爬虫任务报告

爬虫任务

十堰问政 http://m.shiyan.gov.cn/zwhd/web/webindex.action

任务团队

粉红橘子: 范思棋、冯淑杰、高乐天

数据概况

从2007.10.22--2021.11.26 共**十二万**条

任务完成情况

- 成功爬取十二万条问政记录, 存入result.xlsx中
- 更新函数每次更新可以做到将新产生的数据存入原result.xlsx中,并产生一个带有时间的副本文件 , 类似于, newdataNov 17 17-25-35 2021.xlsx

数据说明

• 关于问政信息分别爬取了:编号、提问人、类型、浏览次数、受理单位、标题、提问内容、办结回 复、回复时间、办理部门



早期数据中受理单位办理部门数据不全,并不是爬取问题,是网站本身数据不全。



针对提速做的尝试

由于十堰问政网站的数据比较多,约为十二万条,本小组针对提速做了不同尝试。

- 打开8个jupyter notebook页面同时爬取
 - o 我们首先去获取url,将八千条左右的url分成了8份,然后同时进行爬取。结束后,再将8份内容用concat合并起来。
 - 。 总耗时在3小时以上。
- 使用多进程
 - 使用进程池默认进程数量,即cpu核心数量
 - 。 总耗时在9993.40s, 即166.5分钟
- 寻找合适进程池数量
 - 。 由于进程池默认进程数量比较小, 我们决定增加进程数量。

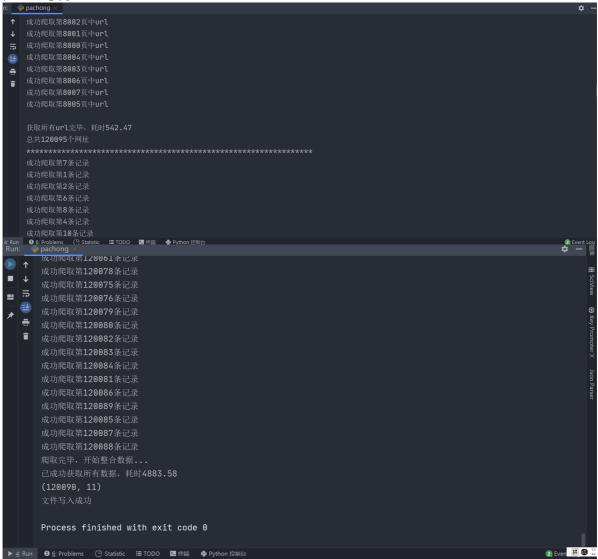
进程数量 \ 网页数量	爬取500个网页	爬取1000个网页	爬取2000个网页
默认	65.04s		
开8个	67.05s		
开10个	55.52s	105.42s	
开12个	47.51s	90.03s	
开14个	44.24s	80.47s	
开16个	39.77s	76.32s	
开18个	39.64s	73.95s	
开20个	40.61s	73.62s	138.89s
开22个	41.93s	72.81s	137.03s
开24个			138.17s

- 。 通过简单的测试, 非常粗略的估计在22个进程数量时, 时间比较短, 可以又可以提速。
- 22进程总耗时: 524.47+4883.58=5408.05s, 即90分钟。
- 相比默认cpu核心数量提速将近一倍

针对异常处理

- 由于爬取网页耗时很长, 所以异常处理和过程输出显得尤为重要。
- 我们使用traceback库,可以即时获取各种异常信息,发现问题。
- 同时如果个别网页出现爬取异常,整个进程并不会被中断。事后回溯还可以找到问题网页的具体报错。

pachong.py文件爬取全部数据时控制台的部分输出展示:



更新函数的控制台输出:

```
成功爬取第328条记录
爬取完毕,开始整合数据...
已成功获取所有数据,耗时12.31
增加了329条数据到newdataNov 28 16-34-35 2021.xlsx中
新旧数据合并完毕
新数据追加到result中完毕
Process finished with exit code 0
```

复杂性比较高的bug

- 1. · 12万条记录爬取下来后,保存为dataframe类型写入文件时报错,报错信息显示数据长短参差不齐。
 - 。 回溯控制台输出的内容,发现极个别网页爬取时报错,显示失败
 - 分析原因: beautifulsoup解析网页时,极个别网页和其他网页源代码有差异,相同的模式爬取不到信息,导致数据长短参差不齐,无法以dataframe类型写入文件之中。
 - 。 最后我们调整beautifulsoup抓取的内容, 白框处改为-1得以解决

```
# 受理单位
unit = data.find_all('span', 'mtp3')
unit = [x.text for x in unit [-1]
unit_.append(unit)
```

- 2. o 12万条记录爬取下来,保存为dataframe后,写入excel时又失败。
 - 。 多方查找发现,原来pandas导出到Excel会报URLS数量超出65530警告。
 - 。 我们尝试过这种方法:

```
writer = pd.ExcelWriter('xxxx.xlsx',engine='xlsxwriter',options=**
{'strings_to_urls': False})
```

将url变成字符串保存进去。但是失败了。

。 最后一致认为,excel中没必要保存每条记录的原网址,可以直接将url删除