

Machine Learning Engineer Nanodegree

Capstone Proposal

梅灵睿

2017 年 8 月 14 日

Proposal

Domain Background

无人驾驶汽车一直都是人工智能领域热议的话题。近年来，随着深度学习技术的发展，无人驾驶汽车也离我们的日常生活越来越接近。自动驾驶汽车的展示系统可追溯至 1920 年代及 1930 年代间，第一辆能真正自动驾驶的汽车则出现于 1980 年代。1984 年，卡内基美隆大学推动Navlab 计划与 ALV 计划；1987 年，梅赛德斯-奔驰与德国慕尼黑联邦国防大学共同推行尤里卡普罗米修斯计划。从此以后，许多大型公司与研究机构开始制造可运作的自动驾驶汽车原型。21 世纪以后，伴随着资讯科技的进步，更是突飞猛进，特斯拉汽车率先推出了特定环境下的自驾车，而汽车工业人士估计 2030 年以前就会量产出现。

无人驾驶汽车需要多种传感器对探测周围环境，其中一项，也是近年来最受关注的热门一项，便是通过摄像头和计算机视觉对周围环境进行感测。卷积神经网络的发展让计算机视觉往前发展了一大步，自 2012 年 AlexNet 赢的 ImageNet 比赛之后，几乎每一个计算机视觉问题的解决方案都有了卷积神经网络的身影，它解决了许多计算机视觉中原本非常困难的问题，比如平移不变性和特征提取等，并且在几乎所有的计算机视觉问题上都有着更好的表现。

Domain Background

我将要利用卷积神经网络预测车辆在道路上行驶时后的转向信号，也就是让车辆能够学会自动转向。由于数据集中的数据已经打好了标签，所以我要利用监督学习训练模型。我将要学习一个模型 f ，输入状态 x ，可以得到转向信号 \hat{y} 。也就是：

$$\hat{y} = f(x|\theta)$$

其中模型输出转向信号 \hat{y} 是模型认为的合理的转向信号， θ 是模型的权重， x 是车辆前方摄像头所采集到的视频图像，即单帧图片。也就是模型输出的转向信号和人类驾驶过程中的转向操作基本一致，能保证车辆在道路上安全平稳地往正确方向行驶。模型 f 通过卷积神经网络实现。我将通过监督学习，通过调整权重，最小化模型输出和数据

集提供的输出的均方误差，从而模型能够学会输出合理的转向信号。模型需要优化的损失函数如下：

$$J(\theta) = \frac{1}{n} \sum_i^n (\hat{y}_i - y_i)^2$$

其中 y 为真实驾驶情形下，对相应 x 所输入的转向信号。

Datasets and Inputs

模型训练基于的是 DeepTesla 数据集，主要包括 tesla 在两种不同驾驶模式（human driving 和 autopilot）下的前置相机录制的视频和车辆的转向控制信号。整个数据集分为十个视频文件和十个相应的 CSV 文件，CSV 文件为行驶过程中的控制信号，初第一个和第二个文件记录的条数是 1500 和 3900 外，其余 8 个文件记录的条数均为 2700。

行驶过程中的控制信号: CSV 格式

ts_micro	frame_index	wheel
1464305394391807	0	-0.5
1464305394425141	1	-0.5
1464305394458474	2	-0.5

其中，`ts_micro` 是时间戳，`frame_index` 是帧编号，`wheel` 是转向角度（以水平方向为基准，+ 为顺时针，-为逆时针）。数据的前面两列皆是规定了 x 的先后顺序（基于时间），而 `wheel` 才是我们所关注的 y 值。我将利用卷积神经网络对汽车行驶过程中的前置摄像头录制的视频作为 x 逐帧进行训练，而每一帧对应的 `wheel` 值则是 y 。

Solution Statement

在对模型训练之前，我还要对数据集进行预处理，包括数据增强和 normalization。

我初步的训练方案是基于 VGG16[1] 修改模型，然后对数据集进行训练。VGG16 网络是牛津大学的 Visual Geometry Group 提出的网络，结构如图 1 所示。我还用利用 keras-vis 对模型进行可视化，包括 activation map 等。单纯地降低损失函数可能还不够，我们还需要知道汽车驾驶是否安全合理，这可能比单纯地降低损失函数更为重要。比如，某些 loss 较小的模型可能会让汽车在某些情况下出现违反交规甚至危险驾驶的行为，而我认为这样的比某些 loss 稍微高一些、仅仅只是行驶不平稳但是相对安全的模型要次一些。所以我认为，评价的时候还要进行视频可视化以供参考，观察车辆是否有危险操作，驾驶是否平稳顺利，是否朝着目的地平稳前进。

