数据分析报告

张驰 2022010754 zhang-ch22@mails.tsinghua.edu.cn

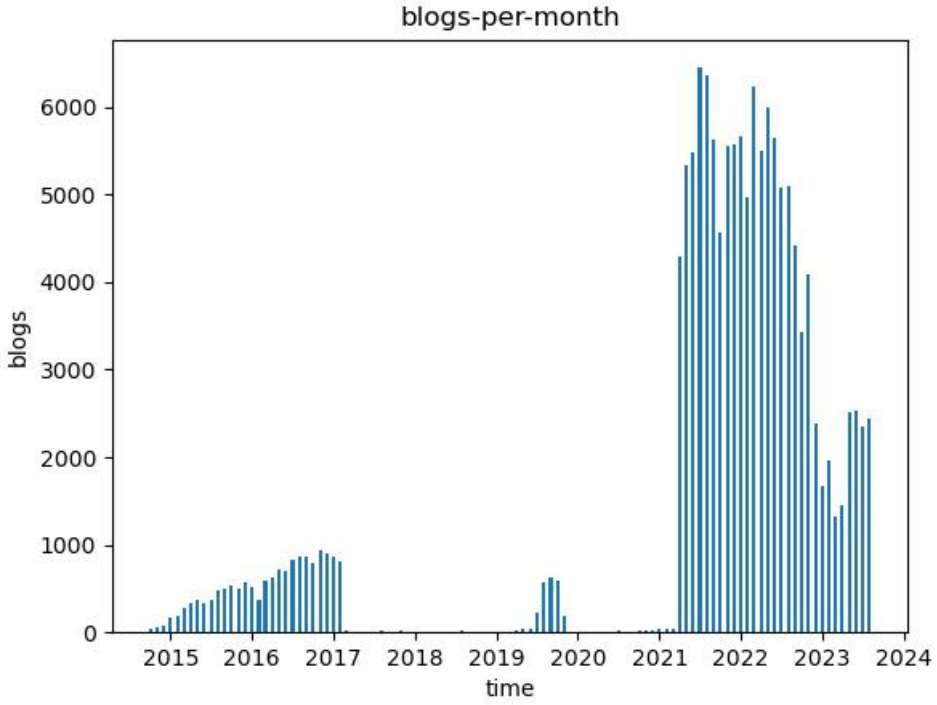
本次数据分析数据来源为从新浪新闻科技板块滚动新闻网页 *https://news.sina.com.cn/roll/#pageid=153&lid=2515&etime={etime}&stime={stime}&ctime={stime}&date={datestr}&k=&num=50&page={page}* 能够获得的**所有新闻，时间范围为从2014-10-16到2023-8-28，共142485条。**下面将从三个方面分析这些数据。

每一部分的代码直接附在这一部分之后。但全文代码需要按顺序运行，各部分代码不能分别运行。

```
import matplotlib.pyplot as plt
import json
import ujson
import jieba
import numpy as np
from wordcloud import WordCloud
from datetime import datetime

# 读取分散的json文件（每条新闻一个）中的信息汇总成一个json文件
bigdic = {
    "cc":[],
    "cu":[],
    "pics":[],
    "len":[],
    "date":[],
}
for i in range(1,142486):
    print(i)
    dic = {}
    try:
        if i<86326:
            with open(f"allfiles/{i}.json", encoding='utf-8') as f:
                dic = ujson.load(f)
        else:
            with open(f"allfiles_86326/{i}.json", encoding='utf-8') as f:
                dic = ujson.load(f)
    except: continue
    try: dt = datetime.strptime(dic["time"],"%Y-%m-%d %H:%M:%S")
    except: continue
    bigdic["cc"].append(int(dic["cc"]))
    if int(dic['cc'])>5000:
        with open("eee.txt",'a') as f:
            f.write(str(i))
            f.write(" ")
            f.write(str(dic['cc']))
    bigdic["cu"].append(int(dic["cu"]))
    bigdic["pics"].append(int(dic["pics"]))
    bigdic["len"].append(len(dic["content"]))
    bigdic["date"].append(str(dic["time"]))

with open("bigdic.json",'w',encoding='utf-8') as f:
    json.dump(bigdic,f,ensure_ascii=False, indent=4)
```

