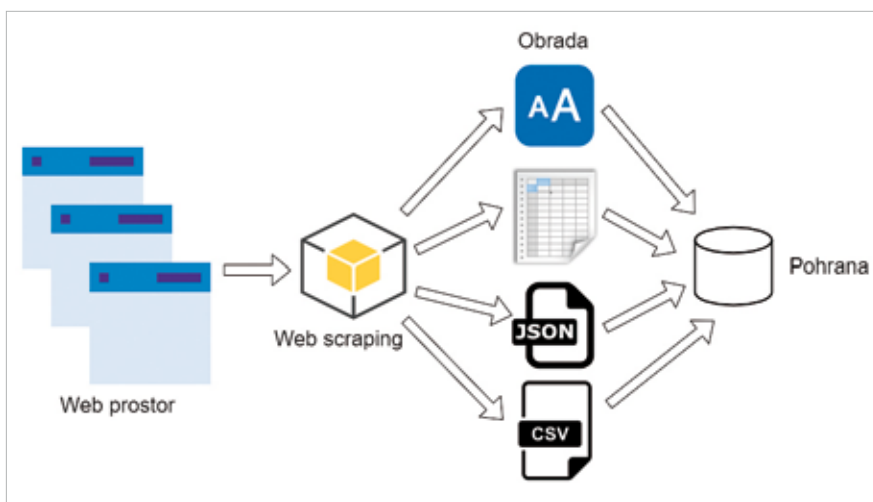


Kada ti život da web, izvuci

Kako količina na webu dostupnih informacija raste, te informacije postaju sve vrednijima, područje automatiziranog sustavnog pretraživanja weba dobiva sve veću važnost. Više informacija i bolje informacije od iznimne su važnosti za donošenje poslovnih odluka, zbog čega je korisno znati kako automatizirano doći do većih količina informacija na Internetu te kako izvući i obraditi tražene informacije

Domagoj Marić, Megatrend poslovna rješenja



Dobivanje strukturiranih podataka web-scrapingom

Što je to *web-scraping*? **Dva glavna pojma u području ekstrakcije web-sadržaja su *web-scraping* i *web-crawling*.** *Web-scraping* jest sistematizirana ekstrakcija tekstualnog ili medijskog sadržaja s web-stranica, postignuta korištenjem alata zvanih *web-scraperi*. Koncept *web-scrapinga* temelji se na korištenju metoda *web-crawlinga*, automatiziranog sustavnog pretraživanja weba, postignutog praćenjem poveznica web-stranica pomoću *web-crawlera*.

Procesi *web-scrapinga* i *web-crawlinga* čine kontinuirani ciklus: *crawlingom* dolazimo do HTML dokumenata iza web-stranica, iz kojih izvlačimo željeni sadržaj i poveznice na ostale web-stranice pomoću *scrapinga*, te dalje vršimo *crawling* po prikupljenim poveznicama.

Zašto *web-scraping*? Poduzećima *web-scraping* pomaže na mnogo načina. Najčešće je riječ o analizi kompetitivnosti

cijena i motrenju konkurencije, te istraživanju tržišnih scenarija (trendova) prije plasiranja usluge ili proizvoda na tržište. Osim toga,

dodatni agregirani podaci s weba uvijek dobro dođu i u raznim područjima umjetne inteligencije i znanosti o podacima.

PODRUČJA PRIMJENE WEB-SCRAPINGA

Kompetitivnost cijena

- Motrenje konkurencije
- Ekstrakcija cijena radi ostvarivanja prednosti na tržištu

Istraživanje tržišnih scenarija (trendova)

- *Brand monitoring*
- Planiranje lansiranja produkta ili usluge na tržište
- *Trademark infringement detection*

Analiza sentimenta

- *Online reputation management*
- Praćenje reakcija potrošača ekstrakcijom ocjena, recenzija i povratnih informacija na forumima i društvenim mrežama

Ljudski resursi

- Selekcija i regrutiranje kandidata
- Ekstrakcija oglasa za poslove

Agregacija sadržaja

- Ekstrakcija i daljnja obrada u organizirane baze znanja

PRISTOJNOST CRAWLINGA

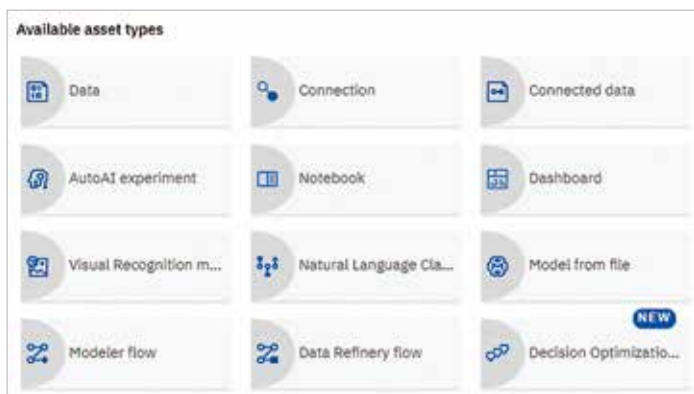
Kada radimo *crawling* po webu, poželjno je poznavanje četiri osnovne politike rada: **Politika selekcije:** koje stranice preuzeti? **Politika ponovnog posjeta:** kada provjeravati promjene na stranicama? **Politika pristojnosti:** kako izbjeći preopterećenje web-stranica? **Politika paralelizacije:** kako koordinirati distribuirane *web-crawlere*?

