Paddlehub 踩坑日志

简介

Paddlehub是一款基于Python的迁移学习框架，可以方便快速地测试并部署深度学习任务。 我们可以通过pip命令将paddlehub安装到本地（如果你有好的GPU的话）， 也可以在百度的 AI Studio 平台上使用Paddlehub。在本日志中，我们默认使用AI Studio平台。

AI Studio由百度开发，我们可以忽略它主页上乱七八糟的课程，奖励之类的链接，直奔它的核心功能————项目一栏（稍后会介绍）。

AI Studio的项目基本上是Colab的翻版，它与Colab一样基于Jupyter Notebook，并且计算任务由百度提供的机器远程完成。总的来说，AI Studio可以看作中文版的Colab。当然，与Colab一样，你可以在上面远程使用百度主机提供的GPU以用于加速深度学习训练。

AI Studio 的利弊比较

Pros:

1. 无需代理即可访问，可以将计算任务整夜挂在上面。作为比较，Colab有时会面临VPN断线而导致计算任务必须全部重来的问题。
2. 可以免费试用强力的计算设备（TESLA V100 显卡）。基于AI Stuido的官方政策，Tesla V100显卡在AI Studio上的价格是1算力卡/小时（算力卡具体价格不明）。然而事实上只要每天运行自己的计算任务用户就可以免费获得12算力卡（即12小时的特斯拉显卡使用时长），并且这一时长不清零，可以每日累加。
3. Paddlehub具有良好的硬件加速能力（同一个命名实体标注任务，Paddlehub的执行时间是Colab上使用Kashgari包所需执行时间的几十分之一。当然，理论上你也可以把Paddlehub安装到Colab上）。
4. 界面操作更加方便一些，当然这点见仁见智。

Cons:

1. AI Studio强制绑定Paddlehub，并且工作区（即Jupyter Notebook中）无法安装tensorflow。
2. AI Studio平台由百度开发，使用它可能会带来道德与良心上的谴责。

AI Studio注册

进入AI Studio的官网后，点击右上角即可注册。注意AI Studio可以直接用github账号注册并登录。

领取免费算力卡

在注册成功后，访问链接 https://aistudio.baidu.com/aistudio/questionnaire?activityid=457

这一链接是AI Studio发布的一份调查问卷，在完成后可以免费获得12小时的GPU使用资格。从理论上来说，对于问卷中令你感到不适的部分（如学校，电话等敏感信息）可以通过乱写规避。百度将不会就这些信息进行验证。

新建项目

在注册并登录后点击右上角头像，进入个人中心，在左侧菜单栏选择“项目”一栏的子目录“创建和Fork的项目”，然后再右侧细节中点击“创建项目”，然后根据指引初始化项目环境（例如是脚本还是Notebook，Python版本等）后点击“进入项目”->“启动环境” 即可进入Notebook工作环境。

当然，在进入工作环境的同时AI Studio将想你询问需要的硬件配置，你可以根据你的计算任务选择使用CPU 还是Tesla V100 GPU 加速。

通过Paddlehub训练深度学习模型

在进入Notebook 工作环境后，我们就可以在里面开展深度学习任务了。本段将取自然语言处理中的NER任务（命名实体识别）作为例子。

环境配置

AI Studio的工作环境分左右两栏，左边栏中是本地文件，你可以将需要的数据上传至左边的文件栏。右边则是标准的Notebook 工作环境。

先安装paddlehub到环境中：

```
#注意版本一定要选择1.8.2 ！！  否则将在
#定义学习任务时出现错误
!pip install paddlehub==1.8.2
import paddlehub as hub
```

预训练模型加载

Paddlehub的另一好处在于，该模块内部已经自带各种预训练模型。因此我们将不再需要额外将与训练模型下载到本地。下面的代码通过paddlehub安装中文BERT：

```
model=hub.Module(name="bert_chinese_L-12_H-768_A-12")
```

通过对Module()函数选取不同的name参数，我们可以加载不同的预训练模型。就NLP任务而言，各种预训练模型的参数可以在网站 https://aistudio.baidu.com/aistudio/projectdetail/147009 中找到。

自定义加载数据集

下面介绍Paddlehub的最为劝退的环节：自定义加载数据集。我们可以通过Paddlehub内部的接口获取Paddlehub模块自带的数据。但显然这并不是我们的任务。我们面临的问题在于，如何将自定义的训练/测试/验证 数据集加载到Paddlehub中。

