

Korištenje velikih jezičnih modela u ocjenjivanju programskog koda studenata

Gašpar, Sara

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:168:560143>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-20**



Repository / Repozitorij:

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1261

**KORIŠTENJE VELIKIH JEZIČNIH MODELA U
OCJENJIVANJU PROGRAMSKOG KODA STUDENATA**

Sara Gašpar

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1261

**KORIŠTENJE VELIKIH JEZIČNIH MODELA U
OCJENJIVANJU PROGRAMSKOG KODA STUDENATA**

Sara Gašpar

Zagreb, lipanj 2024.

ZAVRŠNI ZADATAK br. 1261

Pristupnica: **Sara Gašpar (0036537498)**
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo
Modul: Računarstvo
Mentorica: prof. dr. sc. Ljiljana Brkić

Zadatak: **Korištenje velikih jezičnih modela u ocjenjivanju programskog koda studenata**

Opis zadatka:

Integracija velikih jezičnih modela (engl. Large Language Model - LLM) u proces ocjenjivanja programskog koda studenata može donijeti brojne prednosti, uključujući ubrzanje procesa ocjenjivanja, povećanje konzistentnosti ocjena te pružanje povratnih informacija studentima o njihovim rješenjima. Ovi modeli mogu automatski analizirati sintaksu i semantiku programskog koda, identificirati greške ili nedostatke u kodu te dodatno pružiti prijedloge za poboljšanje koda, što može biti korisno za studente koji uče programiranje. Međutim, njihova sposobnost preciznog ocjenjivanja programskog koda još nije potpuno utvrđena. Potrebno je provesti detaljna istraživanja kako bi se utvrdilo koliko dobro LLM-ovi mogu prepoznati različite vrste grešaka u kodu te koliko su pouzdani u davanju ocjena i povratnih informacija. U ovom završnom radu potrebno je provesti ocjenjivanje i prikupiti povratne informacije za reprezentativan broj studentskih rješenja u obliku programskog koda (rješenja zadatka s ispita iz uvodnog predmeta iz programiranja) koristeći LLM. Nakon toga, potrebno je analizirati kvalitetu i relevantnost ocjena i povratnih informacija koje pruža LLM. Temeljem proučavanja postojeće literature predložite i implementirajte način na koji način ćete procijeniti kvalitetu i relevantnost ocjena i povratnih informacija. Potom usporedite ocjene dobivene od LLM-a s ocjenama koje su dodijelili nastavnici te povratne informacije dobivene od LLM-a s povratnim informacijama koje su dali nastavnici. Predložite mjeru ili više njih kojom bi se ta sposobnost ocijenila.

Rok za predaju rada: 14. lipnja 2024.

Sadržaj

Uvod	1
1. Veliki jezični modeli.....	2
1.1. ChatGPT	2
1.2. Gemini	3
2. Metodologija	4
2.1. Eksperimentalna metodologija.....	4
2.2. Skup podataka.....	5
2.3. Korištene tehnologije	5
2.4. Oblikovanje upita	6
2.5. Način analize i vrednovanja rezultata.....	7
2.5.1. Bodovi.....	7
2.5.2. Komentari	8
3. Analiza rezultata.....	9
3.1. Analiza bodova	9
3.1.1. Ukupna raspodjela bodova na međuispitu	9
3.1.2. Prosječni bodovi i standardne devijacije	10
3.1.3. Detaljnija analiza bodova trećeg zadatka.....	12
3.1.4. Koeficijent korelacije	13
3.1.5. Boxplot graf raspodjele bodova svih ocjenjivača.....	15
3.2. Analiza komentara.....	18
3.2.1. Prvi zadatak	19
3.2.2. Drugi zadatak.....	21
3.2.3. Treći zadatak	23
3.2.4. Četvrti zadatak.....	24
4. Rasprava	25
4.1. Bodovanje programskog koda.....	25

4.1.1.	Mean absolute error	25
4.1.2.	Mean squared error	25
4.2.	Pružanje povratnih informacija	26
4.2.1.	Prvi pristup.....	26
4.2.2.	Drugi pristup	28
	Zaključak.....	30
	Literatura.....	31
	Sažetak	32
	Summary.....	33

Uvod

U današnje vrijeme, sve više procesa iz svakodnevnog života pokušavamo što je više moguće modernizirati i automatizirati. Jedan od alata koji nam u posljednjih nekoliko godina sve više pomaže u tome su upravo veliki jezični modeli (Large Language Models - LLM). Zahvaljujući tome što imaju puno različitih mogućnosti primjene, vrlo često ih koristimo i u obrazovanju. U okviru ovog rada proučena je primjena ovih modela u procesu ocjenjivanja programskog koda studenata. Koristeći napredne pristupe strojnog učenja, ovi modeli mogu analizirati sintaksu i semantiku koda, identificirati greške ili nedostatke te predložiti studentima poboljšanja. Prednosti koje bi ovaj način ocjenjivanja mogao donijeti su ubrzanje procesa ocjenjivanja i pružanje detaljnih i korisnih povratnih informacija studentima o njihovim programskim rješenjima. Pri uobičajenom ocjenjivanju ispita na fakultetu, najčešće sudjeluje nekoliko različitih nastavnika ili asistenata. Iako imaju zadane kriterije ocjenjivanja, konačni dojam koji ocjenjivač stekne pri ocjenjivanju nekog rješenja je ipak subjektivan, i često neujednačen između ocjenjivača. Velikim jezičnim modelima bi mogli pridonijeti konzistentnosti u kriterijima ocjenjivanja tako što više ne bi bilo potrebe imati više različitih ocjenjivača, već samo jednog. Tako bi se na fakultetu riješio i problem nedostatka osoblja. Osim toga, studentima koji žele što prije saznati svoj rezultat pismenog ispita bi isti dan bile dostupne povratne informacije i ocjene. Budući da su LLM-ovi i dalje relativno novi modeli koji svakim danom sve više napreduju, još nije potpuno utvrđena njihova pouzdanost i preciznost u ovakvoj primjeni. Ovim završnim radom provedena je detaljna analiza ocjena i povratnih informacija koje pružaju LLM-ovi na reprezentativnom uzorku studentskih rješenja zadataka iz predmeta Uvod u programiranje. Cilj je bio utvrditi koliko dobro LLM-ovi mogu prepoznati različite vrste grešaka u kodu, primijetiti nedostatke te procijeniti ukupnu kvalitetu koda te pružiti relevantne i korisne povratne informacije.

1. Veliki jezični modeli

Veliki jezični modeli su modeli umjetne inteligencije koji koriste napredne pristupe strojnog učenja kako bi, između ostalog, znali tumačiti i stvarati odgovore pisane svakodnevnim, govornim jezikom. Naziv „veliki“ su dobili jer su „trenirani“ na jako velikim skupovima podataka. Jedan često korišten pristup strojnog učenja koji koriste LLM-ovi je duboko učenje koje je temeljeno na umjetnim neuronskim mrežama i velikoj količini podataka. Umjetne neuronske mreže su slično kao ljudski mozak izgrađene od puno međusobno povezanih „neurona“, odnosno čvorova i sastoje se od više čvorova koji prosljeđuju podatke jedni drugima uz određene uvjete i mutacije. Poseban model neuronskih mreža koji koriste veliki jezični modeli su transformatori. Njihova glavna značajka je da znaju vrlo dobro učiti kontekst što je jako važno u razumijevanju i generiranju ljudskog jezika. To im omogućava razumijevanje obrazaca i struktura prirodnog jezika i zato mogu razumijeti i tekst koji je napisan nejasno ili u kombinacijama s kojima se nisu susreli ranije [1]. OpenAI, Microsoft, Google i mnogi drugi. U ovom istraživanju korištena su dva različita modela, ChatGPT i Gemini. Svaki od ovih modela je zapravo neka vrsta chatbota. Korisnici im postavljaju pitanja, šalju zadatke i različite upite, a modeli im odgovaraju.