一、新浪新闻科技板块滚动新闻网页整体情况

本批数据涵盖了新浪新闻科技板块滚动新闻网站的全部新闻，可以根据不同时间的新闻发布数量和新闻热度（新浪新闻并没有显示热度/阅读量，这里以评论数代替）看出新浪网站整体的热度。其中，网站每个月发布的新闻数量如下图：

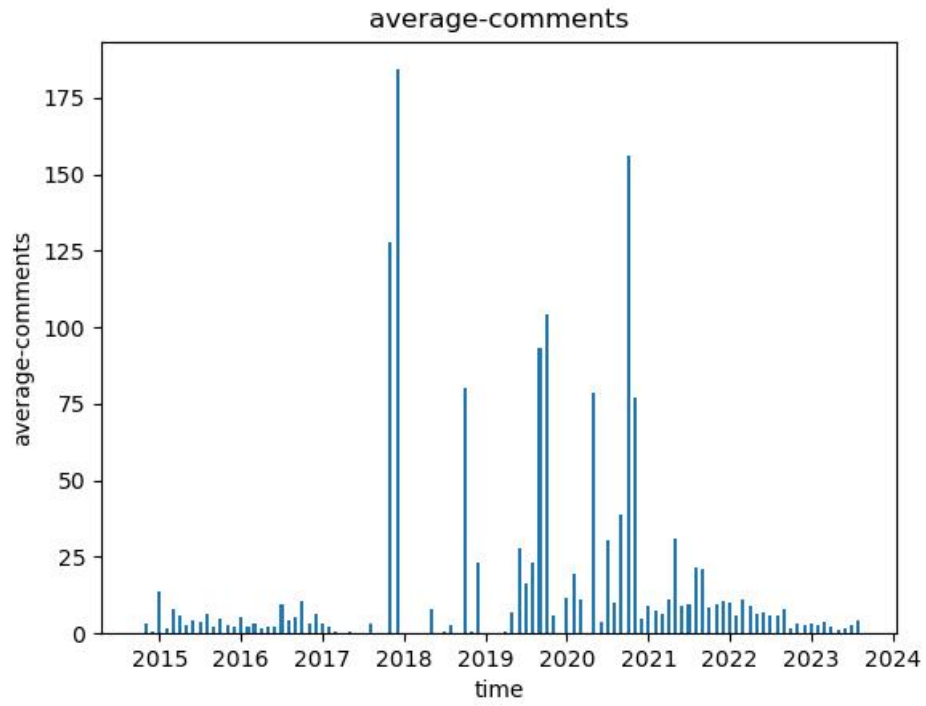


网站从2014年底开始发布新闻，新闻数量逐月增长到2017年初，在2017到2021年这段时间里只有2019年发布了少量新闻。2021年网站重新启用，随后发布新闻数量暴增，达到每月6000余条的最高水平。2022年到2023年新闻数量下降，但近几个月又有反弹。

