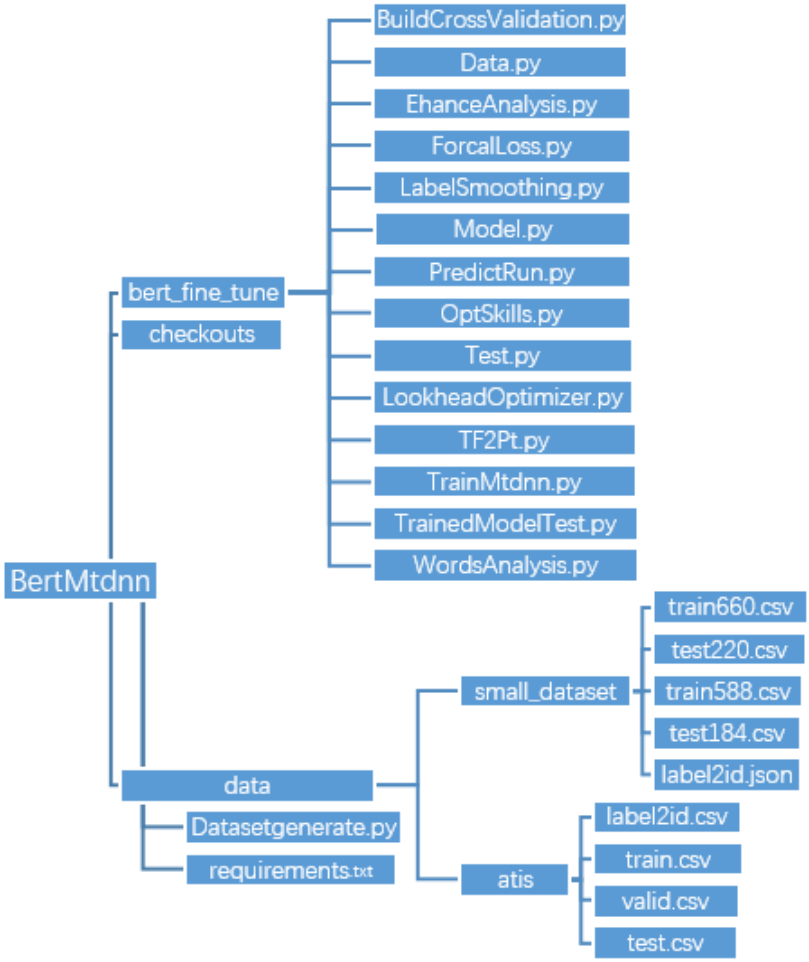
BERT 项目源码运行命令

一、 小数据集场景

在小数据集场景下，源码对应 **BERT_small_dataset**，代码结构如下：



运行环境： **py36**（这里已经将原来的 **py36** 中的 **apex** 卸载，存在冲突）。

文件拷贝：

（1）需将整个文件包拷贝到 **chongshi** 文件夹中。

（2）但 **PretrainedModels** 文件夹中包含了 **bert_base_chinese** 文件夹，该文件夹中含有 **bert-base-chinese** 的模型文件，文件较大，不需拷贝，只需要将 **chongshi/bert_base_chinese** 拷贝到该路径下即可。

（3）**TrainedModels** 路径中，包含重师方已经训练好的模型，文件较大，可

以不拷贝，需将之前拷贝到马上服务器中的小数据集的模型（原文件夹名 PretrainedModels）拷贝到该路径下，即可。

训练与测试：

首先进入 **bert_fine_tune** 中：

（1）激活 py36 环境：source py36/bin/activate

（2）训练 MTDNN：执行 **python TrainMtdnn.py** 即可。命令行参数如下：

参数名	作用	参数设置
seed	随机种子	777
no_cuda	有无使用 GPU	False
loss_function	损失函数	cross_entropy
device_id	GPU id	0,1,2,3,4,5…
do_lower_case	英文大小写转换	True
bert_model	模型名字或路径	../PretrainedModels/bert_base_chinese
local_rank	多卡并行与否	-1
warmup_proportion	Warm up	0.1
classifier_model	可以不用	默认即可
max_seq_length	句子最大长度	70
eval_batch_size	评估批次大小	184
train_batch_size	训练批次大小	16
learning_rate	学习率	5e-5
num_train_epochs	训练轮数	55
train_data_dir	训练集路径	../data/Small/train588.csv

test_data_dir	**测试集路径**	**../data/Small/test184.csv**

若需修改参数设置，如修改 gpu id，测试数据路径，执行以下代码即可：

python TrainMtdnn.py --device_id=3 --test_data_dir=''

(3)**训练单模型 BERT**：执行 **python TrainBert.py** 即可。命令行参数与 MTDNN 的参数相同，具体参数配置见结题报告。

(4) 测试之前的效果，执行 **python PredictRun.py** 即可。命令行参数如下：

参数名	作用	参数设置
seed	随机种子	777
no_cuda	有无使用 GPU	False
device_id	GPU id	0,1,2,3,4,5…
do_lower_case	英文大小写转换	True
classifier_model	可以不用	默认即可
max_seq_length	句子最大长度	70
eval_batch_size	评估批次大小	184
test_type	小或公开数据集	Small，open
pretrained_model	已训练模型路径	见下方

pretrained_model 参数为训练好的模型路径，这里具体的路径包括：

../TrainedModels/Small/besides/ mtdnn/Rcnnbert.bin

../TrainedModels/Small/besides/ mtdnn/Bert_base.bin

../TrainedModels/Small/besides/ mtdnn/Bert_pad.bin

../TrainedModels/Small/besides/ mtdnn/Textcnnbert.bin

../TrainedModels/Small/besides/ mtdnn/Textrnnbert.bin

../TrainedModels/Small/besides/Bert_base.bin

采用命令行进行测试，执行以下代码即可：

python PredictRun.py --test_type='small' --pretrained_model=

' ../TrainedModels/Small/besides/ mtdnn/Rcnnbert.bin'

