豆瓣友邻用户数据分析

摘要：以豆瓣某用户的友邻们的账户数据和观影数据为对象，利用python的相关工具与收集、分析，并进行数据可视化，得到清晰明确的豆瓣友邻画像。

关 键 词: python；爬虫；数据分析；数据可视化

考虑到豆瓣的用户uid并不完全是数字，无法直接使用遍历的方法获取所有用户的信息；以及豆瓣本身存在反爬虫机制，对访问次数具有限制，无法在短时间内获取大量数据；再加上大量的僵尸号会产生占比极高的无用数据，直接对所有豆瓣用户的数据进行收集和分析显得难度过高切耗费较大。若采用随机某分析些用户数据的方式，可能会因为收集了过多的垃圾数据，导致分析结果不具备代表性。于是，本项目将以某一用户的友邻群体为基础，爬取相关数据并进行分析，这样所得数据代表性较高，还能够生成对应用户清的豆瓣友邻画像。

# 数据获取

## 问题描述

目的是获取供分析的数据。数据都来源于目标用户的友邻。

首先是要有目标用户的所有友邻的列表，即该用

户的所有友邻的uid列表。

之后再对每一个友邻的详细数据进行爬取。

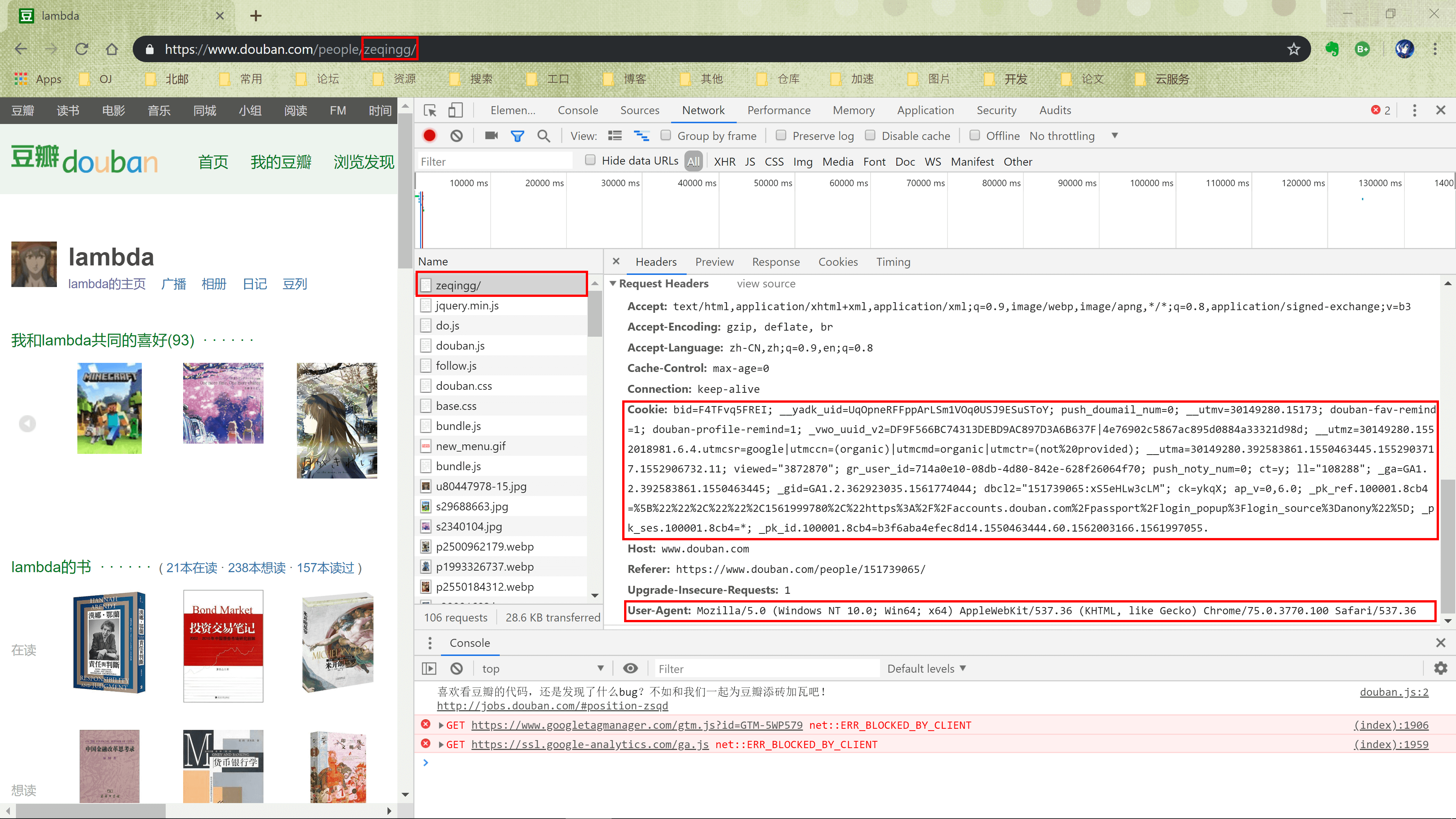
## 友邻列表获取

以[https://www.douban.com/people/](https://www.douban.com/people/zeqingg/rev_contacts)[zeqingg/rev\_contacts](https://www.douban.com/people/zeqingg/rev_contacts)为例，此页面会展示目标用户所有的友邻，其中的zeqingg即是目标用户的uid。

