FAQ 使用说明

1. 运行环境：

   系统：Linux 内核系统，建议 >= Ubantu16.04

   依赖库：（依赖库下文声明的版本可以运行，更高版本目前可以运行，之后不保证）

   python3,version 3.7

   pytorch, version 1.1.0

   pytorch-cuda, version 9.0.0

   Anaconda, version 4.7.12

   transformers,version 2.2.2

2. 代码文件说明：

   所有的代码全部运行在 cuda：0 或者 cpu 中，模型会自动优先选择 cuda：0

   代码针对中文字符进行设计，英文字符会一概拆分成字母，不会保留单词的词义。

   2.1 模型文件：BertForSegment

   模型参数：config 文件夹和 After_training 文件夹

   会在训练和应用的代码中自动会进行调用。config 文件夹下为初始文件。不可修改。

   2.2 训练代码：new_run_faq.py

   代码共包含指令 train1，train2，test1，test2，test3，final

   train1：正反对抗训练，使用数据集 data\final_mix_train_data

   train2：多分类训练，使用数据集 data\train_data

   test1：正反对抗训练的测试

   test2：多分类训练的测试

   test3：针对测试集 A 集的有答案的部分测试，会打印所有不是 top1 正确的句子信息和最终的结果。

   final：对测试集 A 集的所有数据进行测试，会打印所有不是 top1 正确的句子信息和最终的结果。

   通过直接在命令行施加参数来选择运行哪部分程序。

   例：python new_run_faq.py train1

   注：train1 和 train2 不会连续进行调用，在对新数据进行使用时，可以先经过 train1 的训练后观察结果，如果结果不够合适再继续使用 train2 进行训练，train2 并不一定可以提升精度，也可能会造成精度的下降，请谨慎选择做好备份。

   2.3 应用代码：inference.py

   使用数据：data\standard_qa

   data\standard_vec(在首次运行时自动生成，模型参数更新后需要先删除该文件再重新生成)

   直接在命令行输入要进行判断的句子来进行匹配。

   例：python inference.py 什么是 P2P

2.4 应用代码：inference_api.py

该文件是应用文件 inference.py 的 api 版本，保存为函数形式。输入参数为待匹配句，输出为匹配的结果，详细的格式说明见该文件注释。

该文件和 inference.py 共享 standard_vec 文件。

2.5 数据预处理代码：Dataset_preproccess.py

使用数据：data\standard_qa

直接运行该代码，会将 standard_qa 中的所有相似句组转换成模型所训练所需要的格式。生成 train_data,test_data,mix_train_data,mix_test_data,final_mix_train_data,final_mix_test_data.所有含有 test 的文件都不会被使用。

2.6字典维护代码：dictionary.py

人工手动维护模型参数的代码，验证集和训练集的差异性过大问题采取的解决办法之一，对训练集中未出现的相似词组对应进行手动标注添加到模型中。手动添加办法：修改代码第 19 行的字典，key 和 value 分别对应了两个相似的词组。修改后运行，目前功能仅支持增加而不支持删减。

2.7句子结构相似度计算：bleu_pretest_multi_gram.py

该部分不需要单独调用，会在句子进行相似度计算时被自动调用，使用了 config\stopwords 内的文件。

3 数据文件说明。

3.1 文件夹 config：

模型的初始参数保存为止，如果重新进行训练则要在此参数的基础上进行训练。

请勿对该文件夹下的所有文件进行任何修改。

3.2文件夹 After_training

经过训练后的模型参数保存为止，如果重新训练，则该文件夹下的文件会自动进行更新。

3.3data 文件夹

train_data,mix_train_data,final_mix_train_data 作为训练数据。

test_data,mix_test_data,final_mix_test_data 不会被使用

standard_qa，保存着原始问题和其对应答案的数据库。

standard_vec，经过计算后保存下来的所有标准句子的张量数组。

经过处理的标准数据格式：单位对象为字典，整体为数组。

举例：（仅含有一个标准问题的数据库）

下文为 data\standard_qa 文件内数据格式：

[{'class': '理财知识', 'FAQID': '24', 'Q': '工资和花销算资产吗？', 'A': '工资算你的资产，另外花销的话要看具体买了什么，如果是吃吃喝喝，那么算日常消费支出，没有资产，如果是买了个电脑，而你利用电脑开网店赚钱等等，那么这个就算是资产，它为你带来了现金流', 'similar': ['到底买什么东西是资产啊？', '工资是资产的一部分吗？', '哪些消费能够作为资产啊？', '每个月花钱买东西，是不是资产就减少了？', '花销是资产

吗？', '购买一些大型的商品是资产吗？', '哪些花销是不能算作资产的？', '花钱买的东西是资产吗？', '每个月的开支有哪些能当资产呢？', '平常吃饭花的钱算资产吗？', '有现金流的才是资产吗？', '只有工资和花销的话，那我的资产怎么算呢？','工资算我的资产吗？','应该怎么去计算工资和消费的资产有多少？','怎么算工资和花销产生的资产？', '花销也是资产的一部分吗？', '只要买了东西就是资产吗？', '工资是资产那么花销呢？', '拿到工资花掉了是不是资产就减少了？', '每个月的工资是资产吗？', '哪些消费算资产啊？', '公司给的薪水就是我的资产吗？', '日常消费支出算不算资产？', '我的开销里面有哪些算资产啊？', '能够带来现金流的就是资产吗？', '工资是不是资产呢？']}]