Iskanje in ekstrakcija podatkov s spleta - 1. naloga

Anei Makovec, 63150186

April 2020

1 Uvod

Za 1. nalogo sem ustvaril spletnega pajka, ki išče in prenaša podatke s spletnih strani, katerih domene vključujejo *gov.si*. Kot programski jezik sem si izbral Python, uporabil pa sem več različnih knjižnic (za kanonikalizacijo povezav url, za obdelavo robots.txt datoteke, itd.) vendar nobena ne implementira funkcionalnosti spletnih pajkov. Kot glavno knjižnico za prenos podatkov sem uporabil Selenium Wire.

2 Implementacija

Pajek sem oblikoval v obliki večnitnega programa, kjer število niti vnese uporabnik po zagonu programa. Program najprej inicializira podatkovno strukturo za hranjenje povezav url, ki še niso bile pregledane (frontier), v obliki FIFO vrste ter vanjo vnese začetne povezave. Hkrati tudi vnese zapise z oznako FRONTIER o začetnih povezavah v tabelo page v podatkovni bazi. Inicializira se tudi slovar, ki za bo za vsak IP naslov beležil čas dostopa do le-tega, ter slovar, v katerem se bo za vsako domeno skranil IP naslov strežnika. Nato ustvari podano število niti ter vsaki dodeli svojo instanco headless brskalnika iz knjižnice Selenium ter svojo instanco povezave do podatkovne baze za vnos podatkov.

Vnosi v podatkovno bazo se izvajajo ob ekstrakciji povezav z določene spletne strani. Takrat se v tabeli page ustvari vnos z oznako FRONTIER, ki se bo posodobil ob presikavi te povezave.

Vsaka nit se izvaja dokler ni frontier prazen. Najprej vzame novo povezavo iz frontierja, nato preveri, če je že pretekla časovna omejitev dostopa do strežnikov tako, da pogleda če obstaja shranjen IP naslov za trenutno domeno ter nato preveri, če je preteklo dovolj časa (5 sekund), da lahko dostopa do strežnika. Če IP ni shranjen, se izvede DNS poizvedba, ki vrne IP naslov strežnika na kateri se nahaja trenutna domena, ki se nato shrani. Če od zadnjega dostopa do strežnika ni preteklo dovolj časa, se povezava vrne nazaj v frontier, iz nje pa se vzame nova. V nasprotnem primeru pa se nit loti s preiskavo povezave. Najprej preveri, če v podatkovni bazi že obstaja zapis o trenutni domeni. Če ja, se iz podatkovne baze pridobi vsebina datoteke *robots.txt*. Če pa ne, se s pomočjo knjižnice requests preneseta vsebini datotek *robots.txt* ter *sitemap.xml*, v tabeli *site* v podatkovni bazi pa se ustvari nov zapis o domeni. Po obeh zahtevah, se izvede tudi kratek premor (3 sekunde). Sedaj se preveri, če je trenutna povezava dovoljena v datoteki *robots.txt* in če ni, se iz podatkovne baze izbriše vnos v tabelah *page* in *link*, ki zadevajo povezavo. Sledi prenos vsebine na povezavi, nato pa se pregleda vse zahteve, ki jih je Selenium naredil, in se najde tisto, ki zadeva trenutno povezavo. Iz odgovora na to zahtevo se pridobi http statusna koda ter vrsta vsebine (vnos Content-Type v glavi). Iz tabele *page* v podatkovni bazi se pridobi vnos, ki ustreza povezavi. Če je povezava neodzivna, se bo vnos posodobil z vradnostjo TIMEOUT, če ni bilo nobenega odgovora, se bo posodobil z vrednostjo NO RESPONSE, v drugih primerih, pa se bo glede na prejšnja dva parameta odloči, kako se bo posodobil vnos v podatkovni bazi. Če je vsebina binarna, se vnos označi z oznako BINARY, v tabeli *page data* pa se ustvari ustrezen vnos glede na tip vsebine. Poleg tipov datotek, ki

so bili definirani po navodilih, sem dodal še oznake TXT (datoteke .txt), CSS (.css), CSV (.csv), ZIP (.zip) ter UNKNOWN (druge končnice). V primeru, da je tip vsebine slika, se ravno tako vnos označi z oznako BINARY, poleg tega pa se ustvari še vnos v tabeli *image*. Nato nit vzame novo povezavo za preiskavo. Če pa je tip vsebine formata html, se najprej preveri ali je stran duplikat, tako da se preveri ali obstaja že shranjena stran v tabeli *page* z enako vsebino, nato pa se še preveri, če vsebuje element `link` z atributom `rel` nastavljenim na vrednst čanonical". To pomeni, da je trenutna stran duplikat strani na povezavi, ki jo vsebuje ta element v atributu `href`. Če je duplikat, se vnos v tabeli *page* posodobi z oznako DUPLICATE, če pa ni, se posodobi z oznako HTML. Sledi ekstrakcija povezav iz html dokumenta. Pregledajo se vsi elementi, ki imajo v atributu `href` ustrezno povezavo, ter tisti, ki imajo atribut `onclick` ter v njegovi vrednosti vsebujejo znotrajvrstično JavaScript kodo z bodisi vrednostjo `location.href` ali pa vrednostjo `document.location`, ki kaže na ustrezno povezavo. Ekstrairajo se tudi povezave do slik, ki se nahajajo v atributu `src` elementov `img`. Če je dobljena povezava ustrezno formatirana, v njeni bazi vsebuje *gov.si* ter jo datoteka *robots.txt* dovoljuje, se kanonikalizira, nato pa je primerna za nadaljnje preiskovanje. Preveri se samo še, če v tabeli *page* že obstaja vnos s tako povezavo, nakar se ustvari nov vnos z oznako DUPLICATE. V nasprotnem primeru, pa se ustvari nov vnos z oznako FRONTIER, povezavo pa se doda v frontier. Tako lahko dobimo dva različna vnosa z vrednostjo DUPLICATE: s shranjeno povezavo ali brez. Prvi nam pove, da je shranjena povezava duplikat po vsebini ali pa ima element, ki kaže na originalno stran, drugi pa je zgolj duplikat z enako povezavo.

3 Analiza

Pajek je v skupnem našel 272 različnih domen. Na njih pa skoraj 90000 unikatnih spletnih strani. Podrobni podatki o številu posameznih tipov strani so prikazani v tabeli 1. Tabela 2 pa prikazuje podatke o številu najdenih datotek.

	Vrsta strani							
Domena	html	binary	image	duplicate	no response	timeout	undefined	Skupaj
gov.si	2530	1444	885	7858	118	0	0	12835
evem.gov.si	3013	304	112	408	35	0	0	3872
e-uprava.gov.si	12312	406	13	7	72	1	0	12811
e-prostor.gov.si	114	232	63	269	0	0	0	678
vse	68328	10076	5589	2697749	3710	1	137	**2785590**

Slika 1: Podatki o posameznih tipih spletnih straneh na splošno ter glede na začetne domene.

	Vrsta datoteke									
Domena	pdf	doc	docx	ppt	pptx	txt	css	csv	zip	neznana
gov.si	968	199	94	3	17	0	3	1	7	152
evem.gov.si	121	65	0	0	0	58	44	0	5	11
e-uprava.gov.si	173	80	97	0	0	0	33	0	0	23
e-prostor.gov.si	135	20	3	1	0	3	4	0	50	16
vse	4796	900	397	49	54	367	1229	39	373	1873

Slika 2: Podatki o vrstah datotek na splošno ter glede na začetne domene.