

机器学习纳米学位

i. 问题的定义

● 项目概述

学习完课程五的内容并完成图像分类项目后，对深度学习产生了较为浓厚的兴趣，于是选择计算机视觉中的猫狗大战作为毕业项目。计划通过猫狗大战项目对卷积网络有更加深刻的理解，熟悉深度学习框架的配置和使用。

猫狗大战来自于 Kaggle 举办的比赛项目，提供了真实场景采集的 25000 张猫狗图片作为训练数据以及 12500 张猫狗图片作为测试数据。

此项目属于图像分类问题，需要为测试图像打上猫或者狗的标签。图像分类是计算机视觉领域的核心问题之一，许多研究者对此已有了大量研究并且诞生了许多优秀的算法。目前图像分类已在不同的实际场景得到了应用。

● 问题陈述

给定一组猫与狗标注以后的彩色图像，通过使用卷积神经网络对数据集进行建模与学习，将模型用于新图像的分类，判断图像中是猫还是狗。非结构化数据，比如图像，图像特征的提取较为关键，目前卷积神经网络是处理这类问题的很有效的方法。

猫狗大战属于监督学习，目标是构建基于卷积神经网络的深度学习模型，并使用训练数据集训练模型得到合理参数，最后预测测试数据集中图片包含的是猫还是狗。

● 评价指标

使用 Kaggle 猫狗大战比赛的评价指标，Log 损失来评估模型性能：

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

其中，

n 表示测试集的图片数量；

\hat{y}_i 表示预测图像是狗的概率；

y_i 是 1 表示图像是狗，0 则为猫；

$\log()$ 表示自然对数（底为 e ）；

log 损失越低模型性能越好。

ii. 分析

● 数据的探索

使用 Kaggle 猫狗大战提供的数据集。数据集划分为训练集和测试集。

1. 测试集：

- 包含 25000 图像，猫狗图片分别 12500 张。
- 文件夹中每个图片的标签（cat 或者 dog）都作为文件名的一部分。需将训练图片的标签从文件名中提取出来。
- 图片尺寸不一致，训练集图片尺寸散点图如下：