但是需要注意的是，访问该页面必须要登录豆瓣账号，否则会跳转到登录页面，导致网页获取失败。

现在浏览器（以Google Chrome为例）中登录自己的豆瓣账号后进入此页面，使用Developer tools中的network工具获取发送给服务器的Cookie和User-Agent（如图（1）所示）。在python程序中使用requests发送请求时加入Cookie，模拟登录，加入User-Agent模拟浏览器访问，防止被拦截，从而得到目标页面的静态文件。

所得到的Cookie并不是永久有效，并不能一直使用。倘若Cookie失效了，需要重新登录获取新的Cookie。



图（1）

可以发现，每个页面最多只能显示70位友邻，而通过设置在请求的parameter中的start值来指定显示的起始友邻。因此，必须先获取目标用户的所有友邻数量。

通过对多个网页的分析，发现每位用户的友邻数都会显示在固定的地方（如图（2）所示），其对应的css selector为 #db-usr-profile > div.info > h1。使用BeautifulSoup解析之前获取的网页，使用select函数提取出该部分，得到友邻数量。



图（2）

有了友邻数量后，就可以遍历所有页面。对于每一个页面，每个友邻都有一个指向他们自己主页的链接。主页的url对应的css selector都为 #content > div > div.article > dl > dt > a。每个url的形式如<https://www.douban.com/people/zeqingg/>，可以非常方便地提取出uid。

