金融大数据课程实验 ——豆瓣电影短评爬虫

姓名：许方宁 学号：2019210139

1 摘要

1.1 题目概述

本次我选择的实验题目是采用 scrapy 框架爬取 1000 部豆瓣电影短评。

1.2 功能介绍

1. 爬取 豆瓣电影按时间排序的欧美电影和按时间排序的华语电影两个分类下的 1000 部电影 id,
2. 根据电影 id 检索到电影评论页，爬取短评中的评论类型（好评或差评）和评论内容。

1.3 实验环境

操作系统：Windows 10
实验语言：Python 3.6
所用框架：scrapy
使用程序包：re(正则表达式模块)、json(数据交换格式)、csv(建立 csv 类型文件)等

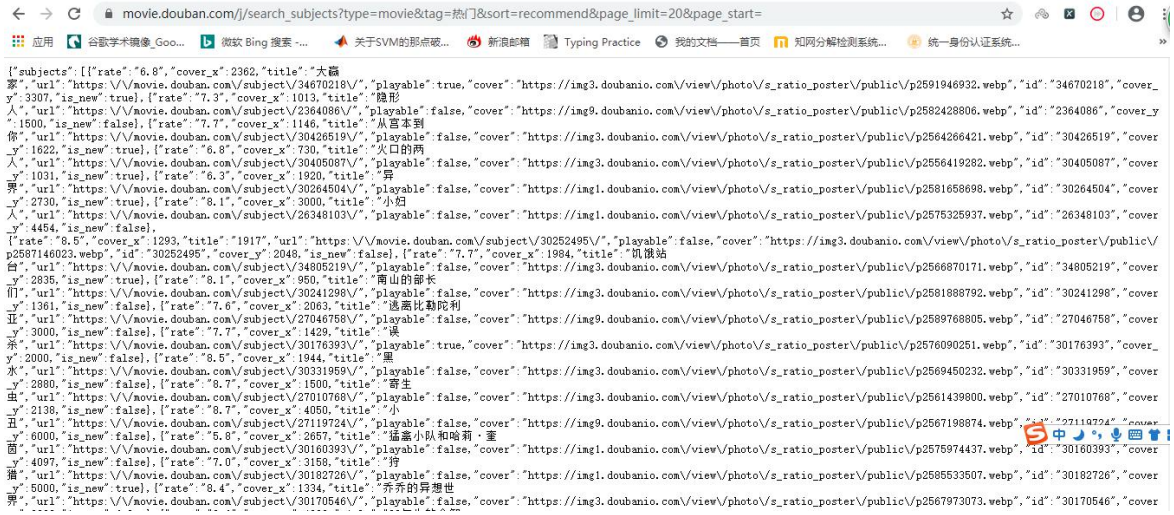
2 工作流程管理

2.1 系统模块设计

2.1.1 信息爬取模块一

本实验的主要思路是先爬取 1000 部电影然后在爬取每部电影的短评。第一个需要爬取的是 1000 部电影的详情页面。如果从豆瓣电影页面入手，代码会比较繁琐。

在各种搜索、学习和尝试后发现豆瓣上有关电影的信息都是采用 ajax 方式加载到 HTML 中，于是在 Network 下找到加载 ajax 的链接（见图 1），可以通过修改该链接查询参数，通过浏览器查看 API 返回的数据格式、数据项和字符编码，比较方便，易于爬取。因此将该链接作为请求地址。



图一 豆瓣加载 ajax 的链接为一个 JSON 字符串

本次实验发现豆瓣每个电影的详情页是 https://movie.douban.com/subject/加上电影的 id。因此在豆瓣加载 ajax 的链接上爬取每个电影的 id，在按时间排序的欧美电影分类下爬取 500 部电影的 id（由于豆瓣页面的限制，最多可以读取 500 部电影），同理在按时间排序的华语电影分类下爬取 500 部电影的 id，并存到 txt 文件中，以便接下来的程序调用。