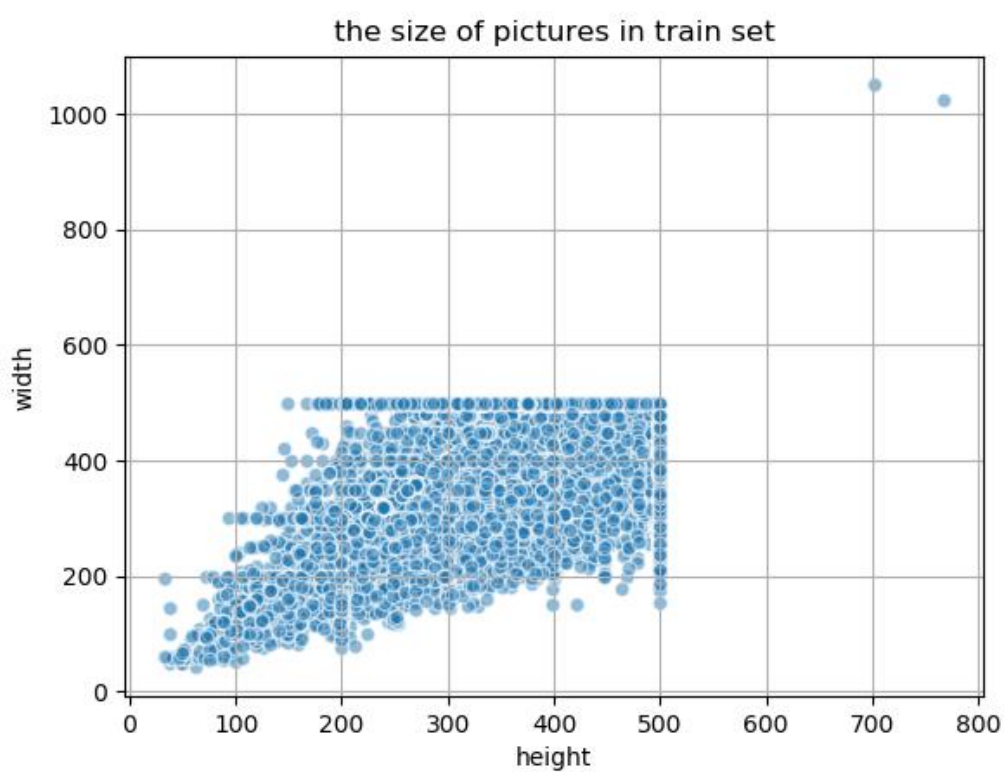


图 1

- 训练集中猫和狗图片都存在异常点，部分异常图片如下：

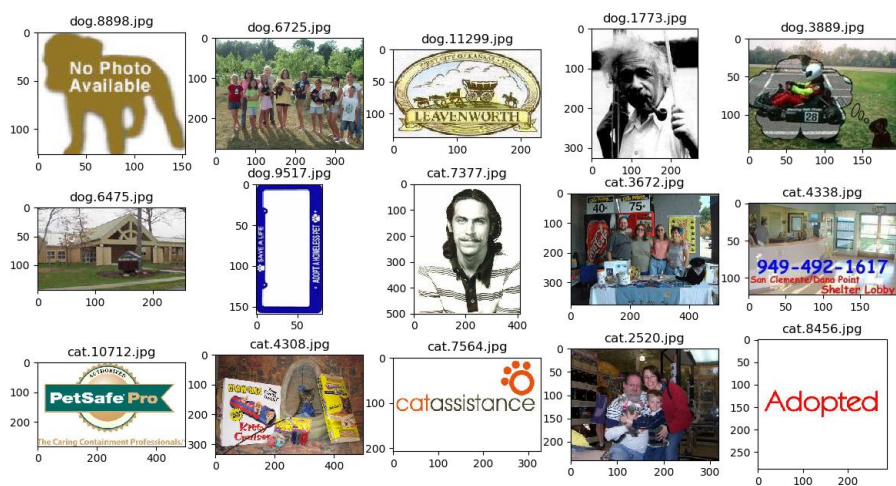


图 2

2. 测试集

- 包含 12500 张图像，以 ID 命名。
- 图片尺寸不一致，测试集图片尺寸散点图如下：

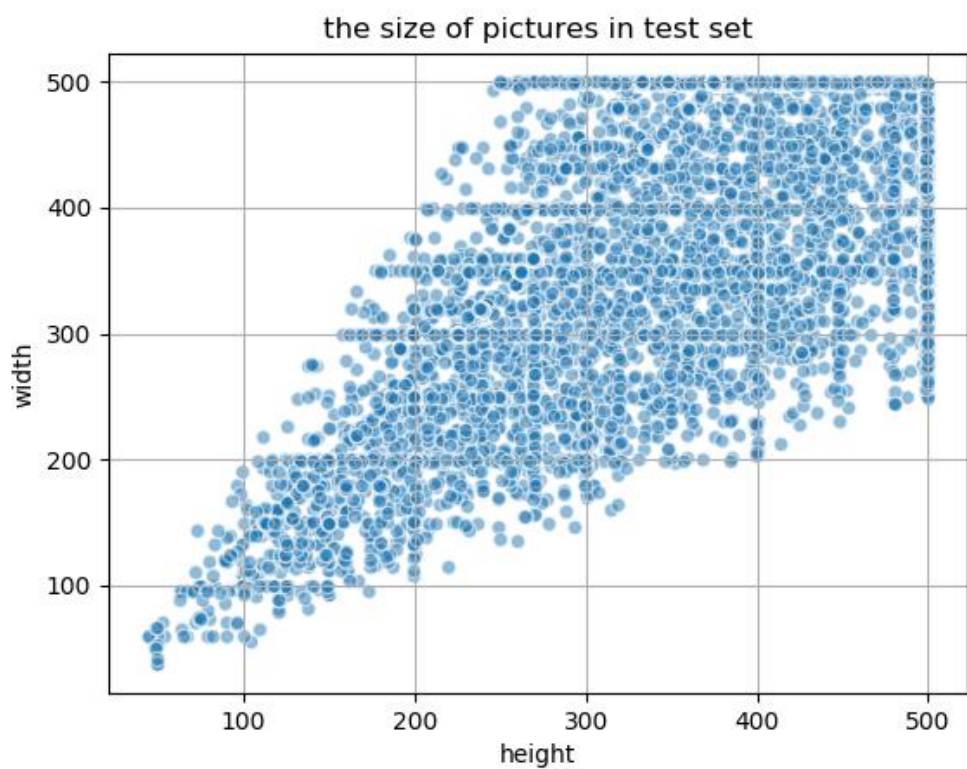


图 3

- 探索性可视化

随机抽取训练集猫狗图片如下：

