课程报告

院 系

专 业

班 级

学 号

姓 名

任课教师

年 月 日

基于大数据爬虫技术对双减政策实施效果进行分析

关键词：爬虫；数据分析；双减政策

# 项目背景

近年来，义务教育阶段学生过重的作业负担和校外培训负担，阻碍了我国教育事业的发展，针对这一问题的教育改革亟需落实。2021年7月，中共中央办公厅、国务院办公厅印发《关于进一步减轻义务教育阶段学生作业负担和校外培训负担的意见》（简称“双减”政策）。“双减”政策内容主要分为两部分：在校内，使学校教育教学质量和服务水平进一步提升，作业布置更加科学合理，学校课后服务基本满足学生需要，学生学习更好回归校园；在校外，使校外培训机构培训行为全面规范，学科类校外培训泛滥，监督不严，资本运作等各种乱象基本消除，校外培训热度逐步降温，以使义务教育阶段学生达到多方面减负。

从既有研究来看，研究者们从“双减”政策的多方参与主体视角出发，对政策设计、执行和评价等问题进行了较为全面的解析，研究方式丰富，研究主题多样。现实已经证明，如果脱离了特定的社会环境与文化，双减政策极易发展为“加压式减负“的模式，所以本研究从社会公众的角度出发，研究公众在特定社会环境下对于双减政策的态度及关注点，进一步探索总结双减政策的改进路径，助力”双减“真正落实，实现教育减负。

## 研究对象

双减政策“相关的网络文本数据

## 研究方法

为获取公众对双减政策的态度与评价，本文借助python爬虫技术对微博平台的文本进行爬取与分析。具体研究过程分为两个阶段，第一阶段是利用python网络爬虫技术获取有关“双减政策“的大数据样本；第二阶段是利用python对获取的样本进行分析，提取并概括总结文本主题，探究公众关注的要点，分析公众的真实需求。