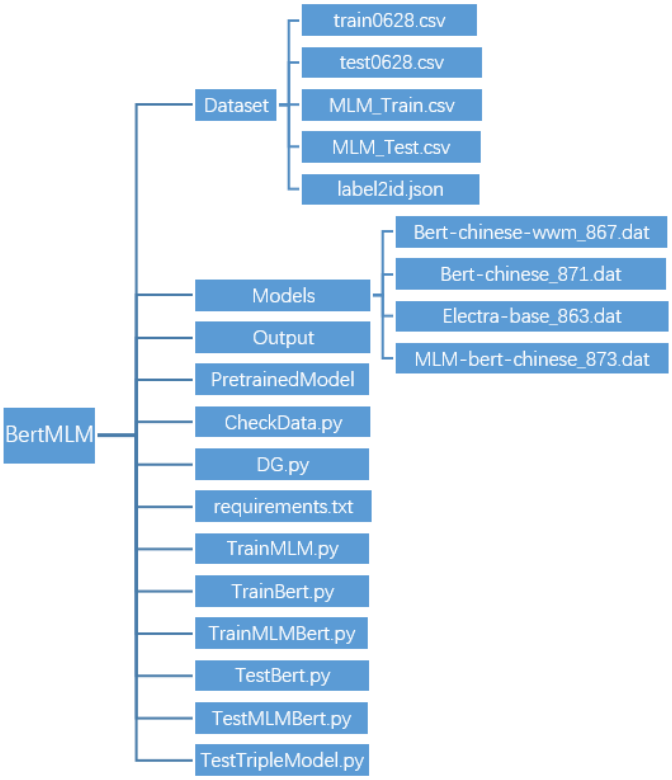
若需采用公司的数据集进行测试，需执行两步：

a) 将待测试 excel 文件放于 **Bert_small_dataset/**下，命名为 **test.xlsx**，执行：**Python DelDataGenerate.py**

b) 再执行：**python PredictRun.py --test_type='small' --pretrained_model=' ../TrainedModels/Small/besides/mtdnn/Rcnnbert.bin'**

二、 大数据集场景

大数据集场景下，源码对应 **BERT_large_dataset**，代码结构如下：



运行环境：

py36_small（py36_small 即为原来罗总的 py36 环境，**未卸载 apex**）；

文件拷贝：

（1）需将整个文件包拷贝到 chongshi 文件夹中。

（2）但 **PretrianedModel** 文件夹中包含了多个预训练模型，文件较大，故可直接将**原来**的 **chongshi/BERT_large_dataset/PretrianedModel** 文件夹拷贝过来，同时将 **bert_base_chinese** 拷贝到 **PretrianedModel** 路径下，即可。

（3）**Models** 文件夹中包含了多个训练好的模型，文件较大，故可直接将**原来**的 **chongshi/BERT_large_dataset** 中的 **Models** 文件夹拷贝过来。

训练与测试：

（1） **激活 py36_small 环境：source py36_small/bin/activate**

（2） 执行 **python TestTimeCost.py** 即可测试在 1000 条数据下，单条数据 cpu 的测试时间。**建议最先执行该步，即可得到测试时间。**

（3）**训练单模型 BERT**：执行 python TrainBert.py，命令行参数如下：

参数名	作用	参数设置
seed	随机种子	777
device_id	GPU id	0,1,2,3,4,5…
bert_model	模型名字或路径	../PretrianedModel/bert_base_chinese
max_seq_length	句子最大长度	40
eval_batch_size	评估批次大小	200
train_batch_size	训练批次大小	64
learning_rate	学习率	2e-5

train_epochs	训练轮数	15
train_data_dir	训练集路径	DataSet/train0628.csv
test_data_dir	测试集路径	DataSet/test0628.csv

若需修改参数设置，如修改 gpu id，测试数据路径，执行以下代码即可：

python TrainBert.py --device_id=3 --train_data_dir='' --test_data_dir=''

（4）两阶段 fine tune 训练：执行 sh sh。

（5）测试集成模型：执行 python TestTripleModel.py，具体命令行参数如下：

参数名	作用	参数设置
seed	随机种子	777
device_id	GPU id	0,1,2,3,4,5…
bert_model	模型名字或路径	../PretrianedModel/bert_base_chinese
max_seq_length	句子最大长度	40
train_data_dir	训练集路径	DataSet/train0628.csv
test_data_dir	测试集路径	test.csv

若需替换测试数据集，只需执行两步：

a) 将待测试 excel 文件放于当前路径下，民命为 **test.xlsx**，执行：**Python DG.py**

b) 执行：**python TestTripleModel.py**