



Web crawler/ Spider

{ Autor: Matija Hemen

Upotreba



- ⌘ Dugo vrijeme pregleda stranice (pregled svih poslova na burzi)
- ⌘ Brzi pomak trendova → jako značajno za poslovne ideje
- ⌘ Velik broj javno dostupnih podataka
 - ⌘ Dohvaćanje tih podataka → Analiza → Veći opseg mogućnosti → Bolji izbor i zaključak
- ⌘ Web crawling – skupljanje podataka s „weba” od tud i naziv *spider*, crawler se spaja na stranicu i skida cijeli html
 - ⌘ *Search indexing*- spideri omogućuju „imenik” sa stranicama – google
 - ⌘ *Robots.txt*
- ⌘ *Web scraping* – obrada ili razvrstavanje podataka

Opis problema



- ⌘ Traženje posla na burzi → vremenski zahtjevno
- ⌘ Slanje e-maila, razvrstavanje poslova
- ⌘ Analiza svih ponuda
- ⌘ Spider treba znati što izdvojiti

Rješenje-Spider

- ⌘ Burza nam čak pruža javni RSS (*Really Simple Syndication – Stvarno jednostavne vijesti*)
- ⌘ RSS → skraćeni web format koji se stalno osvježava
 - ⌘ RSS nekad ne nudi sve što trebamo (email)
- ⌘ BeautifulSoup –python biblioteka
 - ⌘ raščlanjuje DOM tree, html elemente
 - ⌘ Možemo pristupiti i ispisati text elementa
- ⌘ Urllib.request – python biblioteka
 - ⌘ Služi za skidanje sadržaja web stranice

```
from bs4 import BeautifulSoup as soup
from urllib.request import urlopen as ureq
```

```
import json
import pandas as pd
```

- & Koristit ćemo još DataFrame za prikaz poslova
- & Stvaramo klasu Spider s atributima i metodama

```
class Spider:
    def __init__(self,url):
        self.url=url
        self.Poslovi=[]
        self.getPage()
```

- & Metodom getPage i urlopen() skidamo stranicu

```
def getPage(self):
    #uspostavljamo vezu sa serverom i skidamo stranicu
    page=ureq(self.url)
    phtml=page.read()
    page.close()
    #parsiramo tj. raščlanjvanja
    pagesoup=soup(phtml,"html.parser")
    self.getPageInfo(pagesoup)
```

Beautifulsoup raščlanjuje html
i šaljemo to u novu funkciju

```
def getPageInfo(self, pageSoup):
    #html item sadrže info o poslovima u RSS-feedu.
    poslovi=pageSoup.findAll("item")

    for posao in poslovi:
        try:
            pozicija=posao.title.text
            rokIndex=posao.description.text.find('Rok')
            mjestoIndex=posao.description.text.find('Mjesto')
            if(rokIndex <0 or mjestoIndex < 0):
                raise ValueError
            rok=posao.description.text[(rokIndex+16):(rokIndex+26)]
            mjesto=posao.description.text[(mjestoIndex+12)::].split(',')[0]
            link=posao.guid.text
        except ValueError:
            rok='/'
            mjesto='/'
        oglas=Posao(pozicija,rok,mjesto,link)
        self.Poslovi.append(oglas)
```

„parsiran” html, BeautifulSoup nam omogućava pomoću funkcije findAll staviti sve <item> u array.

Soup pronalazi <title> unutar <item> i .text ispisuje samo text od title.

```
<?xml version="2.0"?>
<channel>
  <title>Hrvatski zavod za zapošljavanje</title>
  <description>Pregled slobodnih radnih mjesta</description>
  <link>http://burzarada.hzz.hr</link>
  <language>hr</language>
  <ttl>60</ttl>
  <copyright>
    Copyright 2010, Hrvatski zavod za zapošljavanje,
  </copyright>
  <webMaster>info@hzz.hr</webMaster>
  <managingEditor>info@hzz.hr</managingEditor>
  <image>
    <title/>
    <width/>
    <height/>
    <url/>
    <link/>
  </image>
  <item>
    <title>DADILJA (M/Ž)</title>
    <subject>DADILJA (M/Ž)</subject>
    <pubDate>05.05.2020</pubDate>
    <description>
      Opis posla: DADILJA (M/Ž), Kategorija: PROFESOR
      OSJEČKO-BARANJSKA
    </description>
    <link>
      http://burzarada.hzz.hr/RadnoMjesto_Ispis.aspx
    </link>
    <guid>
      http://burzarada.hzz.hr/RadnoMjesto_Ispis.aspx
    </guid>
  </item>
```

