

深度学习

PaddlePaddle高级技术

DAY09

实现VGG网络

实现VGG网络

概要介绍

主要参数

实现VGG网络

实现VGG网络

实现VGG网络

概要介绍

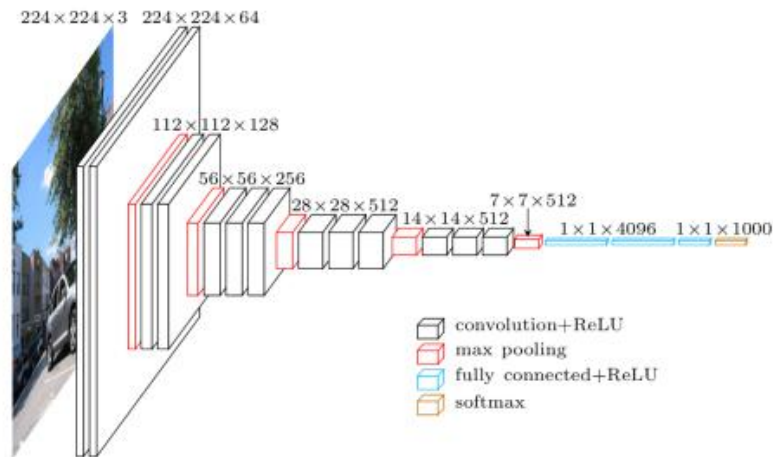
VGG是Visual Geometry Group, Department of Engineering Science, University of Oxford (牛津大学工程科学系视觉几何组) 的缩写, 2014年参加ILSVRC (ImageNet Large Scale Visual Recognition Challenge) 2014大赛获得亚军 (当年冠军为GoogLeNet, 但因为VGG结构简单, 应用性强, 所以很多技术人员都喜欢使用基于VGG的网络)



主要参数

- VGG: 主要参数

- ✓ 网络深度：16~19层
- ✓ 5组卷积-池化层，3个全连接层
- ✓ 三个全连接层，前两层都有4096通道，第三层共1000路及代表1000个标签类别；最后一层为softmax层
- ✓ 所有卷积层有相同的配置，即卷积核大小为 3×3 ，步长为1，填充为1



实现VGG网络

搭建VGG网络

```
def vgg_bn_drop(image, type_size):  
    def conv_block(ipt, num_filter, groups, dropouts):  
        return fluid.nets.img_conv_group(  
            input=ipt, # 具有[N, C, H, W]格式的输入图像  
            pool_size=2,  
            pool_stride=2,  
            conv_num_filter=[num_filter] * groups, # 过滤器个数  
            conv_filter_size=3, # 过滤器大小  
            conv_act='relu',  
            conv_with_batchnorm=True, # 表示在 Conv2d Layer 之后是否使用 BatchNorm  
            conv_batchnorm_drop_rate=dropouts, # 表示 BatchNorm 之后的 Dropout Layer 的丢弃概率  
            pool_type='max') # 最大池化  
  
    conv1 = conv_block(image, 64, 2, [0.0, 0])  
    conv2 = conv_block(conv1, 128, 2, [0.0, 0])  
    conv3 = conv_block(conv2, 256, 3, [0.0, 0.0, 0])  
    conv4 = conv_block(conv3, 512, 3, [0.0, 0.0, 0])  
    conv5 = conv_block(conv4, 512, 3, [0.0, 0.0, 0])  
  
    drop = fluid.layers.dropout(x=conv5, dropout_prob=0.5)  
    fc1 = fluid.layers.fc(input=drop, size=512, act=None)  
  
    bn = fluid.layers.batch_norm(input=fc1, act='relu')  
    drop2 = fluid.layers.dropout(x=bn, dropout_prob=0.0)  
    fc2 = fluid.layers.fc(input=drop2, size=512, act=None)  
    predict = fluid.layers.fc(input=fc2, size=type_size, act='softmax')  
    return predict
```



案例1：利用VGG实现图像分类

- 代码详见：`vgg_classify.py`



中文情绪分析

中文情绪分析

思路及实现

案例目标

数据集介绍

原始数据格式

网络模型介绍

总体步骤

数据预处理

关键代码

训练过程

测试代码

测试结果

思路及实现

案例目标

- 目标：利用训练数据集，对模型训练，从而实现对中文评论语句情感分析。
情绪分为正面、负面两种



数据集介绍

- 中文关于酒店的评论，5265笔用户评论数据，其中2822笔正面评价、其余为负面评价



原始数据格式

1,绝对是超三星标准,地处商业区,购物还是很方便的,对门有家羊杂店,绝对正宗。除了价格稍贵,总体还是很满意的

1,前台服务较差,不为客户着想。房间有朋友来需要打扫,呼叫了两个小时也未打扫。房间下水道臭气熏天,卫生间漏水堵水。

1,1.设施一般,在北京不算好.2.服务还可以.3.出入还是比较方便的.