2.1.2 信息爬取模块二

这一部分是本实验花费最多时间的部分。在上一模块最后生成的 txt 文件中读取电影 id，生成该电影的好评页面和差评页面，每页有 20 个评论。通过对 url 最后一个参数每次加 20 的方法翻页。

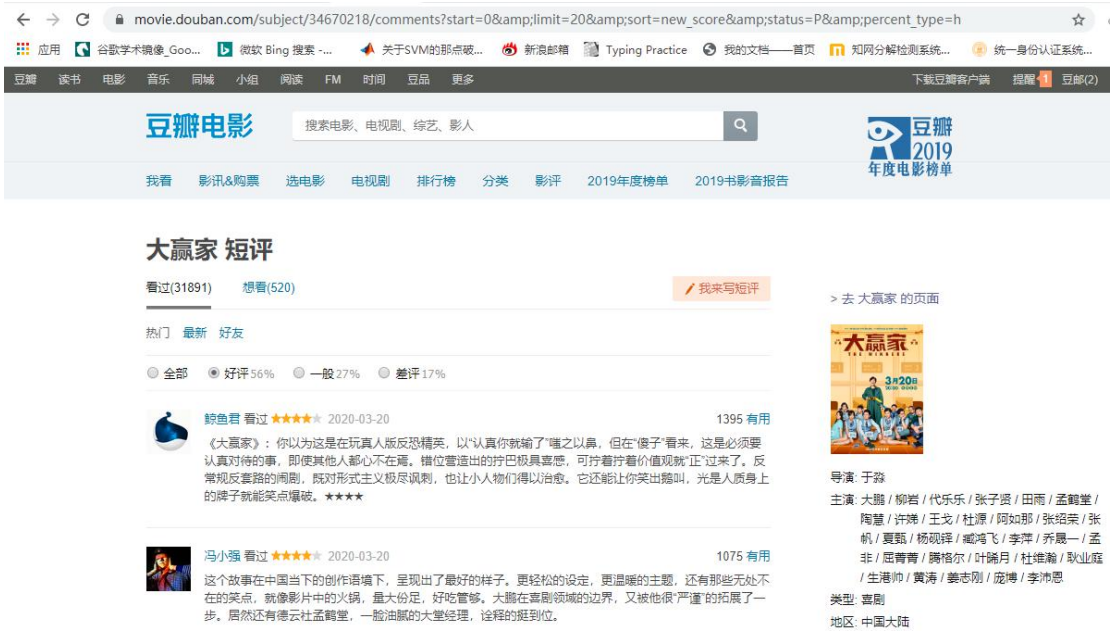


图 2 好评页面（每一页有 20 个评论，读后 20 个需要翻页）

2.2 详细设计

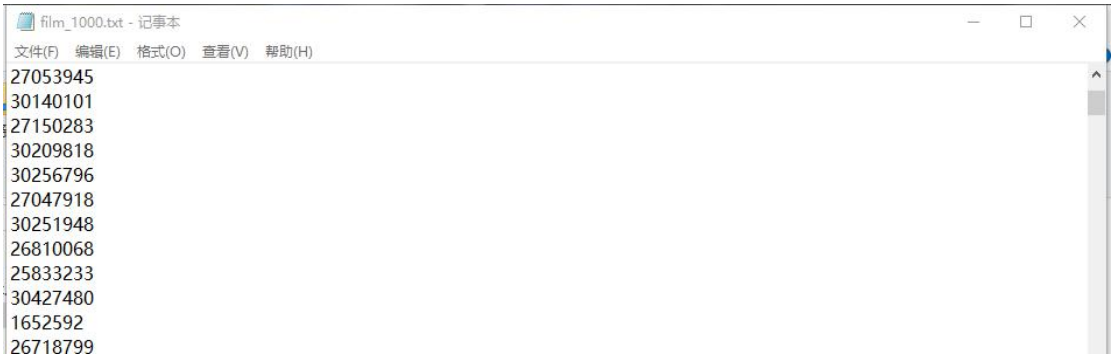
movie_review 文件夹是 scrapy 项目文件夹。spiders 文件夹下有两个 spider，分别是：