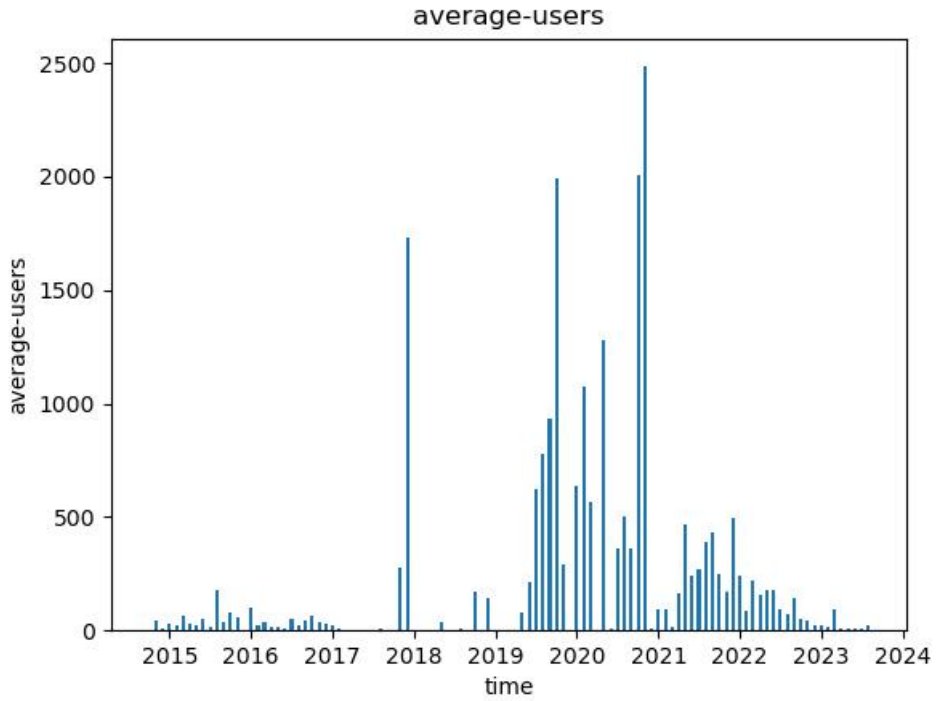
在2017-2021年，新浪新闻为何选择启用/弃用这个滚动新闻板块我们不得而知。不过对比近两年与2015-2017年的数据，可以认为近两年的科技正在高速发展。

而如果从每条新闻的平局评论数和参与用户（评论用户+为评论点赞的用户）数来看：

> 平均评论数：



> 平均参与用户数：



可以看出，2015-2017新闻热度较低，可能是本网页未充分普及。之后一般情况下，发布新闻数量多的月份，新闻的平均点赞量会较少。这可能是因为用户每天浏览的新闻是有限的，每天发布的新闻过多，会导致每条新闻的平均被浏览量下降。但近一年虽然新闻发布量有所下降，而新闻平均热度低迷甚至还在降低，可能是因为更多人选择用其他途径获得新闻。

```
# 绘制按月统计新闻柱状图
with open("bigdic.json", encoding='utf-8') as f:
    bigdic = ujson.load(f)

datestrs = bigdic["date"]
dates = [datetime.strptime(datestr,"%Y-%m-%d %H:%M:%S") for datestr in datestrs]
```

```
yms = [date.replace(day=1, hour=0, minute=0, second=0, microsecond=0) for date in
dates]
ymcount = {}
for ym in yms:
    if ym in ymcount:
        ymcount[ym]+=1
    else:
        ymcount[ym]=1
plt.title("blogs-per-month")
plt.xlabel("time")
plt.ylabel("blogs")
plt.bar(ymcount.keys(),ymcount.values(),width=15)
plt.savefig("blogs-per-month.jpg")


# 绘制评论数统计图
ymcc = {}
for ym,cc in zip(yms,bigdic['cc']):
    if ym in ymcc:
        ymcc[ym]+=cc
    else:
        ymcc[ym]=cc

avcc = [ymcc[ym]/ymcount[ym] for ym in ymcc]
plt.title("average-comments")
plt.xlabel("time")
plt.ylabel("average-comments")
plt.bar(ymcc.keys(),avcc,width=15)
plt.savefig("average-comments.jpg")

# 绘制参与用户数统计图
ymcu = {}
for ym,cu in zip(yms,bigdic['cu']):
    if ym in ymcu:
        ymcu[ym]+=cu
    else:
        ymcu[ym]=cu

avcu = [ymcu[ym]/ymcount[ym] for ym in ymcu]
plt.title("average-users")
plt.xlabel("time")
plt.ylabel("average-users")
plt.bar(ymcu.keys(),avcu,width=15)
plt.savefig("average-users.jpg")
```

二、近年科技热点变化

通过本批数据的词分布可以看出近几年的科技热点。由于2017、2018、2020年的新闻数量较少，本文没有纳入分析。本文统计了其余6年的新闻标题，做成词云如下：

> 2015:


苹果
手机
中国
公司
智能
互联网
推出
索尼
如何
市场
平台
三星
电商
小米
iPhone
微软
网络
创业
移动
用户
曝光
谷歌
服务
Uber
Android
App
O2O
全球
美国
投资
运营商
电视
游戏
收购
体验
专车
iPhone 6s
问题
支持

> 2016:


苹果
手机
iPhone
中国
三星
市场
如何
公司
曝光
互联网
VR
微软
平台
投资
美国
小米
智能
用户
直播
网络
家电
谷歌
乐视
科技
百度
技术
未来
游戏

> 2019:

苹果
华为
中国
回应
5G
亿美元
公司
美国
用户
市场
手机
亚马逊
腾讯
CEO
阿里
谷歌
iPhone
Facebook
特斯拉
直击
员工
电子
企业
技术
收购
百度
投资
互联网
视频
服务
全球
业务
计划

2021:

苹果
特斯拉
华为
中国
马斯克
推出
回应
腾讯
市场
公司
小米
美国
手机
如何
游戏
芯片
微软
宇宙
视频
曝光
全球
宣布
支持
未来
iPhone
亚马逊
上市
用户
计划
问题

2022:

> 2023:

可以看出，这八年间，“苹果手机”始终是热度居高不下的热词。此外，还可以看出“互联网”“电商”等词热度的衰减，标志着以电商为核心的一次互联网浪潮的衰退。2018年中美贸易战打响，于是“华为”、“5G”、“美国”等词成为2019年的热词。2021年特斯拉股值一路飙升超越众多传统车企，使得“特斯拉”、“马斯克”成为年度热词。2022年底ChatGPT发行，开启新一轮AI热潮，使“AI”、“ChatGPT”、“马斯克”成为2023年热词。

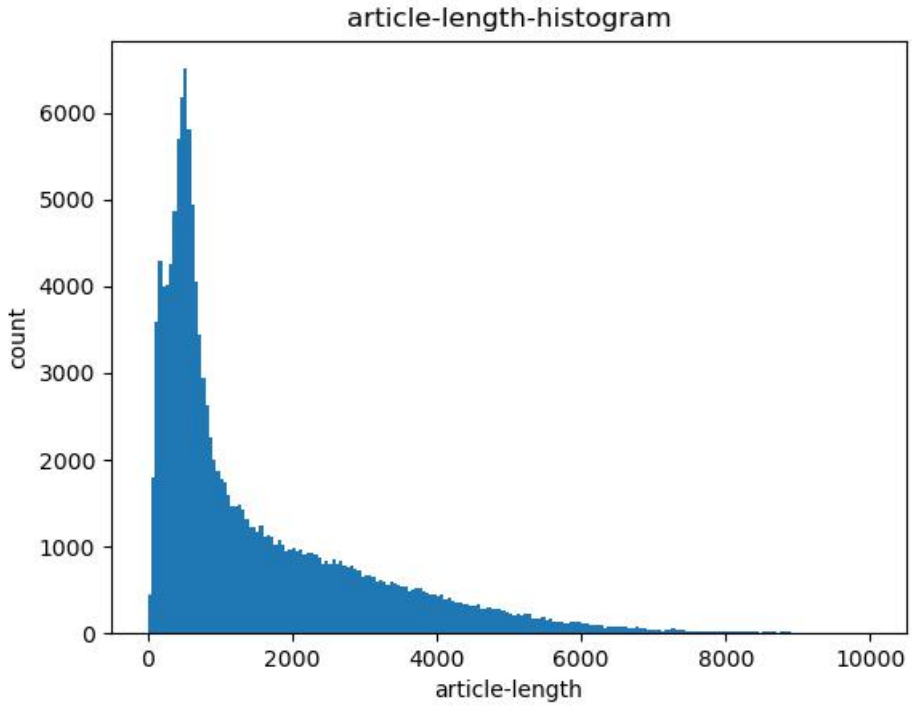
```
# 获取新闻标题，写入一个json文件
titledic = {
    "title":[],
```

```
    "time":[],
}
for i in range(1,142486):
    print(i)
    dic = {}
    try:
        if i<86326:
            with open(f"allfiles/{i}.json", encoding='utf-8') as f:
                dic = ujson.load(f)
        else:
            with open(f"allfiles_86326/{i}.json", encoding='utf-8') as f:
                dic = ujson.load(f)
    except:continue
    try: dt = datetime.strptime(dic["time"],"%Y-%m-%d %H:%M:%S")
    except: continue
    titledic["title"].append(dic["title"])
    titledic["time"].append(dic["time"])

with open("titledic.json",'w',encoding='utf-8') as f:
    json.dump(titledic,f,ensure_ascii=False, indent=4)


# 绘制词云
with open("titledic.json", encoding='utf-8') as f:
    titledic = ujson.load(f)
wordlist = {
    15:[],
    16:[],
    17:[],
    19:[],
    21:[],
    22:[],
    23:[],
}
for title,time in zip(titledic["title"],titledic["time"]):
    for i in wordlist:
        if time[:4]==f"20{i}":
            wordlist[i] += [word for word in jieba.cut(title, cut_all=False) if
len(word)>=2 and word!="发布" and word!="评测"]

wordcloud = WordCloud(font_path="C:\WINDOWS\Fonts\simhei.ttf",
background_color="white", width=800, height=600)
processed_text = {}
for i in wordlist:
    processed_text[i] = " ".join(wordlist[i])

for i in wordlist:
    wordcloud.generate(processed_text[i])
    plt.figure(figsize=(8, 6))
    plt.imshow(wordcloud, interpolation="bilinear")
    plt.axis("off")
    plt.savefig(f"wordcloud20{i}.jpg")
```