1,总的来说可以,总是再这里住,公司客人还算满意。就是离公司超近,上楼上班下楼回家

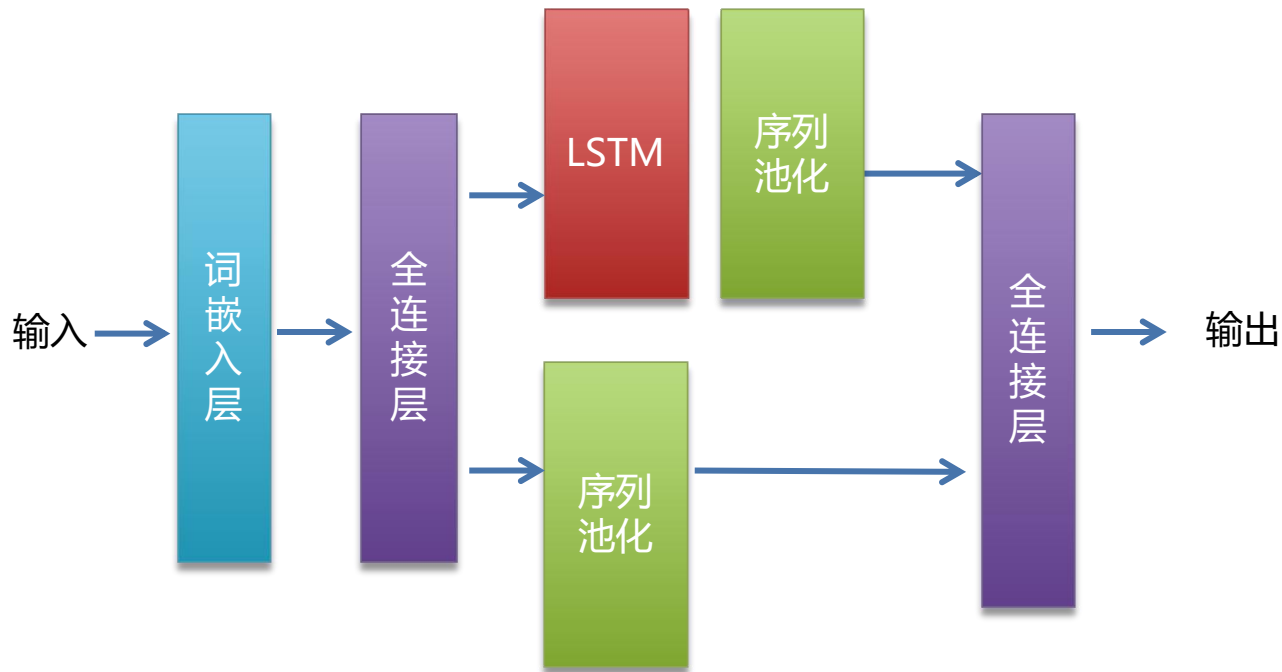
1,房间设施难以够得上五星级,服务还不错,有送水果。

0,标准间太差房间还不如3星的而且设施非常陈旧.建议酒店把老的标准间从新改善.

0,服务态度极其差,前台接待好象没有受过培训,连基本的礼貌都不懂,竟然同时接待几个客人;大堂副理更差,跟客人辩解个没完,要总经理的电话投诉竟然都不敢给。要是没有作什么亏心事情,跟本不用这么怕。



网络模型介绍



总体步骤

- 数据预处理：解析数据文件，编码
- 训练与模型评估
- 输入测试数据，进行预测



数据预处理

- 字典编码

```
{'1': 1, ',': 2, '距': 3, '离': 4, '川': 5, '沙': 6, '公': 7, '路': 8, '较': 9, '近': 10,
```