douban_scrapy.py —— 用于爬取按时间排序的欧美电影和按时间排序的华语电影两个分类下 1000 部电影的 id；

douban.py —— 用于爬取 1000 部电影短评中的好评和差评。

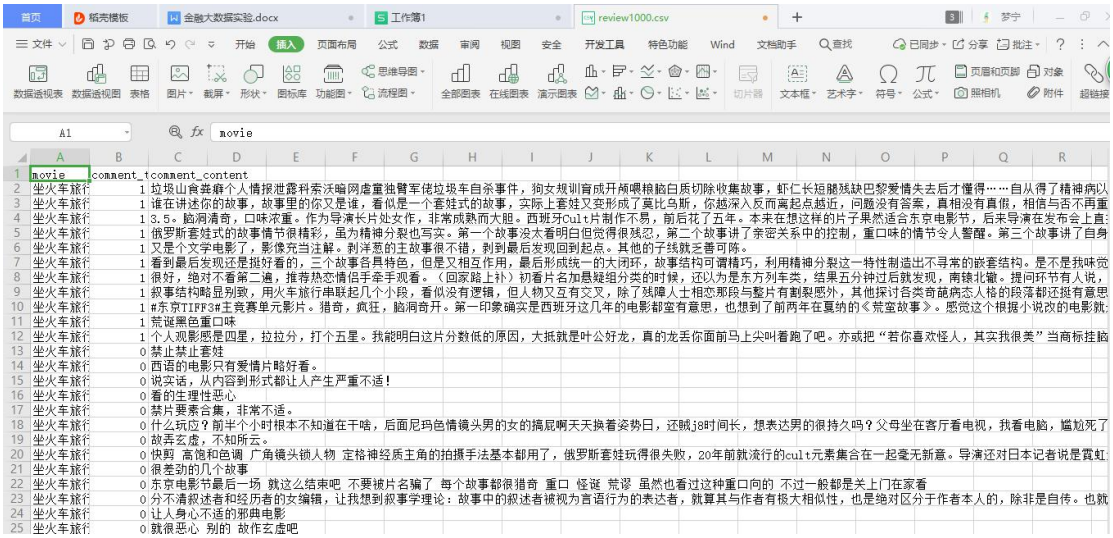
data 文件夹是存数据的文件夹，有两个文件：

film_1000.txt ——1000 部电影的 id；



图三 film_1000.txt 的截图

review1000.csv---电影的评论，有三列：第一列为“电影名 短评”，第二列是评论类型：1 表示好评，0 表示差评，第三列是评论内容。



movie	comment_t	comment_content
坐火车旅行	1	垃圾山食粪癖个人情报泄露科索沃暗网虐童独臂军佬垃圾车自杀事件，狗女规训育成开颅喂根脑白质切除收集故事，虾仁长短腿残缺巴黎爱情失去后才懂得……自从得了精神病以
坐火车旅行	1	谁在讲述你的故事，故事里的你又是谁，看似是一个套娃式的故事，实际上套娃又变形成了莫比乌斯，你越深入反而离起点越近，问题没有答案，真相没有真假，相信与否不再重
坐火车旅行	1	3.5。脑洞清奇，口味浓重。作为导演长片处女作，非常成熟而大胆。西班牙Cult片制作不易，前后花了五年。本来在想这样的片子果然适合东京电影节，后来导演在发布会上直
坐火车旅行	1	俄罗斯套娃式的故事情节很精彩，虽为精神分裂也写实。第一个故事没太看明白但觉得很残忍，第二个故事讲了亲密关系中的控制，重口味的情节令人警醒。第三个故事讲了自身
坐火车旅行	1	又是个文学电影了，影像充当注解。剥洋葱的主故事很不错，剥到最后发现回到起点。其他的子线就乏善可陈。
坐火车旅行	1	看到最后发现还是挺好看的，三个故事各具特色，但是又相互作用，最后形成统一的大闭环，故事结构可谓精巧，利用精神分裂这一特性制造出不寻常的嵌套结构。是不是我味觉
坐火车旅行	1	很好，绝对不看第二遍，推荐热恋情侣手牵手观看。（回家路上补）初看片名加悬疑组分类的时候，还以为是东方列车类，结果五分钟过后就发现，南辕北辙。提问环节有人说，
坐火车旅行	1	叙事结构略显别致，用火车旅行串联起几个小段，看似没有逻辑，但人物又互有交叉，除了残障人士相恋那段与整片有割裂感外，其他探讨各类奇葩病态人格的段落都还挺有意思
坐火车旅行	1	#东京TIFF3#主竞赛单元影片。猎奇，疯狂，脑洞奇开。第一印象确实是西班牙这几年的电影都蛮有意思，也想到了前两年在戛纳的《荒堂故事》。感觉这个根据小说改的电影就