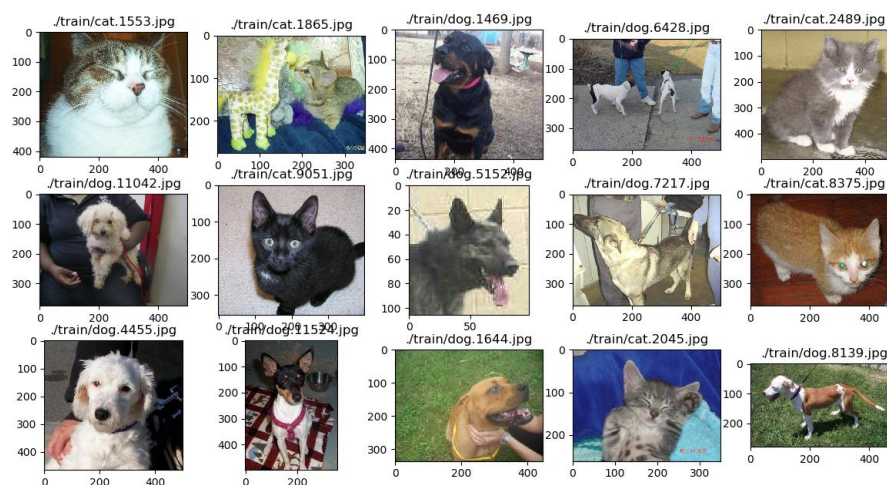


图 4

随机抽取测试集猫狗图片如下

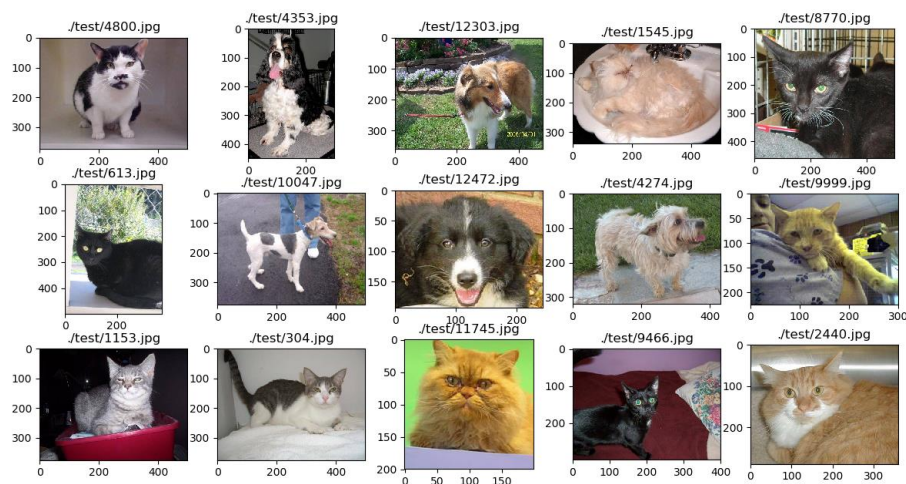


图 5

- 算法和技术

卷积神经网络(CNN)是一种前馈神经网络，它的人工神经元可以响应一部分覆盖范围内的周围单元，对于图像处理有出色表现。CNN 已经成为众多科学领域的研究热点，特别是在图片视频处理等领域，由于该网络避免了对图像的复杂前期预处理，可以直接输入原始图像提取特征，因而得到了更为广泛的应

