bili\_boston.py: 根据链接爬取视频课程数据 并 生成Boson情感评分

bili\_(class).xlsx: **[ID]**(char[]) **[用户]**(char[]) **[评论]**(char[]) **[Boson情感评分]**(char[]) **[点赞数]**(char[])

format.py: 基于BERT方法生成 评分精确且情感分数分布广的 情感评分

1. 原始数据格式整理

输入: 原始数据 bili\_(class).xlsx

作用: 先转为UTF-8格式的.csv文件，再对波士顿情感评分进行分类(大于0则为1,小于0则为0)

输出: bili\_(class).csv多了一列 即**[Boson评分简]**(int)

1. 测试输入用例是否正确， 并删除错误数据

生成文件 test.csv ；格式同上

1. BERT源码运行准备

将test.csv文件放到 F:\桌面\Bert\bert-master\data\ 目录下,替换 test.csv 文件

1. BERT算法概率生成

双击 train.sh 文件，运行代码， 在 train.log里 或者 在git bash下用 "tail -f train.log"查看运行日志

1. 概率转化为BERT评分

将 test\_results.tsv 概率转化为情感评分(0/1)；第二列为bert分数，输出为同目录下的pre\_sample.tsv

1. 格式整理

将test.csv转为final.xlsx并调整格式

1. 数据整合

将pre\_sample.tsv 的第二列拷贝到最后一列 **[Boson评分简] [评论] [用户名] [Boson情感评分] [点赞数] [BERT评分]**(float)

8. 数据筛查

手动删除格式不正确的数据，并新增一列(即在最后一行写入1，用作放置final数据)

9. 数据留存

重新新建一个.xlsx文件，把数据复制进去

10. 最终评分生成

综合考虑两类数据，得到bili\_(class)\_fin.xlsx **[Boson评分简] [评论] [用户名] [Boson情感评分] [点赞数] [BERT评分]**(float) **[最终评分]**(float)

注：以上为原始数据处理部分 与 评分生成 ； 学生端及教师端可视化可参考演示视频。