

Concept Paper

Big Data Analysis of College Entrance Examination Candidates' Volunteer Selection Based on Python

Lu Bei

School of Artificial Intelligence, Jiaozuo University, China
lubeiaa@163.com

Date received: December 13, 2023

Date received in revised form: December 18, 2023

Date accepted: January 28, 2024

Recommended citation:

Bei, L. (2024). Big data analysis of college entrance examination candidates' volunteer selected based on Python. *International Journal of Computing Sciences Research*, 8, 2564-2579. <https://doi.org/10.25147/ijcsr.2017.001.1.175>

Abstract

Purpose – This article adopts a Python crawler technology approach to write crawler program code, extract publicly available college entrance examination volunteer filling data on the internet, and clean and store the data. Then, through analysis of the data, we mainly conducted big data analysis on the indicators used by college entrance examination candidates for voluntary reporting, identify popular majors for college entrance examination candidates to fill in, and providing references for students who are about to participate in the college entrance examination.

Proposed Method – The study will use crawler technology to collect, save, clean, and analyze the data, and finally use data analysis technology to complete data analysis and visual chart display.

Conclusion – The collected data was analyzed and visualized based on the top 10 majors selected for the first batch of undergraduate college entrance examination in Henan Province in 2019, the top 10 majors selected for the second batch of undergraduate college entrance examination, the top 10 majors selected for vocational colleges, the top 10 majors selected for liberal arts, and the top 10 majors selected for science. The analyzed results provide the necessary support for universities to carry out corresponding majors and can also provide a reference for students who are about to participate in the college entrance examination, which is of great significance.

Recommendations – This paper suggests using convenient and efficient Python crawler technology to extract data, store and analyze it, and ultimately display it.



Practical Implication – The analyzed results provide the necessary support for universities to carry out corresponding majors and can also provide a reference for students who are about to participate in the college entrance examination, which is of great significance.

Keywords – Python, crawler, data analysis, college entrance examination volunteer selection

INTRODUCTION

The college entrance examination is a turning point in our lives. The choice of filling out a voluntary application will greatly affect our life trajectory. In terms of the difficulty of the 2021 college entrance examination, our province of Henan still ranks among the top in the country, with a total of 1.2 million candidates. Henan candidates face enormous pressure in the college entrance examination, usually requiring more effort than students from other provinces to attend their ideal schools. The choice of college entrance examination preferences will affect our path to further education. This article takes the voluntary choice of Henan candidates as the starting point, extracts, stores, cleans, and analyzes the big data of candidates' voluntary choice filling, and analyzes the popular majors and the number of choices from different perspectives. This paper intends to provide a reference for many candidates to fill out their applications.

A system is proposed that will be implemented using the Python programming language. Python is a very simple and convenient programming language, with numerous popular third-party packages in various directions, which is also one of the reasons why Python is very popular. This study will discuss the implementation of Python to analyze enrollment data (i.e., degree program selection) to inform universities about which degree programs are popular.

LITERATURE REVIEW

Using Python for data analysis is also very convenient and efficient. We can easily write crawler code to extract useful data from the network, and combine Python's data analysis technology to process and analyze the data, achieving visual display of the data. Pei (2019) researched Python-based web crawling and anti-crawling technologies. Lirong (2017) designed and implemented a Python-based web crawler program. Dongmei and others (2018) explained to us how to use Python programs to obtain web information. Zhonghua and others (2014) used Python's crawler technology to crawl Sina Weibo data, implemented a serial crawler program and a parallel crawler program, and conducted a comparative analysis of execution efficiency, achieving the analysis of users' social graphs. Bian et al. (2020) crawled and analyzed the recruitment position data of Python.

Ma and Zhang's (2021) paper gave some shallow opinions on the Web Crawler, introduces the importance of Web Crawler, and the system crawls medical data through Scrapy frame. The system mainly realizes the crawling and data storage of the doctor's information in the specific area and the designated department. Li's (2021) article takes Lagou.com as an example, uses crawler technology to collect data based on Python and

MySQL, analyzes the collected employment data in various aspects, and uses these data analysis results to help college students in their employment and career planning. Provide a reference basis, provide an objective reference. Zhang's (2021) paper uses Python's requests, BeautifulSoup, Jieba, and Fontools to obtain the relevant data of all public films in 2015-2020 and analyze them. Finally, the data results are displayed graphically through Ajax technology, layui framework, and eckards.

Fu and Li (2021) briefly introduced Python and the hacker attack technology and crawler program based on Python. Wang et al. (2021) introduces the workflow, design, and implementation of the crawler in detail. The experiment proves that the designed crawler can get news quickly and can provide people with the information they need. Wu and Cui (2021) showed a system that is committed to solving the current people to warrant a request for the search of detailed keywords needed. To help make it easier for everyone who needs to rent. The system is mainly composed of data cleaning, data access, algorithm design and implementation, Python implementation of the front and back end of the system, data formatting, and auxiliary decisions.

