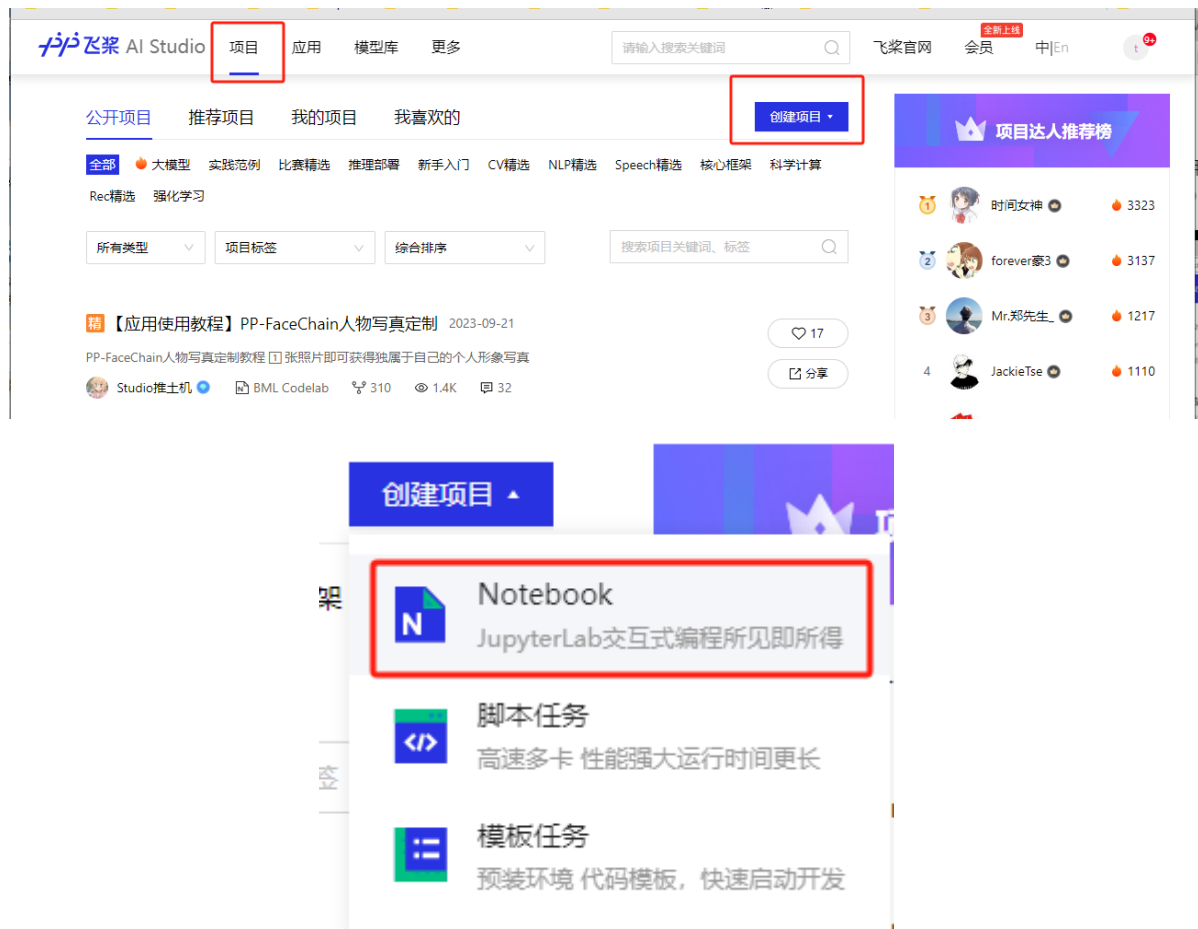


1. 百度AI-studio平台

1.1. 创建项目

搜索百度AI-studio (<https://aistudio.baidu.com/>),选择项目-》创建项目，选择Notebook模式



2、填写项目名称，数据集搜索MINIST数据集 等信息

创建Notebook

×

* 项目名称

手写数字识别

数据集配置

MINIST-官... ×

+ 添加数据集 ?

创建数据集 ?

高级配置 ^

Notebook版本



BML Codelab



AI Studio 经典版

项目框架

PaddlePaddle 2.4.0

请按需选择框架类型，创建后仅支持同类型框架间切换

项目标签

请选择

▼

项目描述

项目描述最多输入100字

3、启动环境，选择免费CPU

飞桨 AI Studio

项目

应用

模型库

更多

请输入搜索关键词

飞桨官网

会员

中|En

10

我的项目 > 手写数字识别

手写数字识别

测试手写数字识别

AI Studio 经典版

Python 2.4.0

Python3

图像分类

2023-10-03 12:00:37

删除 修改 设置为公开

版本内容

数据集

后台任务

启动环境

停止

草稿 2023-10-03 12:00:37

新版Notebook- BML CodeLab上线，点击项目修改可切换为新版进行体验

请选择预览文件

当前Notebook没有标题

In []

查看当前挂载的数据集目录，该目录下的受重置环境后会自动还原

View dataset directory.