用。卷积神经网络通常是由输入层、一个或多个卷积层、池化层和全连接层组成。相比较其他前馈神经网络，卷积神经网络需要的参数更少，使之成为一种颇具吸引力的深度学习结构。使用 CNN 将特征抽取转变为隐式地从训练数据中进行学习，权值共享降低了网络的复杂性。CNN 通过卷积池化和激活函数等计算真实值和预测值之间的损失，再依靠反向传播算法将损失值向前反馈来更新层的参数，如此往复直到模型收敛到预期的结果。

本项目中使用到了 Inception V3、Xception 和 Inception ResNet V2 等经典模型。**Inception V3 模型**主要有两个方面的改造，一是引入 Factorization into small convolutions 的思想。将较大的二维卷积拆分为两个较小的一维卷积。比如 3×3 卷积拆成 1×3 卷积和 3×1 卷积。不仅减少了参数，同时增加了一层非线性扩展模型表达能力。论文中指出，这种非对称的卷积结构拆分，其结果比对称地拆分几个相同的小卷积核效果更加明显，可以处理更多，更丰富的空间特征，增加特征多样性。另一方面优化了 Inception Module 的结构，有 35×35 、 17×17 和 8×8 三种不同结构。Inception V3 除了在 Inception Module 中使用分支，还在分支中使用分支（ 8×8 结构中），可以说是 Network In Network In Network。

Xception 模型是对 Inception V3 的另一种改进，主要是采用 depthwise separable convolution 来替换原来 Inception v3 中的卷积操作。在基本不增加网络复杂度的前提下提高了模型的效果。参数数量和 Inception v3 差不多，Xception 未压缩模型，而是提高模型性能。**Inception ResNet V2 模型**是 Inception V3 模型的变体，该模型借鉴了微软 ResNet 论文中的思路。残差连接（Residual connections）允许模型中存在 shortcuts，可以成功地训练更深的神经网络（能够获得更好的表现），这样也能明显地简化 Inception 块。该网络被认为比先前的 Inception V3 还要深一些。Inception ResNet V2 架构的精确度比之前的模型都高。此外，仅仅要求两倍于 Inception V3 的容量与计算能力。

迁移学习(Transfer learning)是一种机器学习的方法，指的是一个预训练的模型被重新用在另一个任务中。把训练好的模型参数迁移到新的模型来帮助新模型训练。卷积神经网络的前面几层提取到的特征更泛化，后面层提取到的特征与原始数据集更相关。项目中使用的模型都已在些 ImageNet 数据集上有着良好表现，接下来的预训练直接使用这些模型参数。

- 基准模型

Kaggle 猫狗大战比赛使用 Logloss 作为评价指标，目前从它的公共排行榜可知第五十名和二十名得分分别为 0.04842、0.04183，最高得分为 0.03302。通过本项目计划将预测结果提交 Kaggle，并且最终 Logloss 得分能突破达到 0.04000。

iii. 方法

- 数据预处理

通过观察训练集图片并进行分析，发现图片尺寸各有差别，存在一些分辨率很低和不包含猫狗的异常图片，需要把这些异常值的图片和分辨率很低图片删除。

使用 Keras 库中 ImageDataGenerator 类 flow_from_directory 方法处理训练集和测试集中猫狗图片。将训练集图片根据文件名分别放入 dog 和 cat 目录。

训练集和测试集中图片尺寸不一致，需要统一。InceptionV3、Xception 和 InceptionResNetV2 模型输入图片尺寸为 299*299。

- 执行过程

从头开始训练卷积神经网络，需要大规模的数据集，而且会占用大量的计算资源。在实际应用中，通常运用迁移学习和微调方法，将 ImageNet 等数据集上训练的现有模型底部特征提取层结果传递给新分类网络。

single_model1.ipynb 代码中直接使用 Keras 提供训练好的 ResNet50 模型。single_model2.ipynb 代码中使用训练好的 InceptionResNetV2 模型。模型都移除了网络顶端的全连接层，而加入全局平均池化层。输入图片经过预处理后，输入预训练模型得到特征向量。Fine-tuning 阶段，使用导出的特征向量训练网络顶端添加的分类网络。这样将预训练模型的稳定和训练的顶端网络全局结合在一起。应用迁移学习，训练得到稳定的顶端全连接层后，可以用于测试集的预测，并将预测结果提交到 Kaggle，基于 ResNet50 模型和 InceptionResNetV2 模型分别的分数为 0.05255 和 0.03861。虽然使用 InceptionResNetV2 模型的分数超过了 0.04000，但和 Kaggle 公共排行榜相比较还有提升空间。