数据预处理（续）

- 文本编码

```
3,4,5,6,7,8,9,10,2,11,12,7,13,14,15,16,17,2,18,19,12,20,21,22,23,24,2,25,26,27,28,29,30,31,32,33,34,
40,41,42,43,35,44,35,36,45,42,44,43,46,47,48,49,44,50,51,52,53,54,55,56,57,16,58,59,1
60,61,62,63,44,64,65,66,67,68,69,44,70,71,72,16,73,74,75,23,76,77,78,79,80,81,82,83,84,85,86,87,88,8
93,94,95,96,97,98,99,44,16,42,92,100,44,11,101,92,102,103,104,105,106,107,45,67,108,93,94,109,110,11
183,184,185,186,105,2,187,188,189,190,191,78,192,2,193,194,195,46,196,197,198,30,16,199,98,37,190,19
162,23,205,193,44,85,206,23,77,78,207,85,206,23,117,208,101,178,151,152,44,209,210,211,212,213,214,2
117,208,176,176,9,16,58,23,77,78,76,85,222,223,170,224,225,89,44,52,226,227,228,229,41,230,76,35,231
16,58,44,95,106,158,236,222,77,78,186,79,80,12,123,237,164,165,23,238,1
239,240,241,242,44,52,53,45,92,76,119,37,12,234,77,78,44,23,114,46,243,243,23,244,245,159,44,35,36,1
1,76,77,78,176,9,234,44,214,351,352,109,110,101,16,58,44,353,12,35,36,46,219,244,245,159,76,47,76,66
216,240,23,12,180,181,357,36,44,122,358,180,181,44,12,46,219,359,360,282,23,52,53,76,361,362,16,12,3
60,61,45,425,426,44,229,41,72,104,427,44,60,99,45,60,428,35,327,44,227,228,123,429,69,230,430,341,43
85,222,12,287,288,271,23,146,434,42,43,44,162,51,52,53,26,27,16,58,44,127,12,117,208,435,350,436,89,
92,309,111,447,448,449,77,78,83,86,7,335,23,44,122,152,35,36,23,204,208,176,9,450,10,44,127,12,96,83
227,228,331,349,229,41,230,451,16,58,44,35,36,130,131,50,132,44,13,354,355,356,44,182,23,187,188,72,
85,86,93,94,162,51,52,53,101,16,58,44,64,65,142,117,208,260,143,146,260,464,465,44,101,12,466,392,46
117,208,481,436,2,229,41,83,274,30,482,276,204,483,484,485,30,60,61,101,16,58,1
```



关键代码

- 搭建网络

```
105 # 定义长短期记忆网络
106 def lstm_net(ipt, input_dim):
107     ipt = fluid.layers.reshape(ipt, [-1, 1], inplace=True)
108     # 以数据的IDs作为输入
109     emb = fluid.layers.embedding(input=ipt, size=[input_dim, 128], is_sparse=True)
110     # 第一个全连接层
111     fc1 = fluid.layers.fc(input=emb, size=128)
112     # 进行一个长短期记忆操作
113     lstm1, _ = fluid.layers.dynamic_lstm(input=fc1, size=128)
114     # 第一个最大序列池操作
115     fc2 = fluid.layers.sequence_pool(input=fc1, pool_type="max")
116     # 第二个最大序列池操作
117     lstm2 = fluid.layers.sequence_pool(input=lstm1, pool_type="max")
118     # 以softmax作为全连接的输出层, 大小为2, 也就是正面/负面
119     out = fluid.layers.fc(input=[fc2, lstm2], size=2, act="softmax")
120
121     return out
```



训练过程

```
pass_id: 6, batch_id: 40, cost: 0.37168, acc:0.83594
pass_id: 7, batch_id: 0, cost: 0.34798, acc:0.89062
pass_id: 7, batch_id: 20, cost: 0.34973, acc:0.86719
pass_id: 7, batch_id: 40, cost: 0.34333, acc:0.89062
pass_id: 8, batch_id: 0, cost: 0.39762, acc:0.83594
pass_id: 8, batch_id: 20, cost: 0.29967, acc:0.92188
pass_id: 8, batch_id: 40, cost: 0.31670, acc:0.89844
pass_id: 9, batch_id: 0, cost: 0.31817, acc:0.89844
pass_id: 9, batch_id: 20, cost: 0.39566, acc:0.83594
pass_id: 9, batch_id: 40, cost: 0.36204, acc:0.83594
模型训练完成.....
模型保存完成, 保存路径: model/chn_emotion_analyses.model
```



