Fakulteta za računalništvo in informatiko, Univerza v Ljubljani

Beti Piščanec, Sebastjan Mevlja in Lan Zukanović

Spletni pajek

Seminarska naloga pri predmetu Iskanje in ekstrakcija podatkov s spleta

MENTOR: doc. dr. Slavko Žitnik

1. Uvod

Seminarska naloga opisuje implementacijo spletnega pajka za prvo programersko nalogo pri predmetu Iskanje in ekstrakcija podatkov s spleta. Za pridobivanje vsebine spletnih strani smo uporabili Python knjižnici Playwright in Requests, za obdelavo HTML in XML dokumentov knjižnico Beautiful soup. Komuniciranje s podatkovno bazo nam je olajšala uporaba knjižnice SQLAlchemy.

2. Statistika

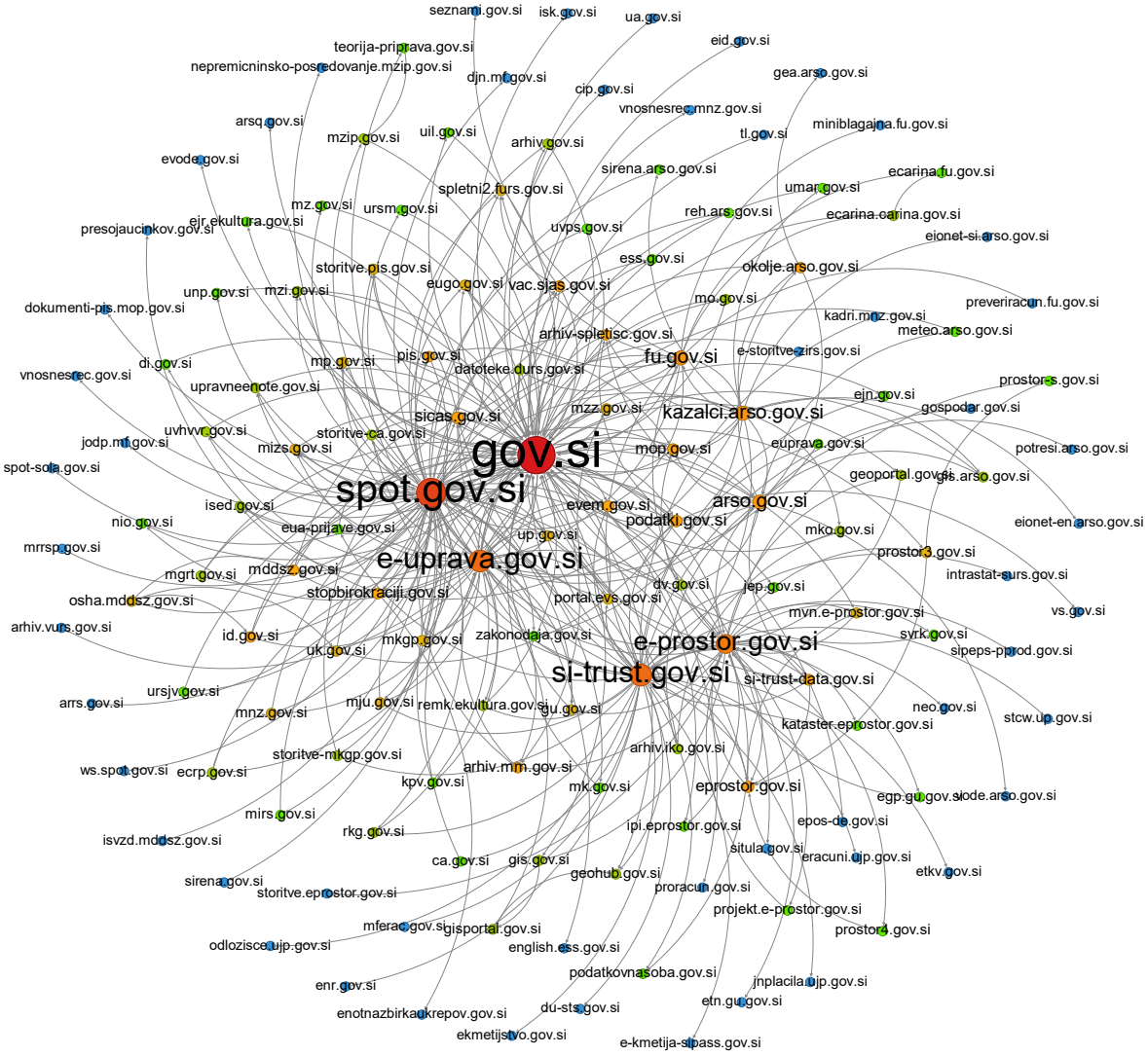
Tabela 1 prikazuje statistiko začetnih in vseh obiskanih strani glede na različne kategorije.

Kategorija	Podkategorija	Začetne strani	Vse strani
Vsa spletna mesta (domene)		4	147
Obiskane HTML strani		4	53385
Duplikati strani		0	403
Nedelujoče povezave		0	467
Še neobiskane povezave		89113	77504
Preusmeritve		1	2778
Binarni dokumenti	**Vsi**	8	37065
	DOC	0	5703
	DOCX	1	8508
	PDF	3	17714
	PPT	0	7
	PPTX	0	116
	ZIP	3	3274
	RAR	0	1
	XLSX	0	1688
	Ostali	0	54
Slike	**Vse**	30	125008
	SVG	21	89758
	JPG	8	22550
	JPEG	0	546
	PNG	1	11730
	GIF	0	424
Povprečno število slik/stran		7,5	2,34
Povprečno število binarnih dokumentov/stran		2	0,69

***Tabela 1**: Statistika obiskanih strani.*

3. Vizualizacija povezav med domenami

Slika 1 prikazuje število povezav med obiskanimi stranmi po domenah. Število povezav, ki kaže na neko domeno, je predstavljeno z velikostjo vozlišča in imena domene ter s samo barvo vozlišča. Na modra vozlišča kaže najmanj povezav, na rdeča pa največ.



***Slika 1**: Vizualizacija obiskanih domen in povezav med njimi.*

4. Težave

Pajka smo od začetka razvijali na modularen način. Programsko kodo smo razdelili na smiselne razrede in metode, saj smo predvidevali, da nam bo to poenostavilo podporo za več niti. Po njeni implementaciji smo ugotovili, da uporabljeno ogrodje za avtomatizirano testiranje Playwright ne nudi ustrezne podpore za večnitno delovanje (angl. Thread safety), kar je vplivalo na učinkovitost delovanja pajka, saj smo morali za vsako nit ustvariti novi Playwright in posledično tudi Chromium instanci. Spoznali smo, da večnitni programi pogosto niso asinhroni, zaradi česa smo imeli tudi nekaj težav pri uporabi knjižnice SQLAlchemy in sodostopu do podatkovne baze, kar smo uspešno rešili.

Po zaključenem zbiranju spletnih strani smo ugotovili, da smo v bazo shranjevali čas shranjevanja strani v podatkovno bazo in ne čas dostopa da strani. Posledično iz podatkov ni mogoče preveriti upoštevanja časovnega zamika spletnega pajka (angl. Crawl delay). Do razlike v času pride, ker več niti želi hkrati zapisati podatke v tabelo, ki jo je že zaklenila druga nit. Spletnega pajka smo ponovno zagnali in bomo na repozitorij oddali tudi popravljen izvoz baze.

Nekaj težav smo imeli tudi s pridobivanjem podatkov o binarnih dokumentih, katerih podatkovnega tipa nismo prepoznali iz naslova, saj jih nismo mogli odpreti kot navadne HTML strani.