在Paddlehub中，所有的训练，测试与验证数据被要求封装在一个被称为DemoDataset的类的对象中（并且你需要自己写这个类）。因此，你需要写一个DemoDataset类将本地数据加载为DemoDataset对象。此外，Paddlehub要求本地数据保存为.tsv 格式，一种较为少见的格式。

因此要自定义数据集一共分为两步：将原有数据转换成.tsv文件，然后将.tsv文件读入Paddlehub。我们将分别介绍这两个任务的方法（官方的教程写得非常之间略粗糙，以至于需要花大量的时间试错以弄懂这两个任务）

保存为tsv文件

首先，我们将文件保存为.tsv格式。在这里我们仅介绍ner任务中的情形。对于NER 任务而言，Paddlehub不仅要求数据集被保存为.tsv格式，同时要求字符与标签间需要用'\002' 标签作为分隔。即假设有字符串"我爱我家"，我们需要将其转换为"我\002爱\002我\002家"（最后一个不用标）。当然这很简单，通过join函数即可完成。

我们同样要将标签用\002分隔符进行分割。这一操作同样可以用join完成，在此不再赘述。

在完成上述预处理之后，我们将语句与标签一一对应保存在一个具有两列k行的pandas的Dataframe中。这里k是全体数据集的基数。并且两列要分别命名为"text_a"与"label"（当然不这么命名也可以，在后面设置参数的时候选择不读header就可以了）

然后执行以下代码即可将数据集保存为.tsv文件（这里我分别保存了全体数据ming_data，训练集，验证集和测试集train，validate，testing。

```
t.to_csv('/home/aistudio/data/train.tsv',sep='\t',columns=
['text_a','label'],encoding='utf_8_sig',index=None)
v.to_csv('/home/aistudio/data/validate.tsv',sep='\t',columns=
['text_a','label'],encoding='utf_8_sig',index=None)
testing.to_csv('/home/aistudio/data/testing.tsv',sep='\t',columns=
['text_a','label'],encoding='utf_8_sig',index=None)
ming_data.to_csv('/home/aistudio/data/dataset.tsv',sep='\t',columns=
['text_a','label'],encoding='utf_8_sig',index=None)
```

当然你还可以将测试集也保存在这一目录下，测试集只需一列数据即可（Paddlehub将自动识别）。

定义DemoDataset

然后，我们定义DemoDataset类。

```
from paddlehub.dataset.base_nlp_dataset import BaseNLPDataset


class DemoDataset(BaseNLPDataset):
    """DemoDataset"""
    #构造函数
    def __init__(self):
        # 数据集存放位置，根据你的情况调整
        self.dataset_dir = "/home/aistudio/data/"
        super(DemoDataset, self).__init__(
        base_path=self.dataset_dir,
        train_file="train.tsv",
        dev_file="validate.tsv",
        test_file="testing.tsv",
        # 如果还有预测数据（不需要文本类别label），可以放在predict.tsv
        predict_file="predict.tsv",
```

```
            train_file_with_header=True,
            dev_file_with_header=True,
            test_file_with_header=True,
            predict_file_with_header=True,
            # 数据集类别集合
            label_list=["O",
        "B-date-reign", "I-date-reign",
        "B-date-year", "I-date-year",
        "B-office-voa", "I-office-voa",
        "B-office-title", "I-office-title",
        "B-place-placename", "I-place-placename"])
    #初始化数据集
    dataset = DemoDataset()
```

上面这段代码将train，testing和validate以及预测集封装在了dataset对象中。

定义训练任务

自定义好数据之后即可开始定义训练任务。这段就没什么复杂的了，挖个坑等会写。。。 如果有迫切需要的话可以参考https://aistudio.baidu.com/aistudio/projectdetail/147009

在定义并执行训练任务后我们就可以开始训练了。训练结束后如何进行模型诊断，如何predict我还没有完全搞清楚（主要是模型诊断）。容我再研究一下

模型诊断，超参数选择与预测

To be continued

结语

总而言之，Paddlehub（基于AI Studio）算是一款好用快速实惠的工作平台。大家的代码在Colab上跑不出来的话（或者跑得很慢）可以去AI Studio试一下Paddlehub，速度也许会快不少。使用Paddlehub会遇到的坑基本都写在这个文件里了。如果你在使用的过程中有什么新的踩坑体会，欢迎编辑这份文档，为其添加新的内容（本文档已经push到了github中CBDB的页面上）