最后，将所有得到的uid都存入本地文件中，方便之后使用，不需要每次都重新获取。

## 账户信息获取

每一个用户的主页都有其对应的数据，但是考虑到其页面结构复杂，处理起来比较复杂，且数据内容较少，所以不是使用。

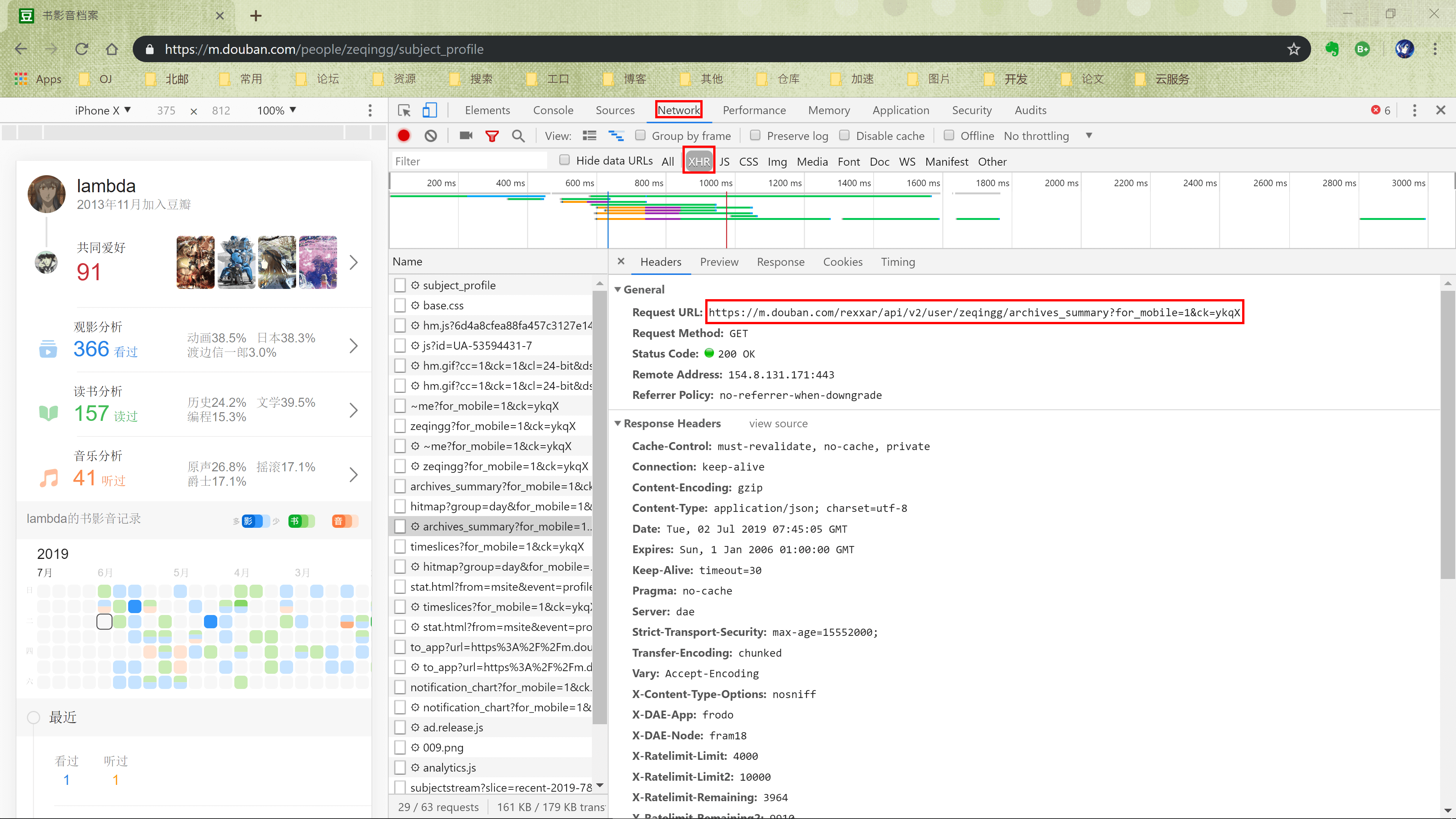
豆瓣手机版的书影音档案页面就有着更加详细的信息，可以支撑更深入的数据分析。

以<https://m.douban.com/people/zeqingg/subject_profile>为例，上面有许多信息可以供分析。但是获取网页后发现并没有所看到的数据。

经过分析后发现，该网页是动态网页，所有的数据都是网页打开后再请求的。再次使用Developer tools中的network工具，查看XHR类别中的项，得到了请求真正的url为<https://m.douban.com/rexxar/api/v2/user/zeqingg/archives_summary?for_mobile=1&ck=ykqX>（如图（3）所示）。请求该链接并不需要模拟登录。同时，返回的数据为json格式，使用python中json模块的loads()函数进行解析。

需要的是账户信息，在json数据中对应的为user项，其中有许多有用的数据项，比如生日birthday，性别gender，常住地loc，注册时间reg\_time，广播数statuses\_count等等。我们都将其保存到一个csv文件中。