Hejing et al. (2020) explored how to develop a crawler method based on the specific framework for the complete interface of steam manufacturers and stores, which should be able to automatically and efficiently crawl the data of specific targets, analyze the dynamic pages, and complete the data cleaning, downloading, saving and other operations, explore the methods of general data analysis, and analyze the downloaded data, extract useful information from it, analyze and summarize the specific crawler method and data analysis method through practical application. Wang's (2019) paper aimed at the problem of low crawl speed of traditional web crawlers, a research and design of distributed multi-topic web crawlers based on Python is proposed.

You et al.'s (2018) article was mainly aimed at the WeChat of the public platform Crawler search technology. Luo (2023) combined Python web crawling technology and precision procurement models, a book recommendation mechanism was constructed to transform book recommendations from traditional manual collection to automated assistance and prediction through the precise procurement model based on the recommendation list, thereby achieving the automation of book procurement. This mechanism improves the efficiency and accuracy of book procurement, provides a solution for the limited resources and funding of libraries, and also provides better services for readers.

Lastly, in Ma and Yan (2021), they take the crawling of second-hand housing information of Anjuke Xi'an as an example. According to the crawler principle and process, the structure of Anjuke's page is first analyzed, using requests to obtain web pages, XML to analyze web pages, and SQL Server 2017 to store data to design and implement a network. The crawler program collects and stores housing information in some cities in East China through this program, and finally analyzes the housing price trend through the collected data through Excel. The results show that this program can automatically obtain housing information from the Internet, which provides a data source for later data analysis.

PROPOSED METHODOLOGY

The so-called web crawler refers to writing code that allows programs to automatically extract useful information from the network according to certain rules. To use a very vivid analogy, the internet is like a huge spider web, and the crawler program we write is a small spider on top of it, which can constantly crawl through the spider web to retrieve the information we need.

Python crawler is a very mature, powerful, and easy-to-use programming technology. There are two main modes of development based on Python crawlers: one is to import the third-party libraries required for crawler development, commonly used are Python+Requests+Beautiful, and the other is to use a more versatile and powerful scraper framework.

The process of Python data analysis will use NumPy, Pandas, and pyECharts libraries. NumPy can facilitate data processing, and the Pandas library is mainly used for data analysis, which can quickly provide structured data results to Python. The PyECharts library is mainly used for visualizing data display.

System Development

1. Data Collection

The data collected in this article is obtained from the admission statistics publicly available by the Henan Provincial Admissions Office, mainly based on the data of candidates' application preferences in Henan Province, to analyze the selection of college entrance examination preferences. The website of the admission statistics publicly available by the Henan Provincial Admissions Office is: <http://www.heao.gov.cn/adc/pzljqtj.shtml> On this webpage, it is convenient to screen for the college entrance examination year and arts and science batches. The publicly available data on this webpage includes the name of the major, the number of recruitment plans, and the actual number of applicants, which is in line with the statistical analysis of our team's voluntary application situation. Therefore, the main goal of data collection is the data on this webpage, as shown in Figure 1.

2. Breakthroughs in anti-crawler technology

In the process of analyzing the content of the webpage, it was found that an anti-crawling technique was used, that is, the application data inside the webpage was loaded by embedding an iframe, as shown in Figure 2. In this way, we cannot obtain the corresponding data using the usual crawling methods.