坐火车旅行	1	荒诞黑色重口味
坐火车旅行	1	个人观影感是四星，拉拉分，打个五星。我能明白这片分数低的原因，大抵就是叶公好龙，真的龙丢你面前马上尖叫着跑了吧。亦或把“若你喜欢怪人，其实我很美”当商标挂脑
坐火车旅行	0	禁止禁止套娃
坐火车旅行	0	西语的电影只有爱情片略好看。
坐火车旅行	0	说实话，从内容到形式都让人产生严重不适！
坐火车旅行	0	看的生理性恶心
坐火车旅行	0	禁片要素合集，非常不适。
坐火车旅行	0	什么玩应？前半个小时根本不知道在干啥，后面尼玛色情镜头男的女的搞屁啊天天换着姿势日，还贼j8时间长，想表达男的很持久吗？父母坐在客厅看电视，我看电脑，尴尬死了
坐火车旅行	0	故弄玄虚，不知所云。
坐火车旅行	0	快剪 高饱和色调 广角镜头锁人物 定格神经质主角的拍摄手法基本都用了，俄罗斯套娃玩得很失败，20年前就流行的cult元素集合在一起毫无新意。导演还对日本记者说是霓虹
坐火车旅行	0	很差劲的几个故事
坐火车旅行	0	东京电影节最后一场 就这么结束吧 不要被片名骗了 每个故事都很猎奇 重口 怪诞 荒谬 虽然也看过这种重口向的 不过一般都是关上门在家看
坐火车旅行	0	分不清叙述者和经历者的女编辑，让我想到叙事学理论：故事中的叙述者被视为言语行为的表达者，就算其与作者有极大相似性，也是绝对区分于作者本人的，除非是自传。也就
坐火车旅行	0	让人身心不适的邪典电影
坐火车旅行	0	就很恶心 别的 故作玄虚吧

图三 review1000.csv 的截图

2.2.1 文件说明

pipeline.py

```
class MovieReviewPipeline(object):
    def process_item(self, item, spider):
        return item
```

middleware.py

和 scrapy 自动生成的文件内容一样，未做改变。

Item.py

```
import scrapy
class MovieReviewItem(scrapy.Item):
    review = scrapy.Field()
    sentiment = scrapy.Field()
    review_name=scrapy.Field()
    filmid=scrapy.Field()
```