不能忽略的是，有的信息用户没有填写或不愿公开，在json中就没有对应的数据。为了防止python程序报错，使用try语句，若不存在，就写入空的字符串。



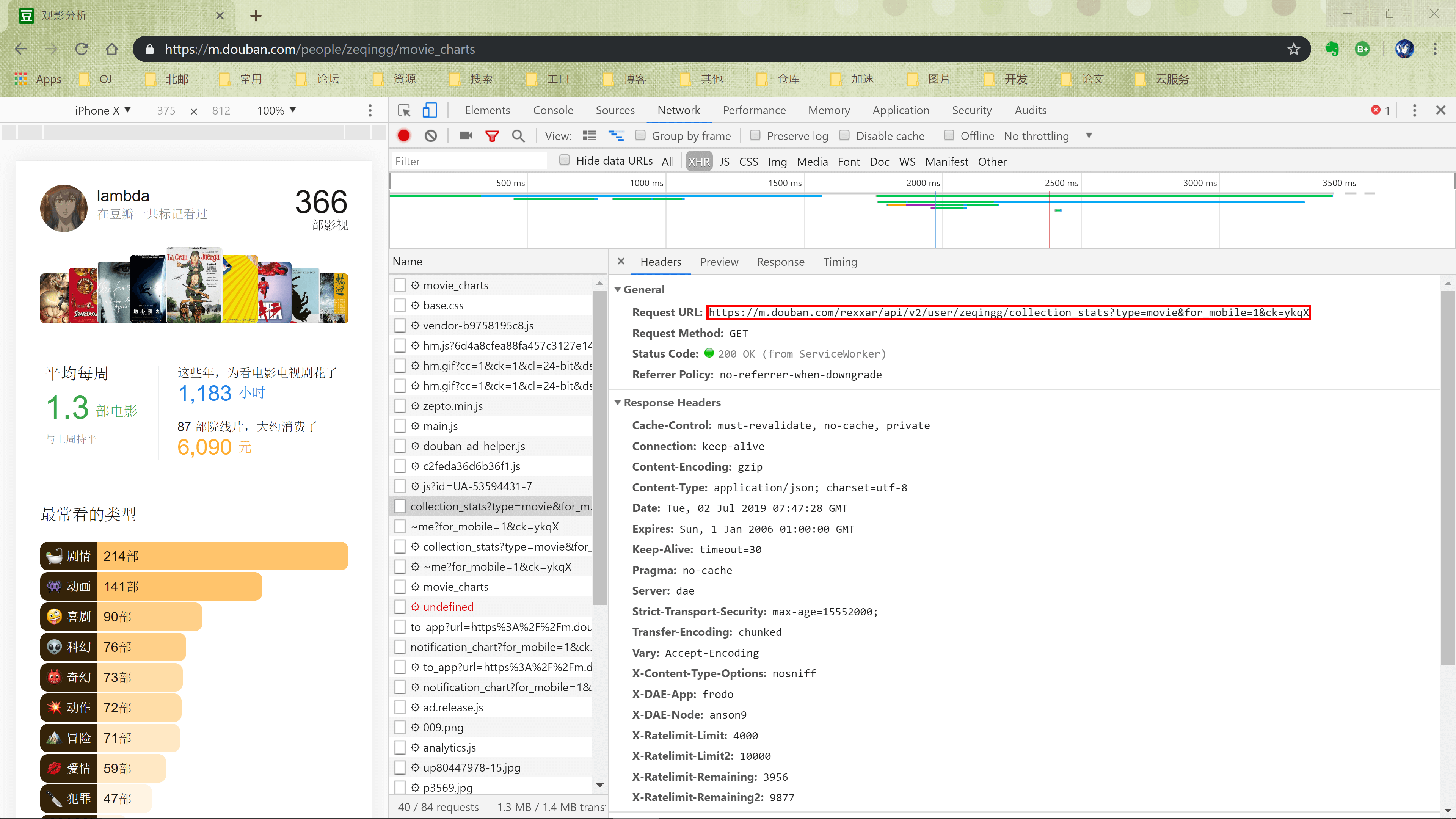
图（3）

## 观影信息获取

观影信息的获取同账户信息的获取一样，以手机端的网页<https://m.douban.com/people/zeqingg/movie_charts>为例。此网页也为动态页面，采用和前一步相同的方法，经过分析得到了真实的请求的url为<https://m.douban.com/rexxar/api/v2/user/zeqingg/collection_stats?type=movie&for_mobile=1&ck=ykqX>（如图（4）所示）。

返回的json数据中有用的数据比如有观影数total\_collections，观看电影电视剧的时间total\_spent，在院线中的消费total\_cost，平均每周观看时间weekly\_avg，最常观看地区countries，最常观看类型genres等等。同样，把它们写入csv文件当中。