Figure 1. The webpage

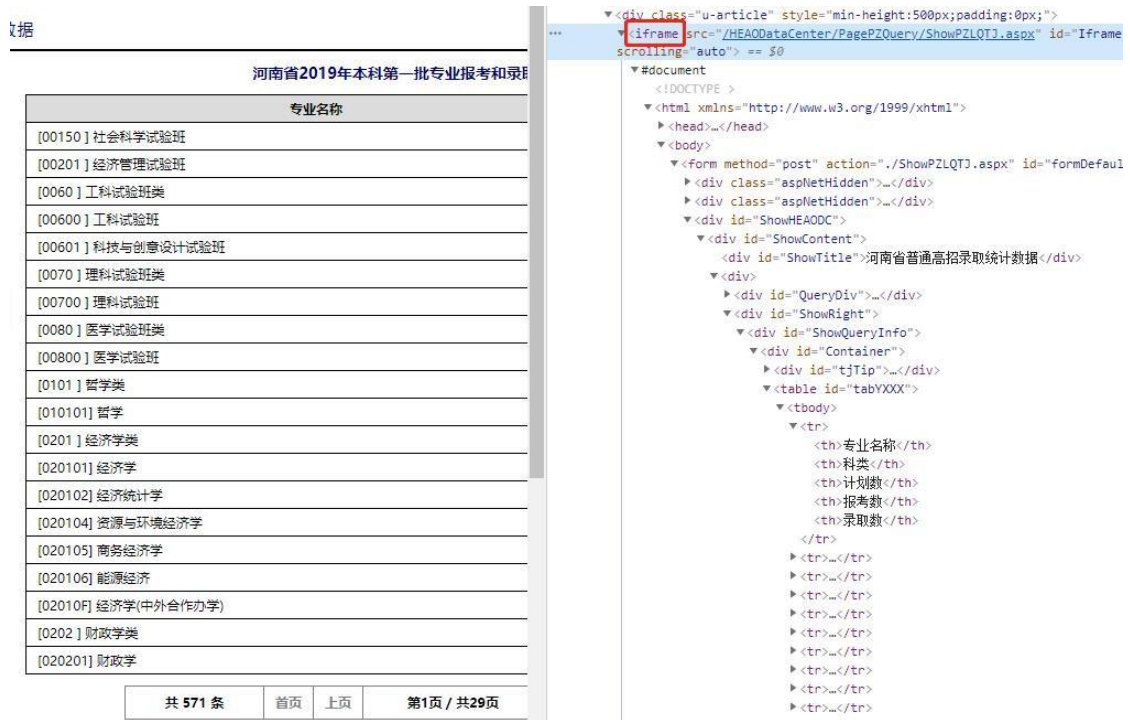


Figure 2. The iframe tab

There are several common forms of anti-crawler and their response methods as follows:

- The website program will detect User Agents, referers, and cookies. We have the program carry the corresponding information when sending the request.
- Anti-crawler based on user behavior. Some websites will detect user behavior, such as multiple visits to the same webpage by the same IP user in

a short period, such as 10000 visits within one second. Or perform the same operation multiple times in a short period. Will be considered as a crawler program. We can reduce the frequency of sending requests. A better solution is to use an IP proxy.

- Use custom fonts to crawl backward. Some websites may use CSS custom fonts to set sensitive data that they do not want others to directly take away and use. Although all the characters we see on the webpage are normal, when the program retrieves the webpage source code, it is indeed an unrecognized encoding. By analyzing the referenced fonts and using font design tools, find the corresponding font encoding, and then compare and re-parse it with the current font encoding on this website.
- Dynamic web page anti-crawling. Nowadays, many websites use Ajax asynchronous transmission technology, and the content inside the pages we see may be dynamically rendered through asynchronous calls. At this point, the source code obtained by the program cannot obtain the dynamically processed content. We can use automated testing tools and the PhantomJS framework to solve the problem. These two frameworks can simulate the process of users operating the browser with programs, allowing programs to operate dynamic websites, load dynamic content, and obtain the desired basic content through corresponding simulated user operations.
- Some website settings directly embed an iframe in the webpage, which can also achieve the goal of making the program unable to obtain the corresponding content source code.

The anti-crawling technique used in the webpage we are extracting is the fifth method. The main breakthrough method used is the Python+selenium framework, which implements code-switching iframe methods. This allows the program to obtain the source code of the corresponding data for data analysis and extraction.

```
# 获取内嵌的iframe
iframe = self.driver.find_element_by_id("Iframe")
# 切换至内嵌的iframe
self.driver.switch_to.frame(iframe)
# 拿到对应的iframe 的源码
frame_source = self.driver.page_source
```

We automatically set the statistical type, statistical year, statistical category, and statistical batch to be extracted, and automatically click the query button to query the corresponding data. Automatically use code to switch between each page, making it easy for us to extract all data.

```

# 统类型
sel_type = self.driver.find_element_by_id('pztjlx')
Select(sel_type).select_by_visible_text(selects['type']) time.sleep(0.5)
# 年份
sel_year = self.driver.find_element_by_id('pztjnf')
Select(sel_year).select_by_visible_text(selects['year']) time.sleep(0.5)
# 统种类
sel_wen_li = self.driver.find_element_by_id('pztjkl')
Select(sel_wen_li).select_by_visible_text(selects['wen_li']) time.sleep(0.5)
# 统批次
sel_batch = self.driver.find_element_by_id('pztjpc')
Select(sel_batch).select_by_visible_text(selects['batch']) time.sleep(0.5)
# 查询结果按钮 点击
btn_query = self.driver.find_element_by_id('QueryBtn')
btn_query.click()
# 解析数据
time.sleep(1)
while True:
    # 分页
    pager = self.driver.find_element_by_id('NewsHeader')
    # 当前页
    cur_page = self.driver.find_element_by_id('UIPageCur')
    print("===== %s =====" % cur_page.text) #
    找到下一页数据
    btn_next_page = pager.find_element_by_id('HEAOPage_N')
    # 是否为最后一页
    last = pager.find_element_by_id('HEAOPage_L') last_color =
    last.value_of_css_property("color")
    # 判断是否包含颜色值为red 的色值 是 说明还不是末页
    if last_color == "rgba(0, 0, 128, 1)":
        # 表格
        table = self.driver.find_element_by_id('tabYXXX')
        self.get_data(table, selects)
        btn_next_page.click()
        time.sleep(randint(10, 20))
    else:
        # 表格
        table = self.driver.find_element_by_id('tabYXXX') self.get_data(table,
        selects) print("=====Work
        Done=====")
        break