douban_scrapy.py

使用 Scrapy 抓取 1000 部豆瓣电影 id

```
class DouBanSpider(scrapy.Spider):
    name = 'douban_scrapy'
    allowed_domains = ["douban.com"]
    start_list = []
```

```
    for i in range(0,25):

        url='https://movie.douban.com/j/search_subjects?type=movie&tag=%E6%AC%A7%E7%BE%8E
        &sort=time&page_limit=20&page_start=' + str(i*20)#按时间排序的欧美电影页面
        start_list.append(url)
    for i in range(0,25):

      url='https://movie.douban.com/j/search_subjects?type=movie&tag=%E5%8D%8E%E8%AF%AD&
      sort=time&page_limit=20&page_start=' + str(i*20)#按时间排序的华语电影页面
        start_list.append(url)
    start_urls = start_list     # 定义 start_urls 为一个存储链接的列表

    def start_requests(self):
        user_agent = 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/74.0.3729.169 Safari/537.36'
        headers = {'User-Agent': user_agent}
        for url in self.start_urls:
            yield scrapy.Request(url=url, headers=headers, method='GET', callback=self.parse)

    def parse(self, response):
        hxs = response.body.decode('utf-8')
        hjson = json.loads(hxs)       # 字典
        for lis in hjson['subjects']:
            item = MovieReviewItem()            # 实例化类
            item["filmid"] = lis['id']    #爬电影 id
           #写入文件
            with open('./data/film_1000.txt', 'a+') as f:
                f.write(item["filmid"])
                f.write('\n')
            print(item["filmid"])
            # filename = item["title"] +item["filmid"]+ '_' + item["score"] + '分' + '.jpg'
            # urllib.request.urlretrieve(item["pic"], filename)
            yield item
```

douban.py

使用 Scrapy 抓取 1000 部豆瓣电影的好评和差评

```
import re
import csv
import scrapy
from scrapy.http import Request
#建立存评论的文件
 with open('./data/review1000.csv','w') as f:
      csv_write = csv.writer(f)
```

```
        csv_head = ["movie","comment_type(1:good,1:bad)","comment_content"]
        csv_write.writerow(csv_head)

class DouBanSpider(scrapy.Spider):
    name = 'douban'

    def start_requests(self):
        #打开存有电影 id 的文件,并将电影 id 一列表的形式存到 film_list
        with open('./data/film_1000.txt', 'r') as f:
            filmid_list = f.readlines()
            # movie_id=top_list[0]
            # start=0
        for movie_id in filmid_list:
            for start in range(0, 200, 20):
                meta = {
                    'sentiment': 1
                }
                movie_id = movie_id.replace('\n', '')
#好评网址
                url = 'https://movie.douban.com/subject/{}/comments?start={}&limit=20&sort=new_score&status=P&percent_type=h'.format(movie_id, start)
                yield Request(url=url, meta=meta)
                meta = {
                    'sentiment': 0
                }
                movie_id = movie_id.replace('\n', '')
#差评网址
                url = 'https://movie.douban.com/subject/{}/comments?start={}&limit=20&sort=new_score&status=P&percent_type=l'.format(movie_id, start)
                yield Request(url=url, meta=meta)

    def parse(self, response):
        #读评论内容
        review_list = response.xpath('//span[@class="short"]/text()').extract()
        #读是哪个电影的短评
        review_name = "".join(response.xpath('//*[@id="content"]/h1/text()').extract())
        print(review_list)
        for review in review_list:
            review = review.strip()
            review = review.replace('\t', '')
            review = review.replace('\n', '')
            review = review.replace('\xa0', '')
```

