Fakulteta za računalništvo in informatiko, Univerza v Ljubljani

Jakob Dekleva

Poročilo za prvo domačo nalogo (Web crawler)

Domača naloga pri predmetu Iskanje in ekstrakcija podatkov s spleta

MENTORJA: prof. dr. Marko Bajec, doc. dr. Slavko Žitnik

1. Uvod

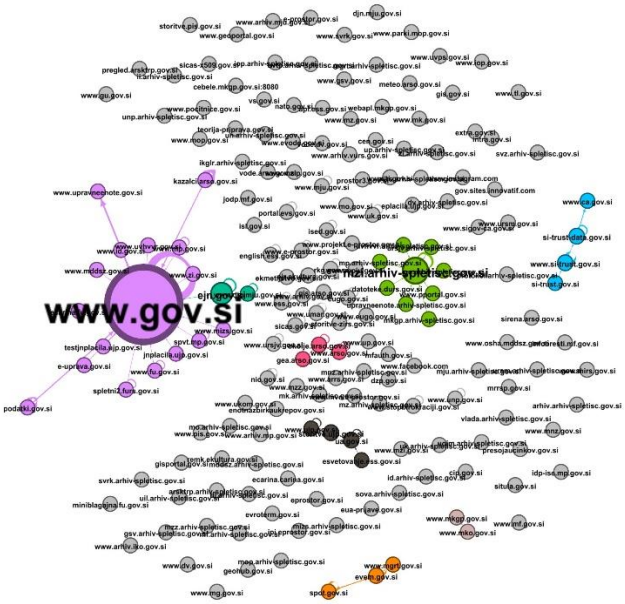
V tem poročilu bomo opisali izdelavo in implementacijo spletnega pajka, namenjenega za strganje spletnih strani z domeno gov.si. Poročilo je sestavljeno iz treh delov. Prvi del obravnava strukturo pajka, drugi del predstavlja statistiko in vizualizacijo povezav, medtem ko zaključni del opisuje težave na katere smo naleteli med izdelavo pajka, trenutne pomanjkljivosti ter možne izboljšave. Tretji del se zaključi z ovrednotenjem celotnega pajka, ki podaja splošno sliko njegove učinkovitosti in uporabnosti.

2. Struktura pajka

Pajka smo želeli implementirati kot sistem klient-strežnik, kjer bi klient opravljal delo strganja in parsiranja, strežnik pa bi upravljal frontir in shranjeval vračane podatke. Na žalost nam ni uspelo izdelati strežniškega dela pajka, zato smo se odločili vključiti strežniške lastnosti v klientov del pajka. Pajek je zasnovan tako, da deluje kot Flask strežnik. Uporabnik pošlje semenske strani, ki jih želi obdelati, na končno točko /scrape na Flash strežniku. Pajek nato sortira ta vnos na URL-je in ustvari več niti, da vsak URL obdeluje svojo nit. Po obdelavi semenskih strani se novi URL-ji shranijo v frontir, ki se generira ob pričetku skripte kot SQLite podatkovna baza. Frontir je razdeljen na dve tabeli: frontir_url in old_frontir. V tabelo frontir_url se shranjujejo novi URL-ji, ki jih pajek še ni obdelal, v tabeli old_frontir pa so shranjeni URL-ji,katere je pajek že obdelal. Po obdelanih semenskih straneh se pajek loti obdelave strani, ki so shranjene v tabeli frontir_url. Med parsiranjem se v bazo SQLite (poimenovano Frontir_db) shranijo tudi razpršilne funkcije (hashi) v tabelo hash_url. Baza beleži tudi možne zakasnitve, ki so lahko implementirane za specifične domene v datoteki robots.txt. Vse zahtevane elemente strani shranjujemo v podatkovno bazo PostgreSQL.

3. Statistika

4. Vizualizacija



Slika 1 Vizualizacija povezav

5. Težave

Med tekočim razvojem pajka sem naletel na veliko težav pri povezavi podatkovne baze s pajkom, predvsem zaradi slabega poznavanja podatkovnih baz. Zato sem se odločil za izdelavo preprostejše baze (SQLite), ki bi hranila vse potrebne podatke za delovanje pajka. Dodatna težava s katero sem se soočil je bila pomanjkanje zadostne strojne opreme za učinkovito delovanje pajka, saj je bila zmogljivost računalnika omejena (premalo delovnega spomina), kar je vodilo v pomankanje časa zaradi dolgotrajnega testiranja.

6. Zaključek

Čeprav spletni pajek ni povsem tak, kot smo si ga zamislili, še vedno pa deluje precej dobro in obdeluje strani hitro. Če bi imel več časa, bi lahko pajka bolj optimiziral (skakanje med domenami in izogibanje čakanja) ter mu dodal nekaj preprostih ukazov, ki bi jih lahko poslal na Flask aplikacijo (start, stop, prikaži število obdelanih strani, nastavi število delavcev).