```

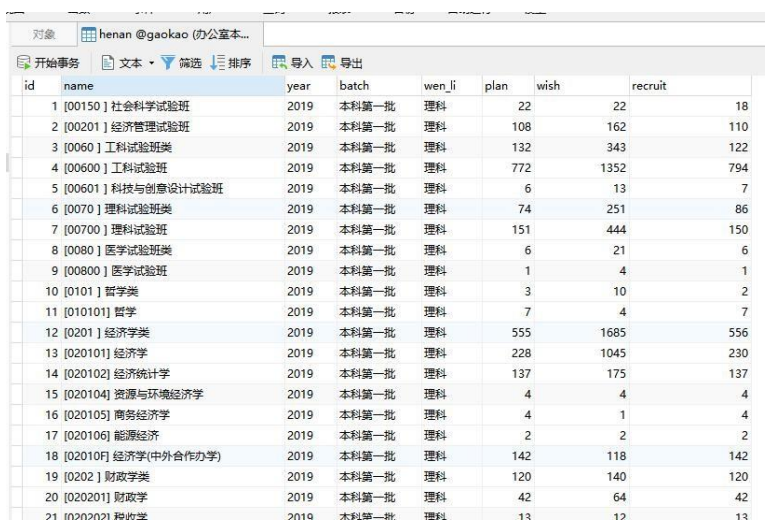

3. Cleaning and storage of data

We analyzed the data in the table, set the extraction fields, clean useless fields, extract useful data, and finally store all useful data in the MySQL database.

```
def get_data(self, table, selects):
    trs = table.find_elements_by_css_selector('tr')
    for i in range(1, len(trs)):
        tds = trs[i].find_elements_by_css_selector('td')
        values = ""
        values += "" + tds[0].text + ","
        values += "" + tds[1].text + ","
        values += tds[2].text + ","
        values += tds[3].text + ","
        values += tds[4].text + ","
        # 组合成sql 语句, 准备入库
        sql = 'insert into henan (name,wen_li,plan,wish, recruit,year,batch)
values (' + values[0:-1] + ',' + \
        selects['year'] + ',' + selects['batch'] + ')
        # print(sql)
        self.to_mysql(sql)

def to_mysql(self, sql):
    self.cursor.execute(sql)
    self.db.commit()
    print("入库成功:" + sql)
```

After careful debugging and trial operation. Finally, we collected and stored data on the humanities and sciences in Henan in 2019, as well as all batches of volunteer application data, totaling 2754 pieces of data. Figure 3 shows some of the data.



id	name	year	batch	wen_li	plan	wish	recruit
1	[00150] 社会科学试验班	2019	本科第一批	理科	22	22	18
2	[00201] 经济管理试验班	2019	本科第一批	理科	108	162	110
3	[0060] 工科试验班类	2019	本科第一批	理科	132	343	122
4	[00600] 工科试验班	2019	本科第一批	理科	772	1352	794
5	[00601] 科技与创意设计试验班	2019	本科第一批	理科	6	13	7
6	[0070] 理科试验班类	2019	本科第一批	理科	74	251	86
7	[00700] 理科试验班	2019	本科第一批	理科	151	444	150
8	[0080] 医学试验班类	2019	本科第一批	理科	6	21	6
9	[00800] 医学试验班	2019	本科第一批	理科	1	4	1
10	[0101] 哲学类	2019	本科第一批	理科	3	10	2
11	[010101] 哲学	2019	本科第一批	理科	7	4	7
12	[0201] 经济学类	2019	本科第一批	理科	555	1685	556
13	[020101] 经济学	2019	本科第一批	理科	228	1045	230
14	[020102] 经济统计学	2019	本科第一批	理科	137	175	137
15	[020104] 资源与环境经济学	2019	本科第一批	理科	4	4	4
16	[020105] 商务经济学	2019	本科第一批	理科	4	1	4
17	[020106] 能源经济	2019	本科第一批	理科	2	2	2
18	[02010F] 经济学(中外合作办学)	2019	本科第一批	理科	142	118	142
19	[0202] 财政学类	2019	本科第一批	理科	120	140	120
20	[020201] 财政学	2019	本科第一批	理科	42	64	42
21	[020202] 税收学	2019	本科第一批	理科	12	12	12