# 实验报告

实验分为数据获取，和数据分析两个大部分。

如下是整个项目结构：



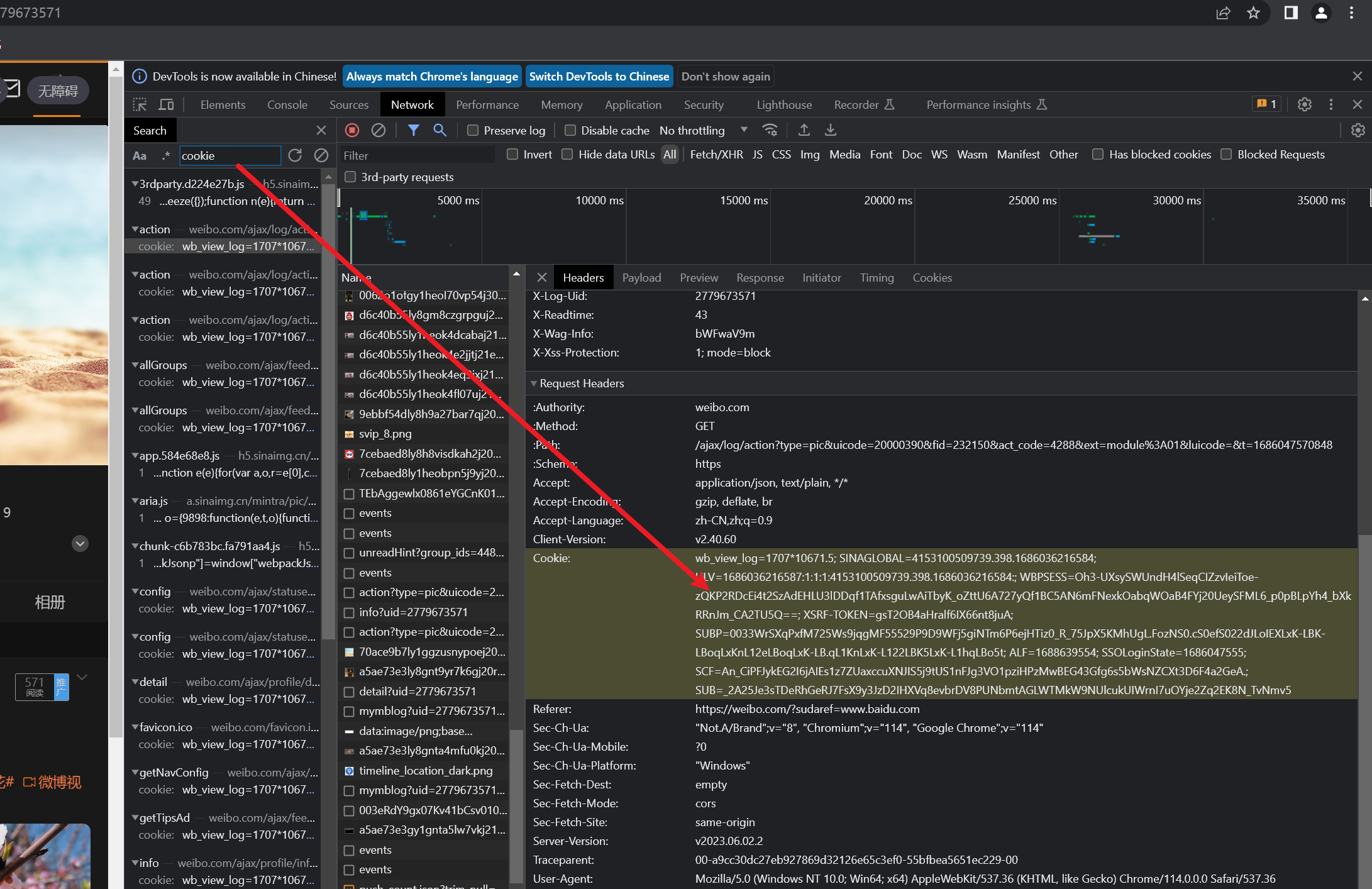
其中，analyse是我们用于后期分析的工具

Spider是我们用于数据获取的爬虫程序

## 2.1 信息获取

在进行数据获取之前，我们需要获得我们的cookie。

如下是获取cookie的过程：



将改cookie存于我们的文件中。

其次，要定义我们的请求头：



## 2.2数据获取

数据获取中，我们拆分为两个任务来完成，其一是获取微相关网页，其二是对相关网页的评论进行解析。网页保存为json格式化，相关数据保存为csv文件。

相关代码如下：

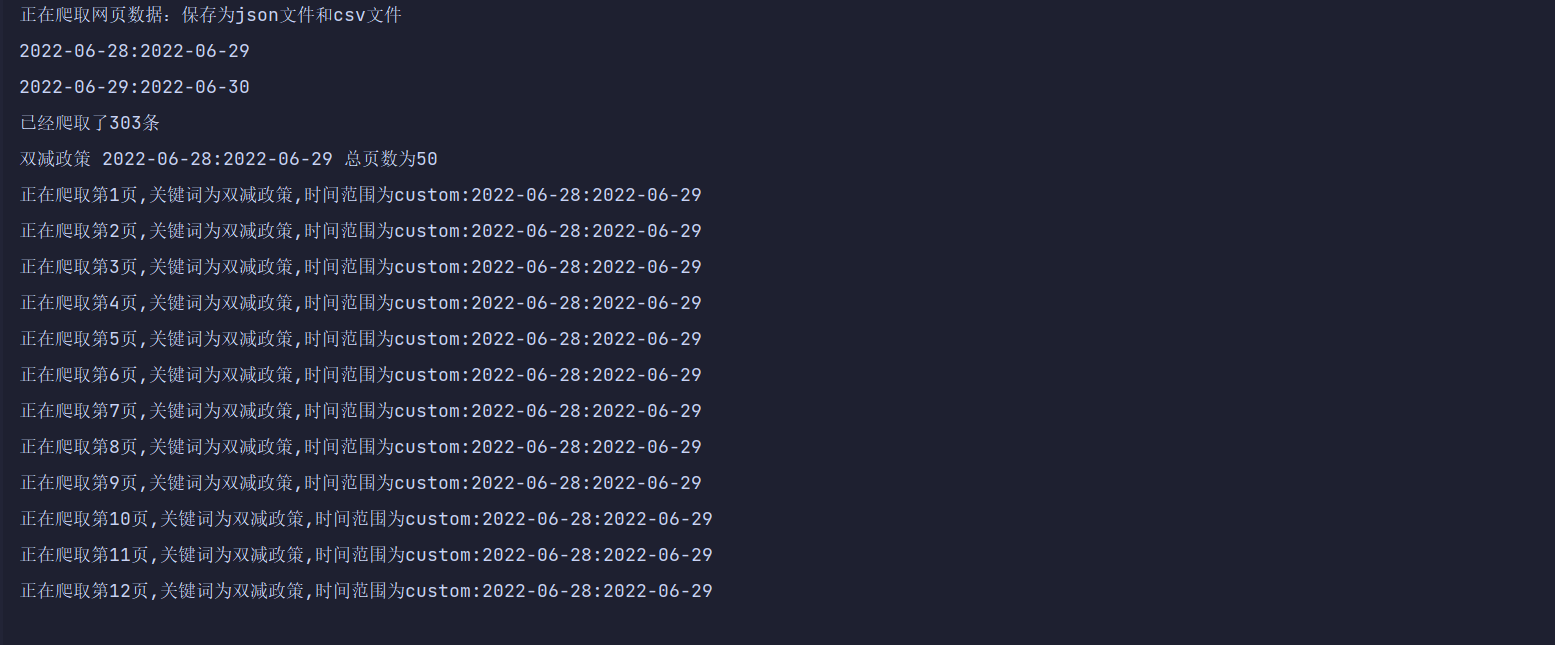
1. 导入所需的库和模块，包括os、get\_weibos、get\_comments、ThreadPoolExecutor、pandas等。
2. 定义了一个名为run\_web的函数，用于爬取指定关键字在特定日期范围内的微博数据，并将其保存到csv文件中。在这个函数中，首先定义了保存csv文件和json文件的路径、起始日期、时间间隔等参数，然后调用get\_date\_range\_list函数获取日期范围。接着调用get\_weibos函数获取微博数据，并将结果保存到csv文件中，同时记录已经爬取的微博数量，以便后续使用。最后将保存的csv文件转换为json格式，并将其读取为DataFrame格式。
3. 定义了一个名为web\_comment的函数，用于爬取指定关键字在csv文件中的微博id对应的评论数据，并将其保存到另一个csv文件中。在这个函数中，首先读取之前保存的csv文件，并获取其中所有微博的id。然后遍历所有微博id，调用get\_comments函数获取评论数据并将结果保存到csv文件中，同时记录已经爬取的评论数量，以便后续使用。当爬取的评论数量达到预设的最大值时，跳出循环。
4. 最后在main函数中调用run\_web和web\_comment函数，并传入指定的关键字"双减政策"作为参数，以开始整个程序的执行。

