# 1网络爬虫

为应对京东网站的反爬虫机制，我们针对其使用动态网页的AJAX数据的特点，基于selenium和pyquery框架，爬取了京东关于华为MateBook D14 2022款产品（等产品，这里光京东光这个产品，是没有几万条数据的，不要出问题，要么多个产品要么多个

）的20935条评论数据，获取了每条评论的评论者、会员、星级、评论内容、大小、版本、点赞数、评论回复数、发表时间等数据。

PS：爬取的中关村1230条数据，有以下数据：发表时间、评论者、主页链接、星级、点赞数、评论回复数、评论内容。**因为数据不是特别统一，为了方便做预处理，中关村的数据没用进去与京东数据一起分析。**

# 2 数据预处理

我们基于jieba、pandas、re等第三方库，进行数据的清洗，包括：

## 数据一致性

由于根据爬取到京东评论的实际情况，我们通过处理了版本（型号类型）、大小（内存容量），使数据保持统一格式，符合使用标准。

为了使星级更加直观，我们对评论星级进行处理，使之归为三类“积极”5分、“中立”3分、“消极”1分。为4的评分，因此，我们将4分归为5分。星级为2的评分，将2分归为1分。

## 数据观测

**（本部分，有部分图，因为太多电脑型号问题，不好看，可忽略）**

我们初步完成了对星级（各评论打分的占比、星级在各型号的分布情况）、评论内容的长度（字数）、评论的发表时间分布情况等数据的统计和观察

基于Matplotlib、seaborn（一个基于matplotlib更便捷画图的Python第三方库），绘制以下数据统计图表：

如下，可以看出，星级处理前，评论为5分好评的占了75.9%。……

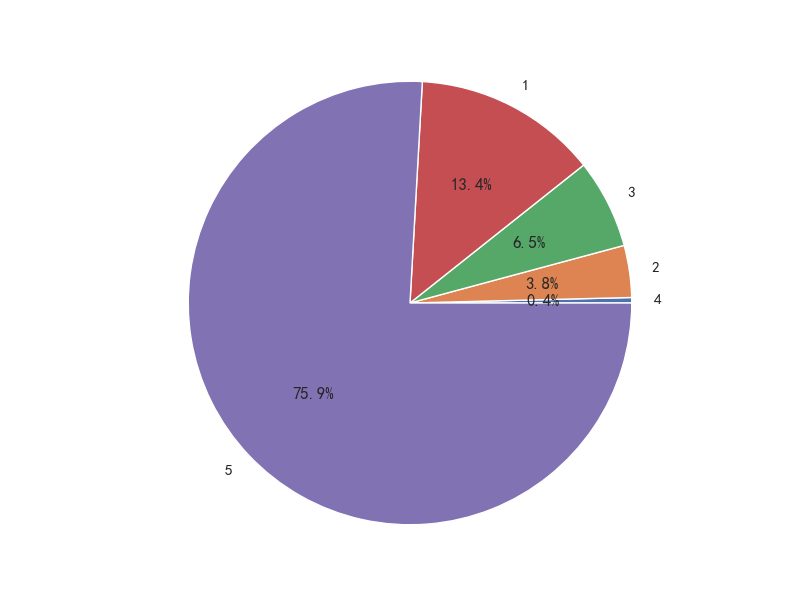


图1 观察评论打分情况