Figure 3. Collected Data

To better analyze data and eliminate duplicate majors (due to different humanities and sciences majors in the same major, the extracted data may have duplicate professional names), we further organized and stored the data in the database, and established two views. One view is specifically used to analyze the voluntary selection situation by recruitment v_batch. A chart specifically used to analyze the selection of subjects according to their preferences in humanities and sciences v_wen_li.

```
CREATE VIEW `v_wen_li` AS SELECT
`henan`.`name` AS `name`,
`henan`.`wen_li` AS `wen_li`,
sum( `henan`.`plan` ) AS `plan`,
sum( `henan`.`wish` ) AS `wish`,
sum( `henan`.`recruit` ) AS `recruit`
FROM
  `henan`
WHERE
  ( `henan`.`year` = '2019' )
GROUP BY
  `henan`.`year`,
  `henan`.`wen_li`,
  `henan`.`name`;
```

For the v_batch's code:

For the v_wen_li's code:

```
CREATE VIEW `v_batch` AS SELECT
`henan`.`name` AS `name`,
`henan`.`batch` AS `batch`,
sum( `henan`.`plan` ) AS `plan`,
sum( `henan`.`wish` ) AS `wish`,
sum( `henan`.`recruit` ) AS `recruit`
FROM
  `henan`
WHERE
  ( `henan`.`year` = '2019' )
GROUP BY
  `henan`.`year`,
  `henan`.`batch`,
  `henan`.`name`;
```

4. Data Analysis

After the data is collected, we can analyze the data. The main packages we use here are NumPy, Pandas, and pyECharts. The NumPy package provides many dimensional arrays and matrix operations, which are often used for scientific calculations. Another is the Pandas package based on this package, which is mainly used for data analysis. The pyEcharts package is mainly used for visualizing data and provides various charts.

```
def top10_ben():
    conn = pymysql.connect(host='localhost', user='root', password='root', port=3306,
db='gaokao', charset='utf8')
    cursor = conn.cursor()
    sql = "select * from v_batch where batch='本科第一批'" db =
    pd.read_sql(sql, conn)
    df = db.sort_values(by="wish", ascending=False) dom =
    df[['name', 'wish']]

    attr = np.array(dom['name'][:10])
    v1 = np.array(dom['wish'][:10])
    attr = ["{}".format(i.replace(" ", "")) for i in attr]
    v1 = ["{}".format(int('%d' % i)) for i in v1]

    bar = (
        Bar()
            .add_xaxis(attr)
            .add_yaxis("报考志愿数 单位: 人", v1)
            # .reversal_axis()
            # .set_series_opts(label_opts=opts.LabelOpts(position="right"))
            .set_global_opts(
                title_opts={"text": "2019 年本科第一批报考志愿 Top10"},
                xaxis_opts=opts.AxisOpts(axislabel_opts=opts.LabelOpts(rotate=-15))
            )
    )
    bar.render("./2019 年本科第一批报考志愿 Top10.html") #render 会生成本地
HTML 文件, 默认是当前目录, 也可以传入路径参数

if __name__ == '__main__':
    top10_ben()
```

The top 10 majors voluntarily selected for undergraduate studies in 2019 were Computer Science, Clinical Medicine, Computer Science and Technology, Electrical Engineering and Automation, English, Electronic Information, Law, Accounting, Software Engineering, and Economics (Figure 4).

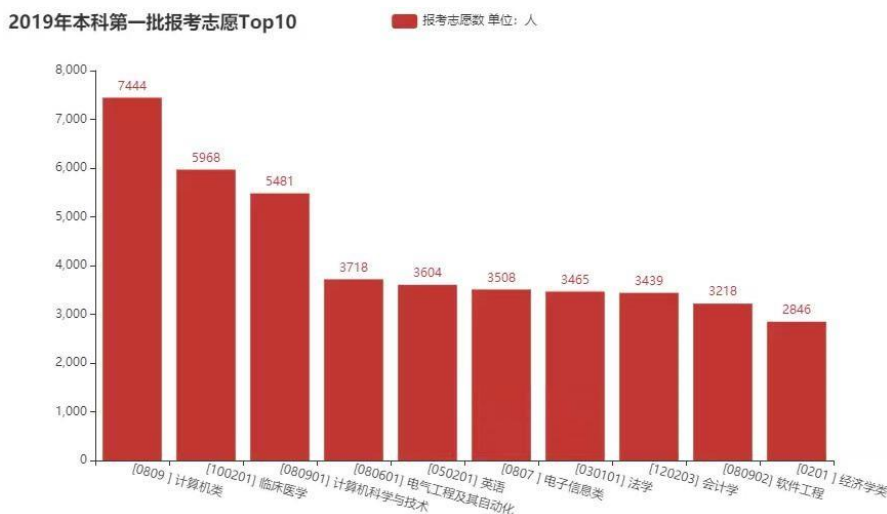


Figure 4. The top 10 majors voluntarily selected for undergraduate studies in 2019

The top 10 majors chosen by the second batch of undergraduate students in 2019 were Accounting, Computer Science and Technology, English, Software Engineering, Chinese Language and Literature, Mechanical Design and Manufacturing and Automation, Law, Primary Education, Electrical Engineering and Automation, and Clinical Medicine (Figure 5).