*import* os  
*from* get\_weibos *import* get\_weibos, get\_date\_range\_list  
*from* get\_comments *import* get\_comments  
*from* concurrent.futures *import* ThreadPoolExecutor  
*import* pandas *as* pd  
  
  
*def* run\_web(keyword="双减政策"):  
 *print*("正在爬取网页数据：保存为json文件和csv文件")  
 save\_csv\_path1 = keyword + ".csv" *# 保存的文件名* save\_json\_path = keyword + ".json" *# 保存json* start\_date = "2022-06-28" *# 开始日期* end\_date = "2022-06-29" *# 结束日期* days = 1 *# 时间间隔* date\_ranges = get\_date\_range\_list(start\_date, end\_date, days) *# 日期范围* count = 0  
 max\_count = 15000  
 *if* os.path.exists(save\_csv\_path1):  
 df = pd.read\_csv(save\_csv\_path1, encoding="utf\_8\_sig")  
 *# 计算有 多少条* count = df.shape[0]  
 *print*("已经爬取了{}条".format(count))  
  
 get\_weibos\_generator = get\_weibos(save\_csv\_path1, keyword, date\_ranges)  
  
 *for* weibos *in* get\_weibos\_generator:  
 df = pd.DataFrame(weibos)  
 df.to\_csv(save\_csv\_path1, mode='a', index=*False*, header=*not* os.path.exists(save\_csv\_path1),  
 encoding="utf\_8\_sig")  
 count += *len*(weibos)  
 *if* count % 10 == 0:  
 *print*("已经爬取了{}条".format(count))  
  
 *if* os.path.exists(save\_csv\_path1):  
 df = pd.read\_csv(save\_csv\_path1, encoding="utf\_8\_sig")  
 *# 计算有 多少条* count = df.shape[0]  
 *print*("已经爬取了{}条".format(count))  
  
 get\_weibos(save\_csv\_path1, keyword, date\_ranges)  
  
 *# 所有页数的数据都保存到同一个JSON文件中* df = pd.read\_csv(save\_csv\_path1, encoding="utf\_8\_sig")  
 df.to\_json(save\_json\_path, orient="records", force\_ascii=*False*)  
  
 *# 读取JSON文件为DataFrame* df = pd.read\_json(save\_json\_path)  
 *# 获取mid属性的值* mids = df["mid"].tolist()  
  
  
*def* web\_comment(keyword="双减政策"):  
 save\_csv\_path1 = keyword + ".csv" *# 保存的文件名* save\_csv\_path2 = keyword + "\_comment.csv" *# 微博评论保存的文件名* save\_json\_path = keyword + ".json" *# 保存json* count = 0  
 max\_count = 1500  
 *# 读取csv文件为DataFrame* df = pd.read\_csv(save\_csv\_path1)  
 *# 获取mid属性的值* mids = df["mid"].tolist()  
 *for* id *in* mids:  
 result = get\_comments(id)  
 get\_comments(*str*(id))  
 comments = []  
 comments.extend(result)  
 count += *len*(comments)  
 df\_comment = pd.DataFrame(comments)  
 df\_comment.to\_csv(save\_csv\_path2, mode='a', index=*False*, header=*not* os.path.exists(save\_csv\_path2),encoding="utf\_8\_sig")  
 *if* count >= max\_count:  
 *break  
  
  
if* \_\_name\_\_ == '\_\_main\_\_':  
 *#keyword = str(input("输入关键字"))* keyword = "双减政策"  
 run\_web(keyword)  
 web\_comment(keyword)

