Pronalaženje skrivenog znanja - Projektni zadatak za junsko-julski rok 2023. godinu

Projektni zadatak se sastoji iz šest celina na kojima se može ostvariti ukupno 60 poena. Zadaci se odnose na prikupljanje podataka, njihovu analizu, vizuelizaciju i implementaciju algoritama mašinskog učenja. Obavezno je uraditi bar jedan zadatak iz skupa {4, 5, i 6} da biste ostvarili prolazan broj poena.

Zadatak 1: Prikupljanje podataka

Realizovati veb indekser (eng. *web crawler/web spider*) sa veb parserom (eng. *web scraper*), koji prikuplja podatke o nekretninama sa jednog ili više od sledećih sajtova:

- https://www.nekretnine.rs/
- https://www.4zida.rs/
- https://www.halooglasi.com/nekretnine
- neki sajt za ponudu nekretnina u Srbiji koji nije u ovoj listi, a ima dovoljan broj zapisa.

Formirati sopstvenu relacionu bazu podataka sa svim relevantnim informacijama o nekretninama koje se izdaju i nekretninama koje se prodaju u Srbiji. Bazu realizovati u tehnologiji *MySQL* ili *PostgreSQL*. Baza treba da ima najmanje 20 hiljada aktuelnih zapisa o nekretninama.

Šta je veb indekser?

Cilj veb indeksera je da se poveže na određenu veb stranu i da preuzme njen sadržaj. Parsiranjem date strane možemo naći linkove, koji vode na neke druge strane, na koje veb-indekser ponovo može da uđe i da ponovi celu proceduru. Pored otkrivanja linkova, parser može da prepozna i druge sadržaje koje veb strana ima. U vašu bazu treba da prikupite informacije o svim nekretninama – tip nekretnine (stan ili kuća), tip ponude (prodaja ili iznajmljivanje), lokacija - grad i deo grada gde se lokacija nalazi, kvadratura nekretnine, godina izgradnje (ukoliko postoji), površina zemljišta (samo za kuće), spratnost (ukupna i sprat na kojoj se nalazi, samo za stanove), uknjiženost (da/ne), tip grejanja, ukupan broj soba, ukupan broj kupatila (toaleta), podaci o parkingu (da/ne) i ostale dodatne informacije (da li ima lift u zgradi, da li ima terasu/lođu/balkon). Podaci koji nisu dostupni u oglasu, u bazi treba da ostanu praznog polja.

Implementaciju veb-indeksera možete raditi u programskim jezicima: C, C++, C#, Java, Python, NodeJS ili PHP. Dozvoljeno je i korišćenje i prilagođavanje neke od postojećih implementacija otvorenog koda: *crawler4j*, *Heritrix*, *Nutch*, *Scrapy* za Python, *PHP-Crawler* za PHP, itd.

Zbog ograničenog broja zahteva na serverima sa iste IP adrese, koristiti rotirajuće proxy-je ili neku drugu tehniku, kako ne biste kršili uslove korišćenja usluga izvornih veb sajtova. Prikupljanje ovih podataka koji nisu orijentisani ka ličnim podacima i koji jesu javno dostupni je dozvoljeno, ali uz molbu da broj zahteva ka serveru prilagodite, i da između zahteva bude vremenske razlike, kako ne biste domaćinu sajta izvršili *Denial-of-service attack (DoS)*.

Šta je veb parser?

Uloga veb parsera je da otkrije potreban sadržaj sa primljenih veb strana. Pri tome potrebno je odrediti značenje sadržaja kako bi se baza podataka popunjavala tačnim podacima. Najčešće tehnike koje se koriste pri implementaciji veb parsera su: HTML parser, DOM parser, tehnika regularnih izraza koji

izdvajaju potreban sadržaj i tehnika prepoznavanja semantičkih anotacija. Za potrebe veb parsiranja takođe možete koristiti neku od postojećih implementacija (npr. biblioteka *Jsoup* – parsira veb stranu kao stablo elemenata, *BeautifulSoap*, *Scrapy*, ili *PySpyder*, za Python, itd.).

Kao rezultat zadatka 1 treba da prikažete realizovanu relacionu bazu podataka popunjenu traženim podacima o nekretninama i da priložite implementacije koje su korišćene za dohvatanje podataka. Podaci treba da budu preuzeti u konačnom vremenskom intervalu.

Zadatak 2: Analiza podataka

Iz navedenih zapisa ubačenih u bazu podataka (iz zadatka 1), potrebno je preprocesirati podatke (odabiciti one koji nemaju veći broj potrebnih vrednosti polja), zabeležiti to u novoj bazi (sa brojem zapisa koji je preostao, ne manji od 15 hiljada), i uraditi sledeće:

a) izlistati koliki je broj nekretnina za prodaju, a koliki je broj koji se iznajmljuju;
b) izlistati koliko nekretnina se prodaje u svakom od gradova (izlistati sve gradove, obuhvatiti i kuće i stanove);
c) izlistati koliko je uknjiženih, a koliko neuknjiženih kuća, a koliko stanova;
d) prikazati rang listu prvih 30 najskupljih kuća koje se prodaju, i 30 najskupljih stanova koji se prodaju u Srbiji;
e) prikazati rang listu prvih 100 najvećih kuća i 100 najvećih stanova po površini (kvadraturi);
f) prikazati rang listu svih nekretnina izgrađenih u 2022. ili 2023. godini, i izlistati ih opadajuće prema ceni prodaje, odnosno ceni iznajmljivanja;
g) prikazati nekretnine (Top30) koje imaju:
    - najveći broj soba unutar nekretnine,
    - najveću kvadraturu (samo za stanove),
    - najveću površinu zemljišta (samo za kuće).