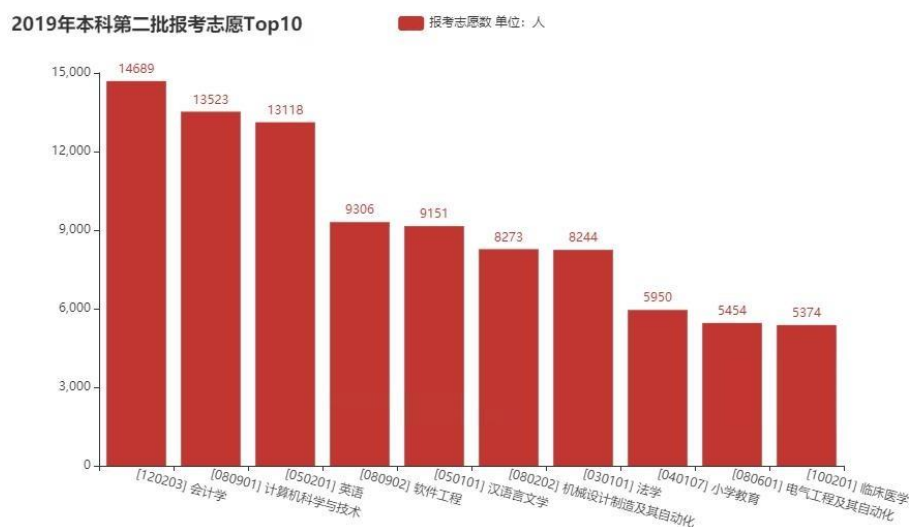


Figure 5. The top 10 majors chosen by the second batch of undergraduate students in 2019

The top 10 majors selected for the 2019 vocational college admission are Accounting, Nursing, Computer Application Technology, Chinese Education, Preschool Education, Computer Network Technology, Software Technology, English Education, Primary Education, and E-commerce (Figure 6).

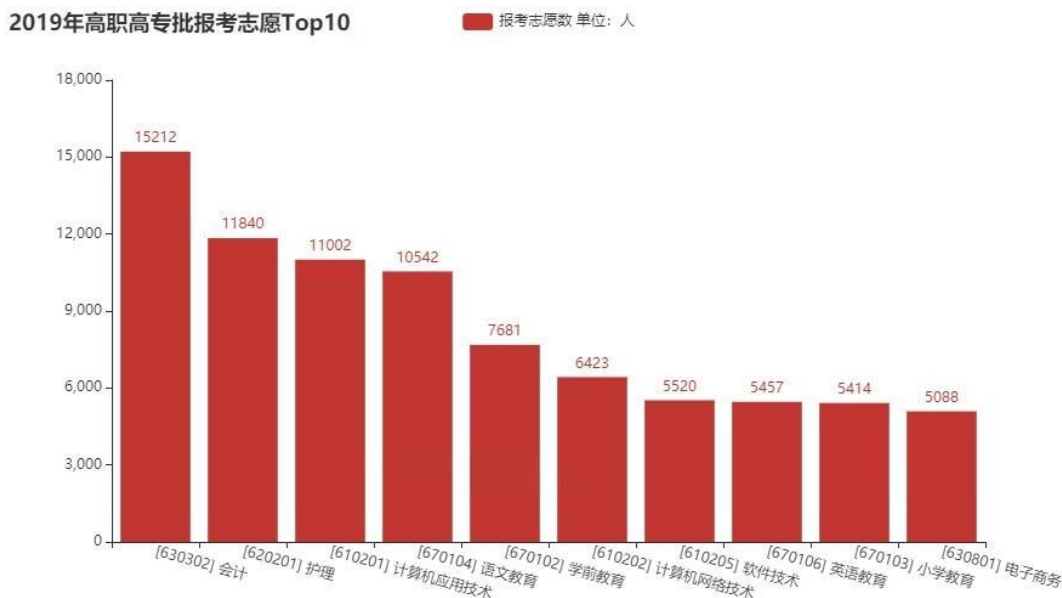


Figure 6. The top 10 majors selected for the 2019 vocational college admission

The top 10 majors voluntarily selected for the entire liberal arts major in 2019 were Chinese education, Chinese language and literature, accounting, English, law, accounting, nursing, preschool education, computer application technology, and English education (Figure 7).