1.1. ChatGPT

ChatGPT je veliki jezični model koji je razvila tvrtka za istraživanje i implementaciju umjetne inteligencije – OpenAI. Prvi dio imena, chat, govori da ovaj model komunicira s korisnicima razgovorom poput ljudskog. Format dijaloga mu omogućuje da odgovori na dodatna pitanja, prizna svoje pogreške i odbije neprikladne zahtjeve. Drugi dio imena, GPT, označava Generativni unaprijed obučeni transformator. Kao što mu ime kaže, ovaj model koristi generativni pristup, što znači da može generirati tekst sličan ljudskom predviđanjem sljedeće riječi u nizu danom kontekstu [3]. Model je treniran metodom podržanog učenja na osnovi povratnih informacija od ljudi. Postoji nekoliko GPT verzija. Nakon prve tri verzije, izlazi nova, značajno bolja verzija GPT-3.5 kojom je pokrenut ChatGPT i koja je sposobna razumjeti i ispisati upute prirodnog jezika i generirati programski kod. Nakon nje izlazi poboljšana verzija GPT-4 koja je dostupna korisnicima ChatGPT Plus opcije koja nije besplatna te kao API za programere koji razvijaju aplikacije

i servise. Prema internim procjenama iz OpenAI-a, postoji 82% manja vjerojatnost da će GPT-4 odgovoriti na zahtjeve za nedopušteni sadržaj i 40% veća vjerojatnost da će proizvesti točne činjenične odgovore od verzije GPT-3.5 [2]. U ovom završnom radu korištena je svima dostupna i besplatna verzija GPT-3.5.

1.2. Gemini

Gemini je veliki jezični model u obliku chatbota razvijen od strane Google-a. Pokrenut je 2023. godine pod nazivom Bard i korišten za razne zadatke kao što su za pripreme za razgovore za posao, otklanjanje pogrešaka u kodu, osmišljavanje novih poslovnih ideja te generiranje slika. Google je s Bardom htio ljudima diljem svijeta omogućiti pristup svojim modelima umjetne inteligencije, a Gemini je njihova najnapredniju obitelj modela. Kako bi to naglasili i odrazili, Bard je preimenovan u Gemini. Model Gemini dostupan je u mnogim Googleovim proizvodima, kao što su Android uređaji, mobilna aplikacija Gemini, Google pretraživanje, Google fotografije i drugi. Google planira dodatno integrirati Gemini u svoje Search, Chrome i druge usluge [4].

2. Metodologija

2.1. Eksperimentalna metodologija

Istraživanje je provedeno na međuispitu predmeta Uvod u programiranje. Ovaj predmet bavi se temeljima programiranja i razvojem osnovnih vještina u jezicima više razine. Znanje gradiva koje se provjerava na međuispitu uključuje proučavanje osnovne sintakse i semantike jezika te koncepata poput varijabli, tipova podataka i operacija. Također se fokusira na pridruživanje tipova podataka varijablama, provjeru tipova podataka i rješavanje pogrešaka uzrokovanih nedosljednom uporabom vrijednosti s obzirom na pridruženi tip podatka. Uz to, predmet uključuje i učenje o izrazima, dodjeljivanjima, te uvjetnim i iterativnim upravljačkim strukturama. U tablici (Tablica 1 Gradivo Međuispita) prikazana su područja predmeta koja su se ispitivala u pojedinom zadatku.

Tablica 1 Gradivo Međuispita

redni broj zadatka	gradivo koje se provjerava
1.	while petlja, funkcija za provjeravanje tipa broja (prim u ovom slučaju), zbrajanje svakim korakom petlje
2.	korištenje jednodimenzionalnog polja, pronalazak najmanjeg i najvećeg broja, ispis polja od indeksa najmanjeg elementa do indeksa najvećeg
3.	učitavanje nizova znakova, uspoređivanje nizova znakova prema abecednom redosljedu, ignoriranje praznina u nizu, pretvaranje velikih slova u mala, ispis nizova znakova
4.	korištenje dvodimenzionalnog polja, manipulacija indeksima polja kako bi se dobila rotirana matrica, ispis matrice

Eksperimentalnim načinom istraživanja cilj je bio utvrditi koliko su Veliki jezični modeli uspješni u ocjenjivanju i ostavljanju komentara programskim rješenjima studenata.

Istraživanje je provedeno na međuispitu predmeta Uvod u programiranje, na prvoj godini preddiplomskog studija FER-a. U ovom poglavlju opisane su metode korištene za prikupljanje i analizu podataka te vrednovanje dobivenih rezultata.

2.2. Skup podataka

Podaci na kojima je provedeno istraživanje bili su podaci o izabranom međuispitu i sadržavali su sve potrebne informacije o rješenjima studenata, te ocjenama i povratnim informacijama. Ocjene i povratne informacije studenti su dobili od strane nastavnog osoblja predmeta (asistenata) i one su poslužile za daljnju analizu. Tablica podataka imala je sljedeće stupce: jedinstveni identifikator ispita, broj zadatka, jedinstveni identifikator asistenta koji je ispravljao određeni ispit, programsko rješenje studenta, broj bodova koji je asistent dodijelio tom rješenju, te komentar koji je asistent ostavio kao obrazloženje za bodove koje je student u određenom zadatku dobio. Istraživanje je provedeno na 100 studenata. Iz početnog skupa podataka je za svakog asistenta koji je sudjelovao u ispravljaju odabrano 25 nasumičnih studenata, čime je osigurano da su studenti jednoliko raspoređeni što se tiče njihovih ispravljača. Tablici su dodana još 4 stupca koji su bili potrebni za bilježenje prikupljenih podataka tijekom eksperimenta. Ti stupci su bodovi ChatGPT-a, komentar ChatGPT-a, te bodovi Geminija i komentar Geminija.

2.3. Korištene tehnologije

Početni podaci bili su smješteni u Excel tablici. Ona je korištena u cijelom procesu prikupljanja ocjena i komentara velikih jezičnih modela. Za lakše manipuliranje podacima, te za odabir 100 studenata za istraživanje, korišten je R Studio. To je sučelje za statističku analizu i obradu podataka koje koristi programski jezik R. Pomoću njega se podaci iz Excel tablice mogu vrlo lako oblikovati u željeni format i vizualno prikazati pomoću raznih grafova. Osim toga, dobar je i za računanje potrebnih statističkih podataka poput prosjeka i standardne devijacije. Izrada jedne tablice sa prosjecima i standardnim devijacijama svakog zadatka prikazana je na slici (Sl. 2.1 Izrada tablice u R Studiju).

```

{r, echo=FALSE}

prosijek <- group_by(podaci, "ID pitanja" = id_question)
prosijek <- summarise(prosijek, "Prosijek bodova - asistenti" = round(mean(score), 2),
"Prosijek bodova - Chat-GPT" = round(mean(chatgpt_score),2), "Prosijek bodova - Gemini" =
round(mean(gemini_score),2), "Standardna devijacija - asistenti" = round(sd(score),2),
"Standardna devijacija - Chat-GPT" = round(sd(chatgpt_score),2), "Standardna devijacija -
Gemini" = round(sd(gemini_score),2))

c <- c("UKUPNO", round(mean(ukupni_bodovi_asistenti$total_score),2) ,
round(mean(ukupni_bodovi_chatGPT$total_score),2),
round(mean(ukupni_bodovi_gemini$total_score),2),
round(sd(ukupni_bodovi_asistenti$total_score),2),
round(sd(ukupni_bodovi_chatGPT$total_score),2),
round(sd(ukupni_bodovi_gemini$total_score),2))

prosijek <- rbind(prosijek, c)

kable(prosijek, format="latex") %>% column_spec(1:7, width = "2cm") %>%
  row_spec(0, bold = TRUE)