RSS feed od burze

Unutar <item> nalaze se informacije koje nas zanimaju

Klasa Posao je opisana kasnije, te se ti objekti stavljaju u atribut array od spider-a.

```
class Spider:
    def __init__(self,url):
        self.url=url
        self.Poslovi=[]
        self.getPage()
```


Za svaki posao koji nađemo na RSS-u kreiramo objekt Posao s odgovarajućim atributima iz spider metode getPageInfo

```
class Posao:
    def __init__(self, pozicija, rok, mjesto, link):
        self.pozicija=pozicija
        self.rok=rok
        self.mjesto=mjesto
        self.link=link

    def __str__(self):
        return ("Pozicija: "+ self.pozicija
                + "\n Rok Za Prijavu: "+self.rok
                + "\n Mjesto Rada: "+self.mjesto
                + "\n Link do Oglasa: "+self.link+"\n"
                )
```

Posao se sastoji od: teksta <title>
teksta <description>
za rok i mjesto
i teksta <guid>
za link

```
def pronjuskaj(self,kriterij):
    vd={"Posao":[],
        "Mjesto":[],
        "Rok":[]}
    try:
        uvjet=kriterij.split(',')
        if(len(uvjet)<2):
            raise IndexError

        for p in self.Poslovi:
            if(p.pozicija.find(uvjet[0].upper())>0 and p.mjesto.find(uvjet[1].upper())>0):
                vd["Posao"].append(p.pozicija)
                vd["Mjesto"].append(p.mjesto)
                vd["Rok"].append(p.rok)

    except IndexError:
        for p in self.Poslovi:
            if(p.pozicija.find(kriterij.upper())>0):
                vd["Posao"].append(p.pozicija)
                vd["Mjesto"].append(p.mjesto)
                vd["Rok"].append(p.rok)

            if (p.mjesto.find(kriterij.upper())>0):
                vd["Posao"].append(p.pozicija)
                vd["Mjesto"].append(p.mjesto)
                vd["Rok"].append(p.rok)

    x=json.dumps(vd)
    val=pd.read_json(x)
    print(val)
```

Ovo je Spiderova metoda koja dodatno pronalazi poslove uz određeni kriterij (mjesto ili posao)

Kriterij može biti jedan ili dva, ako je jedan onda se pokreće except dio koda.


Vd je dict koji se napuni sa poslovima koji odgovaraju kriteriju i pretvori u json zapis

Potom se json zapis pretvara u DataFrame koji ispisuje poslove u 3 stupca

```
#x=json.dumps(vd)
#val=pd.read_json(x)
val=pd.DataFrame(vd, columns=['Posao', 'Mjesto', 'Rok', 'Link'])
print(val)
val.to_csv (r'C:\Users\Hemen\Desktop\export_dataframe.csv',header=True)
```

Ili ako ne želite json, možemo odmah u DataFrame i u CSV file.

Imamo i metodu koja stvara csv datoteku koja nam daje tablični prikaz u excelu. Sep=, \n pomaže da excel koristi zarez kao delimiter.



```
def createCSV(self, ime):  
    f=open(ime+".csv","w")  
    kategorije="sep=,\nPozicija, Rok Za Prijavu, Mjesto Rada, LINK\n"  
    f.write(kategorije)  
    for p in self.Poslovi:  
        f.write(p.pozicija.replace(',','|') + "," + p.rok  
                + "," + p.mjesto + "," + p.link + "\n")  
    f.close()
```

Svaki posao se upisuje u datoteku.

Link do RSS-a od burze za nastavnike

Stvaramo objekt Spider

```
url='https://burzarada.hzz.hr/rss/rsskat1004.xml'  
test=Spider(url)  
#test.createCSV("nastavnici")
```

Stvaramo CSV sa svim poslovima

```
#kriteriji mogu biti po mjestu, po poziciji ili oboje tada oba uvjeta moraju biti ispunjena  
test.pronjuskaj("Matem,Zagreb")  
print("\n\n")  
#Samo za Jupyter notebook  
#display(Markdown('---'))  
test.pronjuskaj("Zagreb")
```