三、关于新闻长度、热度的一些探究

本文统计了本批数据的新闻长度和评论数，对其稍作探究。

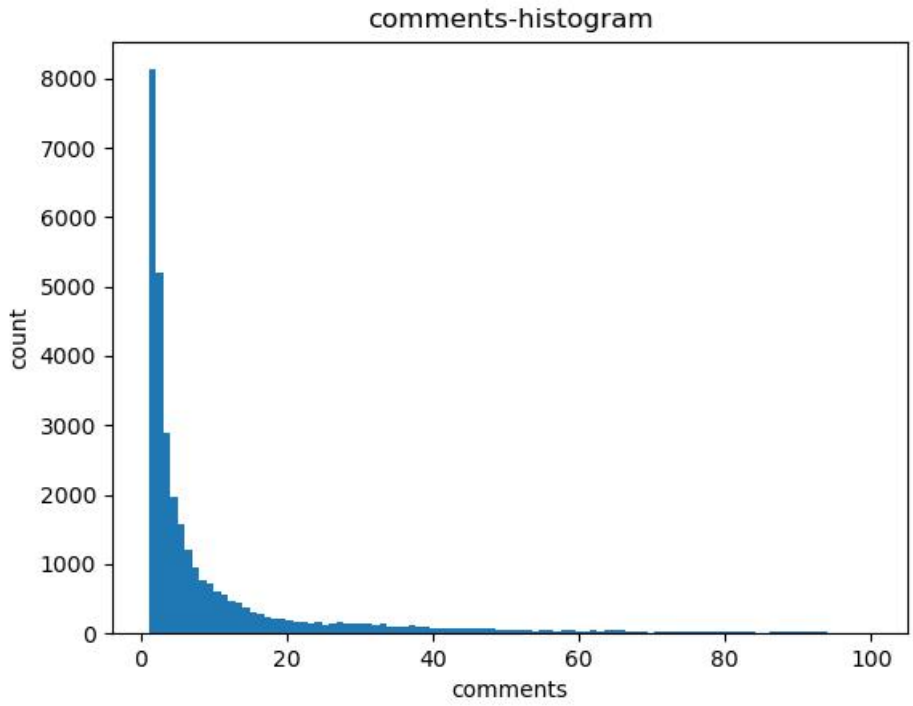
新闻长度的分布如下：



> 说明：上图像仅显示了长度为10000字以内的新闻。这涵盖了绝大多数新闻。在142485条新闻中，只有394条长度大于10000字。其中有3篇长度大于40000字。

