Sistem za preporuku i savete pri heklanju

Autor:

Katarina Mirković SV6/2020

Motivacija:

U poslednjih nekoliko godina heklanje je ponovo postalo popularno kao hobi, ne samo među starijima, nego i mlađima. Nekome ono predstavlja čistu razonodu I opuštanje, dok se drugi odluče heklanjem da zarađuju džeparac, prodavajući svoje kreacije.

Mnogi bi iz istih razloga hteli da se oprobaju o ovu veštinu, ali nemaju konkretan plan, niti ideju šta bi hteli ili mogli da naprave. Čak se može desiti i da znaju šta bi otprilike hteli, ali nisu upućeni koliko i kojeg materijala im treba, niti da li njihova želja odgovara njihovim trenutnim sposobnostima.

Pregled problema:

Tu u pomoć dolazi ova aplikacija. Namenjena je kako početnicima, tako i naprednijim heklašima, da dobijaju preporuke šta da heklaju, šta im je sve potrebno, kao i sistem uz koji mogu da prate svoj napredak.

Često početnici dobiju ideju sa interneta šta bi hteli da heklaju, ali ne znaju šta im je sve potrebno. U tom slučaju bi im sistem na osnovu željenog šablona (igračka, odeća) preporučio koja im vunica i koliko treba, te koji pribor za to ide.

Ako korisniku treba ideja šta da hekla, a ima već nekog starog materijala pri ruci, može da unese koliko kojeg materijala ima, pa će mu sistem na osnovu njegovog iskustva i preferiranog heklanja preporučiti šta bi mogli da heklaju.

Baza znanja:

Kao osnova baza znanja čuvali bi se nekoliko patterna za heklanje, koji svaki zahteva određeni nivo znanja (beginner, intermediate, advanced), određenu vrstu i količinu vunice, kao i odgovarajuću heklaricu.

Za forward-chaining bi se koristila pravila koja kombinuju sve potrebne materijale koje je korisnik naveo da ima pri ruci, i pronalazi se pattern koji odgovara tim materijalima i nivou znanja. U slučaju da se korisniku taj pattern svideo, može da ga favoritizuje, ili čak i celu vrstu patterna (igračke, majice, odeća generalno) da lajkuje. Za ova pravila se lako može iskoristiti **templates** za svaki ponuđeni pattern. Primer template:

when   
**Difficulty = lvl** and **Vunica = type** and **Kolicina = x** and **heklarica in (0.5, 1.0, 1.5)**   
then  
**pattern**

Za backward-chaining bi se iskoristili prethodno sačuvani podaci o favoritnim patternima, tako da kada se sledeći put preporuči pattern, proverava se da li je favorit (\*\*\*Ili da radim obrnuto, ako je nesto dislikeovano, da se ono ne preporučuje?\*\*\*). U ovom slučaju bi postojalo stablo sa korenom heklanje, koje je svakako favorit, heklanje može imati više podvrsta heklanja, koje mogu da se lajkuju, te svaka podvrsta može imati nivo kompleksnosti (onaj koji smo izabrali u pocetku, on je lajkovan), takođe svaki pattern određene kompleksnosti može posebno da se lajkuje.

Dakle stablo bi bilo:  
 heklanje, liked

/ \

Igracke, liked odeca, liked

/ | \ | \ \

Zivotinje, liked Vocke, liked Igracke Kapa, Dzemper, carape,

za ljubimce, liked liked liked….. itd

liked

\*CEP još u izradi\*

Za implementaciju CEP-a je bila ideja da se prilikom heklanja prati napredak korisnika. Ili da uživo korisnik ima help button gde može da označi šta je pogrešio, pa da ako je puno puta istu grešku učinio u poslednjih sat vremena, da mu se dadne preporuka šta bi mogao promeniti da to ispravi (npr. uzmi veću heklaricu, proveri osvetljenje u sobi i sl.)

Takođe bi bila ideja da se prati koliko je patterna jedne vrste i jedne težine korisnik završio u poslednje vreme, i da nakon 3 patterna iste kompleksnosti, mu se preporuči da proba pattern iste vrste ali veće težine.