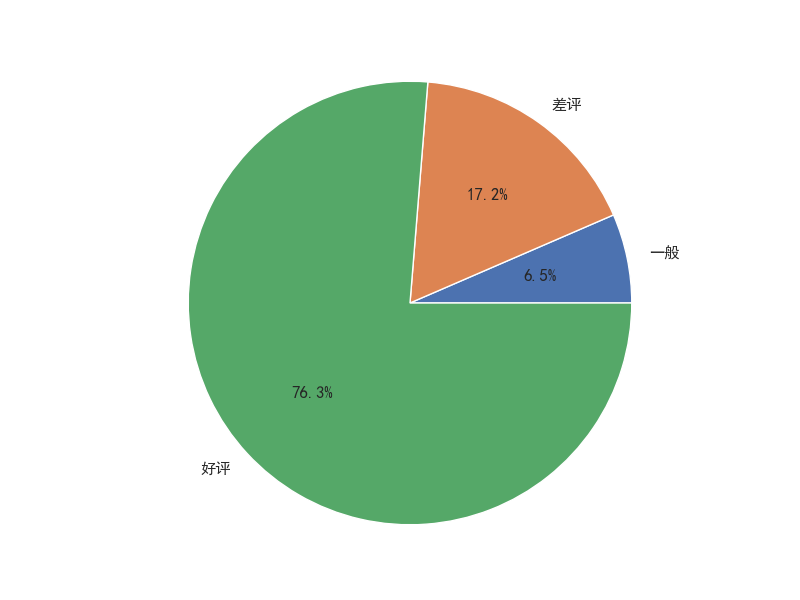


图2 观察评论打分处理情况

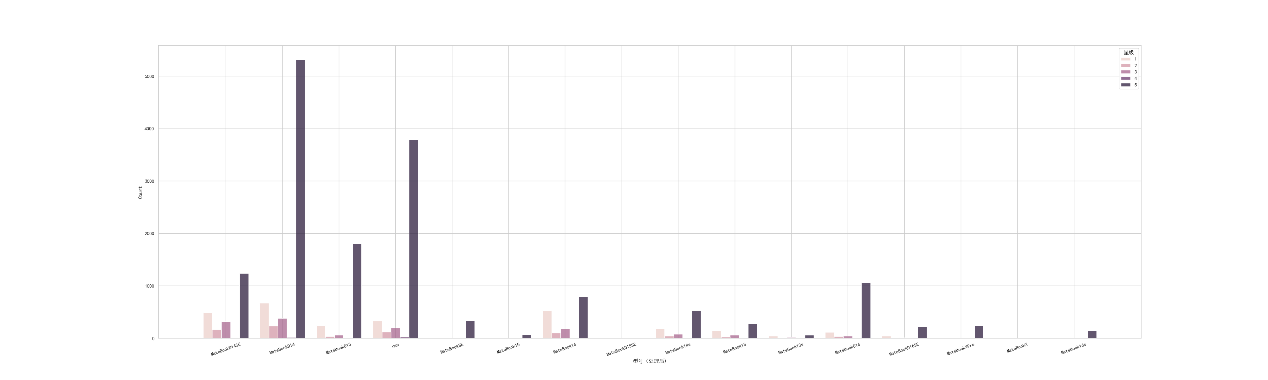


图3 每个型号商品好评与差评的数量分布

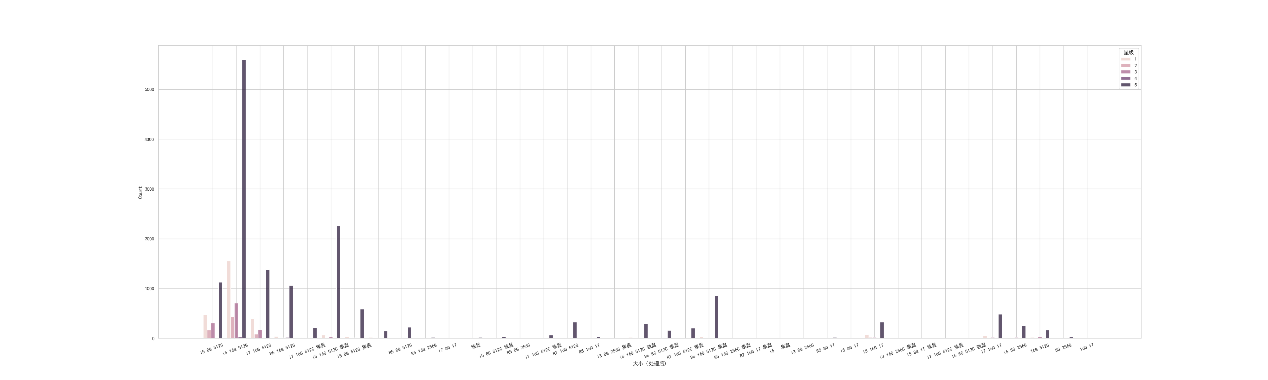


图4 每个大小商品好评与差评的数量分布

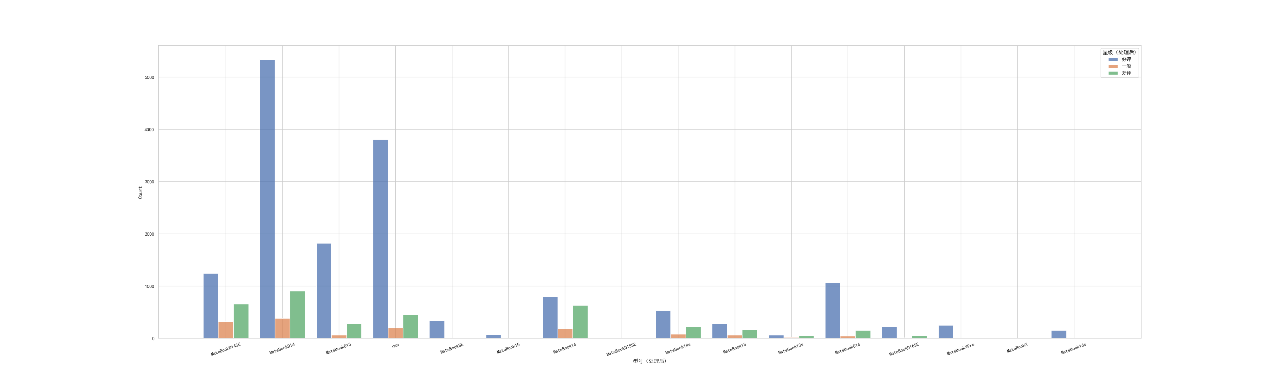


图5 每个型号商品好评与差评的数量分布（星级处理后）

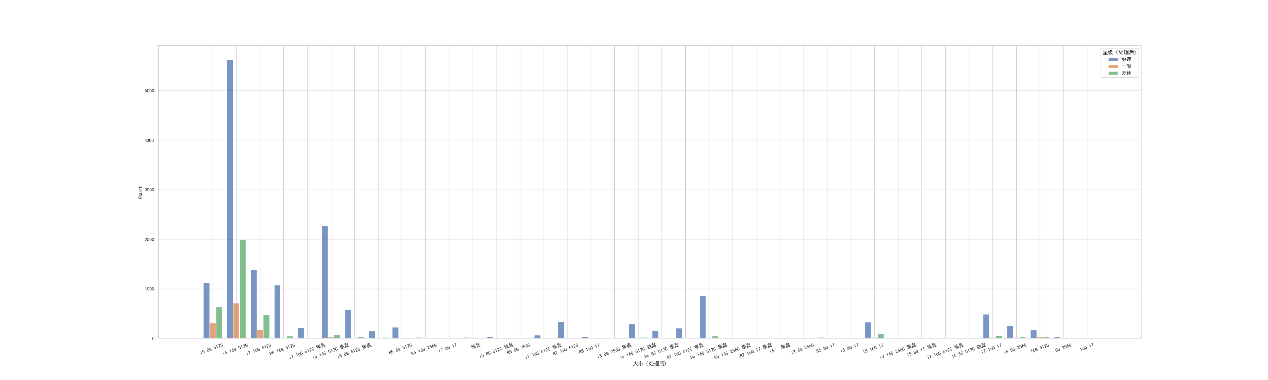


图6 每个大小商品好评与差评的数量分布（星级处理后）

观察每月的商品数量，主要集中在近几年，可以看出，在六月份、11月份有短暂的购买小高峰。由于京东爬取评论数量限制（优先展示近期评论），爬到的数据大量集中在近两个月。