```

Sl. 2.1 Izrada tablice u R Studiju

2.4. Oblikovanje upita

Oblikovanje upita (eng. *Prompt engineering*) je ubrzanim razvojem umjetne inteligencije kojem svjedočimo u današnje vrijeme, postao sve češće korišten pojam. On predstavlja disciplinu koja proučava kako možemo najbolje iskoristiti modele umjetne inteligencije poput ChatGPT-a i Geminija. Glavni cilj Prompt engineeringa je tumačenje kako je najbolje oblikovati upite i izjave koje šaljemo ovim modelima da bi bile jasne, nedvosmislene i specifične, i bismo dobili najrelevantnije i najtočnije odgovore od modela umjetne inteligencije. Ova disciplina je vrlo važna jer nam umjetna inteligencija često može dati netočne odgovore upravo zbog krivog načina postavljanja upita. U promptu koji se šalje, svaka je riječ bitna i čak i najmanja promjena može rezultirati različitim odgovorima i rješenjima umjetne inteligencije [5]. Kod oblikovanja ispravnog upita važan je ispravno postavljen kontekst jer on daje jasne pozadinske ili okolne informacije kako bi odgovor bio relevantan i specifičan, a ne općenit. Osim toga, instrukcije moraju biti jasne, eksplicitne i precizne kako bi ostavile što manje prostora za dvosmislene odgovore. Konačno, korisnik mora specificirati izlazni format odnosno željeni oblik odgovora umjetne inteligencije. Slijedeći ova pravila, oblikovan je upit kojim sam od ChatGPT-a i Geminija tražila ocjenu i povratnu informaciju za rješenje programskog koda studenta. Na slici (Sl. 2.1) nalazi se upit koji je korišten tijekom cijelog istraživanja.

Bok, trebaš mi ocijeniti zadatak sa ispita sa prve godine preddiplomskog studija FER-a iz predmeta Uvod u programiranje. Programira se u programskom jeziku c.
Ocijeni rješenje studenta ocjenom od 0 do 7.5 bodova.
Napiši i povratnu informaciju u kojoj ćeš obrazložiti dodijeljeni broj bodova.
Rješenje mora ispunjavati sve funkcionalnosti tražene u zadatku.
Nemoj za grešku uzimati nedostatak komentara ili imenovanje varijabli.
Ispis treba biti točno kako je traženo u zadatku.

Ovo je tekst zadatka:

{ovdje ide tekst zadatka koji se trenutno analizira}

Ovo je programsko rješenje studenta:

{ovdje ide programsko rješenje studenta za određeni zadatak}

Sl. 2.2 Sadržaj prompta

Odgovor velikih jezičnih modela bio je u traženom formatu – sadržavao je ocjenu od 0 do 7.5 bodova te komentar u kojem je detaljno objasnio dijelove zadatka koji su točno odnosno netočno riješeni. Iako to u upitu nije bilo traženo, ChatGPT je u nekim zadacima predlagao i načine na koji student može popraviti neispravne dijelove programskog rješenja.

2.5. Način analize i vrednovanja rezultata

2.5.1. Bodovi

U analizi bodova, početna pretpostavka bila je da su bodovi asistenata realni i pravedni, te su s njima uspoređivani bodovi dodijeljeni od strane ChatGPT-a i Geminija. Bodovi su prikazani različitim vrstama grafova jer je takvom vrstom vizualizacije rezultate vrlo lako razumjeti i predočiti. Za prikaz ukupne raspodjele bodova svih studenata iz odabranog skupa korišten je stupčasti graf kakav studenti dobiju i na web stranici predmeta nakon svakog ispita, osim što su u ovom grafu prikazani bodovi od tri različita ocjenjivača. Osim toga, korišteni su i boxplot grafovi za prikaz minimalne i maksimalne vrijednosti, donjega i gornjega kvartila te medijana podataka. Za prikaz odnosa između bodova velikih jezičnih modela međusobno, te njih i asistenata, korišten je toplinski graf koeficijenata korelacije.

Podatke o prosjeku i standardnoj devijaciji bodova smješteni su u već prije spomenutu tablicu.

2.5.2. Komentari

Komentari su analizirani zadatak po zadatak, uspoređujući komentare velikih jezičnih modela s komentarima asistenata. U poglavlju 3.2 opisano je kako su bili formatirani komentari LLM-a, koje su bile pogreške studenata na koje su najčešće upozoravali, a koje pogreške nisu primjećivali.

Za ocjenjivanje relevantnosti komentara korištena su dva pristupa.

U prvom pristupu početna pretpostavka je bila da su komentari asistenata potpuni i relevantni. Na temelju toga su onda komentari velikih jezičnih modela označeni relevantnima i s druge strane lažno pozitivnima ili lažno negativnima.

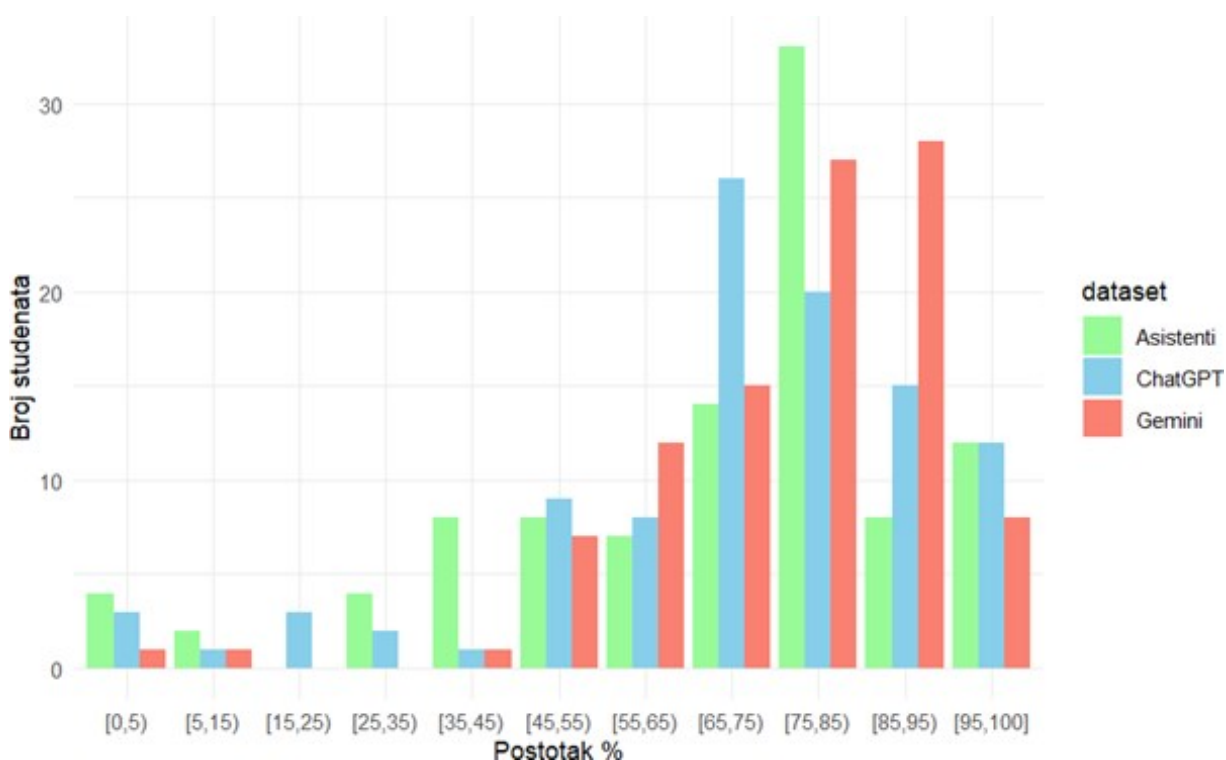
U drugom pristupu, na manjem skupu studenata, komentari sva tri ocjenjivača označeni su relevantnima odnosno nerelevantnima, uspoređujući ih sa „idealnim“ komentarom. Idealni komentar oblikovan je proučavajući programska rješenja studenata i zapisujući sve nedostatke koje kod ima i informacije koje bi studentu bile korisne za razumijevanje dobivene ocjene.