Očigledno najbitnija politika je politika pristojnosti, kod koje je pitanje kako ne biti prevelik teret poslužiteljima stranica koje pretražujemo. Postoji nekoliko mehanizama koji pomažu pri rješavanju tog problema. Prvi je User Agent (korisnički agent), string za identifikaciju poslužitelju (pristojni *crawleri* ostavljaju *mail* adresu ili URL adresu za kontakt u slučaju problema). Najbitnija je ipak datoteka *robots.txt* na određenom poslužitelju, koja *crawlerima* daje pravila koje stranice smiju, odnosno ne smiju pretraživati, te koliko bi trebali čekati između svaka dva posjeta stranicama na poslužitelju. Neke *robots.txt* datoteke imaju i poveznicu na dodatnu, tzv. *sitemap* datoteku, koja sadrži listu svih dopuštenih URL adresa, a često i njihovu prioritetnost za indeksiranje (koliko je sadržaj na određenoj stranici relevantan).

PREDZNAJNA I ALATI

Prije početka rada na *web-scraperima* poželjno je imati dobro pozadinsko znanje o HTTP protokolu (tipovi zahtjeva i status kôdovi) te tipičnoj strukturi HTML dokumenata, što su *tagovi* (oznake), koji *tagovi* imaju koje attribute, što su *id* i *class tagova*, i slično. Jednako bitno je i poznavanje raznih selektora koji mogu poslužiti za navigaciju kroz takvu strukturu, poput XPath i CSS selektora. Osim navedenih selektora, u praksi se koriste i regularni izrazi, sekvence znakova koje označavaju veću skupinu drugih sekvenci znakova, korisne za pronalaženje specifičnog sa-

podatke



Struktura HTML-a, prezentacijskog jezika iza web-stranica

držaja unutar strukture HTML dokumenta.

Najčešći izbor za programiranje vlastitog scraper-a je programski jezik Python, zbog njegove jednostavnosti obrade tekstualnih podataka. Neki popularniji Pythonovi alati su: **Requests**: HTTP zahtjevi za web-stranicama (odnosno HTML dokumentima); **BeautifulSoup i lxml: parseri** HTML datoteka, popularni zbog jednostavnosti i brzine razvoja skripti; **Scrapy**: razne naprednije, već implementirane funkcionalnosti, uključujući i vrlo koristan Scrapy Shell za učenje; te **Selenium WebDriver**: simulacija ljudskog ponašanja u pretraživanju dinamičnih stranica, primjerice, klikova i klizanja mišem te unosa teksta u polje.

OBRADA PRIRODNOG JEZIKA I IBM WATSON

Obrada prirodnog jezika (engl. Natural Language Processing - NLP) jedno je od područja kojemu *web-scraping* najviše pridonosi. NLP je zajedničko potpodručje jezikoslovlja i umjetne inteligencije, kojemu je zadatak kompjutorima dati vještinu čitanja i razumijevanja ljudskih jezika. Koncepti NLP-a koriste se u rješavanju raznih problema razumijevanja ljudskih jezika i obrade tekstualnih podataka.

Ovdje kao **vodeću svjetsku platformu za rješavanje NLP**



Razni tipovi *data asseta* i servisi unutar alata IBM Watson Studio

problema i integraciju NLP-a u vlastite poslovne procese valja istaknuti **IBM Watson**. Watson, kao skup alata, popularan je zbog jednostavnosti njihove primjene u vlastitom poslovnom okruženju. Natural Language Understanding, Tone Analyzer, Assistant i Studio, samo su neki od Watsonovih alata za razne primjene umjetne inteligencije na tekstualnim podacima, a dodatna prednost platforme dolazi iz mogućnosti njihova kombiniranja.

IBM Watson Studio jedan

je od Watsonovih alata koji se koristi za iskorištavanje benefita umjetne inteligencije, točnije, strojnog učenja na vlastitom skupu podataka. Ipak, veća količina i bolje informacije od velike su važnosti za procese strojnog učenja. Stoga valja posegnuti za dodatnim podacima iz web-prostora. Watson Studio (a ni Watson platforma) nema servis namijenjen *web-scrapingu* (što je i logično, s obzirom na raznolikost procesa *web-scrapinga*), ali ima nešto mnogo fleksibilnije – podršku za Jupyter Notebook,

interaktivnu *open-source* web-aplikaciju za kreiranje i dijeljenje dokumenata koji mogu sadržavati programski kôd, tekst i razne vizualizacije. Podržava preko 40 programskih jezika, uključujući Python, najpopularniju opciju za *web-scraping*. Tako dobivamo dodirnu točku *web-scrapinga* s platformom Watson, i pomoću jednostavnih Python skripti možemo obogatiti kolekciju *data asseta* u alatu IBM Watson Studio, kako bismo ih iskoristili za razne procese strojnog učenja i dobili vrijedne uvide.

"KUCHARICA" ZA IZRADU VLASTITOG WEB-SCRAPERA

1. Odaberi sadržaj na web-stranici
 - Radi li se o jednoj ili više stranica?
2. Pronađi odgovarajuću robots.txt datoteku i provjeriti pravila
 - Potpuno/parcijalno dopuštenje/zabrana
3. Ostvariti strategiju *web-crawlinga*
 - Praćenje poveznica ili iteriranje preko *page* parametra
 - Provjeriti postoji li datoteka sitemap
4. Inspect/Inspect Element
 - Je li stranica dinamična?
 - Odabir odgovarajućeg alata
5. Izgradnja *scraper-a*
 - Navigacija kroz HTML strukturu