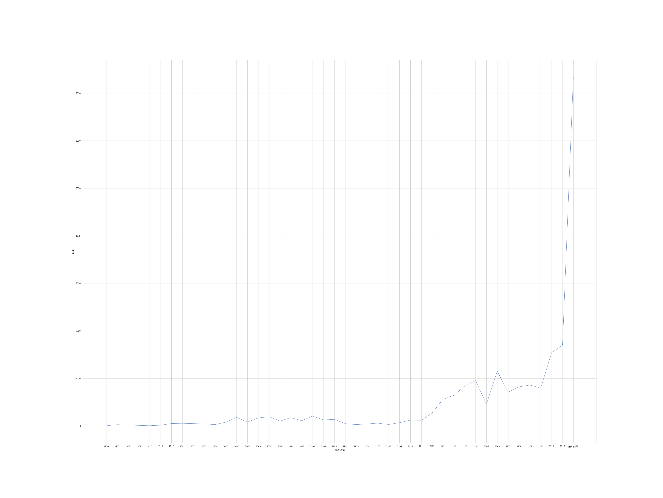


图7 每月的商品数量情况

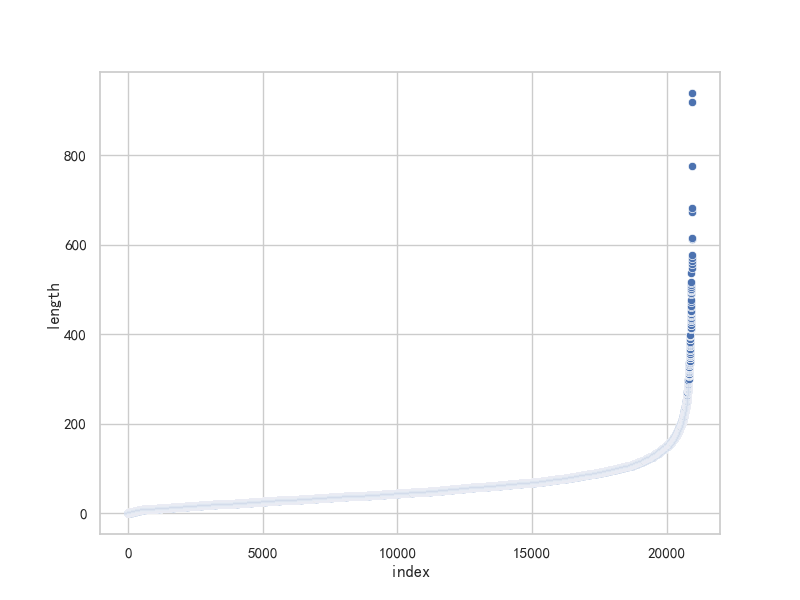
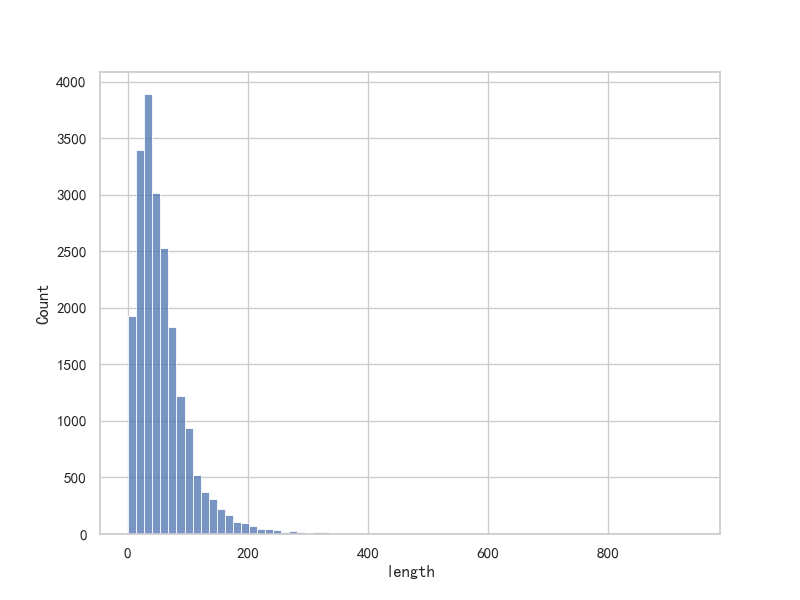