3. Analiza rezultata

3.1. Analiza bodova

3.1.1. Ukupna raspodjela bodova na međuispitu

Analizirani su bodovi koje su studentima dodijelili asistenti te veliki jezični modeli, ChatGPT i Gemini. Ukupna raspodjela bodova na određenom međuispitu studentima se često prikazuje stupčastim grafom kako bi mogli vidjeti kakav je njihov uspjeh u odnosu na ostale studente na predmetu. Na slici (Sl. 3.1) se nalazi stupčasti graf koji prikazuje ukupnu raspodjelu bodova na međuispitu od tri različita ocjenjivača. Zelenom bojom prikazani su bodovi koje su dodijelili asistenti, plavom ChatGPT, a crvenom bodovi koje je studentima dodijelio Gemini. Na osi x nalazi se podjela po postotku bodova koje su studenti ostvarili, a na osi y broj studenata koji se nalazi u pojedinoj skupini.



Sl. 3.1 Stupčasti graf ukupne raspodjele bodova

Budući da je početna pretpostavka da su asistenti pravedno ocijenili studente, primijećeno je da u odnosu na njih, Veliki jezični modeli imaju puno odstupanja.

U intervalu 0-45% ukupnih bodova, Gemini ima zanemarivo mali broj studenata što ukazuje na to da je prilično blag u ocjenjivanju i ne zna prepoznati neke pogreške. ChatGPT je u tom intervalu bio nešto uspješniji, ali također popustljiviji od asistenata.

Asistenti su najviše studenata ocijenili bodovima u intervalu 75-85 %, a ChatGPT u intervalu 65-75 %. Gemini je najviše studenata ocijenio bodovima u intervalu 75-95 %, gdje se nalazi više od polovice ispitanih studenata. Iz toga mogu zaključiti da je Gemini bio vrlo blag u ocjenjivanju.

U zadnjem intervalu gdje se nalaze najuspješniji studenti, sa 95-100% ukupnih bodova, asistenti i ChatGPT imaju jednako mnogo studenata, dok Gemini ima nešto manji broj, što ukazuje na to da su asistenti i ChatGPT dobro usklađeni kada je riječ o prepoznavanju izvrsnosti među studentima, dok Gemini možda ima malo stroži kriterij za dodjelu maksimalnog broja bodova.

Iz analize ovog grafa može se zaključiti da ChatGPT i Gemini generalno ocjenjuju studente s većim brojem bodova nego nastavnici. Kada se u ova dva LLM-a, ChatGPT je u svojem ocjenjivanju bio puno bliži ocjenjivanju asistenata nego što je to bio Gemini. Gemini je uvjerljivo najblaži ocjenjivač od ova tri. Osim toga, važno je istaknuti da razlike u ocjenjivanju možda proizlaze iz različitih pristupa i kriterija koje koriste asistenti, ChatGPT i Gemini.

3.1.2. Prosječni bodovi i standardne devijacije

U proučavanju velikog broja podataka kao što je ovaj zanimljivo je proučavati kojoj vrijednosti je većina podataka bliska. Za to služi računanje prosjeka i standardne devijacije na određenom skupu podataka, a u ovom slučaju to su bodovi koje su studenti dobili za rješenje pojedinog zadatka od asistenata, ChatGPT-a te Geminija. Prosječna vrijednost pruža informaciju o srednjoj vrijednosti bodova koje su studenti dobili, dok standardna devijacija mjeri raspršenost podataka oko tog prosjeka. Veća standardna devijacija ukazuje na veću varijabilnost bodova, dok manja ukazuje na manju varijabilnost i veću konzistentnost u ocjenjivanju. Analiza ovih statističkih mjera može pomoći u razumijevanju kako se ocjene razlikuju između asistenata i velikih jezičnih modela te pružiti kontekst za interpretaciju rezultata.

Tablica (Tablica 3.1) prikazuje podatke za četiri različita zadatka na ispitu. Za svaki zadatak, navedeni su prosječni bodovi i standardna devijacija za svaki od tri načina ocjenjivanja - asistenti, ChatGPT i Gemini.

Tablica 3.1 Prosjek bodova i standardna devijacija

Redni broj zadatka	Prosjek bodova			Standardna devijacija		
	Asistenti	Chat-GPT	Gemini	Asistenti	Chat-GPT	Gemini
1.	5.8	5.5	5.27	2.92	2.54	1.9
2.	6.69	6.3	6.8	1.96	2.15	1.17
3.	1.73	2.76	4.64	2.59	2.79	2.75
4.	5.87	6.38	5.98	2.68	2.15	1.89
UKUPNO	20.08	20.94	22.69	7.28	6.82	5.11

Što se tiče prosjeka bodova, za prva dva i četvrti zadatak se on minimalno razlikuje u sva tri načina ocjenjivanja. Jedino odstupanje koje se ovdje može izdvojiti je prosjek bodova ChatGPT-a za četvrti zadatak koji je pola boda veći od prosjeka asistenata za taj zadatak. U trećem zadatku može se primijetiti najviše odstupanja LLM-ova od ocjenjivanja asistenata. Asistenti daju znatno nižu ocjenu (1.73) u usporedbi s Chat-GPT-om (2.76) i Geminijem (4.64). Iz ovog zaključujem da Gemini ima najveću toleranciju na pogreške ili najblaže kriterije ocjenjivanja. Također, Gemini ima slične prosječne ocjene za sva četiri zadatka, što ukazuje ukazivati na manje varijacije u ocjenjivanju ili na konzistentnost u prilično blagim kriterijima.

Što se tiče standardne devijacije, asistenti imaju veće standardne devijacije za prva dva i četvrti zadatak u usporedbi s LLM-ovima, što ukazuje na veću varijabilnost u ocjenjivanju. Međutim, za treći zadatak, standardna devijacija je najmanja (2.59), što sugerira konzistentnost u ocjenjivanju među asistentima. Chat-GPT ima nešto manju varijabilnost u ocjenama nego asistenti, osim za treći zadatak gdje je varijabilnost (2.79) bliska onoj koju pokazuju asistenti. Standardne devijacije su kod Geminija najmanje za sve zadatke, što ukazuje na najveću konzistentnost u ocjenjivanju. To može značiti da Gemini ima

stabilne kriterije ocjenjivanja ili da je manje osjetljiv na varijacije u kvaliteti rješenja zadataka.

Osim uspoređivanja prosjeka i standardnih devijacija velikih jezičnih modela sa asistentima, zanimljivo je usporediti i vidjeti koliko se razlikuju i te vrijednosti asistenata međusobno. U tablici (Tablica 2) prikazani su prosječni bodovi i standardne devijacije svakog asistenat posebno. Kao i u ukupnom zbroju, svaki od asistenata je s najvećim prosječnim bodovima ocijenio drugi, a s najmanjim treći zadatak. Po prosječnim bodovima može se zaključiti samo da je drugi asistent imao najstrože kriterije ocjenjivanja ili je u njegovom skupu studenata bilo najviše onih s najlošijim rješenjima zadataka. Ostali asistenti nemaju međusobnih odstupanja većih od jednog boda. Što se tiče standardnih devijacija, najmanja je u drugom zadatku gdje su i prosjeci bodova najveći, što znači da su se vrijednosti najviše kretale oko maksimalnog broja bodova.

Tablica 2 Prosjeci bodova i standardne devijacije asistenata

Redni broj zadatka	Prosjek bodova				Standardna devijacija			
	Asistent 1	Asistent 2	Asistent 3	Asistent 4	Asistent 1	Asistent 2	Asistent 3	Asistent 4
1.	6.42	5.12	5.70	5.98	2.56	3.44	3.04	2.58
2.	7.17	6.22	6.30	7.05	1.15	2.40	2.35	1.56
3.	1.77	1.05	1.95	2.14	2.96	1.94	2.68	2.69
4.	5.78	5.28	5.92	6.48	2.95	3.00	2.80	1.81

3.1.3. Detaljnija analiza bodova trećeg zadatka

U poglavlju 3.1.2 je u analizi bodova primijećeno najviše odstupanja LLM-ova od asistenata u trećem zadatku međuispita. Razlog tome bi mogli biti različiti kriteriji ocjenjivača. Ponovljeno je ocjenjivanje od strane LLM-ova za 10 nasumično odabranih studenata i rezultati su prikazani u tablici (Tablica 3). Prikazani su prosjeci bodova za odabrane studente u oba načina ocjenjivanja LLM-ova – s kriterijima i bez.

