豆瓣图书 数据分析：

前几天上网，无意中看到有人抓取了知乎的数据用户信息做数据分析，其实知乎爬虫的数据分析，我早就看到过，不过这次觉得自己也得做出点什么东西出来，零零总总写了许多次爬虫了，都是把数据抓下来放进数据库里，有的就是初步清洗了一下，有的干脆抓下来，往数据库里面一扔，就完事了。

以前在做爬虫时，都把抓取放在第一位，后来我觉得抓取只是爬虫的第一步（当然，抓取的程序设计好，会节约很多时间），爬虫抓的数据，如果后期不做分析和处理的话，都只能算是“垃圾”，没有任何意义，后面的处理，才能将数据变废为宝，当然，也有一种情况是边抓取边处理，不过一般不建议这样设计一个爬虫程序，不但不利于调度与管理，还会导致程序复杂度增大。

这次抓取的是豆瓣所有热门标签下的图书数据，网络上有很多电影方面的分析，所以我换了一个方向，第一次做分析，属于菜鸟类，也很初步，各位看看就好，觉得有帮助那就更好~

我不太习惯使用框架（主要是短时间看不懂），所以程序除使用jsoup解析网页和dom4j加载一下xml文件之外，都是用java原生库实现。对了，后面图表的前端显示也没有用软件，是用java+第三方包jfreechart绘制的~

完整花在上面的时间应该在4~5天，中间还忙活了其他的事情。

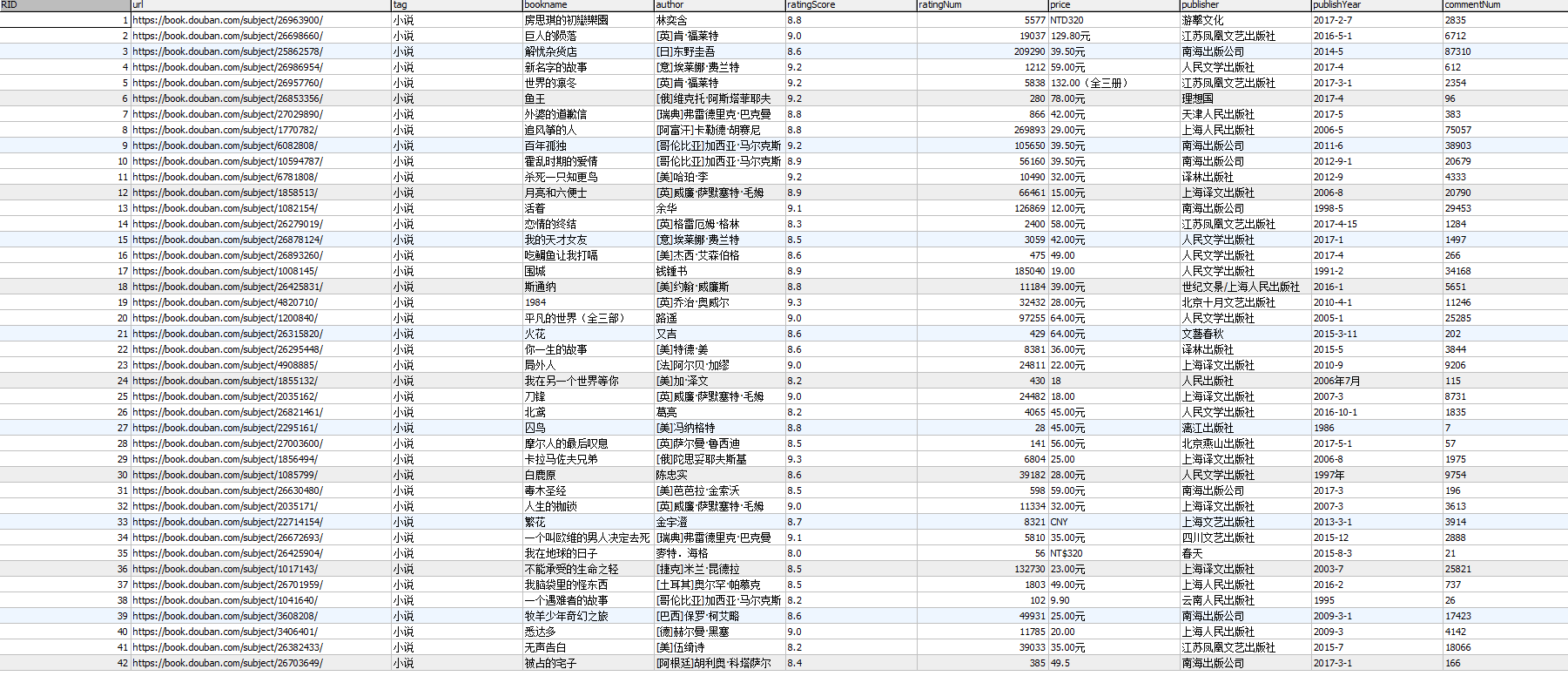
先是将豆瓣读书上，所有热门的图书标签拖下来，每一个大分类下，跟着一系列标签，如下：