图8 评论内容长度分布情况（散点图）



9 评论内容长度分布情况（柱状图）

## 处理无效值

1、删除过长、过短评论（大于400或者小于10字）

2、删除重复评论

3、删除标点符号或英文字母过多的评论

4处理HTML符号

5、处理“\*”号，（可能是违禁语、价格、品牌），避免jieba分词出现多个\*号

6、将“京东”、“电脑”字样去除（所有商品都是电脑，都来自于京东）

7、根据京东评论的实际情况，剔除以下无效内容：“运行速度：屏幕效果：散热性能：轻薄程度：外形外观：其他特色：此用户未填写评价内容”等。**（PS：这些内容是京东自动生成的每行的评论主题，用于提醒当时评论者对每方面进行评论的思考，与评论者内容无关）**

## 删除缺失值

评论内容、星级等重点分析内容缺失的评论，予以剔除

（值得注意的是，发表时间、型号等信息非重点分析内容：在京东的数据中，这些内容常常爬不到，所以大部分数据条都缺少这类内容，不予剔除）

最后，预处理后，删掉了几千条数据吧，京东的20935条，最后分析的时候好像剩下17325条。

## 分词

我们综合运用jieba库，进行分词和删除停用词、词性识别等操作，并统计了每条评论的形容词数、动词数、字数、句子数、句子平均长度、主观句子数、客观句子数。此外，为了避免删除停用词后，评论出现NaN值，在分词后我们还再进行了一次删除缺失值操作。

此部分简单对整条评论进行了分析：（基于第一次做的内容）

|  |  |
| --- | --- |
| **列名** |  |
| 提到产品特征的句子数 |  |
| 提到产品特征的句子 |  |
| 产品特征数/提到产品特征的句子数 |  |
| 提到产品特征数/句子数 |  |
| 提到产品特征句子数/句子数 |  |
| 提到顾客需求数 |  |
| 情感词典预测（整条的） |  |
| 情感词典预测分数（整条的） |  |
| 情感词典预测可信度 |  |
| BERT情感分类 | 这三个预测模型，做得比较早，不管有没有用，先放着。[参考后面的章节](#_（可能没有用）初步情感分析) |
| SVM情感取向 |  |
| 星级有效性分析 | 这个分析是对上面预测的积极消极与，评论者打的星级进行对比，如果差别过大，可能就是无效评论（如，打了五分，但是评论内容说的都是坏话，预测为消极评论，与原来打的星级五分不符）[参考后面的章节](#_（可能没有用）初步情感分析) |
| LSTM情感取向 |  |
| 星级有效性分析 |  |

## 数据描述性统计

我们对点赞数、评论回复数、字数、句子数、句子平均长度等信息进行了数据描述性统计。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **点赞数** | **评论回复数** | **字数** | **句子数** | **句子平均长度** |
| **Min** | 0 | 0 | 3 | 1 | 0.909090909 |
| **25%分位数** | 0 | 0 | 22 | 1 | 14.5 |
| **50%分位数** | 0 | 0 | 37 | 1 | 23 |
| **75%分位数** | 0 | 1 | 57 | 2 | 37 |
| **MAX** | 441 | 215 | 382 | 31 | 382 |
| **均值** | 0.67 | 0.69 | 45.66 | 1.97 | 28.77 |
| **方差** | 37.72 | 6.29 | 1256.44 | 3.06 | 463.68 |
| **标准差** | 6.14 | 2.51 | 35.45 | 1.75 | 21.53 |
| **均值±标准差** | 0.67±37.72 | 0.69±6.29 | 45.66±1256.44 | 1.97±3.06 | 28.77±463.68 |