需要注意的是，豆瓣具有反爬虫机制，对单位时间内的访问次数有限制。如果在短时间内进行了大量访问就会被拦截。所以，使用python中time模块的sleep()函数，在每次请求后加入2秒钟的延迟，避开被拦截的可能。



图（4）

# 数据分析

## 问题描述

经过上一个步骤数据获取，我们已经得到了所需的数据集。现在的目标确定是对其所进行的分析需要什么样的方法。

## 地区分布

之前获取了友邻的常住地，对此可以做一个友邻分布地图，观察那些地区是友邻集中分布的地方。

在可视化方面，在一张真实的地图上绘制散点图、热力图可以算作比较好的方式。

## 男女比例、广播数

豆瓣的广播类似新浪微博的微博和微信的朋友圈，是用户展示自己状态的方式。通过对其进行分析。我们可以得知目标用户友邻的使用豆瓣的活跃度，将男性、女性分开统计，还可以得出该用户友邻中男女的比例，以及两种群体在发广播上的积极程度（从某种程度上也能反应使用豆瓣的积极程度）。

广播发布的数量很重要，所以在可视化时尽可能使用柱状图之类的能够反应数值大小的图表，同时还要能体现男女两个群体之间的不同。

## 注册时间

分析友邻的注册时间，可以知道该用户的友邻主要集中在那个时间注册，还可已看出该用户是否受到新用户的欢迎。

对于注册时间的分布，不太需要知道具体的人数，反而是更加关心其占比。因此，使用饼图来进行可视化会比较合适

## 观影数据

我们能得到的观影数据非常详细，有观影总数、平均每周观影数、观看所花的时间、在院线观影的消费等等。通常来说，这几个数据项具有较强的关联性，可以将这几者统一起来分析，不仅可以得出友邻在观影方面的投入，还可以看出这几项数据之间内在的关系。

所以在数据可视化方面，选择的图表要能够支持多个数据维度，并且能够反映各个数据维度之间的关系。

## 常看电影类型

统计每位友邻常看的几种电影类型，再统计每个类型受到多少友邻的喜爱。

此数据的可视化要求比较简单，只要能突出电影类型的名称，并能简单地反应数据之间数量关系即可。常用的图表如柱形图、饼图都是不错的选择的，而比如词云之类的特殊图表也不失为一种选择。

## 常看电影地区

常看电影地区的分析方式与常看电影类型的分析方式基本相同，都是通过统计得到几个最受欢迎的地区。

可视化的选择也与常看电影类型类似。

# 数据可视化

## 问题描述

确定了数据分析的方法，接下来就要考虑将数据可视化。Python中有许多数据可视化的包，比如著名的使用python中的matplotlib，但考虑到交互性和美观性，matplotlib展现的效果并不好。所以最终选择plotly和pyecharts。

## plotly

plotly是一个在线数据分析和可视化工具，具有python科学图形库。

plotly是一个在线的平台，所有的图表都通过网络传输。优点是图表美观，交互性强且文档详细。缺点是需要提供账号的api key，且不同级别的账号有不同的功能，许多功能需要付费。而绘制地图时还要注册Mapbox账号，提供access token。

在绘制友邻地区分布图方面，plotly不如pyecharts，很大原因是Mapbox对中国地区的支持并不友好。所以必须自己提供坐标的经纬度。为此，可以现在豆瓣账号的设置界面得到所有可设置的常住地，再使用地图类的产品（如高德地图）获取他们的经纬度，并储存到settings.py中。还有一个不足时地图颜色单调，数量关系并不清晰。

绘制友邻广播数图时采用金字塔图。要根据分析得到的数据的最大值来确定坐标的分度值，划分的区间也要随不同的数据集而改变，才能保证可视化得到的图表是有价值的。

绘制友邻注册时间分布图时采用饼图，这块比较简单，只需要统计各个年份的人数即可。

绘制友邻观影数据图时采用气泡图，它可以支持三个数据维度，可以把x 轴的维度设为观看时间，y 轴的维度设为消费，气泡的大小根据用户的观影数。

绘制友邻常看电影类型分布图时从用柱形图，在统计完各个电影类型的受众人数后，选择合适数目的类型来展示。

绘制友邻常看电影地区分布图时仿照豆瓣官方的观影数据图表，使用六个饼图，每个饼图只展示一个地区的占比。一个比较复杂就是六个图表不能够自动排列，需要为每个设置坐标，较为麻烦。