这是main函数的执行逻辑，我们调用了自己写的两个库，从而获得相关的文本数据。

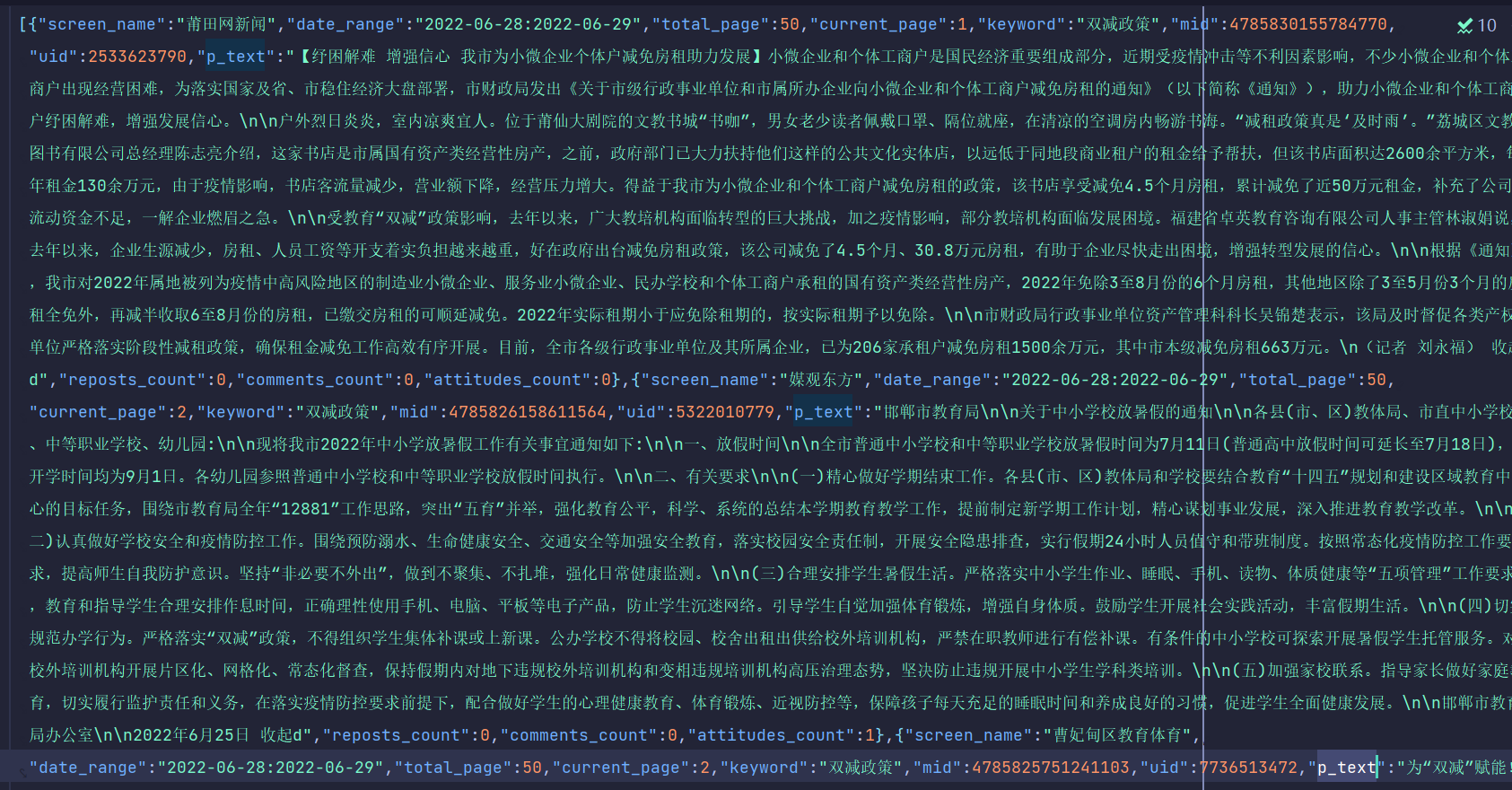
如下为代码的执行过程和结果

爬虫的过程图：

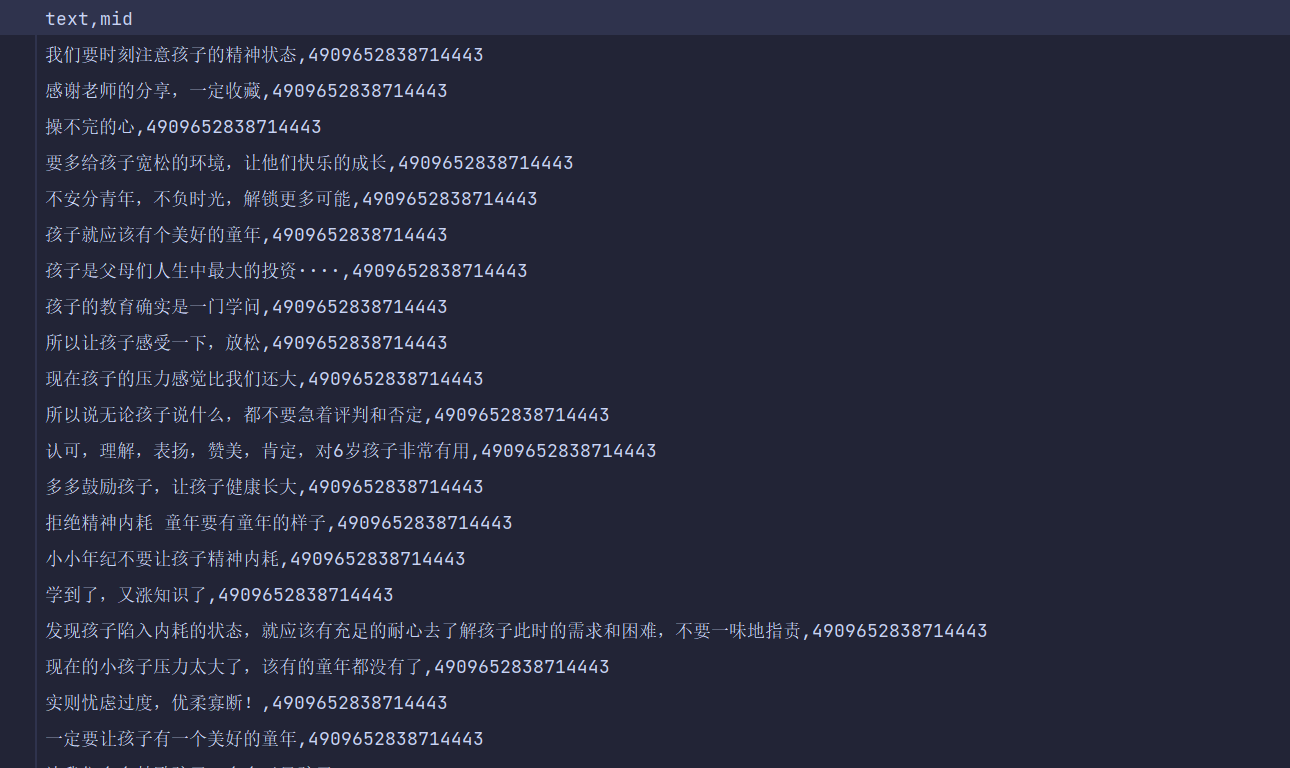


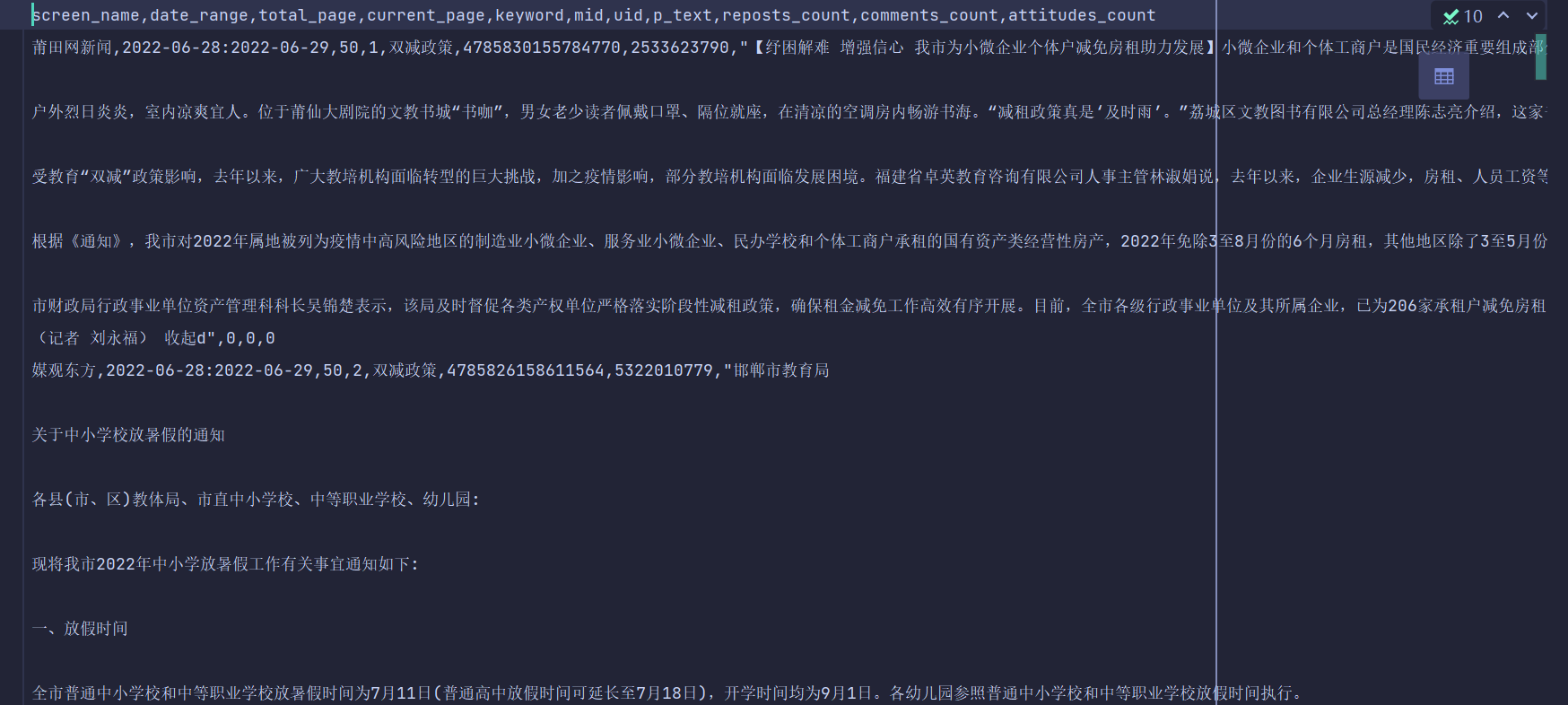
最终的结果，我们得到了一个JSON网页格式文件，微博相关文本数据和评论数据。

JSON文本：



微博文本和相关评论的数据：





## 2.3 数据分析

因为本项目是对微博文本和评论进行数据分析，统计出高频词和云图，所以需要对收集的数据进行合并处理。

如下是处理数据合并的代码：

首先，使用 Pandas 的 read\_csv() 函数分别读取了名为 '双减政策.csv' 和 '双减政策\_comment.csv' 两个文件中的数据。这两个文件中都包含有关微博的信息，其中 '双减政策.csv' 文件包含了微博的 ID 和原始文本内容，而 '双减政策\_comment.csv' 文件包含了微博的 ID 和评论的文本内容。