总共有145个。

然后，就是把所有标签下的图书链接抓下来，这里豆瓣君欺骗了我们，看着上图下，每个标签下都有起码5W+的图书信息，其实标签为作者（东野圭吾、韩寒等），图书一般都没有达到1k以上；还有像小说这种一听就知道有很多书的标签，都是只给你1k的数据封顶，只能抓取50页，其实这个很好理解，一个原因是它防着爬虫之类的机器人，还有一个就是根据用户关注的热度来推送的，50页之后，很少有用户会感兴趣往下翻，其实所有网络上，带有翻页模式的页面，都不会超过100页，google、百度等，超过了都不显示了，因为之后都是很慢的查询（我不是指瀑布模式啊~），

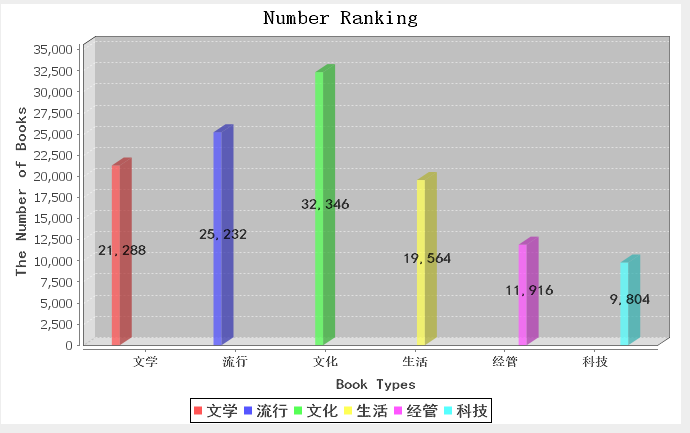
整个抓取的过程，可以说是一把辛酸一把泪~不过最后还是将数据拿下来，汇成一张大表，多次校准后，数据具有较高的完整性。详情请看：



字段较多，不太清晰。

所有热门标签下的图书信息我都拿了下来，每个标签下面都过滤了重复的链接（不保证每个大分类下的重复链接，如东野圭吾的《白夜行》，既属于悬疑类，又属于推理类），最后的获取到的图书数量为：





