Experiment Report

Name: Naixuan Wang
Student Number: 515030910618

Part I
Introduction

Who is more likely to gain a large number of citations? Predicting the future influential researchers in big scholarly networks. If we want to know in some certain research field who is most likely to win a prize, or in the future several years, who will publish most papers in a famous journal, we need to analyze the data of a huge number of researchers in the last several years and make a closest rediction.

This project is to predict the most influential researchers in the future. The definition of it is not definite, and many factors can contribute to it, such as paper numbers, paper citations, research field, co-authors and so on.

To simplify and quantificat the problem, we just use the citation numbers to measure the level of influence.

The problem then becomes how to use the data of the last several years to predict the citation numbers, or other data like H-index and G-index in the next few years.

Part II
Basic thoughts

For any researcher, if we know his or her publishing papers, citations, and where he or she published the paper on in the last few years, we can build models to analyze the changing trade of the data, using the trade of the data growing, we can predict the citations, paper numbers and H-index in the next several years.

So what we need is many researchers' data in some certain years. We believe the data must be changing in some common trade, and we just want to find the general rule of the data growing.

Part III
Data Preperation

We crawled 1000 researchers' data from *www.xueshu.baidu.com*, after throwing the data which is not useful, there are 987 left, so this is our data set. Time limiting, we can't find another larger data set, but with more time, we can gain much more researchers' data.

For the 987 researchers, we want to get most of their information, and as a result, we did the most(See Figure 1., Figure 2. and Figure 3.).

Figure 1: data we have dealedwith

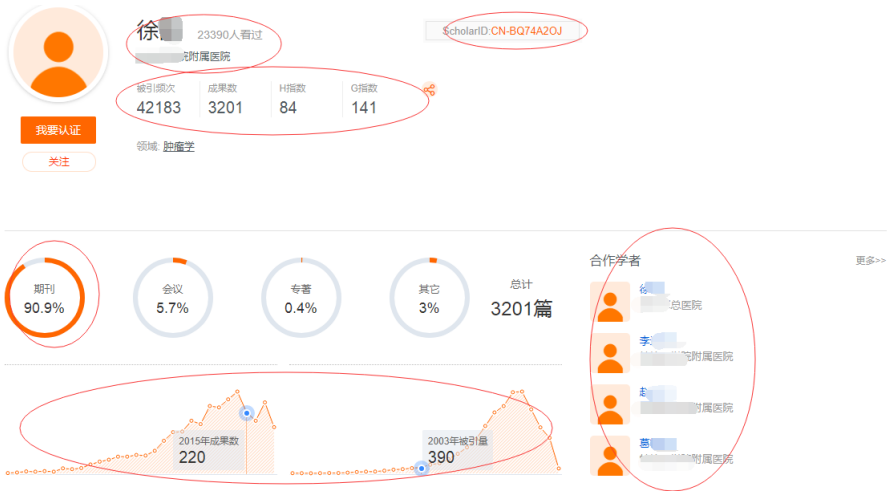

Figure 2: data we get from the researcher's personal main page

Figure 3: data we get from the researcher's personal main page

As we can see, we almost crawled every useful information in the researcher's main page, but there are still some data we want, but have no time to get it(See Figure 4.).

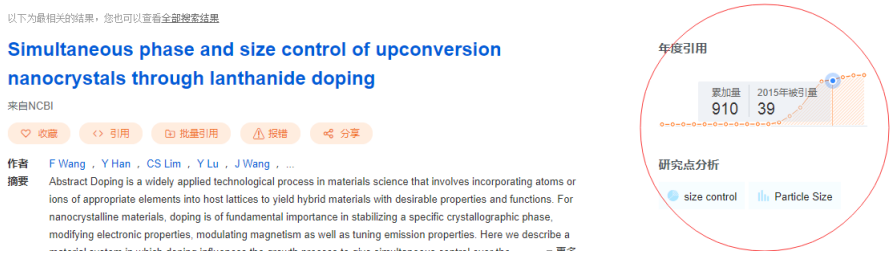

Figure 4: data we want but not get yet

We can get all the researcher's papers' links in his or her main page, and there are many important and useful data we want to get, and particularly, the data is what our model needs, if we have time to get them, we will predict the citation more accurate.

Part IV
Why This Data Set

Before we finally decided to chose this data set, we tried every way to find a good data set.

At the beginning, we are interested in the data sets on $https://www.aminer.cn/citation$, there are many data sets on it, though, we found them useless. For example, most of the data struct is in the following manner(See Figure 5.), there are the paper's title, authors, publish year, publish journal name, index, papers it referring to, and its abstract. First, it doesn't give the citation numbers of every paper, and then, the reference information is among these papers, that is to say, if we added them up, we can't get a paper's all citation information, so we just gave up this data set.

Figure 5: data struct in the data set

Since there are no data sets are ready for us to us, so we decided to crawl the data from some paper websites.

Our first target is *WebofScience*, but after we compared it with *scholar.google.com*, we decided to turn to the latter on. I wrote codes to achieve it, and it goes well at first, I just finished 1000 researcher's infomation collection and his every paper's data(See Figure 6.), however, something wrong then happeded. I met HTTP ERROR 503 and couldn't solve it, even if now it is solved, but at that time, I only had to turn to *xushu.baidu.com*.

Figure 6: some links I have already got

There are also plenty of data on *xueshu.baidu.com*, even more kinds of.

Many useful data is valuable to analyse. This is why we choose it to be our data sets.

Part V
Source Code

I wrote several to crawl the data down, and analyzed them step by step. The key part is using POST to get all the pages' data(See Figure 7.).

```
    return ScholarID, entity_id, pages[1]
def savepapers(ScholarID, entity_id, pages):
    url = 'http://xueshu.baidu.com/scholarID/' + ScholarID
    path = 'authors'
    headers = {
        'Accept': 'text/html, */*; q=0.01',
        'Accept-Encoding': 'gzip, deflate',
        'Accept-Language': 'zh-CN, zh; q=0.9, en-US; q=0.8, en; q=0.7',
        'Content-Length': '131',
        'Content-Type': 'application/x-www-form-urlencoded; charset=UTF-8',
        'User-Agent': 'Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.33E
        'Host': 'xueshu.baidu.com',
        'Origin': 'http://xueshu.baidu.com',
        'Proxy-Connection': 'keep-alive',
        'Referer': url,
        'X-Requested-With': 'XMLHttpRequest'
    }
    for page in range(1, int(pages)+1):
        from_data = {
            'cmd': 'academic_paper',
            'entity_id': entity_id,
            'bsToken': '5242892358eaf2dfdb9aeaa0a65d33f0',
            'sc_sort': 'sc_time',
            'curPageNum': str(page)
        }
        time.sleep(random.randint(0, 3))
        result = requests.post('http://xueshu.baidu.com/usercenter/data/author', data=from_data, headers=headers)
        content = result.content.decode()
        name = validateTitle(url) + str(page)
        if not os.path.exists(path + '/' +ScholarID):
```

Figure 7: using POST to get the data in the next page

There are several steps of collecting data. Firstly, I crawled the pages from the web and saved them, then I can get all the useful information in the pages, and from the first page, I can get page 2, page 3 and so on. I saved all of them because there are page links on them. After having the pages, I can read the paper links in them, and get the paper pages, in which there are every paper's detail infomation.

All detail codes can be viewed on github, $https://github.com/NashWang1997/Citation-Prediction$. Some data sets are big and I may put them into my netdisc, which is not ready yet.