数据科学项目报告

##毕业项目 **狗分类识别** 丁道华 18188241@qq.com 2018 年 6 月 6 日

一、问题的定义

项目概述

该项目的目标是将狗狗的图片根据品种进行分类。Udacity已经提供模板代码及其他项目所需资源。熟悉图像识别分类流程,使得接受任何用户提供的图片,以识别是狗还是人。

问题陈述

创建图像中检测人脸和小狗后,需要预测品种。熟悉CNN模型,搭建模型,并了解如何使用迁移学习。

评价指标

通过迁移学习,选择<u>VGG-19</u>, <u>ResNet-50</u>, <u>Inception</u>, <u>Xception</u>其中一类,达到85%以上识别率。

二、分析

数据的获取

```
def load_dataset(path):
    data = load_files(path)
    dog_files = np.array(data['filenames'])

dog_targets = np_utils.to_categorical(np.array(data['target']), 133)

return dog_files, dog_targets

# load train, test, and validation datasets

train_files, train_targets = load_dataset('../../../data/dog_images/train')

valid_files, valid_targets = load_dataset('../../../data/dog_images/valid')

test_files, test_targets = load_dataset('../../../data/dog_images/test')
```

检测人脸

我们使用 OpenCV 实现的 <u>Haar 特征级联分类器(Haar feature-based cascade classifiers)</u>来检测图片中的人脸。

检测小狗

我们使用预训练的 ResNet-50 模型来检测图片中的狗狗。我们的第一行代码下载了 ResNet-50 模型,以及在 ImageNet 预训练后的权重信息。ImageNet 是一个庞大且常用的 数据集,用于图像分类和其他视觉任务。ImageNet 包含 1000 万以上的 URL,每个都链接 到包含某个对象的图像,这些对象分成了 1000 个类别。输入一个图片,预训练的 ResNet-50 模型预测图片中包含了什么物体,并返回对应的标签(标签是 ImageNet 内置的一些类别)。

创建分类小狗品种CNN

创建好从图像中检测人脸和小狗的函数后,我们需要预测图像中的小狗品种。

Layer (type)	Output	Shape	Param #
conv2d_1 (Conv2D)	(None,	223, 223, 16)	208
nax_pooling2d_1 (MaxPooling2	(None,	111, 111, 16)	0
conv2d_2 (Conv2D)	(None,	110, 110, 32)	2080
max pooling2d 2 (MaxPooling2	(None,	55, 55, 32)	0
			-
conv2d_3 (Conv2D)	(None,	54, 54, 64)	8256
max_pooling2d_3 (MaxPooling2	(None,	27, 27, 64)	0
global_average_pooling2d_1 ((None,	64)	0
dense_1 (Dense)	(None,	133)	8645
Total params: 19,189.0 Trainable params: 19,189.0			
Non-trainable params: 0.0			

基准模型

识别率达到60%以上。

三、方法

迁移学习特征提取

```
bottleneck_features = np.load('/data/bottleneck_features/DogXceptionData.
npz')
train_Xception= bottleneck_features['train']
valid_Xception = bottleneck_features['valid']
```

```
4 test_Xception = bottleneck_features['test']
```

模型架构

```
### TODO: Define your architecture.

2    Xception_model = Sequential()

3    Xception_model.add(GlobalAveragePooling2D(input_shape=train_Xception.shape[1:]))

4    Xception_model.add(Dense(133, activation='softmax'))

5    Xception_model.summary()
```

训练模型

```
### TODO: Train the model.
checkpointer = ModelCheckpoint(filepath='saved_models/weights.best.Xception.hdf5',
verbose=1, save_best_only=True)

Xception_model.fit(train_Xception, train_targets,
validation_data=(valid_Xception, valid_targets),
epochs=10, batch_size=20, callbacks=[checkpointer], verbose=1)
```

四、结果

识别率达到85%。

合理性分析

在与基准模型进行对比,该结果合理,已经预训练过的模型。

V. 项目结论

该报告,只是根据Udacity学城的已构建框架,对相关流程,CNN搭建,迁移 学习的了解等,对整个图像识别和分类做了了解,在未来遇到实际问题时,可以 找到解决方案的方向。

需要作出的改进

由于对 python和深度学习的不熟悉,代码能力需要进一步的提升,并且可以尝试使用Pytorch等进行尝试。

最后,感谢 udacity,感谢审阅论文的老师。