	Posao	Mjesto	Rok
0	NASTAVNIK/ICA MATEMATIKE	ZAGREB	12.05.2020

DataFrame iz metode pronjuskaj

	Posao	Mjesto	Rok
0	2. DEFEKTOLOG/INJA /LOGOPED/INJA	ZAGREB	18.05.2020
1	AGENT ZA KORISNIČKU PODRŠKU NA FRANCUSKOM JEZI...	ZAGREB	12.05.2020
2	ASISTENT/ICA (DOKTORAND/ICA)	ZAGREB	18.05.2020
3	EDUKACIJSKI/A REHABILITATOR/ICA	ZAGREB	16.05.2020
4	EDUKACIJSKI/A REHABILITATOR/ICA	ZAGREB	15.05.2020
5	IZVANREDNI/A PROFESOR/ICA (NASLOVNO ZVANJE)	ZAGREB	25.05.2020
6	NASTAVNIK/ICA MATEMATIKE	ZAGREB	12.05.2020
7	ODGOJITELJ/ICA	ZAGREB	22.05.2020
8	POSLIJEDOKTORAND/ICA	ZAGREB	18.05.2020
9	REDOVITI/A PROFESOR/ICA PRVI IZBOR	ZAGREB	18.05.2020
10	REDOVITI/A PROFESOR/ICA U TRAJNOM ZVANJE (NASL...	ZAGREB	25.05.2020