Kao rezultat zadatka 2 treba priložiti bazu podataka (revidiranu i prečišćenu, iz zadatka 2), realizovane upite i generisane rezultate.

Zadatak 3: Vizuelizacija podataka

Iz navedenih zapisa ubačenih u bazu podataka (iz zadatka 2), potrebno je vizuelizovati sledeće podatke:

a) 10 najzastupljenijih delova Beograda koji imaju najveći broj nekretnina u ponudi (i u sekciji za prodaju, i u sekciji za iznajmljivanje, zbirno).
b) Broj stanova za prodaju prema kvadraturi, u celoj Srbiji (do 35 kvadrata, 36-50, 51-65, 66-80, 81-95, 96-110, 111 kvadrata i više).
c) Broj izgrađenih nekretnina po dekadama (1951-1960, 1961-1970, 1971-1980, 1981-1990, 1991-2000, 2001-2010, 2011-2020)[1], a obuhvatiti i sekcije za prodaju i za iznajmljivanje.
d) Broj (i procentualni odnos) nekretnina koje se prodaju i nekretnina koje se iznajmljuju, za prvih 5 gradova sa najvećim brojem nekretnina (za svaki grad posebno prikazati grafikon BROJ_ZA_PRODAJU : BROJ_ZA_IZNAJMLJIVANJE).
e) Broj (i procentualni odnos) svih nekretnina za prodaju, koje po ceni pripadaju jednom od sledećih opsega:
    - manje od 49 999 €,

[1] Za sajtove koje nemaju godinu izgradnje, umesto ovoga izlistati koliko pripada nekoj klasi (novogradnja, starogradnja, itd.)

- između 50 000 i 99 999 €,
- između 100 000 i 149 999 €,
- između 150 000 € i 199 999 €,
- između 200 000 € i 499 999 €,
- 500 000 € ili više.

f) Broj nekretnina za prodaju koje imaju parking, u odnosu na ukupan broj nekretnina za prodaju (samo za Beograd).

Kao rezultat zadatka 3 treba priložiti bazu podataka (iz zadatka 2), realizovane upite i generisane rezultate u vidu grafikona (*charts*). Za grafikone možete koristiti bilo koji alat / biblioteku.

Zadatak 4: Implementacija regresije

Iz dobijenih rezultata izdvojiti samo stanove koji se prodaju u Beogradu (samo 5 centralnih opština – Vračar, Stari grad, Savski venac, Zvezdara, Novi Beograd, bez prigradskih naselja) sa kojima ćete obučavati model. Realizovati malu aplikaciju koja na osnovu zapisa iz Vaše filtrirane baze podataka primenjuje višestruku linearnu regresiju na nekoliko nezavisnih ulaznih promenljivih i pravi što bolji model zavisnosti između prediktora i ciljne (izlazne) promenljive. Podatke podeliti na skup za treniranje i skup za testiranje, a obučavanje realizovati korišćenjem gradijentnog spusta. Ulazni atributi (*features*) koje možete analizirati mogu biti: lokacija - numerički pokazatelj udaljenosti kvarta/ulice od centra grada (ne naziv lokacije/mikrolokacije!), kvadratura nekretnine, godina starosti nekretnine, broj soba, spratnost – da li je prizemlje ili poslednji sprat, ili nije, i slično).

Ciljna promenljiva treba da bude cena nekretnine (stana) za prodaju. Aplikacija treba da na osnovu ulaznih promeljivih koje korisnik (prodavac stana) treba da unese preko forme i realizovanog modela, prikaže prediktivnu vrednost nekretnine za prodaju.

U ovom zadatku nije dozvoljeno korišćenje gotovih funkcija iz neke biblioteke programskog jezika, osim u cilju provere ispravnosti sopstvenih rezultata. Sve funkcije treba da budu samostalno napisane.

Zadatak 5: Implementacija klasifikacije

U okviru iste aplikacije, primeniti još i algoritam K-najbližih suseda (kNN) na osnovu istih ili sličnih ulaznih promenljivih (atributa nekretnine) i na osnovu potpuno iste (filtrirane) baze podataka, kao u zadatku 4. Opseg izlazne vrednosti (cene nekretnine) podeliti na nekoliko klasa, kao što je na primer navedeno u zadatku 3.e). Faktor K odrediti automatski (kao najoptimalnije na osnovu broja nekretnina za prodaju u skupu Vaših podataka), ali dozvoliti i manuelnu promenu tog faktora, na ulazu, pre pokretanja samog algoritma. Realizovati i bar 2 različite funkcije rastojanja suseda.

Zadatak 6: Implementacija klasterizacije

U neophodnom broju iteracija, primeniti metod K-srednjih vrednosti (*k-Means*), na najmanje 3 ulazna atributa koje ćete posmatrati iz filtrirane baze podataka sa stanovima (iz Zadatka 4). Klasteri ne treba da se grupišu po atributu za lokaciju, gde se ponuđeni stanovi nalaze, već analizirati druge atribute.

Kao rezultat zadataka 4, 5 i 6 treba priložiti programski kod realizovane aplikacije ili aplikacija (implementacije realizovanih finalnih modela, procedure za obučavanje, sve realizovane, i eventualno pomoćne funkcije i klase, koje su korišćene). Takođe, priložiti izveštaje sa kratkim komentarom o realizovanim implementacijama, šta ste sve probali da biste došli do finalne implementacije i koji su sve dobijeni rezultati. U ova tri zadatka takođe je poželjno koristiti vizuelizaciju podataka i grafikone (samo potpuno urađeni zadaci donose najveći broj poena!).