## 词频统计

华为 4049

不错 3533

速度 3347

很快 2841

开机 2537

运行 2383

喜欢 2336

屏幕 2220

外观 2183

轻薄 2137

笔记本 2034

办公 1997

收到 1736

客服 1728

流畅 1701

满意 1518

好看 1501

没有 1470

散热 1273

物流 1262

鼠标 1231

清晰 1223

感觉 1165

颜值 1087

真的 1087

手机 1066

值得 1038

特别 1020

购买 1009

支持 997

包装 869

键盘 855

手感 836

……（后面还有，不一一列了）

# （可能没有用）初步情感分析

构建了基于SVM（支持向量机）的二分类模型，将评论内容初步分类为积极和消极两类，用于主题分析。训练SVM模型，目前是直接采用网络上别人已经经过大量训练集数据训练好的模型（较完备）。

类似的，构建了基于BERT模型的三分类模型

LSTM（循环神经网络）的三分类模型

比较分析来看：BERT模型预测的较为保守，多中性评论

似乎SVM模型正确率略高一点，但是三者各有千秋，预测准确是跟他们训练的数据集有很大关系。

此部分用途，想着，要么取一个模型要么三个模型一起（有点多），是可以作为预处理一部分的，就是无效评论的剔除，不知道有没有用，但是实际操作也没有去剔除了，所以这部分保留，作为参考。

此外，这是对整条评论的预测，也可以用于后面根据顾客需求分类后的单条评论预测，可以再思考需不需要。

# （可能没有用）初步主题分析

此部分，可以看着玩玩。刚学，不是搞得很好。

对数据初步的主题词提取，利用词频和TF-IDF*(term frequency-inverse document frequency)*挖掘出正负文本中的关键点情况，并绘制云词图。



图10 好评主题词云图



图11 差评主题词云图

使用sklearn中的TruncatedSVD进行文本主题分析。通过主题分析，我们可以得到一个语料中的关键主题，并得到主题对应的关键词以及代表性文本，如下。

此外，主题分析后还可以看出，各个词语在主题中的重要程度，各个文章在各个主题上的倾向程度。

积极主题

主题1:不错 速度 很快 华为 运行 喜欢 外观 开机 轻薄 好看

主题2:华为 支持 不错 笔记本 手机 国货 值得 产品 国产 品牌

主题3:不错 感觉 好看 手感 东西 质量 喜欢 整体 收到 客服

主题4:喜欢 特别 好看 客服 超级 颜色 收到 满意 真的 购买

主题5:很快 物流 值得 客服 收到 满意 购买 速度 质量 包装

Counter({0: 9804, 1: 1119, 3: 878, 2: 778, 4: 758})

结论：

运行速度快

外观好看、手感轻薄

国货品牌

质量、包装不错

客服满意

物流速度快

消极主题

主题1:没有 鼠标 差评 客服 华为 垃圾 降价 不好 真的 体验

主题2:鼠标 没有 配件 鼠标垫 裸机 充电器 连个 标配 不配 不送

主题3:降价 刚买 保价 差评 几天 一个月 差价 星期 昨天 鼠标

主题4:填写 内容 用户 评价 降价 鼠标 保价 刚买 没有 三个

主题5:垃圾 鼠标 华为 不要 东西 真的 降价 刚买 玩意 不带

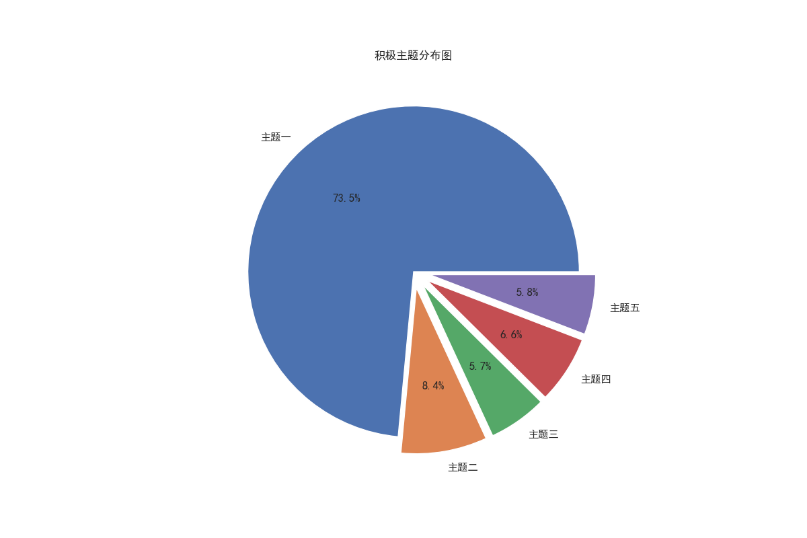
Counter({0: 2386, 2: 254, 4: 171, 3: 127})

