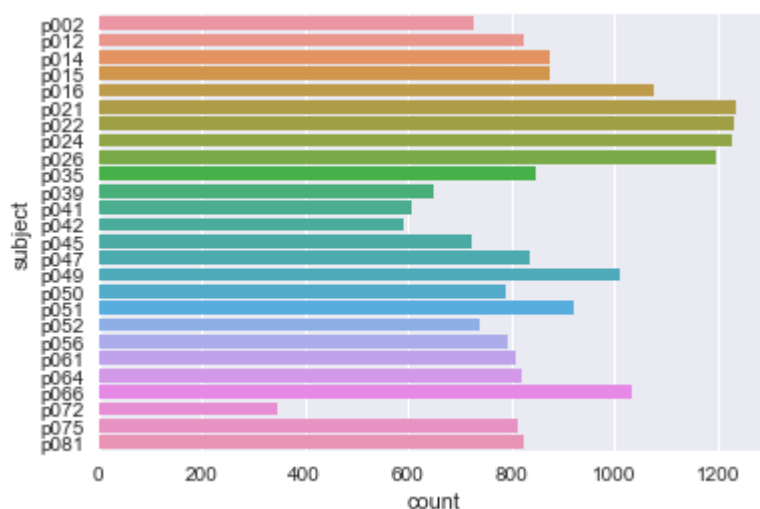
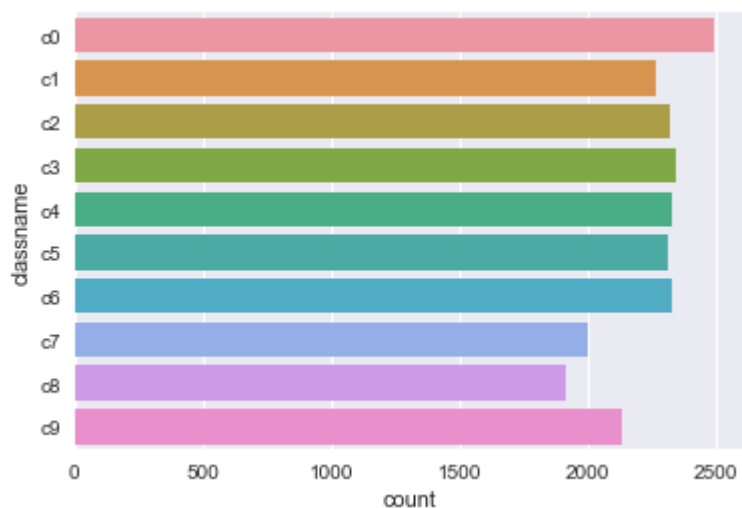
模型的评估采用 **cross entropy loss**，对比测试图片的预测分类概率和实际分类的误差。

## 数据集和输入

[数据集](#)包含测试和训练两部分。训练集 22424 张图片已经分类存放在 10 个文件夹内，文件夹名就是 **label**。一共分为 10 类。具体如下：

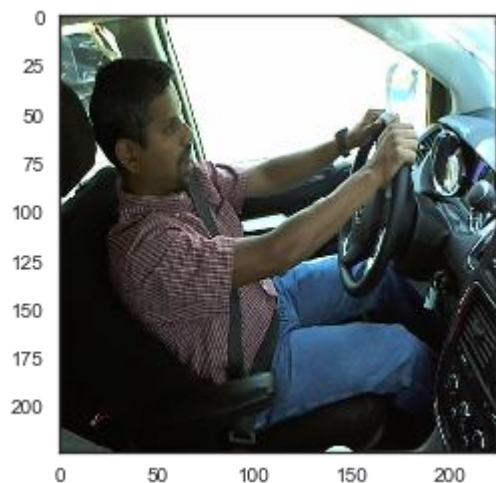
- c0: 安全驾驶
- c1: 右手打字
- c2: 右手打电话
- c3: 左手打字
- c4: 左手打电话
- c5: 调收音机
- c6: 喝饮料
- c7: 拿后面的东西
- c8: 整理头发和化妆
- c9: 和其他乘客说话

每类包含 2000 多张司机驾驶中的 RGB 图片。所有 10 类图片都是对 26 名不同的司机拍摄的，每名司机在每一类文件夹下都有若干张图片。司机的 id 号，对应的图片文件名，以及对应的类别在 **driver\_imgs\_list.csv** 文件中提供。司机 id 和 10 种分类的分布分别如下图，可见分别还是比较均匀的：