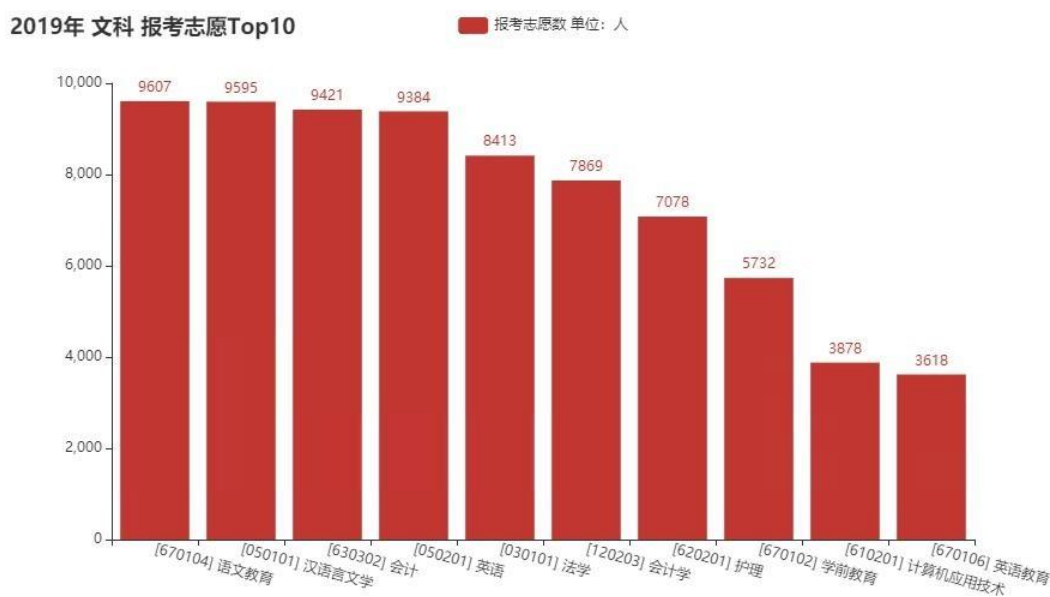


Figure 7. The top 10 majors voluntarily selected for the entire liberal arts major in 2019

The top 10 majors voluntarily selected for the entire science major in 2019 are Computer Science and Technology, Software Engineering, Clinical Medicine, Accounting, Mechanical Design and Manufacturing and Automation, Computer Science, Electrical Engineering and Automation, English, Computer Application Technology, Mathematics and Applied Mathematics (Figure 8).

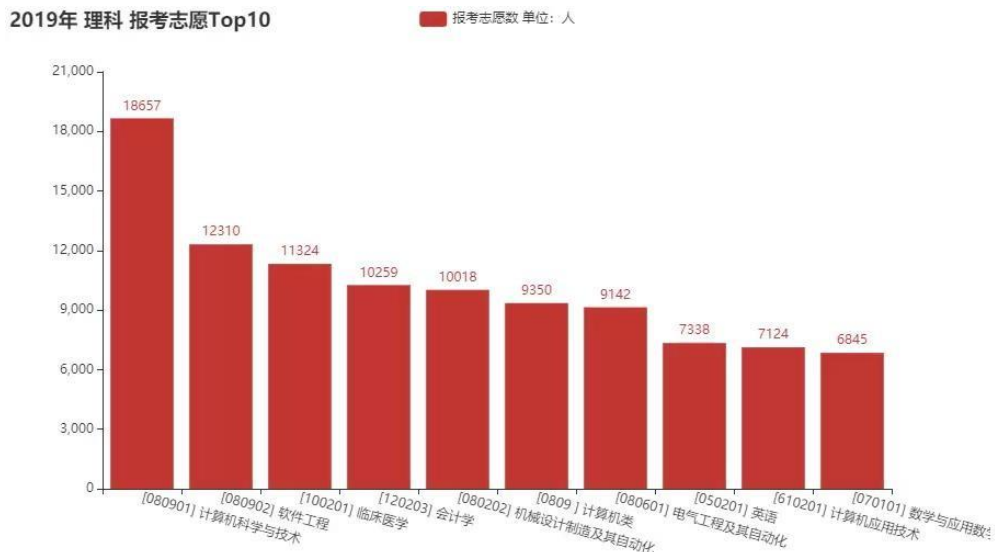


Figure 8. The top 10 majors voluntarily selected for the entire science major in 2019

CONCLUSIONS AND FUTURE RESEARCH

This article uses Python's crawling and data analysis techniques, based on the voluntary selection data publicly available on the official website of the Henan Provincial Admissions Office, to carry out page analysis, code writing, breakthroughs in anti-crawling, data extraction, cleaning, storage, analysis, and display. The collected data was analyzed and visualized based on the top 10 majors selected for the first batch of undergraduate college entrance examination in Henan Province in 2019, the top 10 majors selected for the second batch of undergraduate college entrance examination, the top 10 majors selected for vocational colleges, the top 10 majors selected for liberal arts, and the top 10 majors selected for science. The analyzed results provide the necessary support for universities to carry out corresponding majors and can also provide a reference for students who are about to participate in the college entrance examination, which is of great significance.

PRACTICAL IMPLICATIONS

This study will help students see the popular majors. They can use this as a decision on whether to take the popular majors or the less popular ones. In the university, they can use the information to plan for additional rooms, teachers, or facilities.