This directory will be recovered automatically after resetting environment.

!ls /home/aistudio/data

1.2. 编写手写字体网络训练程序

在Botebook中输入以下代码

```
1  import paddle
2  from paddle.vision.transforms import Normalize
3
4  transform = Normalize(mean=[127.5], std=[127.5], data_format='CHW')
5  # 下载数据集并初始化 DataSet
6  train_dataset = paddle.vision.datasets.MNIST(mode='train',
7  transform=transform)
8
9  # 打印数据集里图片数量
10 #print('{} images in train_dataset, {} images in
11 test_dataset'.format(len(train_dataset), len(test_dataset)))
12
13 # 模型组网并初始化网络
14 lenet = paddle.vision.models.LeNet(num_classes=10)
15
16 # 可视化模型组网结构和参数
17 paddle.summary(lenet, (1, 1, 28, 28))
18
19 # 封装模型，便于进行后续的训练、评估和推理
20 model = paddle.Model(lenet)
21
22 # 模型训练的配置准备，准备损失函数，优化器和评价指标
23 model.prepare(paddle.optimizer.Adam(parameters=model.parameters()),
24               paddle.nn.CrossEntropyLoss(),
25               paddle.metric.Accuracy())
26
27 # 开始训练
28 model.fit(train_dataset, epochs=5, batch_size=64, verbose=1)
29
30 # 保存模型，文件夹会自动创建
31 model.save('./output/mnist')
```

会得到以下输出信息

```

26 # 开始训练
27 model.fit(train_dataset, epochs=5, batch_size=64, verbose=1)
28
29 # 保存模型，文件夹会自动创建
30 model.save('./output/mnist')

```

运行时长: 2分钟3秒430毫秒 结束时间: 2023-10-03 12:30:18

```

-----
Layer (type)      Input Shape      Output Shape      Param #
-----
Conv2D-9          [[1, 1, 28, 28]] [1, 6, 28, 28]    60
ReLU-9            [[1, 6, 28, 28]] [1, 6, 28, 28]    0
MaxPool2D-9       [[1, 6, 28, 28]] [1, 6, 14, 14]    0
Conv2D-10         [[1, 6, 14, 14]] [1, 16, 10, 10]   2,416
ReLU-10           [[1, 16, 10, 10]] [1, 16, 10, 10]   0
MaxPool2D-10      [[1, 16, 10, 10]] [1, 16, 5, 5]     0
Linear-13          [[1, 400]]        [1, 120]           48,120
Linear-14          [[1, 120]]         [1, 84]            10,164
Linear-15          [[1, 84]]          [1, 10]            850
-----
Total params: 61,610
Trainable params: 61,610
Non-trainable params: 0
-----
Input size (MB): 0.00
Forward/backward pass size (MB): 0.11
Params size (MB): 0.24
Estimated Total Size (MB): 0.35
-----

The loss value printed in the log is the current step, and the metric is the average value of previous steps.
Epoch 1/5
step 938/938 [=====] - loss: 0.0125 - acc: 0.9405 - 25ms/step
Epoch 2/5
step 938/938 [=====] - loss: 0.0205 - acc: 0.9784 - 25ms/step
Epoch 3/5
step 938/938 [=====] - loss: 0.0021 - acc: 0.9819 - 26ms/step
Epoch 4/5

```

1.3. 编写测试程序

```

1 import paddle
2 from paddle.vision.transforms import Normalize
3 import numpy as np
4
5 transform = Normalize(mean=[127.5], std=[127.5], data_format='CHW')
6 test_dataset = paddle.vision.datasets.MNIST(mode='test',
7       transform=transform)
8
9 # 模型组网并初始化网络
10 lenet = paddle.vision.models.LeNet(num_classes=10)
11 # 封装模型，便于进行后续的训练、评估和推理
12 model = paddle.Model(lenet)
13
14
15 # 加载模型
16 model.load('./output/mnist')
17 # 从测试集中取出一张图片
18 img, label = test_dataset[0]
19 # 将图片shape从1*28*28变为1*1*28*28，增加一个batch维度，以匹配模型输入格式要求
20 img_batch = np.expand_dims(img.astype('float32'), axis=0)
21
22 # 执行推理并打印结果，此处predict_batch返回的是一个list，取出其中数据获得预测结果
23 out = model.predict_batch(img_batch)[0]
24 pred_label = out.argmax()
25 print('true label: {}, pred label: {}'.format(label[0], pred_label))

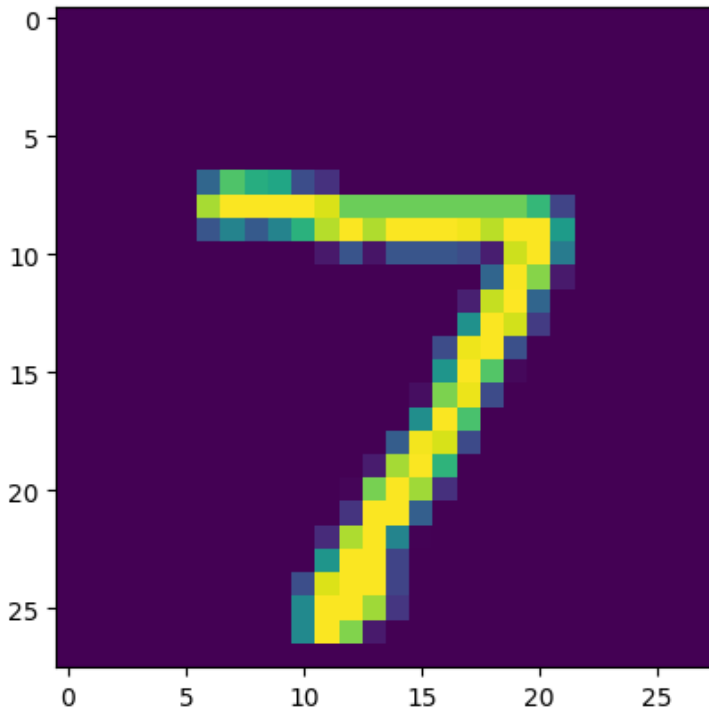
```

```
26 # 可视化图片
27 from matplotlib import pyplot as plt
28 plt.imshow(img[0])
```