我们将这两个数据的内容，进行一一对应的合并。

*import* pandas *as* pd  
  
df1 = pd.read\_csv('D:\spider\双减政策.csv', encoding='utf-8')  
df1 = df1[['mid', 'p\_text']]  
*# 重命名*df1.rename(columns={'p\_text': 'text'}, inplace=*True*)  
df2 = pd.read\_csv('D:\spider\双减政策\_comment.csv', encoding='utf-8')  
df2 = df2[['mid', 'text']]  
  
df = pd.concat([df1, df2], axis=0)  
df.to\_csv('双减政策\_合并.csv', index=*False*, encoding='utf-8-sig')

合并完数据，我们需要对数据进行分词处理（对语言进行分词处理的作用是将一段长文本切分成一个个独立的词语，使得计算机能够更好地理解和处理文本。分词可以帮助计算机准确地识别出文本中每个单词的含义，进而进行文本分类、信息检索、情感分析等自然语言处理任务。此外，分词也有助于提高机器翻译的准确性和流畅度，因为它可以帮助计算机更好地理解源语言和目标语言的语法结构）

使用 Pandas 的 read\_csv() 函数读取名为 '双减政策\_合并.csv' 的 CSV 文件，并将其转换为 DataFrame 对象 df。

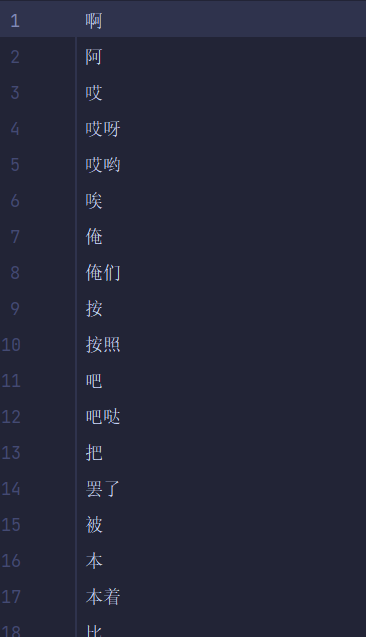
导入了 jieba 库用于对文本进行分词，并打开名为 '中文停用词表1.txt' 的文件，读取其中的停用词列表。

然后对数据进行处理，使用列表推导式对文本进行了分词，并去掉了停用词和单个汉字，将每条微博和评论分好词后的结果保存到一个新的列 'text\_cut' 中，并转换为字符串格式。

最后，使用 Pandas 的 to\_csv() 函数将处理后的结果保存到原始的 CSV 文件 '双减政策\_合并.csv' 中，以便后续使用。

*import* re  
  
*import* pandas *as* pd  
  
*# 读取数据*df = pd.read\_csv('双减政策\_合并.csv', encoding='utf-8-sig')  
  