关于plotly的使用，更详细的官方文档请看参考文献[1]。

## pyecharts

pyecharts 基于python和Echarts。具有良好的交互性，精巧的图表设计。

pyecharts的优点是可以离线操作，支持链式调用，并且支持原生百度地图，在地理数据可视化方面效果出众。缺点是文档比较难读，图表个性化比较复杂，不同设备显示的图表可能会有些许不同。

在绘制地区分布图方面，pyecharts中国省市的数据全面，所以效果出色。加上还有热力图之类地图类型，展现出来的效果更好，更直观。

因为pyecharts没有金字塔图，所以绘制友邻广播数图使用层叠柱形图，效果也很好，甚至在男女总数的对比上会比金字塔图更加直观。

友邻注册时间分布图也采用采用饼图。在这方面，两个可视化操作基本相同。

在绘制友邻观影数据图方面，因为pyecharts没有气泡图，就采用3D散点图，依然是有三个维度。因为可以旋转，可以单独分析某两个维度的数据之间的关系。

绘制友邻常看电影类型分布图时也使用柱形图。

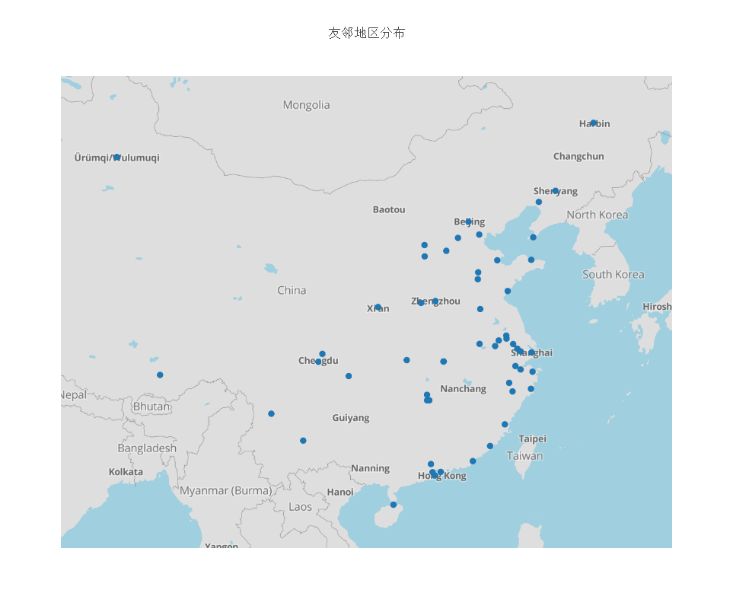
绘制友邻常看电影地区分布图时也使用六个饼图，但pyecharts显然更加复杂，而且不能单独设置颜色，在面对不同的数据集时可能要有不同的处理。

关于pyecharts的使用，更详细的官方文档请看参考文献[2]。

# 分析结果

分析的数据来自uid为zeqingg的用户的友邻，可视化以使用plotly为例。

## 地区分布

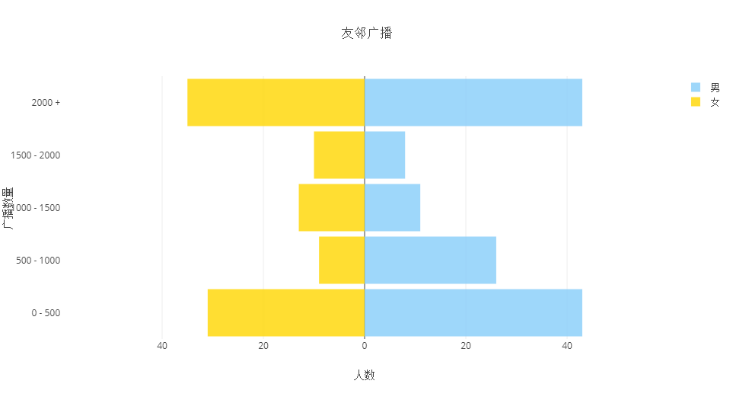


图中的蓝色标记代表该地区有友邻分布。可以清楚的看到，大部分友邻都分布与东部沿海省份，其中长三角和珠三角更是集中；中部地区零散地有友邻分布，但是西部地区，尤其是西北地区基本没有友邻分布。

