

“猫狗大战”开题报告

项目背景：

近年来，深度学习技术的提出与发展，让人工智能领域再度成为了人们的焦点，各个方向百花齐放，其中“计算机视觉”也取得了不错的进展。如何利用好深度学习技术，自动学习和识别图像的特征，让机器具有人类的“视觉”也成了亿万学者追求的目标。

“猫狗大战”项目来自于 Kaggle 于 2013 年起举办的比赛，属于图像分类领域的热门话题，同时，深度学习的兴起也带动了该领域的长足发展。在深度学习之前，图像分类技术主要利用传统机器学习模型（如 SVM[1]）来学习手工特征（如灰度直方图特征[2]、纹理特征[3]等），近年来，随着深度学习的兴起，图像分类领域也出现了各种各样的比赛，其中比较出名的如 ImageNet，从当初火热的 AlexNet[4]到后来的 VGGNet[5]、InceptionNet[6]到如今 SENet[7]等，在图像分类领域做出了卓越贡献。

问题描述：

“猫狗大战”项目要求根据给定的数据集（包含很多张不重复的猫和狗的图片）和对应的标签（猫或狗）来训练一个端到端的模型，使得模型能够自动识别出未知图片（测试集）是猫还是狗，即二分类问题。此模型可以是深度学习中的常用的神经网络及其变更，而识别效果好坏可以利用 logloss 来衡量。

数据或输入：

项目的数据集分为训练集和测试集。训练集共 25000 张图片，其中猫狗图片共 12500 张，并且给定了类别标签。测试集一共 12500 张没有给定标签。图像均为真实拍摄的图像，但图像大小不一致，拍摄地点也多种多样，已发现的异常图片名为 cat.7377.jpg，它是一个人的照片而不是动物照片。

由于实验中需要设置验证集，因此每次选 20% 作为验证集进行 KFold 交叉验证效果会更好；此外，图片大小不一致，应当 resize 到同一大小，使得神经网络输入层维度可控。另外，原始的图像像素每个通道的值范围在 0-255，这样的数据变化范围太大，不利于模型参数的学习，一般我们要将它归一化到 0-1 之间。

模型在学习过程中，输入归一化后的像素值，输出每个类别的概率，并根据设定好的损失函数及训练集中的标签来学习和调整模型的参数，得到一个二分类模型。

评估标准：

可以使用 **logloss** 来作为评估标准，**logloss** 公式如下：

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

其中 n 表示样本数， y_i 表示第 i 个样本真实标签，其中 **1** 为狗，**0** 为猫， \hat{y}_i 表示第 i 个样本预测为狗的概率。此公式得到的值越小越好。

基准模型：

由于项目要求为 **Kaggle** 前 **10%**，因此我们可以直接使用前 **10%**排名的成绩作为基本阈值，根据其结果，我们可以很好的根据结果进行对比。

项目设计：

1.数据预处理：对于当前数据集，预处理需要做以下工作：

1.1 异常值剔除：首先要找出样本中所有异常值，并把它删掉

1.2 图片大小归一化：将所有图片 **resize** 到同一尺寸，使得模型输出层维度可控制，并且维度不同的图片经过卷积池化操作后维度也不同，无法学习。

1.3 图片像素归一化：原始图像像素值在 **0-255** 之间，数值浮动过大，应当使用最大最小归一化（**max min normalization**），将其映射到 **0-1** 之间。

1.4 验证集划分：为避免模型处于过拟合状态，且考虑参数的选择或调整，数据集中还需要有验证集，这个过程可以使用 **sklearn** 进行 **KFOLD** 交叉验证，验证集大概选择 **10%-20%**左右即可。

另外，测试集除了 **1.4** 以外，均做相同处理。

2.模型搭建：在确认好自己的模型架构后，使用深度学习框架（如 **tensorflow**、**keras** 等）搭建自己的模型。以 **keras** 为例，分为以下过程：

2.1 输入层：输入层指定 **Input** 的 **shape**:**input = Input(shape=[])**

2.2 隐藏层：我们把输出层和输入层中间所有的层称为隐藏层，隐藏层的输入来自于上一层的输出。输出的维度实验中进行适当调整，以找到合适的维度。

2.3 输出层：输出层指定输出维度，这里是 **2** 分类，只需要 **1** 个神经元，经过 **sigmoid** 激活即可得到 **0-1** 之间的概率值。即 **prob_dog = Dense(1,activation = 'sigmoid')(last_hidden_output)**

2.4 生成模型对象：搭建好层以后，需要生成模型对象，指定参数为输入和输出：**model = Model(inputs=input,outputs=prob_dog)**

2.5 编译模型：**compile** 模型时，需要指定训练所需要的一些相关指标，如：损失函数为 **logloss**（以单个 **loss** 为例），评价标准 **metric** 也为

logloss（训练时也可以用准确率 accuracy），优化器可选择 SGD、Adam 等等。

3.模型训练：keras 模型训练时，调用 fit 方法，可以指定验证集的大小作为参数，这样模型训练时都会随机选择相应大小的数据集作为验证集对模型进行测试：model.fit(x,y,validation_split=0.2),同时，额外需要设定的参数还有 batch size 和 epoch。在内存允许情况下，通常选大一点的 batch size，而 epoch 通常不能选太大，会造成过拟合，太小会导致模型没收敛，因此可以通过验证集上的测试来观测 epoch 是否合理，进而进行调整。

4.模型调参：模型一般含有的参数有优化器的学习率、初始化参数的方差等等，还有些模型中设定的额外超参数（如若指定的损失函数带有多项，损失函数中各项权重等等。）在调整一些重要参数时，应当保证其它参数一致，将对比实验结果保存起来，再调整其它参数。当然，也可以通过 sklearn 中的 GridResearch 函数自动调整参数（如果可用）。

5.模型评估：在训练过程中，keras 模型会根据设定好的 metric 对验证集进行评估；在测试过程中，应当使用预处理后的测试集图片作为模型输入，得到每张图片对于狗的分类概率，将其保存起来。

参考文献：

- [1]Fortuna J. ICA feature extraction and support vector machine image classification[J]. <https://www.lap-publishing.com/>, 2005.
- [2]Raghu P P, Yegnanarayana B. Multispectral Image Classification Using Gabor Filters and Stochastic Relaxation Neural Network[J]. Neural Networks, 1997, 10(3):561-572.
- [3]Sung J, Bang S Y, Choi S. A Bayesian network classifier and hierarchical Gabor features for handwritten numeral recognition[J]. Pattern Recognition Letters, 2006, 27(1):66-75.
- [4]Yann LeCun, Gradient-Based Learning Applied to Document Recognition, 1998
- [5]Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.
- [6]Szegedy C, Vanhoucke V, Ioffe S, et al. Rethinking the Inception Architecture for Computer Vision[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2016:2818-2826.
- [7]Hu J, Shen L, Sun G. Squeeze-and-Excitation Networks[J]. 2017.