*# 分词  
import* jieba  
  
*# 停用词  
with open*('中文停用词表1.txt', encoding='utf-8') *as* f:  
 stopwords = f.read().splitlines()  
*# 分词  
# 去掉非中文字符 去掉@网名*df['text\_cut'] = df['text'].apply(*lambda* x: ''.join(re.findall('[\u4e00*-*\u9fa5]', x)))  
*# 去掉收起 这两个字*df['text\_cut'] = df['text\_cut'].apply(*lambda* x: x.replace('收起', ''))  
*# 去掉停用词*df['text\_cut'] = df['text\_cut'].apply(*lambda* x: [i *for* i *in* jieba.cut(x) *if* i *not in* stopwords])  
*# 去掉单字*df['text\_cut'] = df['text\_cut'].apply(*lambda* x: [i *for* i *in* x *if len*(i) > 1])  
df['text\_cut'] = df['text\_cut'].apply(*lambda* x: ' '.join(x))  
  
*# 保存*df.to\_csv('双减政策\_合并.csv', index=*False*, encoding='utf-8-sig')

这里，我们使用了网上收集的中文停用表。



最后我们将得到数据，进行统计分析，生成云图和词频图。

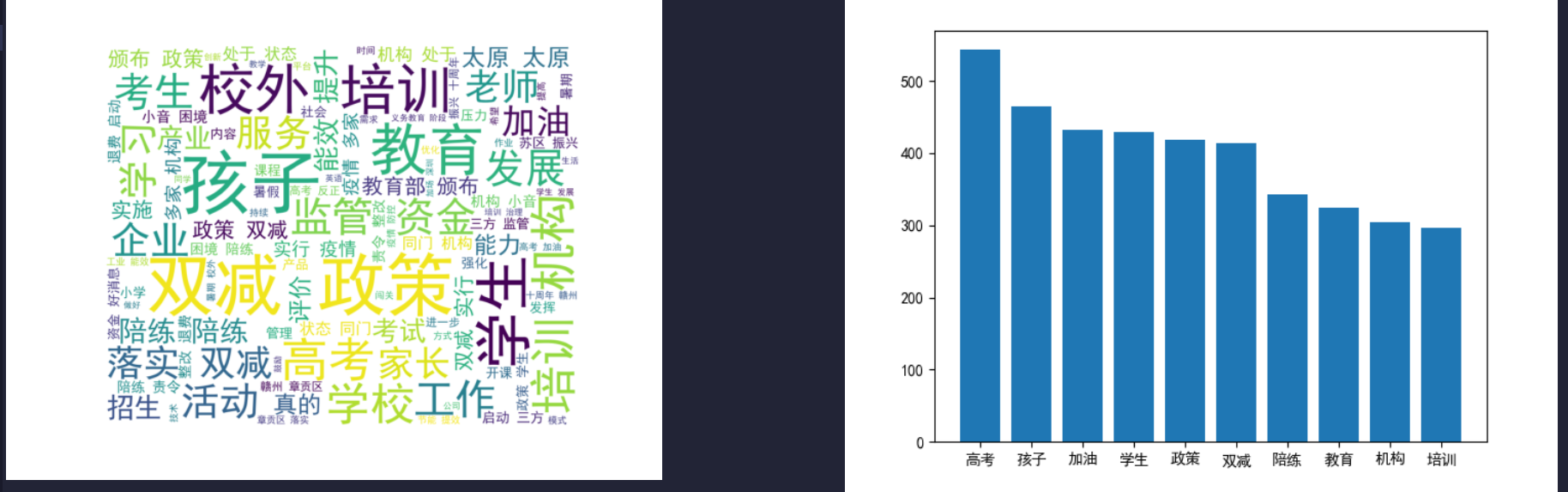
如下是代码是相关代码：

首先，使用 Pandas 的 read\_csv() 函数读取名为 '双减政策\_合并.csv' 的 CSV 文件，并将其转换为 DataFrame 对象 df。并导入相关的模块。

然后就进行两种可视化操作：词云图和词频图

*import* pandas *as* pd  
*from* wordcloud *import* WordCloud  
*import* matplotlib.pyplot *as* plt  
*from* collections *import* Counter  
  
*# 中文乱码的处理*plt.rcParams['font.sans-serif'] = ['SimHei']  
plt.rcParams['axes.unicode\_minus'] = *False*df = pd.read\_csv('双减政策\_合并.csv', encoding='utf-8-sig')  
  
*# 1.词云图  
# 统计词频  
# 1.1 df 将所有的text\_cut拼接成一个字符串*text = ''  
*for* i *in* df['text\_cut']:  
 *if type*(i) == *str*:  
 text += i  
  