Tablica 3 Prosjeci bodova ocjenjivača

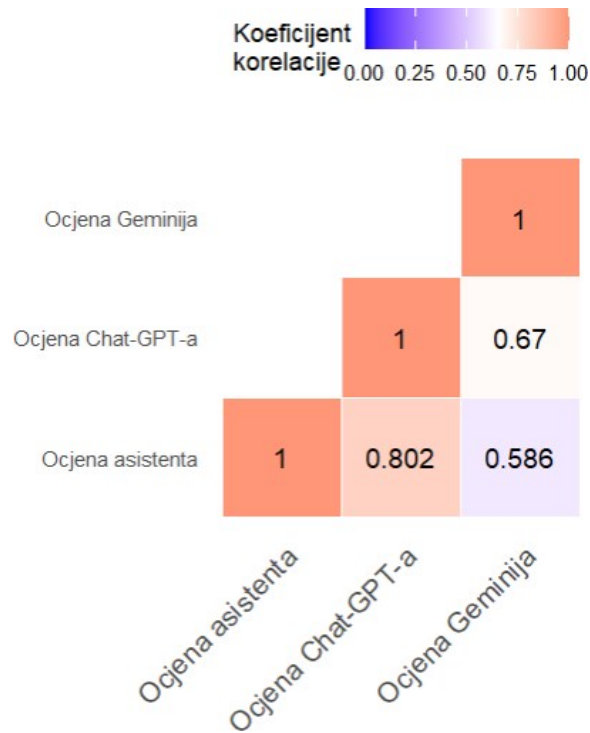
Asistenti	ChatGPT bez kriterija	ChatGPT s kriterijima	Gemini bez kriterija	Gemini s kriterijima
1.225	4.3	2.6	5.7	6.75

Prosjek bodova ChatGPT-a se dodavanjem kriterija približio prosjeku asistenata, dok se prosjek Geminija s kriterijima još više povećao i udaljio od prosjeka asistenata.

Ovime se može zaključiti da Gemini uopće ne prepoznaje kriterije ocjenjivanja čak ni kada su mu eksplicitno zadani. Razlog nižem prosjeku bodova u trećem zadatku u odnosu na ostale zadatke u poglavlju 3.1.2 bili su studenti koji su predali prazno rješenje – njih je bilo čak 21%. Bez njih bi prosjek bodova Geminija bio 5.87 bodova što je slično kao i u ostalim zadacima.

3.1.4. Koeficijent korelacije

Korelacija je statistički postupak kojim se izračunava povezanost dviju varijabli. Vrijednost korelacije se brojčano iskazuje koeficijentom korelacije koji pokazuje u kojoj su mjeri promjene vrijednosti jedne varijable povezane s promjenama vrijednosti druge varijable. Vrijednost koeficijenta korelacije općenito se kreće od -1 do 1, gdje -1 označava savršenu negativnu korelaciju, 0 da između varijabli nema nikakve korelacije, a 1 označava savršenu pozitivnu korelaciju. Koeficijent korelacije vizualno se najbolje može prikazati toplinskim grafom kao na slici (Sl. 3.2).



Sl. 3.2 Toplinski graf koeficijenata korelacije

U kontekstu ovog grafa, koeficijent korelacije mjeri koliko slično svaka od dviju skupina ocjenjivača dodjeljuje bodove. Korelacija između bodova koje su dodijelili asistenti i ChatGPT je visoka i pozitivna, 0.802, što ukazuje na to da Chat-GPT i asistenti imaju slične kriterije u ocjenjivanju. Kada asistenti dodijele više bodova, i Chat-GPT ima tendenciju dodjeljivanja više bodova, i obratno. Međutim, koeficijent nije 1, što znači da postoji određeni prostor gdje se njihove ocjene razlikuju.

Koeficijent korelacije između ocjenjivanja asistenata i Geminija iznosi 0.67. Ovdje vidimo umjerenu pozitivnu korelaciju. To sugerira da iako Gemini dodjeljuje ocjene slično asistentima, postoji veći stupanj razlike u njihovim ocjenjivanjima u odnosu na onaj između Chat-GPT-a i asistenata. To znači da u nekim slučajevima, kada asistenti dodijele visoke ocjene, Gemini može dati niže i obrnuto, ali u većini slučajeva idu u istom smjeru. Koeficijent korelacije između Chat-GPT-a i Geminija je također pozitivan, što ukazuje na to da generalno postoji sklonost dodjeljivanja sličnih ocjena. Ipak, koeficijent nije izuzetno visok, što pokazuje da iako postoji sličnost u ocjenjivanju, postoje i primjetne razlike. U zaključku, visoki koeficijent korelacije između ocjena Chat-GPT-a i asistenata pokazuje sličnost u ocjenjivanju, dok niži koeficijent između Geminija i asistenata te između Geminija i Chat-GPT-a ukazuje na različite kriterije ili sklonosti pri ocjenjivanju.

3.1.5. Boxplot graf raspodjele bodova svih ocjenjivača

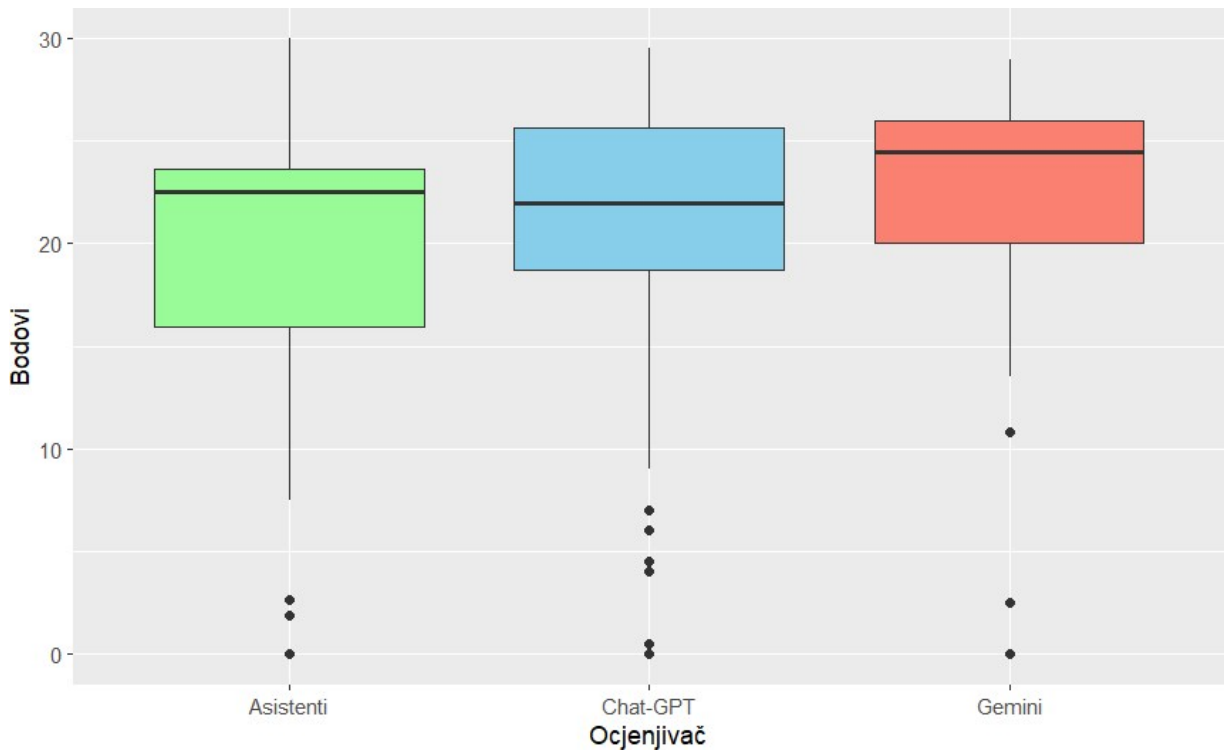
Kutijasti dijagram (engl. box plot ili box-and-whisker plot) prikazuje odnos minimuma, maksimuma, donjega i gornjega kvartila te medijana podataka.