测试代码

```
lods.append(encode_by_dict("周围太吵了", new_dict))
lods.append(encode_by_dict("环境不错,周围比较干净", new_dict))
lods.append(encode_by_dict("干净,卫生,安静", new_dict))
lods.append(encode_by_dict("房间很旧,价格偏贵,服务态度不好", new_dict))
# lods.append(encode_by_dict("不喜欢", new_dict))
lods.append(encode_by_dict("什么垃圾酒店哦", new_dict))

# 获取每句话的单词数量
base_shape = [[len(c) for c in lods]]

# 生成预测数据
place = fluid.CPUPlace()
infer_exe = fluid.Executor(place)
infer_exe.run(fluid.default_startup_program())

tensor_words = fluid.create_lod_tensor(lods, base_shape, place)

infer_program, feed_target_names, fetch_targets = fluid.io.load_inference_model(dirname=model_save_dir,
                                                                                   executor=infer_exe)

# tvar = np.array(fetch_targets, dtype="int64")
results = infer_exe.run(program=infer_program,
                        feed={feed_target_names[0]: tensor_words},
                        fetch_list=fetch_targets)
```



测试结果

1. 周围太吵了
2. 环境不错,周围比较干净
3. 干净,卫生,安静
4. 房间很旧,价格偏贵,服务态度不好
5. 什么垃圾酒店哦

负面: 0.51424, 正面: 0.48576

负面: 0.02990, 正面: 0.97010

负面: 0.16727, 正面: 0.83273

负面: 0.54619, 正面: 0.45381

负面: 0.63514, 正面: 0.36486



案例2：中文情绪分析

- 代码见： `chn_emotion.py`



AIStudio介绍

AIStudio介绍

AIStudio简介

什么是AIStudio

为什么使用AIStudio

AIStudio的使用

项目管理

使用注意事项

AI Studio简介

什么是AIStudio

- AI Studio是百度针对AI学习者的在线一体化开发实训平台
- 内容
 - ✓ AI教程
 - ✓ 示例项目
 - ✓ 经典数据集
 - ✓ 云端存储、计算资源（GPU算力卡）
 - ✓ AI程序编辑、运行环境（预置了PythonPaddlePaddle框架，同时可以自行加载Scikit-Learn , numpy, matplotlib等常用库）



什么是AIStudio（续）

- 平台分为项目、数据集、比赛、教育四大部分
 - ✓ 项目：2000+优质公开项目，覆盖CV、NLP、推荐算法等众多AI热门领域
 - ✓ 数据集：1000+开放数据集，种类多样，支持数据集预览、下载、上传，单次上传容量高达100G
 - ✓ 比赛：新手练习赛、常规赛、高级算法大赛
 - ✓ 教育：合作高校300+，为线上教学提供从教学项目、AI在线实训环境、教学管理



为什么使用AIStudio

- 预置AI程序编辑、运行环境，在线使用，不用单独搭建环境
- GPU算力资源，图形处理速度快
- 学习PaddlePaddle开发平台的优秀环境
- 地址：<https://aistudio.baidu.com/>



AIStudio的使用

项目管理

• 项目主页

Baidu 百度 | AI Studio

项目

数据集

比赛

教育

English

帮助文档

访问飞桨官网

公开项目

我的项目

我的收藏

创建项目

所有专题

所有类型

所有项目标签

综合排序

搜索列表里项目关键词、标签

波士顿房价预测

波士顿房价预测从经典的线性回归 (Linear Regression) 模型入手, 你将使用真实的波士顿房价数据集建立起一个房价预测模型, 并...

AIStudioHelper

Notebook

初级

深度学习

回归

51

17

2019-07-02

AI Studio基本操作(一)

由于AI Studio使用了Notebook为主要编辑器, 因此也引入了一些Python操作, 特别示例说明

小度AIStudio

Notebook

初级

34

12

2019-01-19

AI Studio基本操作(二)

本文主要介绍如何使用PDB来进行代码调试, 可以打印变量和单步执行

小度AIStudio

Notebook

初级

深度学习

3

2

2019-06-11

VisualDL使用Demo项目

VisualDL使用Demo项目, 介绍VisualDL用法, 使用PaddlePaddle内嵌MNIST数据集进行演示

小度AIStudio

Notebook

初级

深度学习

可视化

28

3

2018-12-18

百度推荐学习项目

- Python入门-豆瓣高分电影爬取
- 机器学习入门-鸢尾花分类
- PaddlePaddle入门-房价预测
- 深度学习入门(CV)-手写数字识别
- 深度学习入门(NLP)-文本分类
- 卷积神经网络实践-猫狗分类
- 卷积神经网络实践-人脸识别初探
- 循环神经网络实践-电影评论情感分类
- 深度学习进阶实践(CV)-目标检测
- 深度学习进阶实践(NLP)-机器翻译