如下为实验效果

运行时长: 850毫秒 结束时间: 2023-10-03 12:27:44

```
➤ true label: 7, pred label: 7
<matplotlib.image.AxesImage at 0x7fed894170d0>
```



1.4. 利用自己的图片进行识别

程序功能，利用opencv读取图片，传入网络进行预测

```
1 import paddle
2 from paddle.vision.transforms import Normalize
3 import numpy as np
4 import cv2
5
6 transform = Normalize(mean=[127.5], std=[127.5], data_format='CHW')
7 test_dataset = paddle.vision.datasets.MNIST(mode='test',
8       transform=transform)
9
10 # 模型组网并初始化网络
11 lenet = paddle.vision.models.LeNet(num_classes=10)
12 # 封装模型，便于进行后续的训练、评估和推理
13 model = paddle.Model(lenet)
14
15 # 加载模型
16 model.load('output/mnist')
17 # 从测试集中取出一张图片
18 img, label = test_dataset[0]
```

```

19 # 将图片shape从1*28*28变为1*1*28*28, 增加一个batch维度, 以匹配模型输入格式要求
20 img_batch = np.expand_dims(img.astype('float32'), axis=0)
21
22 # 输入图像处理
23 # img_6 = cv2.imread("my_data/6.png")
24 # img_6 = cv2.cvtColor(img_6, cv2.COLOR_RGB2GRAY)
25 # img_6 = cv2.resize(img_6, (28,28))
26 # img_6 = np.expand_dims(img_6, 0)
27 # img_batch = np.expand_dims(img_6, 0)
28
29
30 img_6 = cv2.imread("my_data/6.png", cv2.IMREAD_GRAYSCALE)
31 img_6 = cv2.resize(img_6, (28, 28))
32 img_6 = 255 - img_6;
33 # # 可视化图片
34 from matplotlib import pyplot as plt
35 # img_6=cv2.cvtColor(img_6,cv2.COLOR_BGR2RGB) # 将颜色通道从BGR改变成RGB
36 # plt.imshow(img_6)
37
38 img_6_np = np.reshape(img_6, (1,28, 28)) # reshape图片数据为模型能识别的格式
39 print(img_6_np.shape)
40 plt.imshow(img_6_np[0], cmap='gray')
41 img_batch = np.expand_dims(img_6_np.astype('float32'), axis=0)
42
43 print(img_batch.shape)
44
45 # 执行推理并打印结果, 此处predict_batch返回的是一个list, 取出其中数据获得预测结果
46 out = model.predict_batch(img_batch)[0]
47 pred_label = out.argmax()
48 print('pred label: {}'.format(pred_label))

```

2. 本地平台部署

环境 Ubuntu20.04

python 3.8.10

百度飞桨官方平台: <https://www.paddlepaddle.org.cn/>

这里根据自己的电脑环境生成安装命令:

快速安装

本地快速安装，开发灵活
推荐有深度学习开发经验、有源代码和安全性需求的开发者使用

飞桨版本	2.5 (推荐, 稳定版)				develop (Nightly build)			
操作系统	Windows		macOS		Linux		其他	
安装方式	pip		conda		docker		源码编译	
计算平台	CUDA12.0	CUDA11.8	CUDA11.7	CUDA11.6	CUDA11.2	CUDA10.2	ROCm 4.0	CPU
安装信息	<div><div><div>• 执行以下命令安装:</div><div><pre>python -m pip install paddlepaddle==2.5.1 -i https://pypi.tuna.tsinghua.edu.cn/simple</pre></div></div></div>							