CSV datoteka koju stvara Spider

C58			
f _x			
	A	B	C
			D
1	Posicija	Rok Za Prijavu	Mjesto Rada
2	DADILJA (M/Ž)	15. 5. 2020	OSIJEK
3	AGENT ZA KORISNIČKU PODRŠKU NA FRANCUSKOM JEZIKU (M/Ž)	12. 5. 2020	ZAGREB
4	ASISTENT/ICA (DOKTORAND/ICA)	18. 5. 2020	ZAGREB
5	DOCENT/ICA IZ PODRUČJA BIOMEDICINE I ZDRAVSTVA	8. 5. 2020	PULA-POLA
6	DOCENT/ICA IZ ZNANSTVENOG PODRUČJA BIOTEHNIČKIH ZNANOSTI ZNANSTVENOG POLJA POLJOPRIVREDA ZA RAD NA KATEDRI ZA HERBOLOGIJU I FITOFARMACIJU	25. 5. 2020	OSIJEK
7	DOCENT/ICA IZ ZNANSTVENOG PODRUČJA DRUŠTVENIH ZNANOSTI ZNANSTVENOG POLJA PSIHLOGIJA	17. 5. 2020	OSIJEK
8	DOCENT/ICA IZ ZNANSTVENOG PODRUČJA HUMANISTIČKIH ZNANOSTI ZNANSTVENOG POLJA FILOZOFIJA	17. 5. 2020	OSIJEK
9	EDUKACIJSKI/A REHABILITATOR/ICA	15. 5. 2020	ZAGREB
10	IZVANREDNI/A PROFESOR/ICA (NASLOVNO ZVANJE)	25. 5. 2020	ZAGREB
11	IZVANREDNI/A PROFESOR/ICA IZ ZNANSTVENOG PODRUČJA DRUŠTVENIH ZNANOSTI ZNANSTVENOG POLJA INFORMACIJSKE I KOMUNIKACIJSKE ZNANOSTI	17. 5. 2020	OSIJEK
12	LOGOPED/ICA	13. 5. 2020	LOBOR
13	LOGOPED/LOGOPETKINJA	12. 5. 2020	KRIŽEVCI
14	NASTAVNIK/ICA MATEMATIKE	12. 5. 2020	ZAGREB
15	NASTAVNIK/NASTAVNICA POVIJESTI	12. 5. 2020	VUKOVAR
16	ODGOJITELJ/ICA	13. 5. 2020	VIROVITICA
17	ODGOJITELJ/ICA PREDŠKOLSKJE DJECE	15. 5. 2020	SESVETE
18	ODGOJITELJ/ICA PREDŠKOLSKJE DJECE	7. 5. 2020	ZAGREB-DUBRAVA
19	ODGOJITELJ/ICA PREDŠKOLSKJE DJECE	7. 5. 2020	ZAGREB-DUBRAVA
20	ODGOJITELJ/ODGOJITELJICA	19. 5. 2020	VARAŽDINSKE TOPLICE
21	ODGOJITELJ/ODGOJITELJICA	12. 5. 2020	LUBEŽICA
22	ODGOJITELJ/ODGOJITELJICA PREDŠKOLSKJE DJECE	12. 5. 2020	MAGADENOVAC
23	POMOĆNIK/ICA U NASTAVI	7. 5. 2020	ZAGREB
24	POSUJEDOKTORAND/ICA	18. 5. 2020	ZAGREB
25	POSUJEDOKTORAND/ICA IZ ZNANSTVENOG PODRUČJA BIOTEHNIČKIH ZNANOSTI ZNANSTVENOG POLJA POLJOPRIVREDA	25. 5. 2020	OSIJEK
26	POSUJEDOKTORAND/ICA IZ ZNANSTVENOG PODRUČJA BIOTEHNIČKIH ZNANOSTI ZNANSTVENOG POLJA POLJOPRIVREDA	25. 5. 2020	OSIJEK
27	POSUJEDOKTORAND/ICA IZ ZNANSTVENOG PODRUČJA DRUŠTVENIH ZNANOSTI ZNANSTVENOG POLJA PEDAGOGIJA	17. 5. 2020	OSIJEK
28	POSUJEDOKTORAND/ICA IZ ZNANSTVENOG PODRUČJA DRUŠTVENIH ZNANOSTI ZNANSTVENOG POLJA PSIHLOGIJA	17. 5. 2020	OSIJEK
29	POSUJEDOKTORAND/ICA IZ ZNANSTVENOG PODRUČJA TEHNIČKIH ZNANOSTI ZNANSTVENOG POLJA ELEKTROTEHNIKA U ZAVODU ZA PROGRAMSKO INŽENJERSTVO	29. 5. 2020	OSIJEK
30	PROFESOR / ICA ENGLESKOG JEZIKA	7. 5. 2020	VALPOVO
31	PROFESOR / PROFESORICA ENGLESKOG JEZIKA	6. 5. 2020	POŽEGA
32	PROFESOR / PROFESORICA NJEMAČKOG JEZIKA	6. 5. 2020	POŽEGA
33	PROFESOR / ICA NJEMAČKOG JEZIKA	7. 5. 2020	VALPOVO
34	PROFESOR / ICA TALIJANSKOG JEZIKA	7. 5. 2020	VALPOVO
35	PROFESOR/ICA ENGLESKOG JEZIKA	6. 5. 2020	KUTINA
36	PROFESOR/ICA NJEMAČKOG JEZIKA	6. 5. 2020	KUTINA
37	PROFESOR/PROFESORICA ENGLESKOG JEZIKA	6. 5. 2020	BJELOVAR
38	PROFESOR/PROFESORICA NJEMAČKOG JEZIKA	6. 5. 2020	BJELOVAR
39	PROFESOR/PROFESORICA NJEMAČKOG JEZIKA	6. 5. 2020	SLATINA
40	PROFESOR/PROFESORICA NJEMAČKOG JEZIKA	7. 5. 2020	NAŠICE
41	PROFESOR/PROFESORICA TALIJANSKOG JEZIKA	7. 5. 2020	NAŠICE
42	PROFESOR/PROFESORICA TALIJANSKOG JEZIKA	7. 5. 2020	OSIJEK
43	PROFESOR/PROFESORICA TALIJANSKOG JEZIKA	6. 5. 2020	BJELOVAR
44	PSIHOLOG/PSIHOLOGINJA	/	/
45	RAVNATELJ/RAVNATELJICA Dječjeg vrtića Cipelica	7. 5. 2020	ČAKOVEC
46	REDOVITI/A PROFESOR/ICA PRVI IZBOR	18. 5. 2020	ZAGREB
47	REDOVITI/A PROFESOR/ICA U TRAJNOM ZVANJE (NASLOVNO ZVANJE)	25. 5. 2020	ZAGREB

Analiza rješenja

- ⌘ Spider jako brzo prikuplja podatke čak ako i nema RSS-a (prilagodba)
- ⌘ Puno oglasa na burzi čak traži pismenu molbu, ako to zanemarimo i napravimo dodatnu metodu za slanje maila vrlo brzo ćemo poslati CV na sve radne pozicije koje nas zanimaju te tako povećati %.
- ⌘ Pruža dobru analizu javno dostupnih podataka
- ⌘ Različitost stranica
 - ⌘ Spider se mora modelirati po svakoj stranici posebno
 - ⌘ Neke stranice mogu blokirati botove
 - ⌘ Privatni podaci

