Implementacija Regex Golf igre pomoću Genetskog programiranja

Projekat u okviru kursa Računarska inteligencija

Anđela Ilić
mi17105@alas.matf.bg.ac.rs
Mina Milošević
mi17081@alas.matf.bg.ac.rs

Matematički fakultet
Univerzitet u Beogradu

Februar 2021

Uvod

Opis problema

Cilj *Regex Golf* igre je pronaći najkraći regularni izraz kojim se mogu zapisati sve reči iz skupa M, ali kojim se ne može zapisati nijedna reč iz skupa U.

Za date skupove M i U ne možemo sa sigurnošću da tvrdimo da postoji rešenje koje zadovoljava prethodne uslove. Takođe, ako dobijemo regularni izraz koji zadovoljava prva 2 uslova, ne možemo za svaki primer znati da li postoji i bolje rešenje tj. kraći regularni izraz.

Implementacija

Terminal i Function skupovi

Svaka jedinka će biti predstavljena kao drvo. U listovima nalaze elementi koje ćemo jednim imenom zvati *Terminali* (terminal set), a u unutrašnjim čvorovima su elementi koje nazivamo *Funkcije* (function set).

Terminal i Function skupovi

Skup funkcija sadrži operatore koji se mogu javiti u regularnim izrazima. Primeri takvih operatora su: $.*$, $.+$, $.?$, $.\{.,.\}+$, $(.)$, $[.]$, $[\ \hat{}.]$, $..$, $.|..$ Tačka . je mesto na kome se nalaze deca u drvetu.
Skup terminala čine elementi koji zavise i koji ne zavise od ulaznih skupova M i U. Elementi koji su nezavisni - opsezi malih i velikih slova, brojeva u regularnim izrazima, karakteri $\hat{}$ i $\$$, wildcard karakter '$\%$'. Elementi skupa terminala koji su zavisni - skup karaktera iz M, parcijalni opsezi karaktera iz M i n-grami.

Genetsko programiranje - jedinke

Svaka jedinka se predstavlja preko stabla. U korenu stabla se nalazi karakter '.' i koren uvek ima dva deteta. Elementi stabla se biraju nasumično iz skupova *Function* i *Terminal*.
Od kreiranog drveta se dobija niska koja predstavlja validan regularni izraz.
Za svaku jedinku računamo i *fitnes* funkciju po formuli:

$$f(x) = w_i * (n_m - n_u) - length(r)$$

gde je w_i statistički određena konstanta, n_m i n_u brojevi reči iz skupova M i U, redom, koje su opisane dobijenim regularnim izrazom r. Ovako definisanu funkciju maksimizujemo.

Genetsko programiranje - selekcija

Za *selekciju* koristimo turnirsku selekciju veličine 7. Jedinke za selekciju biramo nasumično i uzimamo najbolju jedinku tj. onu koja ima najveći fitnes među odabranim.

Genetsko programiranje - ukrštanje

Koristimo *jednopoziciono ukrštanje*.
Najpre obilazimo stabla roditelja BFS-om i numerišemo čvorove.
Biramo jedan broj koji će predstavljati indeks čvora. Tražimo podstabla u roditeljima koja treba da razmene mesta.
Na kraju treba proveriti da li su regularni izrazi ovako dobijene dece validni. U slučaju da neko dete nije validno, vraćamo odgovarajućeg roditelja umesto njega.

Genetsko programiranje - mutacija

Mutaciju radimo sa verovatnoćom 0.1. Biramo indeks čvora koji želimo da promenimo. Razlikujemo dva slučaja - kada je čvor iz skupa terminala i kada je iz skupa funkcija.
U prvom slučaju, samo izaberemo nasumično novi terminal i ažuriramo vrednost čvora.
Ako je čvor bio iz skupa funkcija, biramo novu vrednost iz istog skupa. Treba obratiti pažnju da li se kardinalnost (broj dece) čvora promenila.
Na kraju opet treba proveriti da li je dobijena jedinka validna. Ako nije, vraćamo jedinku za koju smo i pokrenuli mutaciju.

Genetsko programiranje - parametri

Parametri su postavljeni na osnovu podataka iz [1]:

- populaciju čini 500 jedinki
- pravimo 1000 generacija
- turnirska selekcija je dimenzije 7
- verovatnoća mutacije je 0.1
- elitizam iznosi 20%
- algoritam pozivamo za 30ak populacija

Izlaz programa

Iz svake populacije čuvamo sve jedinke kod kojih važi:

$$n_m - n_u = |M|$$

Dakle, želimo da sačuvamo sve jedinke kod kojih su ispunjena prva 2 uslova ovog problema. Ako u trenutnoj populaciji ne postoji takva jedinka, čuvamo samo najbolju jedinku na osnovu celokupnog fitnesa.
Na kraju rada algoritma biramo najbolju jedinku, od svih sačuvanih, po njenom ukupnom fitnesu i nju proglasavamo za rešenjem datog problema.

Rezultati

PRIMER	NAŠE REŠENJE	SCORE	NAJBOLJE REŠENJE	SCORE
Plain strings	foo	207	foo	207
Anchors	k$	208	k$	208
It never ends*	u$	28	u\b	27
Ranges	ff|.b|d[a-f]	178	^[a-f]*$	202
Abba	st|.+u|z	142	^(?!(.)+ \1)|ef	196
A man, a plan	^r|x|gg|oo	90	^(.)[^p].*$	177
Presidents**	Bu|am|Ta|Har|N+i|vel	120		390
Movies***	m |B?R|[AB?]	48	m | [tn]|b	40
Regions****	br|-|os|L|P|t?il	104		200

Reference

- Bartoli, A., Medvet, E., Lorenzo, A. D. & Tarlao, F. *Playing Regex Golf with Genetic Programming.* (2014).
- *Regex Golf Examples.* https://alf.nu/RegexGolf.
- *Regex Golf Solutions.* https://gist.github.com/Davidebyzero/9221685.
- *Regions of Italy vs. States of USA.* https://codegolf.stackexchange.com/questions/17855/regex-golf-regions-of-italy-vs-states-of-usa.
- *Star Wars and Star Trek titles.* http://zegnat.github.io/xkcd1313/.
- *USA Election winners and losers.* https://pastebin.com/EvycCQTB.

KRAJ