飞桨热门模型项目

- 图像类模型-图像分类- GoogleNet
- 图像类模型-图像分类- ResNet
- 图像类模型-图像分类- MobileNet
- 图像类模型-图像分类- VGG



项目管理（续1）

• 创建项目

Baidu 大脑 | AI Studio

项目 数据集 比赛 教育 English 帮助文档 访问飞桨官网

公开项目 我的项目 我的收藏 创建项目

所有权限 所有类型 运行时间排序 搜索列表里项目关键字、标签

鸢尾花分类
鸢尾花分类（四种方法）
Notebook 私有 2019-07-10 未运行

波士顿房价预测2
波士顿房价预测
Notebook 初级 深度学习 回归 私有 2019-07-09 未运行

图像分类-VGG1.0
图像分类-VGG
Notebook 中级 计算机视觉 深度学习 ... 私有 2019-07-09 未运行

PaddleHub 情感分析教程1

百度推荐学习项目

- Python入门-豆瓣高分电影爬取
- 机器学习入门-鸢尾花分类
- PaddlePaddle入门-房价预测
- 深度学习入门(CV)-手写数字识别
- 深度学习入门(NLP)-文本分类
- 卷积神经网络实践-猫狗分类
- 卷积神经网络实践-人脸识别初探
- 循环神经网络实践-电影评论情感分类
- 深度学习进阶实践(CV)-目标检测
- 深度学习进阶实践(NLP)-机器翻译

飞桨热门模型项目

- 图像类模型-图像分类-GoogleNet



项目管理（续2）

- 设置项目属性

The screenshot shows the '创建项目' (Create Project) dialog box in the Baidu AI Studio interface. The dialog is titled '创建项目' and has a close button (X) in the top right corner. It contains the following fields:

- * 项目名称** (Project Name): A text input field with a placeholder '项目名称最多输入40字' (Project name can input up to 40 characters).
- * 项目标签** (Project Tag): A dropdown menu with the placeholder '请选择' (Please select).
- * 配置资源** (Configure Resources): A dropdown menu with the value 'Notebook' selected.
- * 预加载项目框架** (Pre-load Project Framework): A dropdown menu with the value 'PaddlePaddle-v1.5.0' selected.
- * 项目环境** (Project Environment): A dropdown menu with the value 'python 2.7' selected.
- * 项目描述** (Project Description): A text input field with a placeholder '项目描述最多输入100字' (Project description can input up to 100 characters).

At the bottom of the dialog, there is a section for '数据集' (Dataset) with a '+ 添加数据集' (Add Dataset) button and a '创建数据集' (Create Dataset) button. Below this, there are two buttons: '创建' (Create) and '取消' (Cancel).

The background interface shows the Baidu AI Studio sidebar with sections like '公开项目' (Public Projects) and '我的项目' (My Projects). The main content area displays a list of recommended learning projects, including '波士顿房价预测' (Boston Housing Price Prediction), 'AI Studio基本操作(一)' (AI Studio Basic Operation (1)), and 'AI Studio基本操作(二)' (AI Studio Basic Operation (2)).



项目管理（续3）

- 运行、停止、部署项目

我的项目 > 目标检测YOLOv3

目标检测YOLOv3

目标检测

Notebook 中级 计算机视觉 深度学习 私有 Python2 2019-07-19 10:52:23 [删除](#) [修改](#) [设置为公开](#)

版本内容 数据集 在线服务

请选择版本 草稿 2019-10-14 11:35:25

```
In[1] # 解压螺丝螺母数据
!cd data/data6045/ && unzip -qo lslm.zip && unzip -qo lslm-test.
```

运行 停止 部署



项目管理（续4）

- 设置运行环境



项目管理（续5）

• 编辑界面说明

The screenshot shows the JupyterLab interface with several red boxes and arrows pointing to specific components:

- 项目浏览区域** (Project Browser Area): Points to the left sidebar containing the file explorer and environment information.
- 操作菜单** (Action Menu): Points to the top toolbar containing icons for running, saving, and other actions.
- 说明区** (Documentation Area): Points to the top right area containing the 'Notebook' and 'Terminal' tabs.
- 代码编辑区** (Code Editing Area): Points to the main area containing the code editor and the output area.