查看了各个地区的人数，发现友邻主要集中于上海、北京、深圳、广州、杭州、南京、武汉这些一线城市。

这写数据表面该用户地友邻主要分布于经济发达，环境条件优异地大都市里。甚至可以推测，豆瓣所有用户的分布也是遵循此规律

## 男女比例、广播数



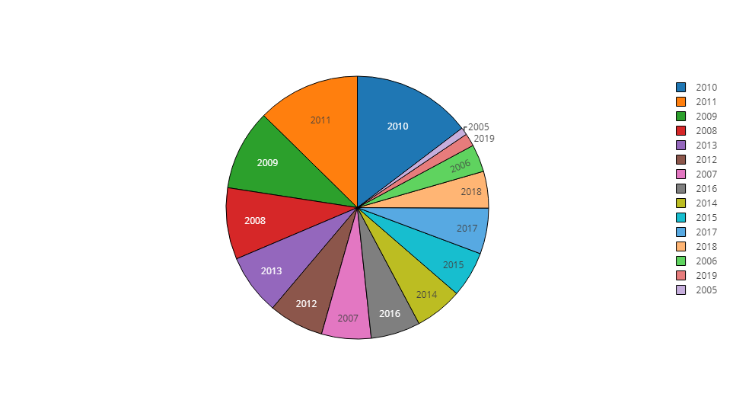
该图中蓝色代表男性，黄色代表女性，广播数的区间分别为[0, 500)，[500, 1000)，[1000, 1500)，[1500, 2000)，[2000, ∞)，人数的分度为20。

广播数主要集中于[0, 500)和 [2000, ∞)这两个区间，就是要么基本不使用广播功能（很可能是“僵尸号”），要么经常使用广播功能。

男性总体多于女性，各个区间总体来看比例都相差不大，除了[500, 1000)这个区间男性较多。

总体来说，该用户的男性友邻更多，且更加活跃。总共有三分之一的友邻发的广播数超过2000，属于较活跃的用户。

## 注册时间



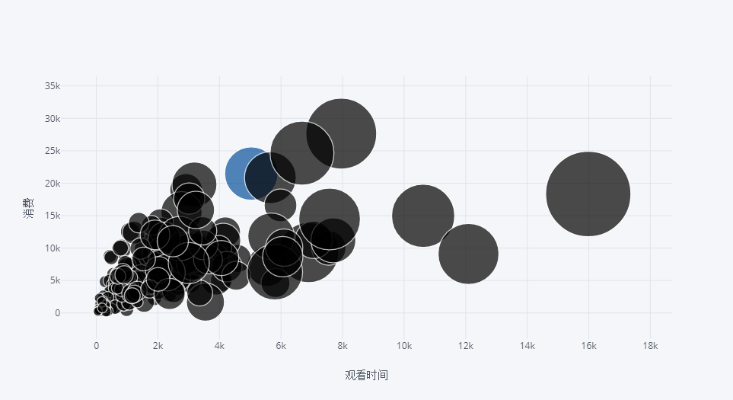
总体上来看，友邻的注册年份分布比较平均，其中2008、2009、2010、2011这四年人数较多，2005、2019这两年人数较少。