*# 1.2 词云图*wc = WordCloud(  
 background\_color='white',  
 width=1000,  
 height=800,  
 *# 黑体* font\_path='simhei.ttf',  
 *# 设置最大词数* max\_words=100,  
 *# 设置字体最大值* max\_font\_size=150,  
 *# 设置有多少种随机生成状态，即有多少种配色方案* random\_state=30  
)  
wc.generate\_from\_text(text)  
plt.imshow(wc)  
plt.axis('off')  
plt.show()  
  
*# 2.词频图  
# 2.1 统计词频  
# 2.1.1 将所有的text\_cut拼接成一个字符串*words = []  
*for* i *in* df['text\_cut']:  
 *if type*(i) == *str*:  
 words += i.split(' ')  
*# 2.1.2 统计词频*word\_count = Counter(words)  
*# 2.1.3 排序*word\_count = *sorted*(word\_count.items(), key=*lambda* x: x[1], reverse=*True*)  
*# 2.1.4 取前10个*word\_count = word\_count[:10]  
*# 2.1.5 画图*plt.bar(*range*(*len*(word\_count)), [i[1] *for* i *in* word\_count], tick\_label=[i[0] *for* i *in* word\_count])  
plt.show()

最终的运行结果如下：



# 调研结果与展望

## 3.1 调研结果分析

根据对双减政策相关信息的调研，我们发现高考、孩子和加油是人们关注的高频词汇。这说明在社会各界，特别是家长和学生中，高考仍然是一个备受关注的话题，孩子的教育问题也备受关注，并且大家都希望孩子能够顺利升学上进，因此也表达出了“加油”的祝福。

另外，我们注意到政策、双减等词汇也受到了广泛关注。在最近的一些政策中，政府开始着手推行双减政策，旨在从源头上控制培训机构的超前教育，遏制面向小学生开展的竞争性课程和学科类培训等不良趋势。这使得培训机构成为了公众讨论的重要话题之一。双减政策的实施影响着多个领域，尤其是涉及到孩子教育问题的家庭和教育机构。

同时，陪练、教育、机构、培训等词汇也与双减政策密切相关。基于双减政策的背景，家长和学生有意识地开始关注正规的培训机构，并注重培训的质量。更多的家长选择通过陪练的方式对孩子进行教育，而不是仅仅依赖于培训机构的辅导。教育机构也因此受到了影响，需要调整自身的业务模式，适应政策变化。

总之，双减政策的出台引起了社会各界广泛的关注。在教育问题上，越来越多的人开始意识到合理平衡的重要性，并且注重培养孩子的自主学习能力，避免过度压力和学习焦虑。同时，在政策层面上，政府也在推进相关措施，切实保障学生的权益，为孩子们的健康成长和未来发展提供更好的环境和条件。

## 3.2 项目改进计划

本项目未来我们计划使用LDA主题模型对“双减”政策相关的网络评论文本数据进行分析与研究，以深入探讨该政策对社会各界的影响。通过运用LDA主题模型的方法，我们可以将文本数据所涉及的主题按概率分布进行提取，并生成文档主题模型进行主题聚类或文本分类。

我们将利用Python的自然语言处理Gensim训练库中的LDAModel函数，构建LDA主题模型，并对文本数据进行训练，以困惑度(perplexity)来评估模型的拟合程度。在训练模型之后，我们将使用主题可视化工具对生成的主题进行可视化，以更直观地展现主题之间的关系和分布情况。

这样的分析方法可以帮助我们更全面地了解人们对于双减政策的态度、意见和看法，以及受政策影响的各个方面，例如培训机构、教育机构、学生和家长等。同时，还可以从多个角度出发，挖掘出潜在的问题和需求，为政府部门和相关机构提供更有针对性的政策建议和服务。

**参考文献**

[1] [Python爬虫实战，完整的思路和步骤（附源码） - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/149527289)

[2] [读懂“双减”背后的教育逻辑 - 中华人民共和国教育部政府门户网站 (moe.gov.cn)](http://www.moe.gov.cn/jyb_xwfb/moe_2082/2021/2021_zl53/mtgc/202112/t20211222_589201.html)

[3] [Python 标准库 — Python 3.11.3 文档](https://docs.python.org/zh-cn/3/library/index.html)

附录1.

课程报告评分表

|  |  |  |  |
| --- | --- | --- | --- |
|  | 各项满分 | 评分标准 | 得分 |
| 评分标准一 | 20 | 第一部分源代码：项目源代码材料提交完整且符合要求，程序可正确运行，能够连接数据库 |  |
| 评分标准二 | 20 | 第一部分实验报告，能清晰阐述爬虫程序的功能逻辑，有实验过程、实验结果 |  |
| 评分标准三 | 40 | 第二部分调研报告，选题符合课程范围，内容结构完整，层次清晰，能够对一项Python爬虫技术进行完整介绍 |  |
| 评分标准三 | 10 | 语言通顺、无病句、错别字等 |  |
| 评分标准四 | 10 | 论文格式符合模板要求 |  |
| 总分 | 100 | |  |
| 评阅人 |  | | |