结论：

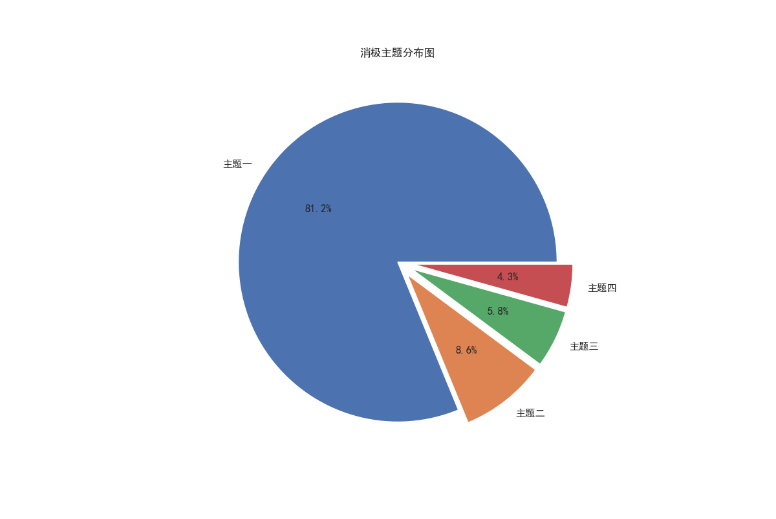
不保价，买完就降价

标配只有充电器，裸机没有配鼠标

此外，我们对各主题分布，制作饼图。



13 积极主题分布图



14 消极主题分布图

# 3 产品特征提取

产品特征数、其他产品数、提到产品特征的句子数、产品特征数/提到产品特征的句子数、提到产品特征数/句子数、提到产品特征句子数/句子数等信息

特征词分类为顾客需求

|  |  |  |  |
| --- | --- | --- | --- |
| 顾客需求 | 提到此顾客需求的评论条数 | | 特征词 |
| 显示效果良好 | 3867 | 屏幕 桌面 触摸屏 显示屏 触屏 分辨率 亮度 显示 蓝光 界面 色彩 画质 清晰度 画面 内屏 分屏 饱和度 色泽 像素 | |
| 网络信号稳定 | 876 | 网络 信号 上网 WLAN 蓝牙 双频 WIFI 红外 无线 有线 网速 路由器 宽带 网卡 连接 | |
| 电池耐用度高 | 2700 | 电池 待机 充电器 充电 快充 电量 耗电 用电 时长 续航 发热 发烧 毫安 电源 省电 发烫 散热 | |
| 外观简洁好看 | 7508 | 外观 外形 体积 缝隙 工艺 颜色 外表 线条 机身 样子 造型 设计 个性 外壳 尺寸 机型 颜值 美观 漂亮 好看 美 丑 划痕 厚度 边框 后盖 后壳 大小 耐看 简单 简洁 时尚 商务 重量 | |
| 运行流畅 | 7610 | 速度 处理器 CPU 骁龙 英特尔 四核 八核 线程 多核 GPU 内存 容量 储存 空间 RAM ROM 反应速度 延迟 存储 SD 系统 操作 流畅 window 黑屏 Bug 兼容 闪退 死机 启动 卡机 不卡 应用 程序 功能 软件 导航 网页 浏览器 地图 娱乐 信息 玩游戏 卡顿 显卡 | |
| 零配件齐全 | 3569 | 配件 配备 壳 膜 玻璃膜 保护膜 套 电脑包 耳机 数据线 电源适配器 鼠标 键盘 插头 充电器 包装 | |
| 机身轻巧纤薄 | 6098 | 手感 重量 质感 触感 材质 做工 材料 薄 轻 边框 | |
| 人机界面友好 | 1749 | 指纹 解锁 人脸识别 语音识别 多屏协同 移动应用引擎 云服务 云盘 HMS 电视剧 音箱 听歌 爆音 杂音 听筒 通话 噪音 音效 闹铃 摄像 弹窗 风扇 扬声器 音量 | |
| 售后服务 | 4778 | 售后 服务 物流 客服 态度 口碑 顺丰 卖家 维修 保修 质保 包装 送货 及时 收货 上门 配送 | |