Ako u podacima postoje stršeće vrijednosti (podaci čije su vrijednosti puno veće ili manje u odnosu na druge podatke), one se također označuju na tom tipu dijagrama.

Ovaj grafički prikaz uključuje tri osnovna elementa: kutiju (box), whiskere i odstupanja (outliers). Kutija predstavlja interkvartilni raspon (IQR) podataka, što je raspon između gornjeg i donjeg kvartila (25. i 75. percentil) skupa podataka. Srednja linija u kutiji obično predstavlja medijan. Whiskeri su linije koje se protežu iz kutije prema van, obično predstavljaju minimalnu i maksimalnu vrijednost podataka unutar “whisker” ograničenja. Postoje različite strategije za određivanje dužine whiskera. Odstupanja su pojedinačne točke koje se nalaze izvan opsega whiskera. Oni mogu ukazivati na prisutnost ekstremnih vrijednosti ili potencijalnih anomalija u podacima.

Boxplot omogućuje brzu vizualnu procjenu raspodjele podataka, kao i identificiranje eventualnih odstupanja i ekstremnih vrijednosti. To ga čini korisnim alatom za analizu skupa podataka, posebno kada se radi s velikim količinama podataka ili kada je potrebno uspoređivati više skupova podataka.

Na grafu na slici (Sl. 3.3) prikazana je ukupna raspodjela bodova svakog ocjenjivača.



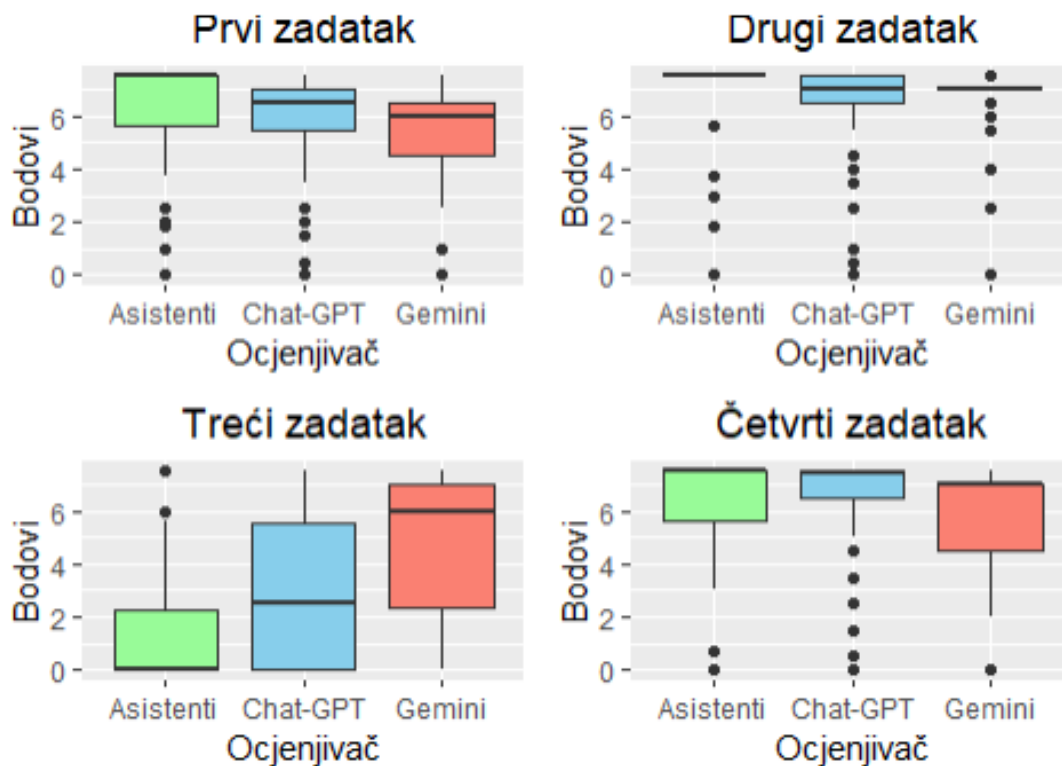
Sl. 3.3 Boxplot graf ukupne raspodjele bodova

Medijan ocjena za asistente je malo iznad 20 bodova. Medijan je jako blizu gornje granice kutije što znači da je 25% studenata imalo oko 24 boda. Postoji nekoliko izvanrednih vrijednosti koje su puno niže od većine ostalih ocjena i spuštaju se skoro do nule.

Medijan za Chat-GPT je također malo iznad 20 bodova, ali kutija ovog boxplota je više izdužena prema gore, što znači da postoji veći raspon ocjena iznad medijana nego što je to slučaj kod ocjene asistenta. I ovdje vidimo nekoliko izvanrednih nižih ocjena.

Medijan ocjena za Geminija je najviši i nalazi se blizu 25 bodova. Raspon ocjena je sličan kao kod ocjene Chat-GPT-a, ali s manje izvanrednih vrijednosti na donjem kraju što ukazuje na to da je Gemini rijetko ocjenjivao studente s niskim brojem bodova.

Na slici (Sl. 3.4) vidimo 4 grafa koji prikazuju bodove koji su dodijelili svi ocjenjivači pojedinom zadatku.



Sl. 3.4 Boxplot grafovi raspodjele bodova pojedinog zadatka

U prvom zadatku prevladavaju vrlo visoki bodovi. Kod asistenata su bodovi najviši, a kod Geminija i ChatGPT-a nešto niži iz čega mogu zaključiti da su LLM u ovom zadatku imali nešto strože kriterije od asistenata kada su u pitanju student koji su najbolje riješili ovaj zadatak.

U drugom zadatku vidimo da kutija u slučaju asistenata i ChatGPT-a niti nema, a kod Geminija je vrlo uska, što znači da je polovica studenata ocijenjena s vrlo sličnim i to vrlo visokim ocjenama. U slučaju asistenata je to čak 7.5 – maksimalni broj bodova koji je bilo moguće dobiti.

Najveća odstupanja ocjenjivanja LLM-ova od ocjenjivanja ChatGPT-a vidimo u trećem zadatku. Medijan asistenata je 0 bodova i nalazi se na samom dnu kutije što upućuje na to da čak 25% studenata nije dobilo niti jedan bod u ovom zadatku. Medijan ChatGPT-a je nešto veći od dva boda, a Geminija 6 bodova. Ovo upućuje na to da su ChatGPT i Gemini ocjenjivali studente blažim kriterijima ili da ne znaju dobro prepoznati lose riješen zadatak. Osim razlike u medijanima, ChatGPT i Gemini imaju i značajnije veće kutije od asistenata što znači da su imali veću varijabilnost prilikom ocjenjivanja nego asistenti.

Za četvrti zadatak su medijani sva tri ocjenjivača vrlo visoki. Međutim, razlikuju se varijabilnosti u ocjenjivanju studenata u IRQ rasponu. Najveću varijabilnost vidimo kod u Geminija, zatim kod asistenata, a najmanju varijabilnost pokazuje ocjenjivanje ChatGPT-a.

3.2. Analiza komentara

Osim ocjene, LLM-ovi su za priloženo rješenje studenta napisali i svoje komentare u kojima su obrazložili ocjenu te istaknuli pozitivne i negativne strane rješenja.

Komentari ChatGPT-a i Geminija većinom su podijeljeni u nekoliko odlomaka koji opisuju pojedinu funkcionalnost koju je zadatak zahtijevao. Puno su duži i opširniji od komentara asistenata, kako za slabije, tako i za zadatke ocijenjene s maksimalnim brojem bodova.

Primjer komentara ChatGPT-a za rješenje zadatka ocijenjeno maksimalnim brojem bodova prikazano je na slici (Sl. 3.5).