ACKNOWLEDGEMENT

First, I would like to give my heartfelt thanks to all the people who have ever helped me in this paper. My sincere and hearty thanks and appreciations go firstly to my supervisor, Mr. Rex P. Bringula, Ph.D, College of Computer Studies and Systems, University of the East, whose suggestions and encouragement have given me much insight into this paper. It has been a great privilege and joy to study under his guidance and supervision. Furthermore, it is my honor to benefit from his personality and diligence, which I will treasure my whole life. I am also extremely grateful to all my colleagues who have kindly provided me assistance and companionship while preparing this paper. In addition, many thanks go to my family for their unfailing love and unwavering support. Finally, I am grateful to all those who devote much time to reading this thesis and give me much advice, which will benefit me in my later study. This paper is an extended version of the previous published in *Computer Campus* (2019, issue 10).

FUNDING

The study did not receive funding from any institution.

DECLARATIONS

Conflict of Interest

No potential conflict of interest was reported by the authors.

Informed Consent

Written informed consent was obtained from all the participants before the enrollment of this study.

Ethics Approval

Written informed consent for publication of this paper was obtained from the Jiaozuo University of Arts and Science and all authors.

REFERENCES

Bian, Q., Wang, Z., & Ku, Z. (2020). Design and Implementation of a Data Analysis System for Recruitment Positions Based on Python. *Microcomputer Applications*, 9, 18-19.

- Dongmei, W., Zhongxiu, H., & Jianmei, T. (2018), Research on web information acquisition methods based on Python. *Software Guide*, 1, 41-43.
- Fu, R., & Li, X. (2021). Malicious attacks on the web and crawling of information data by Python technology. *Security and Privacy*, 4(5), e173.
- Hejing, W., Fang, L., Long, Z., Yabin, S., & Ran, C. (2020). Application research of crawler and data analysis based on Python. *International Journal of Advanced Network, Monitoring and Controls*, 5(2), 64-70.
- Li, L. (2021). Employment Data Analysis based on Python Crawler Technology. In *Proceedings of 4th International Workshop on Education Reform and Social Sciences (ERSS 2021)*(pp. 164-168). College of Information Engineering, Yunnan Vocational College of Mechanical and Electrical Technology. doi:10.26914/c.cnkihy.2021.049800.
- Lirong, G. (2017). Python-based web crawler program design. *Electronic Technology and Software Engineering*, 23, 248-249.
- Luo, K. (2023). A Study and Implementation of an Optimized University Library Book Recommendation System Based on Artificial Intelligence and Python Crawler Scraping Technology. *Journal of Artificial Intelligence Practice*, 6(2), 9-17.
- Ma, X., & Yan, M. (2021, September). Design and Implementation of Crawler Program Based on Python. In *Journal of Physics: Conference Series* (Vol. 2033, No. 1, p. 012205). IOP Publishing.
- Ma, T., & Zhang, Z. (2021, December). Medical Consultation System based on Python Web crawler. In *2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT)* (pp. 737-741). IEEE. doi:10.26914/c.cnkihy.2021.065511.
- Pei, L. (2019). Research on Python-based web crawler and anti-crawler technology. *Computer and digital engineering*, 47(6), 1415-1420.
- Wang, A. (2019). Research and Design of Distributed Multi-topic Web Crawler Based on Python. *Computer Informatization and Mechanical System*, 6.
- Wang, J., Li, Q., Zhou, J., Wen, Y., Zhang, Y., & Lin, P. (2021). News crawling based on Python crawler. In *Journal of Physics: Conference Series* (Vol. 1757, No. 1, p. 012117). IOP Publishing.
- Wu, H., & Cui, R. (2021). Research on Data Collection and Analysis of Second Hand House in China Based on Python. *International Journal of Advanced Network, Monitoring and Controls*, 6(2), 37-47.
- You, F., Gong, H., Guan, X., Cao, Y., Zhang, C., Lai, S., & Zhao, Y. (2018). Design of data mining of wechat public platform based on python. In *Journal of Physics: Conference Series* (Vol. 1069, p. 012017). IOP Publishing.
- Zhang, N. (2021, July). Design of Movie Data Visualization System Based on Web Crawler. In *Journal of Physics: Conference Series* (Vol. 1971, No. 1, p. 012029). IOP Publishing.
- Zhonghua, Z., Zhang, H., & Xie, J. (2014). Python-based data crawler for Sina Weibo. *Computer Applications*, 34, 3131-3134.

Author's Biography

Mr. Lu Bei, a member of the Communist Party of China, is of Han nationality and hails from Jiaozuo, Henan Province. Born in August 1986, he holds a master's degree and currently serves as the Director of the Office of the School of Artificial Intelligence at Jiaozuo University. His primary research focuses on areas such as mobile application development, data mining, and artificial intelligence. Presently, he is actively involved in teaching and research, particularly in courses related to mobile application development, data mining, and artificial intelligence.