Indeed Jobs Scrapping

March 31, 2021

[2]:
```
from bs4 import BeautifulSoup
import requests

import csv
from datetime import datetime
```

[3]:
```
def get_url(position, location):
    # generate a url from position and location #
    url_temp = 'https://ca.indeed.com/jobs?q={}&l={}'
    url = url_temp.format(position, location)
    return url
```

[4]:
```
url = get_url('data analyst', 'toronto on')
```

0.1 Extract Raw HTML

[5]:
```
response = requests.get(url)

soup = BeautifulSoup(response.text, 'html.parser')

cards = soup.find_all('div', 'jobsearch-SerpJobCard')
```

0.2 Prototype the model with a single record

[6]:
```
card = cards[0]
```

[7]:
```
card
```

[7]:
```
<div class="jobsearch-SerpJobCard unifiedRow row result" data-ci="360139833"
data-empn="1213061919789664" data-jk="e88963836eff09d5"
id="pj_e88963836eff09d5">
<style>
.jobcard_logo{margin:6px 0}.jobcard_logo img{width:auto;max-width:80px;max-
height:30px}.jasxrefreshcombotst .jobcard_logo img{max-height:2rem;max-
width:100%}
</style>
<h2 class="title">
```


Data Analyst
new</h2>

AgencyAnalytics

Toronto, ON
•
Remote

$70,000 a year

<ul style="list-style-type:circle;margin-top: 0px;margin-bottom: 0px;padding-left:20px;">
<li style="margin-bottom:0px;">Create dashboards in hubspot to efficiently analyze data.
<li style="margin-bottom:0px;">Use hull.io to fetch, unify and push data.
Create a system to monitor and keep data organized.

5 days ago <button aria-expanded="false" aria-haspopup="true" aria-label="save or dislike" class="job-reaction-kebab" data-ol-has-click-handler=""