“Unos i provjera brojeva: Student ispravno implementira unos elemenata polja i provjeru da li su uneseni brojevi u rasponu od 0 do 9.

Pronalaženje najmanjeg i najvećeg broja: Student ispravno pronalazi indekse najmanjeg i najvećeg broja u polju. Varijable min i max se inicijaliziraju prilikom prvog prolaska kroz petlju, što je ispravno. Tijekom prolaska kroz polje, ažuriraju se indeksi najmanjeg i najvećeg broja.

Ispis elemenata u intervalu: Ispis elemenata u intervalu od indeksa najmanjeg do indeksa najvećeg broja je ispravan. Ovisno o tome koji je indeks manji od drugog, ispisuje se niz elemenata polja od manjeg prema većem ili od većeg prema manjem.

Ispis u skladu s traženim formatom: Ispisuje se niz brojeva bez praznina ili skokova u novi red, što je u skladu s traženim formatom.

Sveukupno, rješenje studenta zadovoljava sve funkcionalnosti tražene u zadatku i nema grešaka.”

Sl. 3.5 Komentar ChatGPT-a

U komentaru se vidi da ChatGPT slijedno opisuje kako se kod izvodi i tako navodi je li određena funkcionalnost zadovoljena – prvo se provjerava unos brojeva, zatim pronalaženje najvećeg i najmanjeg broja i na kraju ispis. Iz komentara LLM-ova mogu zaključiti kako su se „oglušili“ na neke upute iz upita koje sam im poslala te na neke zahtjeve zadataka. Nekim rješenjima studenata asistenti su dodijelili maksimalan broj bodova jer su ispunili sve zahtjeve tražene u zadatku. Međutim, LLM-ovi su istim

studentima u nekim slučajevima dodijelili nešto manji broj bodova, između 5 i 7. Ovo su neki primjeri koji su se pojavljivali u sva 4 zadatka promatranog međuispita:

- “Međutim, moguće je poboljšati čitljivost koda dodavanjem komentara kako bi se olakšalo razumijevanje njegove svrhe i logike.” - ChatGPT
- „Nedostaju komentari u kodu, što otežava razumijevanje namjene različitih dijelova koda.“ - Gemini
- „Varijable nisu najbolje imenovane. Npr., nmax i nmin bi mogli biti nazvani jednostavno max i min.” – ChatGPT
- „Imenovanje varijabli je nejasno i ne odražava njihovu funkciju (npr. minind, maxind).“ - Gemini
- „Optimizirati kod korištenjem ternarnog operatora.” – Gemini

3.2.1. Prvi zadatak

Rješenja prvog zadatka imaju vrlo veliki prosjek bodova.

Što se tiče komentara asistenata, oni su većinom vrlo kratki. Ono što se može istaknuti je da su neki studenti s maksimalnim brojem bodova na ovom zadatku dobili komentar „Optimalnije je prvo provjeriti sadrži li broj unesenu znamenku pa onda provjeravati je li prim“. Dakle, u zadatku se nije ocjenjivala optimalnost pa se za manje optimalna rješenja nisu oduzimali bodovi, ali su asistenti svejedno htjeli ukazati studentima na važnost optimizacije kodova u njihovom budućem programiranju. ChatGPT je također slično komentirao ovakve pogreške “Kod nije najefikasnije implementiran jer se u unutarnjoj petlji provjerava primarnost svakog broja. Bolje bi bilo koristiti funkciju za provjeru primarnosti koja bi se pozivala samo za brojeve koji zadovoljavaju uvjet troznamenkastih brojeva s unesenom znamenkom.”

ChatGPT je većinom dobro znao prepoznati jako loše riješene zadatke. Primjer komentara ChatGPT-a za zadatak kojem je dodijelio 0 bodova nalazi se na slici (Sl. 3.6).

„Ocjenjujući rješenje studenta, evo nekoliko ključnih točaka:

Ispravnost izračunavanja prostih brojeva: Student koristi algoritam Eratostenovog sita za generiranje i označavanje prostih brojeva do 999, ali nije implementirao provjeru prostih brojeva koji zadovoljavaju uvjet da sadrže unesenu znamenku. Umjesto toga, student provjerava jesu li brojevi neparni. To nije ispravan pristup jer neki prosti brojevi nisu nužno neparni (npr. 2).

Prikupljanje prim brojeva koji sadrže unesenu znamenku: Student koristi složeni pristup provjere je li znamenka prisutna u broju, što može biti nejasno i manje efikasno od jednostavne aritmetike modulo operacije.

Ispisivanje prim brojeva i zbroj: Ispisivanje prim brojeva i zbroj je ispravan i odgovara zahtjevima zadatka.

Komentari i čitljivost koda: Kod je relativno čitljiv, ali nedostaju komentari koji bi pomogli u razumijevanju.“

Sl. 3.6 Komentar ChatGPT-a

Niti jedan LLM nije znao prepoznati pogrešku obrnutog redoslijeda ispisa brojeva.

Komentar koji su studenti često dobivali od asistenata je „Nema potrebe obilaziti sve moguće djelitelje; dovoljno je pronaći jedan djelitelj kako bismo dokazali da broj nije prost, nakon čega možemo prekinuti petlju.“. Tu grešku u optimizaciji ChatGPT je također znao prepoznati i spomenuti u komentaru: “Iako je općenito funkcionalan, postoji prostor za poboljšanje efikasnosti. Na primjer, moglo bi se zaustaviti provjeru dijeljenja čim se pronađe djelitelj.”.

Gemini nije niti jednom prepoznao ovakvu vrstu propusta u optimizaciji koda.

Kada ChatGPT prepozna neku grešku u implementaciji, uvijek preporuči neki konkretan način poboljšanja, npr. „Kôd bi se mogao malo optimizirati u smislu efikasnijeg pronalaženja prim brojeva, na primjer, preskakanjem parnih brojeva osim 2.” Međutim, neke preporuke su bile besmislene:

- „Iako rješenje koristi osnovni algoritam za provjeru je li broj prost, moglo bi se poboljšati preskakanjem neparnih brojeva“
- „petlja ide unatrag od 999 prema 100, što može biti manje efikasno od petlje koja ide unaprijed od 100 prema 999“
- „Postoji mogućnost optimizacije algoritma tako da se izbjegne provjera primarnosti za sve troznamenaste brojeve koji sadrže unesenu znamenku.“

Iako je u zadatku napisano da korisnikov unos ne treba provjeravati, LLM-ovi su za to svejedno oduzimali bodove studentima.

- “Nedostaje provjera unosa kako bi se osiguralo da je unesena znamenka u rasponu od 0 do 9.“ – ChatGPT
- „Nedostatak provjere unosa: Rješenje ne provjerava je li unesena znamenka valjana (npr. slovo ili decimalni broj).“ – Gemini

Budući da je Gemini najblaži ocjenjivač, kao i broj bodova, komentari koje je dodijelio različitim rješenjima su vrlo jednolični i nije znao prepoznati većinu pogrešaka. Dva komentara koji su se više puta ponavljali:

- „Ispis brojeva ne koristi praznine kao što je traženo u zadatku.“ – u zadatku to nije bilo traženo, ispis je trebao biti bez praznina između brojeva
- “Koristiti funkciju za provjeru sadržaja znamenke u broju umjesto dupliranja koda.” – u slučaju dupliranja koda gotovo uvijek predlaže korištenje funkcije što je dobra praksa.

3.2.2. Drugi zadatak

Drugi zadatak imao je najveću prosječnu ocjenu od asistenata. Budući da LLM-ovi imaju tendenciju ocjenjivanja većim ocjenama, njihova odstupanja od ocjena asistenata su u ovom zadatku bila najmanja. Međutim, nisu znali prepoznati neke greške, ali su znali i smanjiti ocjenu kada to nije bilo potrebno.

LLM-ovi nisu znali prepoznati grešku u ispisu u slučaju obrnutog redoslijeda ispisivanja od zadanog. Osim toga, nisu primjećivali pogrešku u određivanju najvećeg i najmanjeg elementa do koje je došlo jer varijable za indekse nisu bile inicijalizirane.

