BERT 项目交付

1. 小数据集与开放数据集的模型

- **Docker 环境**

  环境： Cuda10.2 + Cudnn7.0 + Python 3.6;

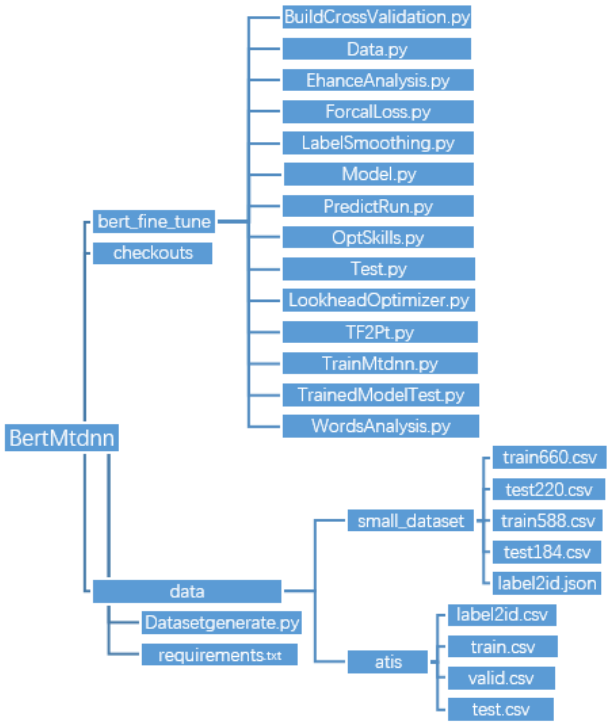
  重要依赖包： torch==1.5.1

  Pytorh-pretrained-bert==0.6.2

  包含： Docker 中已包含模型代码，数据集，已训练好的模型。

- **模型代码**

  代码结构如下:

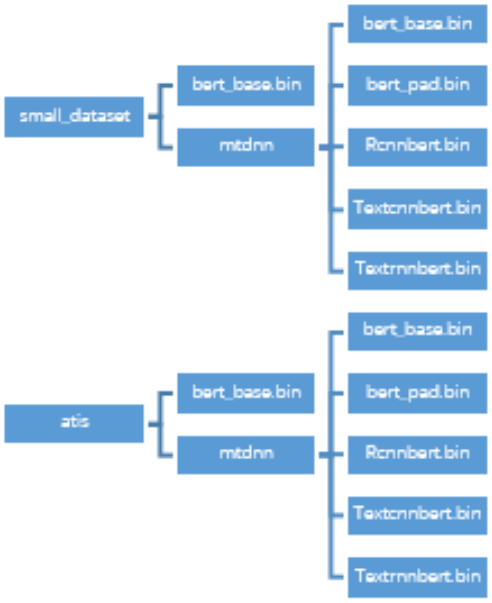


脚本说明:

脚本	作用
BuildCrossValidation.py	用于构建交叉验证数据集
Data.py	用于模型读取数据
EhanceAnalysis.py	用于数据增强
ForcalLoss.py	损失函数
LabelSmoothing.py	损失函数
Model.py	用于定义模型
PredictRun.py	模型效果测试
OptSkills.py	优化的技巧
Test.py	用于模型训练中的评估
LookheadOptimizer.py	优化器

TF2Pt.py	Tensorflow 模型转成 Pytorch 模型
TrainMtdnn.py	用于训练 MTDNN
TrainedModelTest.py	用于测试模型效果
wordAnalysis.py	用于数据集的单词分析
Datasetgenerate.py	将 excel 文件转成训练和测试集

- **训练好的模型**



- **数据集（训练，和我们自己的测试集）**

1) **小数据集：**

完整小数据集:

train660.csv
Test220.csv
label2id.json

去掉错误类的数据集:

train588.csv
test184.csv
label2id.json

2) **Atis 公开数据集：**

train.csv
valid.csv
test.csv

- **优化方案**

2. 大数据集模型

- **Docker 环境**

  环境：Cuda10.2 + Cudnn7.0 + Python3.6.10;