onclick="toggleKebabMenu('e88963836eff09d5', true, event); return false;"
tabindex="0"></button><button class="job-
reaction-kebab-item job-reaction-save" data-ol-has-click-handler=""
onclick="changeJobState('e88963836eff09d5', 'save', 'linkbar', true, '');return
false;"><svg focusable="false" height="16" viewbox="0 0 24 24"
width="16"><g><path d="M16.5,3A6,6,0,0,0,12,5.09,6,6,0,0,0,7.5,3,5.45,5.45,0,0,0
,2,8.5C2,12.28,5.4,15.36,10.55,20L12,21.35,13.45,20C18.6,15.36,22,12.28,22,8.5A5
.45,5.45,0,0,0,16.5,3ZM12.1,18.55l-0.1.1-0.1-.1C7.14,14.24,4,11.39,4,8.5A3.42,3.
42,0,0,1,7.5,5a3.91,3.91,0,0,1,3.57,2.36h1.87A3.88,3.88,0,0,1,16.5,5,3.42,3.42,0
,0,1,20,8.5C20,11.39,16.86,14.24,12.1,18.55Z"
fill="#2d2d2d"></path></g></svg>Save
job</button><button class="job-reaction-kebab-item job-reaction-dislike"
data-ol-has-click-handler="" onclick="dislikeJob(false, false,
'e88963836eff09d5', 'unsave', 'linkbar', true, '');"><span class="job-reaction-
dislike-icon">Not
interested</button><button class="job-reaction-kebab-item job-reaction-
report" onclick="reportJob('e88963836eff09d5');"><span class="job-reaction-
report-icon">Report
job</button> <span class="result-link-bar-
separator">·<a class="sl resultLink save-job-link" href="#"
id="sj_e88963836eff09d5" onclick="changeJobState('e88963836eff09d5', 'save',
'linkbar', true, ''); return false;" title="Save this job to my.indeed">Save
job <script>if (!window['sj_result_e88963836eff09d5'])
{window['sj_result_e88963836eff09d5'] =
{};}window['sj_result_e88963836eff09d5']['showSource'] = false;
window['sj_result_e88963836eff09d5']['source'] = "Indeed";
window['sj_result_e88963836eff09d5']['loggedIn'] = false;
window['sj_result_e88963836eff09d5']['showMyJobsLinks'] =
false;window['sj_result_e88963836eff09d5']['baseMyJobsUrl'] =
"https://myjobs.indeed.com";window['sj_result_e88963836eff09d5']['undoAction'] =
"unsave";window['sj_result_e88963836eff09d5']['relativeJobAge'] = "5 days
ago";window['sj_result_e88963836eff09d5']['jobKey'] = "e88963836eff09d5";
window['sj_result_e88963836eff09d5']['myIndeedAvailable'] = true;
window['sj_result_e88963836eff09d5']['showMoreActionsLink'] =
window['sj_result_e88963836eff09d5']['showMoreActionsLink'] || false;
window['sj_result_e88963836eff09d5']['resultNumber'] = 0;
window['sj_result_e88963836eff09d5']['jobStateChangedToSaved'] = false;
window['sj_result_e88963836eff09d5']['searchState'] = "q=data
analyst&l=toronto+on";
window['sj_result_e88963836eff09d5']['basicPermaLink'] =
"https://ca.indeed.com"; window['sj_result_e88963836eff09d5']['saveJobFailed'] =
false; window['sj_result_e88963836eff09d5']['removeJobFailed'] = false;
window['sj_result_e88963836eff09d5']['requestPending'] = false;
window['sj_result_e88963836eff09d5']['currentPage'] = "serp";
window['sj_result_e88963836eff09d5']['sponsored'] =
true;window['sj_result_e88963836eff09d5']['reportJobButtonEnabled'] = false;
window['sj_result_e88963836eff09d5']['showMyJobsHired'] = false;

```
window['sj_result_e88963836eff09d5']['showSaveForSponsored'] = true;
window['sj_result_e88963836eff09d5']['showJobAge'] = true;
window['sj_result_e88963836eff09d5']['showHolisticCard'] = true;
window['sj_result_e88963836eff09d5']['showDislike'] = true;
window['sj_result_e88963836eff09d5']['showKebab'] = true;
window['sj_result_e88963836eff09d5']['showReport'] = true;</script></div></div>
</div>
</div>
<div class="tab-container">
<div class="sign-in-container result-tab"></div>
<div class="tellafriend-container result-tab email_job_content"></div>
</div>
</div>
```

[8]:
```
atag = card.h2.a
```

[38]:
```
job_title = atag.get('title')
```

[12]:
```
job_url = 'https://ca.indeed.com' + atag.get('href')
```

[19]:
```
company = card.find('span', 'company').text.strip()
```

[21]:
```
job_location = card.find('div', 'recJobLoc').get('data-rc-loc')
```

[25]:
```
job_summary = card.find('div', 'summary').text.strip()
```

[28]:
```
post_date = card.find('span', 'date').text
```

[33]:
```
today = datetime.today().strftime('%Y-%m-%d')
```

[37]:
```
try:
    job_salary = card.find('span', 'salaryText').text.strip()
except AttributeError:
    job_salary = ''
```

0.3 Generalize the model with a function

[40]:
```
def get_record(card):
    """ Extract job data from a single record """
    atag = card.h2.a
    job_title = atag.get('title')
    job_url = 'https://ca.indeed.com' + atag.get('href')
    company = card.find('span', 'company').text.strip()
    job_location = card.find('div', 'recJobLoc').get('data-rc-loc')
    job_summary = card.find('div', 'summary').text.strip()
    post_date = card.find('span', 'date').text
    today = datetime.today().strftime('%Y-%m-%d')
```

```
    try:
        job_salary = card.find('span', 'salaryText').text.strip()
    except AttributeError:
        job_salary = ''

    record = (job_title, company, job_location, post_date, today, job_summary,␣
↪job_salary, job_url)

    return record
```

[41]:
```
records = []

for card in cards:
    record = get_record(card)
    records.append(record)
```