目标用户是在2013年注册，人数较多的年份都在这附近，可以推测该用户注册后就受到了不少其他用户的关注，或者也可以认为这几年豆瓣用户增长比较快速。

对于人数较少的两年。其中2005年时豆瓣刚成立的一年，本身的用户数就少。而2019才过去一半，用户量也不会太多。

而其他年份的占比也都不会少，说明该用户一直有受到新用户的关注，对应的很可能时因为其有持续的产出。

## 观影数据



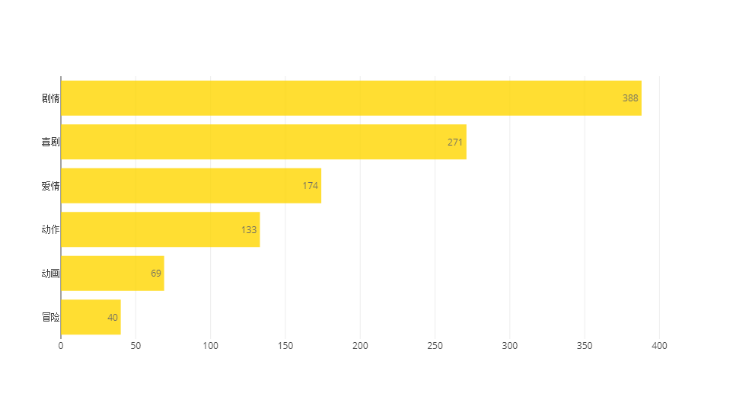
改图x 轴的维度设为观看时间，y 轴的维度设为在影院的消费，气泡的大小根据用户的观影数。

观察x轴和y轴的关系大概一个一次函数的关系，说明友邻的观看时间与在影院的消费是一个正比的关系。而因为观影数的增长，气泡也随之变得越来越大。

但是可以看到有些靠右上的气泡却比较小，说明这类友邻相比于其他人，大部分电影都是在影院观看的。

偏向右下的气泡说明那些友邻可能比较喜欢观看一老片，而不是去影院观看新片。

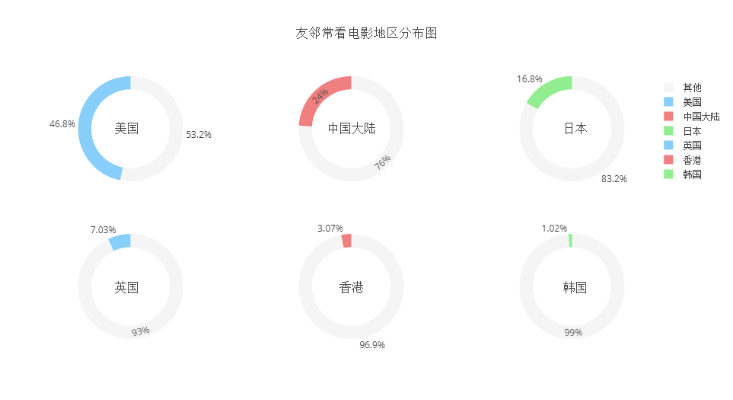
## 常看电影类型



剧情、喜剧、爱情、动作、动画、冒险都属于常见的电影类型。大部分友邻喜欢看剧情片，也喜欢看喜剧，可以说明他们既可以接受较严肃的电影，也可以接受比较娱乐的电影。

注意到其中有一部分的用户喜欢观看动画，可以认为这些友邻的年龄会比其他人小。

## 常看电影地区



有近一半的友邻最喜欢观看美国的电影。这个结果不难想到。因为美国的电影工业发展较早，体系完备，放眼整个世界，好莱坞都算是数一数二的电影中心。

豆瓣用户基本都为中国人，所以大陆的电影占到四分之一也算是正常的结果。

排第三的地区是日本，同为亚洲地区的国家且是电影大国，日本的电影有很大一部会被引进国内。再加上之前常看电影类型的分析，知道该用户有很多友邻喜欢观看动画，而这些动画很可能大部分都是日本动画。

之后受欢迎的地区是英国、香港、韩国。

# 结束语

这个项目能够对豆瓣的友邻群体做一个简要的分析，并且得到一些有实际意义的图表。

但是因为时间关系和个人能力的问题，还有许多的数据没有去挖掘。比如友邻观看电影的年份分布，观看日历，喜欢的导演，喜欢的演员，用户之间的相同兴趣爱好等等。甚至可以进一步拓展，用户群体可以拓展到所有豆瓣用户，分析的内容可以从电影拓展到读书、音乐等。

而在数据的爬取方面，可以编写一个模拟登录程序（还可以加入验证码的识别），而不是通过复制Cookie的方式模拟登录，使得程序使用起来更加方便。

在数据的可视化方面，可以加入更多的包，比如matplotlib，以及更多的图表。

参考文献:

1. Python Graphing Library, Plotly. <https://plot.ly/python/>, 2019.
2. pyecharts - A Python Echarts Plotting Library. [https://pyecharts.org/](https://pyecharts.org/#/), 2019.