- 完善

以上实现过程是基于单模型导出的特征而展开的迁移学习。基于 InceptionResNetV2 模型的结果优于 ResNet50 模型。查看 Keras 文档可发现如下预训练模型信息。

Model	Size	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth
Xception	88 MB	0.790	0.945	22,910,480	126
VGG16	528 MB	0.715	0.901	138,357,544	23
VGG19	549 MB	0.727	0.910	143,667,240	26
ResNet50	99 MB	0.759	0.929	25,636,712	168
InceptionV3	92 MB	0.788	0.944	23,851,784	159
InceptionResNetV2	215 MB	0.804	0.953	55,873,736	572
MobileNet	17 MB	0.665	0.871	4,253,864	88
DenseNet121	33 MB	0.745	0.918	8,062,504	121
DenseNet169	57 MB	0.759	0.928	14,307,880	169
DenseNet201	80 MB	0.770	0.933	20,242,984	201

图 6

从列表可知 InceptionResNetV2 模型已经是单模型中准确度最高。因此尝试使用多模型融合的提取特征进一步的提升迁移学习模型的性能。根据列表选取准确度前三的模型 InceptionV3、Xception 和 InceptionResNetV2 模型。multi_model1.ipynb 代码中首先使用 InceptionV3 和 Xception 模型导出特征并进行融合，然后类似于单模型 Fine-tuning 阶段，使用导出融合的特征向量训练网络顶端添加的分类网络。来进一步的提升结果。最终提交的结果成绩是 0.03882，此结果并不如基于 InceptionResNetV2 单模型的成绩。于是尝试使用 InceptionV3、Xception 和 InceptionResNetV2 三模型进行导出特征的融合，进一步提升模型性能。果然最后提交 Kaggle 成绩为 0.03709。

iv. 结果

- 模型的评价与验证

Logloss 最低的方案是采用 InceptionV3、Xception 和 InceptionResNetV2 三种模型提取特征并且融合，网络顶端增加 Dropout 和带有 Sigmoid 激活的全连

接层的分类器。Dropout 可以比较有效地减轻过拟合的发生，一定程度上达到了正则化的效果。通过迁移学习和 Fine-tuning 顶端分类器，模型最终达到比较好的效果。在最终方案选取前尝试过单模型和双模型的尝试，参数和效果如下表：

预训练模型	顶端分类器	优化器	训练轮数	得分
ResNet50	Dropout+Dense+Sigmoid	rmsprop	10	0.05255
InceptionResNetV2	Dropout+Dense+Sigmoid	adadelata	10	0.03861
InceptionV3 和 Xception	Dropout+Dense+Sigmoid	adadelata	10	0.03882
InceptionV3、Xception 和 InceptionResNetV2	Dropout+Dense+Sigmoid	adadelata	10	0.03709

表 1

● 合理性分析

首先分别使用 ResNet50 和 InceptionResNetV2 单模型提取特征的方法尝试迁移模型的效果。然后使用 InceptionV3 和 Xception 双模型提取特征并融合的方式尝试效果。最后使用 InceptionV3、Xception 和 InceptionResNetV2 三模型提取特征并融合。

从图 6 的列表可以发现 InceptionResNetV2 模型虽然规模是最大的，但是它的准确度也达到最高。其次是 Xception 和 InceptionV3，最后是 ResNet50。在单模型 ResNet50 和 InceptionResNetV2 对比中，发现 InceptionResNetV2 模型本身的性能更加明显，并且在后面 InceptionV3 和 Xception 双模型特征提取融合后的结果也比不上 InceptionResNetV2 单模型的效果。在机器学习的有监督学习算法中，目标是学习出稳定且在各个方面表现都较好的模型，但实际情况往往不这么理想，有时只能得到多个有偏好的模型。组合多个模型以得到一个更好更全面的强监督模型。虽然使用单模型 InceptionResNetV2 可以打到比较好的效果，但在 InceptionV3 和 Xception 双模型基础上融合 InceptionResNetV2 模型提取的特征，可能会达到更加稳定且全面的效果。当然最后的实验效果也印证了以上推断。