注意：

1、这个表，有部分重复的特征词在里面，具体移步excel。

2、有些选取的特征词，后面统计出来也没有任何次数，所以，要不要考虑删掉

此外，特征词选取，自我感觉，可能不够全面。

# 按顾客需求分类后的情感分析

按顾客需求分类，这部分，基本参照师姐的思路，不多赘述。

分类后

注意列名：

|  |  |
| --- | --- |
| **pos** | **neg** |

这两列是师姐的算法

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **情感词典预测** | **情感pos数** | **情感nes数** | **情感总分数** | **情感词典预测可信度** | **归一化处理pos** | **归一化处理nes** | **取整处理pos** | **取整处理nes** |

这个是我自己弄的

本部分中使用情感词典进行情感分析的思路为：

1. 对文档分词，找出文档中的情感词、否定词以及程度副词

2. 然后判断每个情感词之前是否有否定词及程度副词，将它之前的否定词和程度副词划分为一个组

3. 如果有否定词将情感词的情感权值乘以-1，如果有程度副词就乘以程度副词的程度值

4. 最后所有组的得分加起来，大于0的归于正向，小于0的归于负向。(得分的绝对值大小反映了积极或消极的程度)'''

**重点在于，找出文章的情感词，对应的在一个BosonNLP情感词典，找到对应得分去计算。**

关于BosonNLP情感词典可参考：

<https://blog.csdn.net/liuxueyingwxnl/article/details/119860962>

里面算法没细看，但是基本流程应该是一样的

# 数据归一化和取整

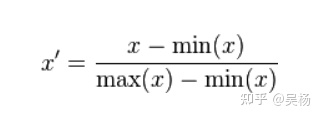
**此次采用的是：****最大最小标准化（Min-Max Normalization）的方法**

数学原理，将数组归一化到[a,b]区间

1.首先找到原样本数据x的最小值Min以及最大值Max

2.计算系数：k = (b - a) /(Max - Min)

3.得到归一化到[a,b]区间的数据 : y = a + k(x-Min) 或者 y = b + k(x-Max)



参考：

1、[python将数据归一化到任意区间的数学原理以及实现方法\_归一化到某个区间\_qq\_45034164的博客-CSDN博客](https://blog.csdn.net/qq_45034164/article/details/128172279)

2、[pandas数据归一化方法\_AI界扛把子的博客-CSDN博客\_pandas 归一化](https://blog.csdn.net/qq_16792139/article/details/117112650)

如有不妥，可参考其他归一化方法

参考：[如何理解归一化（normalization）? - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/424518359)

**我的思路比较简单：**

采用了最大最小标准化方法，这个用excel公式也能解决的，就是在整体里面，把数据映射在我们想要的区间里面。如果得出数据分布不妥也可以采用其他方法归一化，不过我不知道有没有效果。

取整处理，也是excel公式=CEILING可以做的，

值得注意的是：这里的取整是大于零的按每0.5取整，小于零的按每-0.5取整。

所以零的数量会比较少，我估计，就是只有本身等于零，取整后才能是等于零吧。

一般情感计算应该很少是零的吧，所以很少。

这避免了全部按0.5取整，导致-0.5到0的数据都是被取整为情感值为0。

大概就这么多。。。