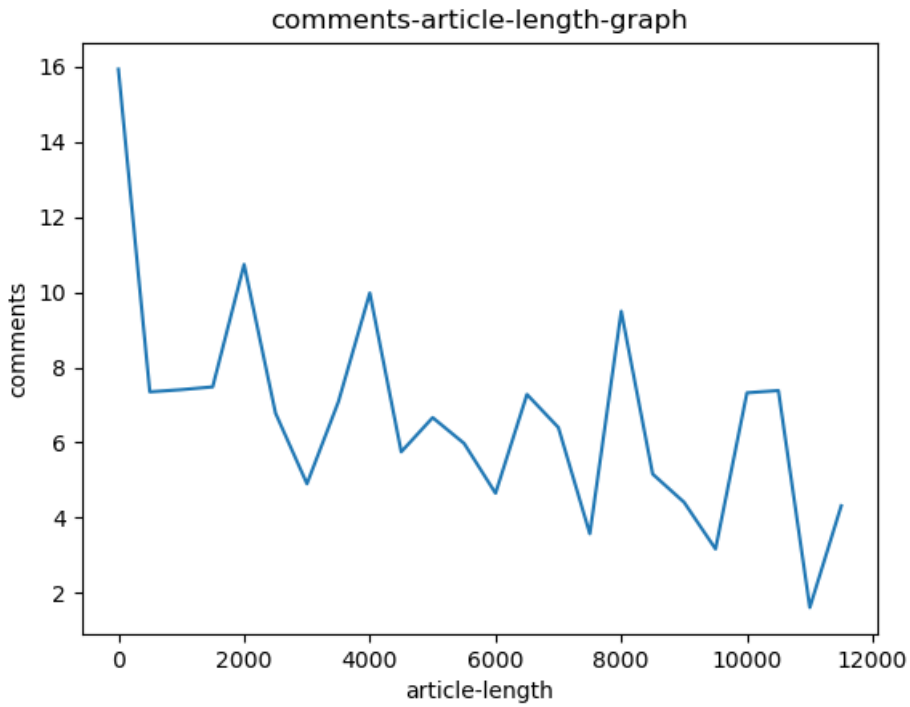
由图可见，大多数新闻篇幅在2000左右浮动，约700字长度的新闻数量最多。短新闻可以非常短，而网站上也有少量长文。

新浪新闻科技板块滚动页面网站上的大多数新闻都没有评论。在总数142485条中，109570条没有评论，占比76.9%。在有评论的32915中，31915条的评论数在100以内，占比96.7%。然而也有少数新闻拥有大量评论，最多的有五万余条。下面这张直方图只展示了评论数在1-100范围内的分布情况：



可见，在76.9%的零评论之后，新闻分布随着评论数的上升快速严格递减。

本文进而探讨了新闻长度和热度（评论数）的关系：



> 仅统计了文章长度在12000以内的新闻（占新闻总数99.7%以上）

图像表明，新闻热度和新闻长度总体上呈负相关的关系，这一点令笔者比较意外。可能的原因是：用户在互联网上没有足够的精力和耐心阅读过长的文字，太长的新闻通常没有被看到结尾，所以评论数较少。

```
# 绘制评论数和新闻长度的关系图
data1 = bigdic['len']
data2 = bigdic['cc']
bin_width = 500
```

```
min_value = 0
max_value = 12000
ranges = np.arange(min_value, max_value + bin_width, bin_width)

averages = []
for start in ranges[:-1]:
    end = start + bin_width
    values_in_range = [data2[i] for i in range(len(data1)) if start <= data1[i] <
end]
    if values_in_range:
        avg = sum(values_in_range) / len(values_in_range)
        averages.append(avg)
    else:
        averages.append(0)

plt.plot(ranges[:-1], averages)
plt.title('comments-article-length-graph')
plt.xlabel('article-length')
plt.ylabel('comments')
plt.savefig('comments-article-length')
```