对于超参数的调节，我会使用 GridSearch 对大多数合理的超参数取值进行搜索，而合理超参数的取值范围会参考经验取值和各类文献。

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					

图 1: VGG

我将在我的网络中增加正则化、权值衰减 [2]、batch normalization[3] 和 dropout[4] 等操作。在模型训练之前，我还要对数据进行预处理，以便模型能够很好地进行训练。

Benchmark Model

我要将数据集划分为训练集和测试集，训练集占 80%，测试集合占 20%；其中训练集中的 20% 作为验证集。测试集将用于测试模型的泛化能力，而验证集则用于在测试阶段对模型的性能进行评估，以便对超参数和模型结构进行调整。我将要使用 AlexNet[5] 作为基准模型，与我使用的模型进行性能对比。AlexNet 结构如图 2 所示。

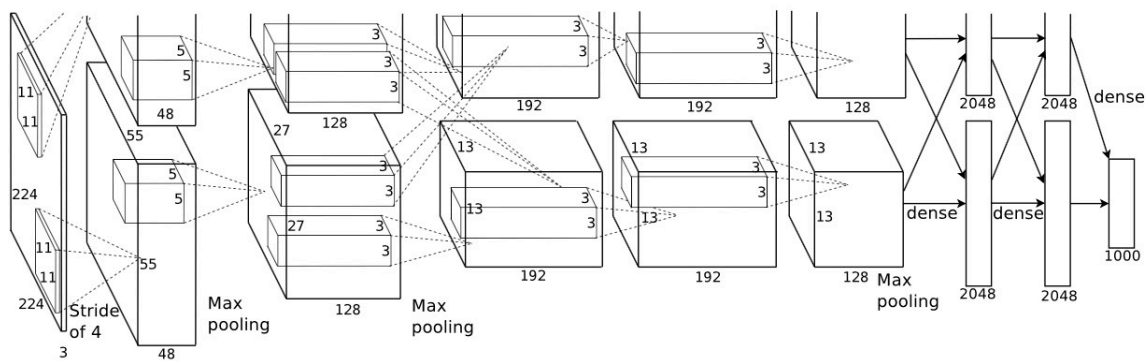


图 2: AlexNet

Evaluation Metrics

由于这不是一个单纯地分类问题，所以模型的评估方法主要是 RMSE:

$$\sqrt{J(\theta)} = \sqrt{\frac{1}{n} \sum_i^n (\hat{y}_i - y_i)^2}$$

DProject Design

Workflow:

- 下载数据，整理数据
- 对部分数据进行可视化
- 划分数据集，包括测试机、验证集和测试集
- 数据预处理。包括对数据进行 normalization，对数据集进行翻转增强，以及裁剪不必要的图片内容等
- 设计神经网络模型。此处要查阅大量文献，如 Bojarski 2016[6]，参考别人的设计
- 训练模型，调整超参数；训练过程中，可能要对模型结构进行反复修改
- 可视化模型结果，再对模型进行调整训练。
- 可视化所有模型结果并整理。

参考文献

- [1] SIMONYAN K, ZISSERMAN A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. CoRR, 2014, abs/1409.1556.
- [2] KROGH A, HERTZ J A. A simple weight decay can improve generalization[C] // ANON. Advances in neural information processing systems. 1992: 950–957.
- [3] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C] // ANON. International Conference on Machine Learning. 2015: 448–456.
- [4] SRIVASTAVA N, HINTON G E, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural networks from overfitting.[J]. Journal of machine learning research, 2014, 15(1): 1929–1958.
- [5] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet Classification with Deep Convolutional Neural Networks.[J]. Annual Conference on Neural Information Processing Systems, 2012: 1106–1114.
- [6] BOJARSKI M, TESTA D D, DWORAKOWSKI D, et al. End to End Learning for Self-Driving Cars[J/OL]. CoRR, 2016, abs/1604.07316. <http://arxiv.org/abs/1604.07316>.