v. 项目结论

● 结果可视化

图 6 到图 7 分别展示了 ResNet50、InceptionResNetV2 单模型，InceptionV3 和 Xception 双模型，InceptionV3、Xception 和 InceptionResNetV2 三模型训练在训练集和验证集上 Loss、Accuracy 随着训练轮数的变化。从图中可知模型的收敛速度较快，在第二轮前就有较大收敛。图 7 和图 8 的对比可知，InceptionV3 和 Xception 双模型在训练集和验证集上的 Loss 比 InceptionResNetV2 更高，并且在训练的后期验证集的 Loss 和训练集的 Loss 差距较大，双模型的泛化能力慢慢走低。从图 9 可知随着在双模型中加入 InceptionResNetV2 模型，三模型的 Loss 变得更加低，并且验证集 Loss 和训练集 Loss 差距降低。有效改善模型泛化能力。

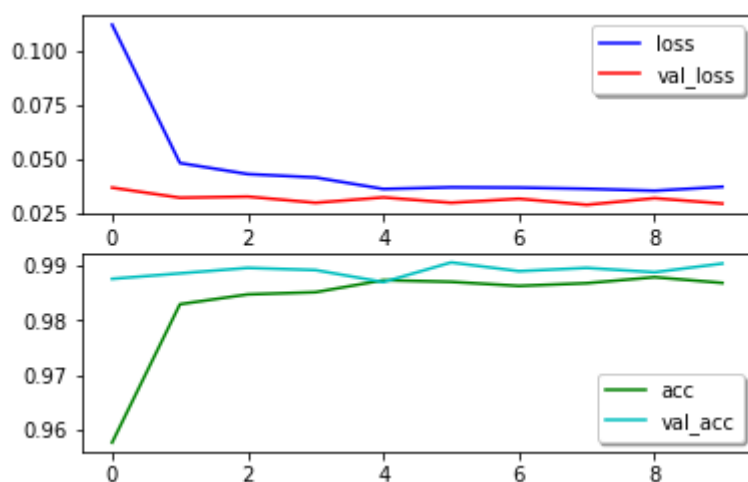


图 6

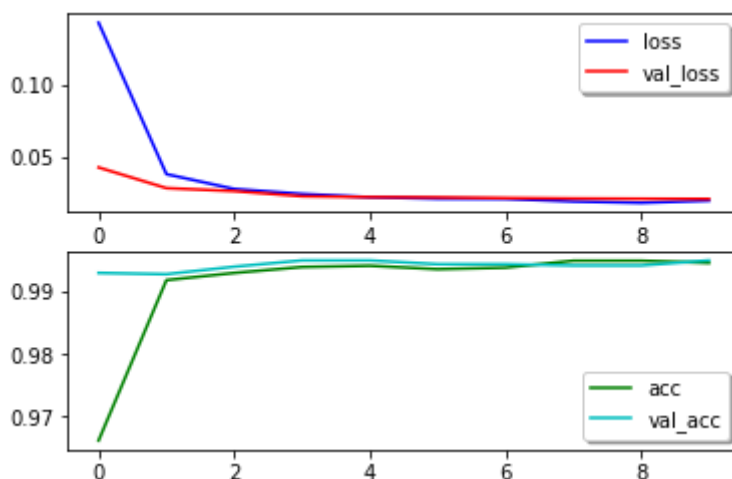


图 7

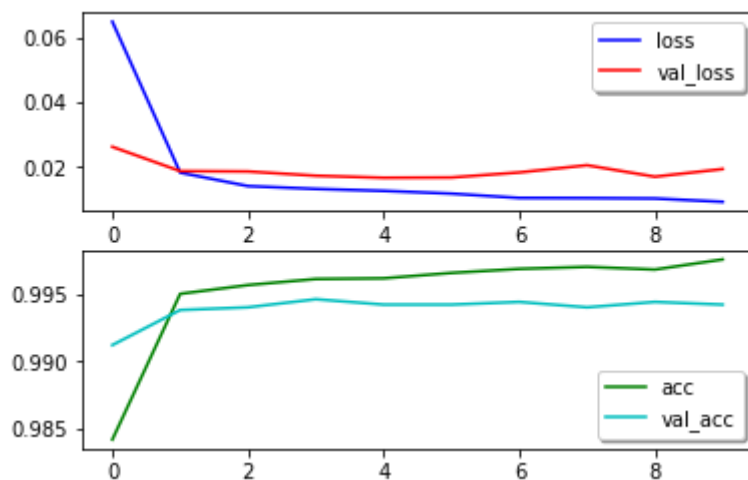


图 8

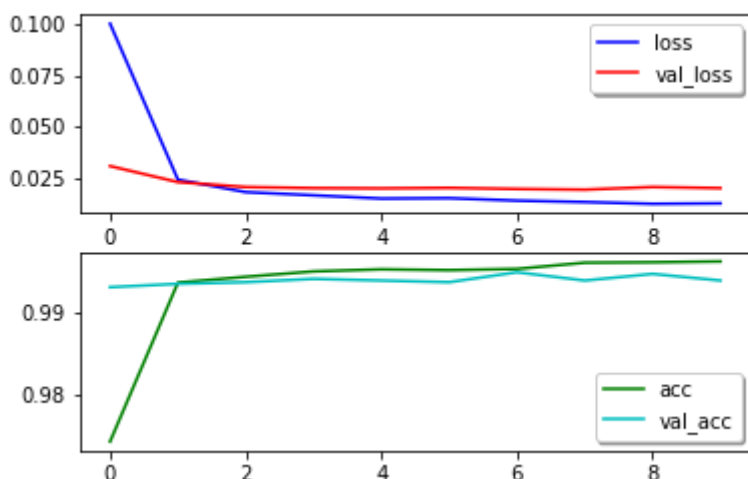


图 9

● 对项目的思考

本项目主要是使用 Keras 框架完成了 Kaggle 猫狗大战项目，并且取得了不错的效果。期间熟悉了 Kaggle 平台的使用和提交流程。对于 Kaggle 实战和交流有着很大的帮助。

本项目主要使用了迁移学习和微调。迁移学习的过程激发了对 ImageNet 数据集和图片分类中先进模型的了解和学习，对于卷积神经网络的发展和演进有着深入理解。Keras 框架的使用让理论与实践有效结合，框架本身的使用对于初学者比较友好，根据开发文档很快可以上手实践。它本身对优秀模型的集成也大大方便了迁移学习的展开。

