博学谷
www.boxuegu.com
黑马程序员线上品牌

# 基于BERT+PET方式数据预处理介绍

一样的教育，不一样的品质

博学谷
www.boxuegu.com
黑马程序员线上品牌

# 目录

Contents

博学谷
www.boxuegu.com
黑马程序员线上品牌

01

# 查看项目数据集

# 数据集介绍

数据存放位置：/Users/***/PycharmProjects/llm/prompt_tasks/PET/data

data文件夹里面包含4个txt文档，分别为：train.txt、dev.txt、prompt.txt、verbalizer.txt

# train.txt

train.txt为训练数据集，其部分数据展示如下

| | |
|---|---|
| 水果 | 脆脆的，甜味可以，可能时间有点长了，水分不是很足。 |
| 平板 | 华为机器肯定不错，但第一次碰上京东最糟糕的服务，以后不想到京东购物了。 |
| 书籍 | 为什么不认真的检查一下， 发这么一本脏脏的书给顾客呢！ |
| 衣服 | 手感不错，用料也很好，不知道水洗后怎样，相信大品牌，质量过关，五星好评！！！ |
| 水果 | 苹果有点小，不过好吃，还有几个烂的。估计是故意的放的。差评。 |
| 衣服 | 掉色掉的厉害，洗一次就花了 |

train.txt一共包含63条样本数据，每一行用`\t`分开，前半部分为标签(label)，后半部分为原始输入（用户评论）。

如果想使用自定义数据训练，只需要仿照上述示例数据构建数据集即可。

# dev.txt

dev.txt为验证数据集，其部分数据展示如下

| | |
|---|---|
| 书籍 | ”一点都不好笑,很失望,内容也不是很实用” |
| 衣服 | 完全是一条旧裤子。 |
| 手机 | 相机质量不错，如果阳光充足，可以和数码相机媲美．界面比较人性化，容易使用．软件安装简便 |
| 书籍 | 明明说有货，结果送货又没有了。并且也不告诉我，怎么评啊 |
| 洗浴 | 非常不满意，晚上洗的头发，第二天头痒痒的不行了，还都是头皮屑。 |
| 水果 | 这个苹果感觉是长熟的苹果，没有打蜡，不错，又甜又脆 |

dev.txt一共包含590条样本数据，每一行用`\t`分开，前半部分为标签(label)，后半部分为原始输入（用户评论）。

如果想使用自定义数据训练，只需要仿照上述示例数据构建数据集即可。

# prompt.txt

prompt.txt为人工设定提示模版，其数据展示如下

```
这是一条{MASK}评论：{textA}。
```

其中，用大括号括起来的部分为「自定义参数」，可以自定义设置大括号内的值。

示例中 {MASK} 代表 [MASK] token 的位置，{textA} 代表评论数据的位置。

你可以改为自己想要的模板，例如想新增一个 {textB} 参数：

```
{textA}和{textB}是{MASK}同的意思。
```

# verbalizer.txt

- verbalizer.txt 主要用于定义「真实标签」到「标签预测词」之间的映射。在有些情况下，将「真实标签」作为[MASK]去预测可能不具备很好的语义通顺性，因此，我们会对「真实标签」做一定的映射。
- 例如

```
"中国爆冷2-1战胜韩国"是一则[MASK][MASK]新闻。        体育
```

- 这句话中的标签为「体育」，但如果我们将标签设置为「足球」会更容易预测。
- 因此，我们可以对「体育」这个 label 构建许多个子标签，在推理时，只要预测到子标签最终推理出真实标签即可，如下：

```
体育 -> 足球,篮球,网球,棒球,乒乓,体育
```

# verbalizer.txt

项目中标签词映射数据展示如下

| | |
|---|---|
| 电脑 | 电脑 |
| 水果 | 水果 |
| 平板 | 平板 |
| 衣服 | 衣服 |
| 酒店 | 酒店 |
| 洗浴 | 洗浴 |
| 书籍 | 书籍 |
| 蒙牛 | 蒙牛 |
| 手机 | 手机 |
| 电器 | 电器 |

- verbalizer.txt 一共包含10个类别，上述数据中，我们使用了1对1
- verbalizer，如果想定义一对多的映射，只需要在后面用","分割即可，
  eg：水果　　苹果,香蕉,橘子
- 若想使用自定义数据训练，只需要仿照示例数据构建数据集

博学谷
www.boxuegu.com
黑马程序员线上品牌

02

# 编写Config类项目文件配置代码

# 项目文件简介

## 代码路径

/Users/***/PycharmProjects/llm/prompt_tasks/PET/pet_config.py

## config文件目的

配置项目常用变量，一般这些变量属于不经常改变的，比如：训练文件路径、模型训练次数、模型超参数等等

# 具体代码实现

```
# coding:utf-8
import torch
import sys
print(sys.path)

class ProjectConfig(object):
    def __init__(self):
        # 是否使用GPU
        self.device = 'cuda:0' if torch.cuda.is_available() else
'cpu'
        # 预训练模型bert路径
        self.pre_model = '/home/prompt_project/bert-base-
chinese'
        self.train_path =
'/home/prompt_project/PET/data/train.txt'
        self.dev_path = '/home/prompt_project/PET/data/dev.txt'
        self.prompt_file =
'/home/prompt_project/PET/data/prompt.txt'
        self.verbalizer =
'/home/prompt_project/PET/data/verbalizer.txt'
        self.max_seq_len = 512
        self.batch_size = 8
        self.learning_rate = 5e-5
        # 权重衰减参数（正则化，抑制模型过拟合）
        self.weight_decay = 0
        # 预热学习率(用来定义预热的步数)
        self.warmup_ratio = 0.06
        self.max_label_len = 2
        self.epochs = 50
        self.logging_steps = 10
        self.valid_steps = 20
        self.save_dir =
'/home/prompt_project/PET/checkpoints'


if __name__ == '__main__':
    pc = ProjectConfig()
    print(pc.prompt_file)
    print(pc.pre_model)
```

博学谷
www.boxuegu.com
黑马程序员线上品牌

03

# 编写数据处理相关代码

# 代码介绍

代码路径：/Users/***/PycharmProjects/llm/prompt_tasks/PET/data_handle.

data_handle文件夹中一共包含三个py脚本：template.py、data_preprocess.py、data_loader.py

# template.py

目的：构建固定模版类，text2id的转换

导入必备工具包

```
# -*- coding:utf-8 -*-
from rich import print # 终端层次显示
from transformers import AutoTokenizer
import numpy as np
import sys
sys.path.append('..')
from pet_config import *
```

定义HardTemplate类

# data_preprocess.py

目的：将样本数据转换为模型接受的输入数据

导入必备的工具包

```
from template import *
from rich import print
from datasets import load_dataset
# partial：把一个函数的某些参数给固定住（也就是设置默认值），返回一个新的函数，调用这个新函数会更简单
from functools import partial
from pet_config import *
```

定义数据转换方法convert_example()

# data_loader.py

目的：定义数据加载器

导入必备的工具包

```
# coding:utf-8
from torch.utils.data import DataLoader
from transformers import default_data_collator
from data_preprocess import *
from pet_config import *

pc = ProjectConfig() # 实例化项目配置文件
tokenizer = AutoTokenizer.from_pretrained(pc.pre_model)
```

定义获取数据加载器的方法get_data()

博学谷
www.boxuegu.com
黑马程序员线上品牌

# Thanks!

扫码关注博学谷微信公众号