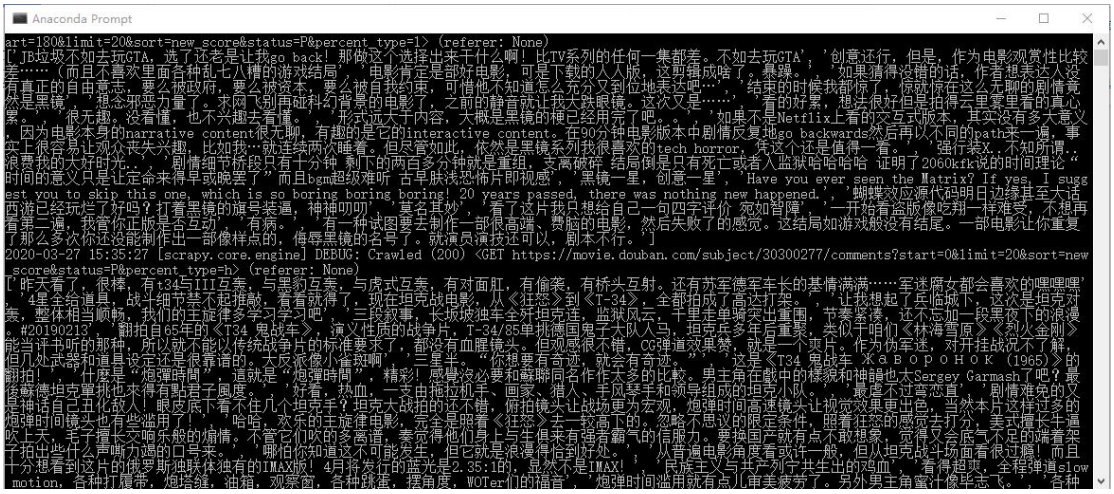
```
        review = review.replace('\ufeff', '')
        review = review.replace('\u200b', '')
        #写入文件
        if review:
            with open('./data/review1000.csv', 'a+',newline='') as f:
                csv_write = csv.writer(f)#可以理解为初始化
                data_row = [review_name,response.meta['sentiment'], review]
                csv_write.writerow(data_row)
```

2.2.2 运行方式

获取豆瓣 1000 部电影 ID：scrapy crawl douban_scrapy
抓取豆瓣 1000 部电影短评：scrapy crawl douban

3 最终结果

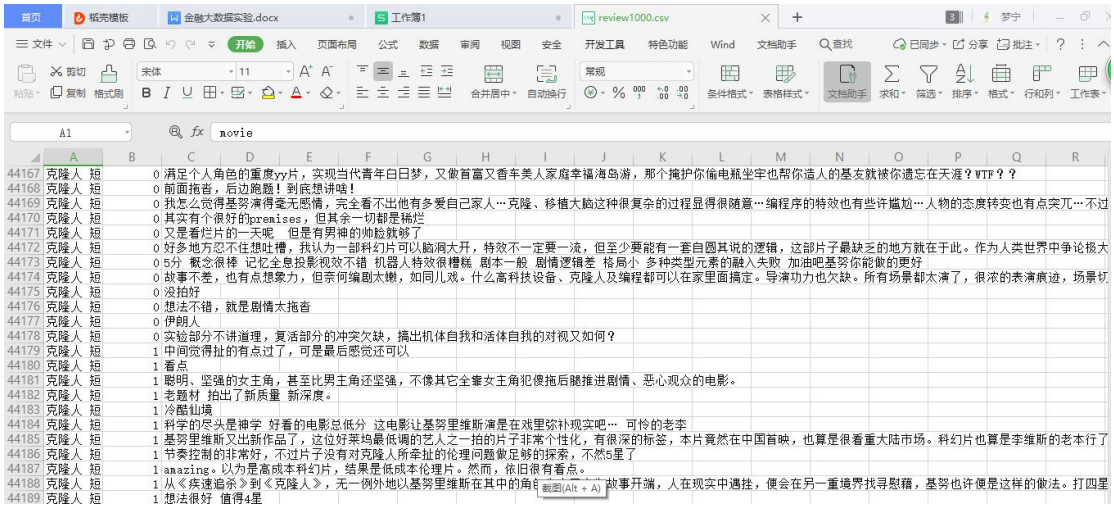
3.1 运行截图



3.2 运行结果

第一个爬虫运行顺利，能一次成功读 1000 个电影的 id。

第二个爬虫可以正常运行，但是运行运行大约 40 分钟-2 小时后，ip 会被禁，需要暂停 40 分钟后才能继续爬虫。data 文件夹中的 review1000.csv 是我运行三次 douban 爬虫后爬取的评论，读取了 187 部电影 44286 条评论。