迁移学习和 Fine-Tuning 是站在巨人肩上优秀的方法。在 ImageNet 上预训练好的卷积网络，删除顶部全连接层，然后将卷积网络作为新数据集的特征提取层。Fine-Tuning，重新训练卷积网络顶部的分类器，当然也还可以通过反向传播算法调整预训练网络的权重。

● 需要做出的改进

对于模型得分的改进还可以从两方面尝试。1) 是模型本身，不局限于 Keras 集成的预训练模型，可尝试使用其他的模型提取特征并且融合。修改顶端分类器结构和参数。2) 是在数据集基础上，进行数据增强。通过数据转换来人为增加样本。数据增强不仅能增加训练数据集的样本量，更能增大网络模型的泛化能力。

参考文献：

- [1] 卷积神经网络, <http://sina.lt/fvRM>
- [2] Karen Simonyan, Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.
- [3] Kaiming He, Xiangyu Zhang, Shaoqing Ren. Deep Residual Learning for Image Recognition, 2015.
- [4] Keras Documentation, <https://keras.io/>
- [5] 迁移学习和微调, <https://zhuanlan.zhihu.com/p/26693647>
- [6] Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, <https://arxiv.org/pdf/1602.07261.pdf>
- [7] Rethinking the Inception Architecture for Computer Vision, <https://arxiv.org/pdf/1512.00567.pdf>
- [8] Xception: Deep Learning with Depthwise Separable Convolutions, <https://arxiv.org/pdf/1610.02357.pdf>