[43]:
```
records[1]
```

[43]:
```
('Data Analyst',
 'Parmida E-commerce',
 'Toronto, ON',
 '16 days ago',
 '2021-03-05',
 'To do well in this role you need a fine eye for detail, experience as a data
analyst is an asset, and a deep understanding of the popular data analysis
tools…',
 '$45,000 a year',
 'https://ca.indeed.com/pagead/clk?mo=r&ad=-6NYlbfkN0AN7Y4eKAyl_DmVMU9zc2RZJG70w
-w9zONODxlRVZc__K7M27SNOPPWFlvNpjd6b56Kn2W5SUywiTkg8AsULtw1riiUPhOBjdqTtsOWNM-83
rQGhLNUo2h4OJII5GjRD42dFhgP758yPrFGaSFcLAYDvq3gVYmR6BEc2SS2YTLmUurppnRM5oSmXRG9X
saoZMbTBP1wIlCHqNbwkBpSyObwN0Ho613-45nTNH2JK03BCjE4NLvIUyqe2IEbj6ZHNub5oQ9XdQcF_
E5EOQrxK5Gtq2BZm1Huiye9gcVk4KALHR5jY-EB4NyXHB8m-_zP6goBit1xdaUlb0tznxM9SmDczZHeL
HAfdJcC8Yhcwt8XEXLKYL9D4j212VrC6z3ujnOu8IPqp_vD3YRMZEO4WXyKPCgo1UjuZJ33V4-eP9H32
_rmN1TnvdMfMAXQnes9FVI=&p=1&fvj=1&vjs=3')
```

0.4 Getting Next Page

[51]:
```
while True:
    try:
        url = 'https://ca.indeed.com' + soup.find('a', {'aria-label': 'Next'}).
↪get('href')
    except AttributeError:
        break

    response = requests.get(url)
    soup = BeautifulSoup(response.text, 'html.parser')
```

```
    cards = soup.find_all('div', 'jobsearch-SerpJobCard')

    for card in cards:
        record = get_record(card)
        records.append(record)
```

[52]:
```
len(records)
```

[52]: 786

0.5 Putting it all together

[57]:
```
from bs4 import BeautifulSoup
import requests
import csv
from datetime import datetime

def get_url(position, location):
    # generate a url from position and location #
    url_temp = 'https://ca.indeed.com/jobs?q={}&l={}'
    url = url_temp.format(position, location)
    return url

def get_record(card):
    """ Extract job data from a single record """
    atag = card.h2.a
    job_title = atag.get('title')
    job_url = 'https://ca.indeed.com' + atag.get('href')
    company = card.find('span', 'company').text.strip()
    job_location = card.find('div', 'recJobLoc').get('data-rc-loc')
    job_summary = card.find('div', 'summary').text.strip()
    post_date = card.find('span', 'date').text
    today = datetime.today().strftime('%Y-%m-%d')
    try:
        job_salary = card.find('span', 'salaryText').text.strip()
    except AttributeError:
        job_salary = ''

    record = (job_title, company, job_location, post_date, today, job_summary,␣
 ↪job_salary, job_url)

    return record


def main(position, location):
    """ Run the Main Program Rountine """
    records = []
```

```
    url = get_url(position, location)

    # Extract the job data
    while True:
        response = requests.get(url)
        soup = BeautifulSoup(response.text, 'html.parser')
        cards = soup.find_all('div', 'jobsearch-SerpJobCard')

        response = requests.get(url)
        soup = BeautifulSoup(response.text, 'html.parser')
        cards = soup.find_all('div', 'jobsearch-SerpJobCard')

        for card in cards:
            record = get_record(card)
            records.append(record)

        try:
            url = 'https://ca.indeed.com' + soup.find('a', {'aria-label':␣
↪'Next'}).get('href')
        except AttributeError:
            break

    # Save the job data
    with open('job_results.csv', 'w', newline='', encoding='utf-8') as f:
        writer = csv.writer(f)
        writer.writerow(['Job_Title', 'Company', 'Job_Location', 'Post_Date',␣
↪'Today', 'Job_Summary', 'Job_Salary', 'Job_URL'])
        writer.writerows(records)
```

0.6 Run the main program

[59]:
```
# run the main program
main('software developer', 'toronto on')
```

[ ]:
```

```