44167	克隆人	短	0	满足个人角色的重度yy片，实现当代青年白日梦，又做首富又香车美人家庭幸福海岛游，那个掩护你偷电瓶坐牢也帮你遣人的基友就被你遗忘在天涯？WTF？？
44168	克隆人	短	0	前面拖沓，后边跑题！到底想讲啥！
44169	克隆人	短	0	我怎么觉得基努演得毫无感情，完全看不出他有多爱自己家人…克隆、移植大脑这种很复杂的过程显得很随意…编程序的特效也有些许尴尬…人物的态度转变也有点突兀…不过
44170	克隆人	短	0	其实有个很好的premises，但其余一切都是稀烂
44171	克隆人	短	0	又是看烂片的一天呢　但是有男神的帅脸就够了
44172	克隆人	短	0	好多地方忍不住想吐槽，我认为一部科幻片可以脑洞大开，特效不一定要一流，但至少要能有一套自圆其说的逻辑，这部片子最缺乏的地方就在于此。作为人类世界中争论极大
44173	克隆人	短	0	5分　概念很棒　记忆全息投影视效不错　机器人特效很糟糕　剧本一般　剧情逻辑差　格局小　多种类型元素的融入失败　加油吧基努你能做的更好
44174	克隆人	短	0	故事不差，也有点想象力，但奈何编剧太懒，如同儿戏。什么高科技设备、克隆人及编程都可以在家里面搞定。导演功力也欠缺。所有场景都太演了，很浓的表演痕迹，场景切
44175	克隆人	短	0	没拍好
44176	克隆人	短	0	想法不错，就是剧情太拖沓
44177	克隆人	短	0	伊朗人
44178	克隆人	短	0	实验部分不讲道理，复活部分的冲突欠缺，搞出机体自我和活体自我的对视又如何？
44179	克隆人	短	1	中间觉得扯的有点过了，可是最后感觉还可以
44180	克隆人	短	1	看点
44181	克隆人	短	1	聪明、坚强的女主角，甚至比男主角还坚强，不像其它全靠女主角犯傻拖后腿推进剧情、恶心观众的电影。
44182	克隆人	短	1	老题材　拍出了新质量　新深度。
44183	克隆人	短	1	冷酷仙境
44184	克隆人	短	1	科学的尽头是神学　好看的电影总低分　这电影让基努里维斯演是在戏里弥补现实吧…　可怜的老李
44185	克隆人	短	1	基努里维斯又出新作品了，这位好莱坞最低调的艺人之一拍的片子非常个性化，有很深的标签，本片竟然在中国首映，也算是很看重大陆市场。科幻片也算是李维斯的老本行了
44186	克隆人	短	1	节奏控制的非常好，不过片子没有对克隆人所牵扯的伦理问题做足够的探索，不然5星了
44187	克隆人	短	1	amazing。以为是高成本科幻片，结果是低成本伦理片。然而，依旧很有看点。
44188	克隆人	短	1	从《疾速追杀》到《克隆人》，无一例外地以基努里维斯在其中的角色 故事开端，人在现实中遇挫，便会在另一重境界找寻慰藉，基努也许便是这样的做法。打四星
44189	克隆人	短	1	想法很好　值得4星

图 5 最后生成的存储评论文件的截图

4 总结与体会

这学期的金融大数据实验是我第一次正式接触信息爬取的内容。在这两个星期，我借助书《Python3 网络爬虫开发实战教程》和网上的爬虫视频和教程学习了爬虫知识。 虽然补习了一下知识，但在动手时还是遇到了很多实践上的问题，所以中间费了一番力气。这次的实验题目中的豆瓣电影也是我们日常生活中经常浏览的网站，所以做实验的热情是很大的，感觉做出来的成果很有意思。也正是因为这个题目，我在课余生活中能够拿出时间学习自己感兴趣的知识。在接下来的空闲时间内，我将继续优化我这次的 scrapy 框架，让爬取更加自动化和持久。