值得注意的是测试集图片中的司机和训练集中的不同，也就是 26 名用于训练的司机不会出现在测试集中。这是不同于其他图片分类项目比如猫狗大战的地方。对于 1 名司机建立好的模型，必须能成功的预测另一名没见过的司机的分类。这也是符合实际需求的，在实际侦测中肯定要适用于陌生的司机。为此必须先按照司机 id 分类，在训练和验证中使用不同的司机，这样才能让模型学到在不同的司机身上都能有效分类，而不是只认识这 26 个司机。

另外 OpenCV 通道顺序读取时需要注意，把 BGR 转换成 RGB，才能读出原图如下，否则色彩是不对的：



## 解决方案

这是一个图片分类项目，计划采用迁移学习的方法进行。使用已经在 ImageNet 上训练好的模型和权重，在最后接自己的分类器并训练，得出分类结果。

由于不同的司机在 10 个分类中都有图片，我只能采取自己写的 generator 来进行图片输入和预处理。将 22 个司机的图片作为训练集，乱序输入 generator。

将另外 4 个司机的图片作为验证集。采用 opencv 读取图片，缩放到  $224 \times 224 \times 3$  的图片，然后把分类标签使用 one hot code 编码。设置适合自己内存大小的 batch size，每次读取一个 batch 的图片作为输入。

## 基准模型

这个问题最简单的基准模型就是等概率的预测图片状态属于 10 类中的一类，也就是每类的概率都是 0.1。这个模型没有对数据集进行任何学习，有 10% 的正确率。Sample submission.csv 文件就是做了这样一个预测，提交到 kaggle 后损失函数为 2.3。我的目标是通过学习至少降低损失函数到 0.225，也就是大约 private leaderboard 前 100 名。

## 评价指标

本项目是一个多分类问题，典型的评价指标可以用 cross entropy loss。模型的输出层激活函数是 softmax，也就是 10 类的概率  $p_{ij}$ 。图片的正确分类是  $y_{ij}$ 。i 代表图片数量一共 N 张。j 代表类别数一共 M=10 类。损失函数可以用如下公式表示：

$$logloss = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^M y_{ij} \log(p_{ij})$$

事实上 kaggle 会先做结果的预处理再计算损失。首先会做一步归一化，再采取以下方式避免出现极端情况  $p=0$  或 1：

$$\max(\min(p, 1 - 10^{-15}), 10^{-15}).$$

具体说明引自这里：

<https://www.kaggle.com/c/state-farm-distracted-driver-detection#evaluation>

用这个指标评价基准模型，得到的损失是 2.3，如果我的模型学到东西，应该会小于这个结果。

## 项目设计

本项目使用 OpenCV 库来读取图片，并 resize 到 224\*224\*3 的尺寸。取决于使用的预训练模型，决定是否要进一步预处理。用 inception 或者 xception 的话需要把数据归一化到 (-1,1) 范围内。

Keras 的 ImageDataGenerator 可以作为图片生成器，自动从不同种类的文件夹中读取图片，并且可以实现数据增强功能。但是由于我要按照司机的 id 把不同

的司机分开，所以只能自己写 generator。先按照 driver\_imgs\_list.csv 里的索引把图片根据司机 id 分开，分为训练集（22 名司机）和验证集（4 名司机）。在训练集 generator 中随机选取一个 batch size 的训练集图片名，根据图片的类别标签找到路径并导入图片。将类别标签经过 one hot encode 处理后，和图片数据一起 yield 给 model。验证集的 generator 同理。

本项目使用基于 TensorFlow 后端的 keras 构建模型并训练。计划使用迁移学习，在 keras 提供的经过 ImageNet 预训练的模型后训练自己的全连接分类器。Keras 提供了几种表现比较好的预训练模型，在此我准备选用其中几种进行导入。锁定模型中的层，在最后添加一个 GlobalAveragePooling 层，一个 dropout 层，一个 dense 层。最后输出维度为 10，激活函数采用 softmax。模型编译用 adadelta 优化器，categorical\_crossentropy 损失函数，指标 categorical accuracy 是分类精确度用来参考。构建完模型后，用 fit\_generator 进行训练。

训练完成后，用 generator 将测试集图片导入并进行预测。最后将预测结果写入 csv 文件提交到 kaggle 查看结果。

为了利用类激活图动态可视化训练过程，在训练时采用一个 callback 记录权重的信息。将预训练模型导出的激活图乘以权重并叠加得到类激活图 CAM，然后使用 OpenCv 库进行可视化并制成视频。这部分参考了这篇文章：

<https://zhuanlan.zhihu.com/p/29567314>