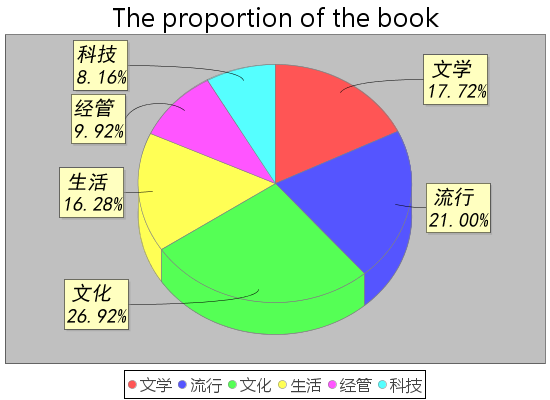
去掉所有大分类下重复链接后：



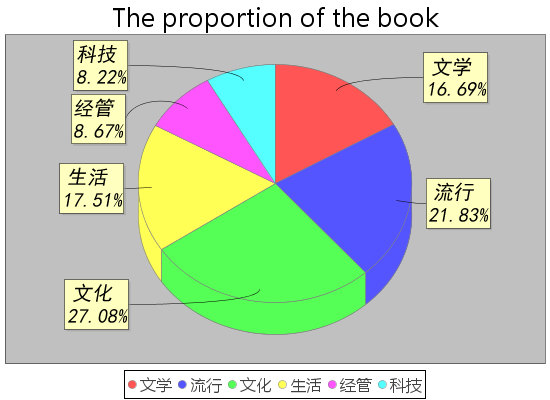
这应该是所有不重复的图书数量了，在豆瓣允许的范围内，这个数据还是比较完整的。

下面是一些分析（比较初步）：

1：这:是不重复的图书数量每个大分类占比：



这是所有不重复的图书数量每个大分类占比，可以看到文化类占比最多。这是不去重的，其实大致没有差别，



2：下面3张图，是截止到我抓取为止的数据，热门的图书评论数量实时都在增长，分别是：

①：评分>=8.0且评论人数10w+以上的图书（16本）；②：仅仅评论人数超过10w+的图书（18本）；③：根据图书评论数量排名的TOP20；



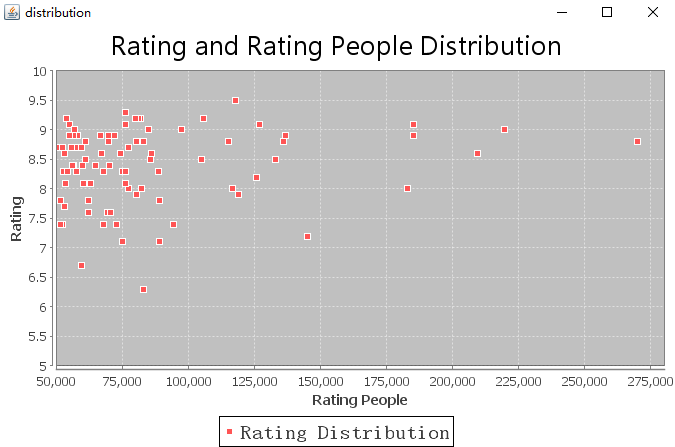




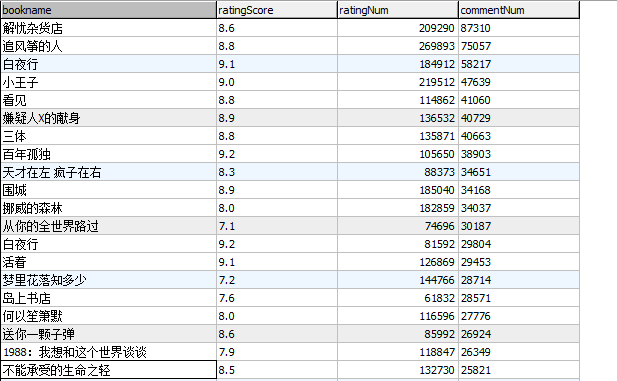
比较后，发现，一些值得我们一读再读的名著，始终都在名单里面，名单里面也大部分是开卷即有益的好书。降低标准后，某些牛鬼蛇神涌出，同时某些有益书籍（《平凡的世界》之类）涌现。

由此可得出，数据分析算法应该是综合多种数据后得到的权重模型（这是我从某位豆瓣电影数据分析的一位仁兄那看来的一句话（当然我也是这么认为的），这样形容比较贴切，我稍做润色，还是谢谢那位仁兄），所以不一定评论人数多或者评分高的书籍值得去阅读，综合考虑得出的结果才可算作公平。典例请参考郭敬明的《梦里花落知多少》和路遥的《平凡的世界》。

3：这是评论人数5w+以上的图书分布情况：



从分布情况来看，图书评论人数5w+以上，评分大部分集中在7.5~9.0之间，但是也有例外，有2本是低于7.0分的，没错，还是郭敬明的~（《悲伤逆流成河》6.2分和他的《小时代》6.7分），大家的品味还是有的。在这里面，只有国学名著《红楼梦》达到了9.5的高分，经典总是不容辜负。评论人数最多，且分数也达到了8.8高分，想必大家都知道，是《追风筝的人》~

4：这里还有一些图书短评数量等的排名TOP20：

其实大部分与前面的名次相同，但是无法准确的反应有效信息，因为未获取短评具体内容，各位看看就好~

其实还可以像豆瓣上的统计那样，评出TOP250什么的，只不过再弄就没有多大意义了，上面的几个分析已经有了初步判断，况且数据都有，只是换一下数据库的查询和筛选而已。我的数据量不大，字段数量也一般，数据库的查询筛选都比较快，需要思考的在于SQL语句的书写。

闲话一下感想：

想起前段时间学校实验室的老师跟我说，所谓的数据挖掘和流行的大数据分析，到最后，就是一个数据库而已，之前一知半解，知其然而不知其所以然，现在才觉得摸到点门道了。

代码的话，没什么难度，爬虫的基本思路，接触过的人应该都了解，我中间花在抓取上的时间较多，主要是想在了解了对方的反爬策略后，找出一个最好的抓取策略，快速而全面的获取数据。

现在流行的一个思想就是分布式，但是我觉得爬虫的设计应该视情况而定吧，不同数据的抓取有不同的抓取策略，分布式在大型数据的拖曳与分析中，优势很大。我之前做过全国google地图的拖曳下载，写过一个分布式的爬虫，后面会再总结分析。于我个人的理解来说，分布式爬虫的设计主要在于二个问题：控制与调度。控制即多次测试后，找出最适合的抓取策略，在数据抓取的来源网站限制下，控制爬虫最大程度的获取更多数据；调度即在各种意外情况（断网、断电等），服务器对不同爬虫客户端任务的处理。

豆瓣图书的分析到此为止（也挺简单的，主要是我觉得没啥好分析的了），后面继续，对豆瓣的用户数据分析（时间允许的话），有兴趣可以跟进了解一下~

如果对代码感兴趣的话，有时间我会贴出来。