The interface displays a notebook titled "人脸（水果）识别" (Face (Fruit) Recognition). The left sidebar shows the environment information (环境信息) and the file explorer (文件夹) with a tree structure:

- 名称 (Name)
- 操作 (Action)
- data
- work
- model

The main area shows the code editor with the following code:

```
[3] 1 # 上传数据集fruits.zip
    2 # 解压
    3 !cd data/
    4 !unzip fruits.zip
```

Below the code editor, the output area shows the results of the code execution, including the command prompt and the output of the unzip command.

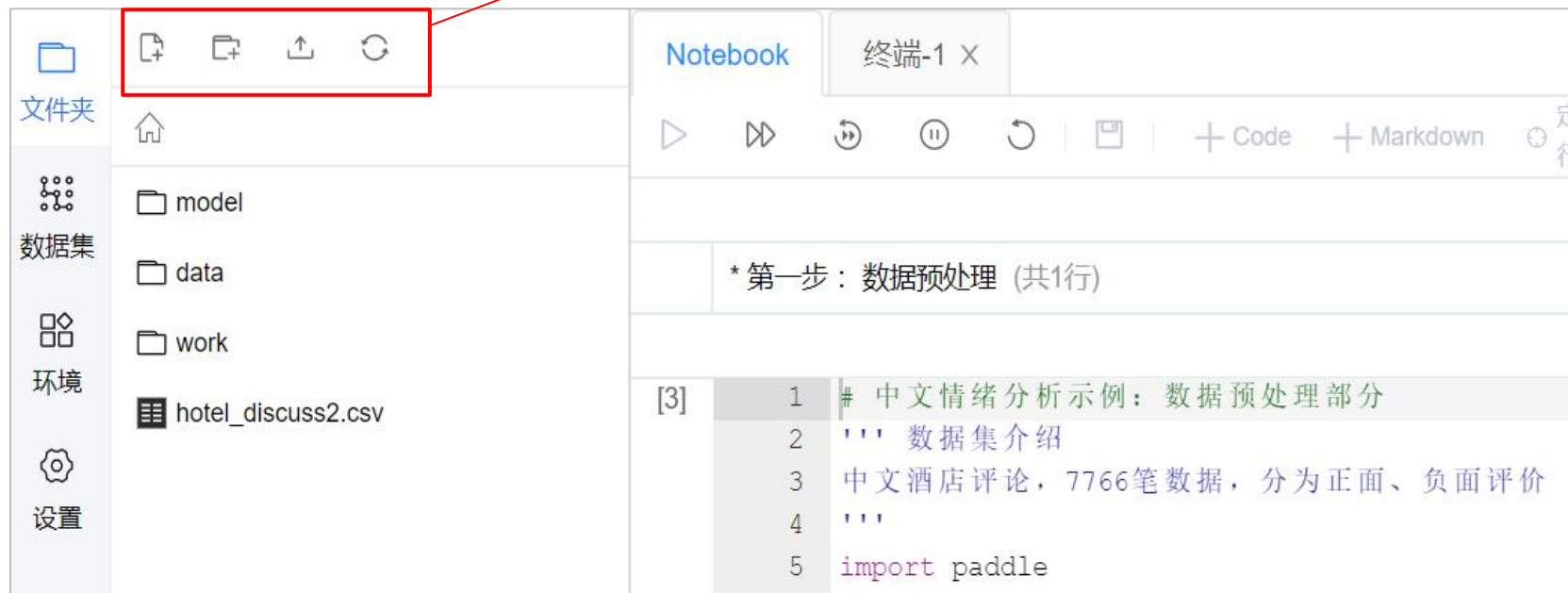
```
[1] 1 # 人脸（水果）识别示例：数据预处理
    2 import paddle.fluid as fluid
    3 import os
    4 import json
    5
    6 # name_dict = {'jiangwen': 0, 'pengyuyan': 1, 'zhangziyi': 2} # 人名字典
    7 # data_root_path = "data/data8205/images/face/" # 数据路径
    8
    9 name_dict = {"apple":0, "banana":1, "grape":2, "orange":3, "pear":4
   10 data_root_path = "data/fruits/" # 数据路径
```



项目管理（续6）

• 文件与目录操作

新建文件、文件夹、上传文件、刷新



使用注意事项

- 编辑器没有PyCharm好用，建议在PyCharm下编辑，然后拷贝到AIStudio执行
- 项目退出重新进入后，文件目录会本清空（如果自建了目录、上传了数据集需要重新操作）
- 不使用的时候，停止项目运行，不然可能消耗算力
- windows下和AIStudio上执行结果可能不一样
- 同一个项目下，不同代码块不要做重复import



今日总结

- VGG网络实现
- 中文情绪分析实现
- AIStudio使用