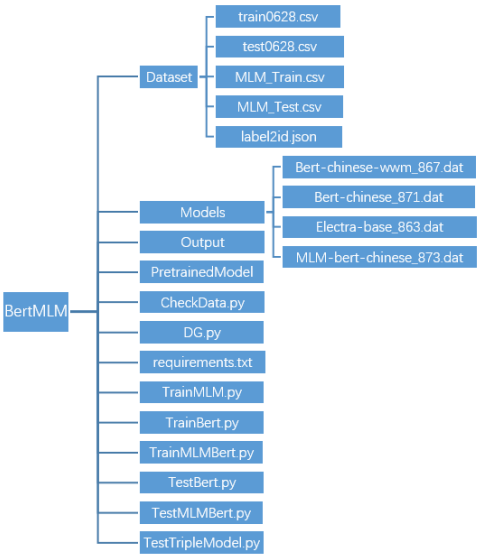
  重要依赖包：torch==1.5.1

  Transformers==2.11.0

  包含：Docker 中已包含模型代码，数据集，已训练好的模型。

- **模型代码**

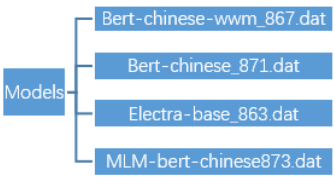
  代码结构如下:



  脚本说明:

脚本	作用
CheckData.py	用于检查是否可用于训练
DG.py	用于从 excel 生成数据集
TrainMLM.py	用于训练 Mask Language Model
TrainBert.py	用于训练 Bert
TrainMLMBert.py	用于训练 MLM fine tune 后的 Bert
TestBert.py	测试 BERT 效果
TestMLMBert.py	测试 MLM 后的的 BERT 效果
TestTripleModel.py	测试模型集成的效果

- **训练好的模型**



- **数据集（训练，和我们自己的测试集）**

大样本数据集：

train0628.csv（13 万条）

test0628.csv（1000 条）

- **优化方案**

一方面，前期试验已经验证采用 APEX 后，可节省训练模型所需 GPU 现存，提升训练效率，同时，可一定程度提升模型效果，故本次优化依然采用 APEX。

Model	Accuracy	
	used	unused
Albert-xxlarge + apex	75.9	75.8
Albert-base + apex	73.12	60.8
Bert-base-chinese + apex	75.6	74.9

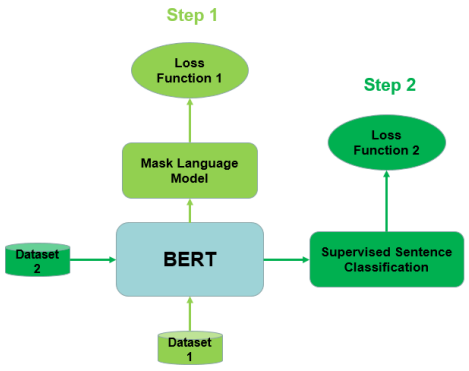
Model	Time(s)		Cuda(M)	
	used	unused	used	unused
Albert-base + apex	12	22	1917	2343
Bert-base-chinese + apex	19	27	3803	3867

另一方面，按照之前预设方案，采用 Multi-Task 两阶段 Fine Tune 的方式：

Step 1：采用 Mask Language Model 来 Fine Tune bert-base-chinese 模型；

Step 2：采用 Sentence Classification 的监督学习模型，进一步对 Step 1 中 Fine Tune 之后的 BERT 进行微调；

如图：



最终多个模型，以及集成模型效果对比如下（以下模型都采用 APEX 训练）：

Model	Accuracy
Bert-base-chinese	87.19
MLM-Bert-base-chinese	87.39
MLM-Bert + Electra + Bert-wwm	88.59

注：在 1000 条的验证集上，MLM-Bert + Electra + Bert-wwm 的集成模型单条数据预测需 30 毫秒。