Pogreška koju je ChatGPT znao dobro prepoznati i ocijeniti s vrlo malo bodova, a Gemini ju nije uopće primjećivao je bila u rješenjima koja su inicijalizirala polje duljine n dok varijabla n još nije bila definirana.

Gemini je oduzimao samo pola boda ako kod nije uopće ispisivao brojeve u traženom intervalu dok su asistenti i ChatGPT bili puno stroži kod ocjenjivanja zadataka s takvom pogreškom.

Postojali su i dijelovi komentara koji su bili besmisleni, a navedeni su kao razlog snižene ocjene:

- „Naime, uvjet za pronalaženje najvećeg broja trebao bi biti `Polje[i] > maks` umjesto `Polje[i] > maks`.”
- „U petlji za pronalaženje najmanjeg i najvećeg broja, nedostaju vitičaste zagrade oko blokova koda unutar if-statementa. Ovo uzrokuje da se samo sljedeća linija koda izvrši uvjetno, što dovodi do pogrešnih rezultata.” – ChatGPT
- „Pronalaženje svih prvih pojava najmanjeg i najvećeg broja: Student treba izmijeniti pristup kako bi pronašao sve prve pojave najmanjeg i najvećeg broja u polju ako se oni ponavljaju. Trenutno, samo se jednom pronalazi najmanji i najveći broj.” – ChatGPT

Osim oduzimanja bodova za loše imenovanje varijabli, nedostatak komentara i optimizacije, LLM-ovi su oduzimali bodove i za neke funkcionalnosti koji nedostaju ali nisu ni bili u zahtjevu zadatka:

- „Međutim, student nije u potpunosti provjerio jesu li uneseni brojevi u rasponu od 0 do 9. Iako je u tekstu zadatka navedeno da korisnikov unos ne treba provjeravati, uvijek je dobro osigurati unos ispravnih podataka.” - ChatGPT
- „Međutim, nedostatak ispravnog navođenja kako unositi brojeve i nedostatak ispisa novog reda ili razmaka nakon ispisa elemenata rezultira malim padom ocjene.” – ChatGPT

Primjer komentara Geminija za rješenje ocijenjeno maksimalnim brojem bodova prikazano je na slici (Sl. 3.7).

Pozitivne strane:

Kod je pravilno formatiran i razumljiv. Varijable su imenovane na jasan način. Koriste se pravilne petlje za pronalaženje indeksa najmanjeg i najvećeg broja. Ispis je ispravan i u skladu s primjerima u zadatku.

Manje greške:

U uvjetnoj grani za ispis elemenata polja ne koristi se `else` naredba. To znači da će se kod izvršiti i u slučaju kada su `najmanjiIndex` i `najveciIndex` jednaki. Ipak, u tom slučaju se ispisuje prazan string, što je u skladu s traženim funkcionalnostima.”

Sl. 3.7 Komentar Geminija

3.2.3. Treći zadatak

U slučaju vrlo loše riješenih zadataka koji su ocijenjeni s 0 bodova, asistenti su najčešće pisali komentare „Program ne radi za javne testove“ ili „Skroz krivo“, dok su LLM-ovi pisali puno opširnije komentare. Primjer komentara ChatGPT-a nalazi se na slici (Sl. 3.8).

„Ocjena za rješenje studenta:

Učitavanje nizova: Ispravno se učitavaju dva niza znakova, ali nedostaje provjera je li unos prekoračio 100 znakova.

Uklanjanje novog retka: Ispravno se uklanja novi redak iz učitanih nizova.

Usporedba nizova: Logika usporedbe nije ispravna. Student pokušava zamijeniti sve znakove u oba niza istim znakom, što nije korisno za usporedbu. Također, petlje za usporedbu su neispravno implementirane i rezultat nije očekivan.

Ispis rezultata: Ispis rezultata nije u potpunosti ispravan. Razlika na indeksima se ispisuje samo u jednom slučaju, a ne uvijek kada su nizovi različiti. Osim toga, nedostaje razmak između dijela teksta i razlike na indeksima u slučaju kada su nizovi različiti.“

Sl. 3.8 Komentar Chat-GPT-a

ChatGPT dobro prepoznaje pogrešku u usporedbi nizova kod koje bi se za nizove „abcd“ i „efgh“ ispisalo da su jednaki, te kada se radi razlika između velikih i malih slova što nije zahtjev zadatka. Gemini ne prepoznaje niti jednu od ovih grešaka.

Niti jedan LLM ne prepoznaje pogrešku u pronalaženju točnih indeksa, te u ignoriranju praznina u nizu što je traženo u zadatku.

Ističe se jedno rješenje zadatka koje je od asistenata dobilo 0 bodova, dok je od ChatGPT-a dobio čak 4 iako je uočio prilično veliku pogrešku. Ovako glasi dio njegovog komentara:

“Student pokušava uspoređivati znakove s ignoriranjem razlika u veličini slova, ali postoji nekoliko problema u logici koja to radi. Na primjer, uvjeti `if (s1[i]<96 && s1[i]>123)` i `if (s2[j]<96 && s2[j]>123)` nikada neće biti ispunjeni jer niti jedan znak ne može biti manji od 96 i veći od 123.”

Oba LLM-a su kao pogrešku uzimali nedostatak provjere duljine niza, iako je u zadatku navedeno da to nije potrebno provjeravati:

“Nedostatak provjere granica nizova: Nema provjere granica nizova, što bi moglo uzrokovati prekoračenje ako se unesu nizovi duži od 100 znakova.” - ChatGPT

3.2.4. Četvrti zadatak

Za ovaj zadatak su također česti komentari asistenata da rješenje ne radi za javne testove dok LLM-ovi vrlo rijetko navode neke primjere za koje rješenje ne bi radilo.

Asistenti i LLM-ovi su pri određivanju točnosti provjere radi li se o pozitivnom ili negativnom broju uzimali u obzir to u koju je skupinu student smjestio nulu, ali nisu bili konzistentni u tome gdje bi nula trebala pripadati.

- „Pogrešno se prebrojavaju pozitivni i negativni brojevi (npr. nula se tretira kao jedan od ta dva broja).“ - asistenti
- „Kod provjere pozitivnih i negativnih elemenata, student nije provjerio jesu li brojevi točno prepoznati kao pozitivni ili negativni. Na primjer, nula bi trebala biti prepoznata kao pozitivan broj prema zahtjevima zadatka.“ – ChatGPT

Niti ChatGPT niti Gemini nisu znali prepoznati rotiranje matrice u krivu stranu te djelomično rotiranje matrice, kao ni razliku između zamjene redaka matrice i rotiranja matrice. LLM-ovi su oduzimali bodove ako je postojala mogućnost optimizacije i pojednostavljenja koda:

- „Razmotriti korištenje funkcija za kopiranje i rotiranje matrice.“ - Gemini
- “U petlji za rotiranje matrice koristi se suvišna varijabla z.” - Gemini
- „Optimizirati algoritam za rotiranje matrice (npr. korištenjem transponiranja i zamjene stupaca).” - ChatGPT

Česti komentar Geminija bio je da je bolje koristiti do-while petlju umjesto while petlje, te je primjećivao greške u implementaciji ali ih nije detaljnije objasnio: “Greška u implementaciji rotiranja matrice”

4. Rasprava

4.1. Bodovanje programskog koda

4.1.1. Mean absolute error

U vrednovanju ocjenjivanja ChatGPT-a i Geminija, pretpostavka je da su asistenti pravedno ocijenili studente. Kao evaluacijske alate korišteni su MAE i MSE.

MAE (Mean Absolute Error) je prosječna apsolutna razlika između bodova koju su dodijelili asistenti i bodova koje su dodijelili LLM-ovi. To znači da ne uzima u obzir smjer pogreške tj. je li dodijeljeno previše ili premalo bodova, već samo veličinu pogreške – broj bodova odstupanja. Formula gdje je x vrijednost ocjene određenog LLM-a, y vrijednost ocjene asistenta, a n broj ispitanih studenata:

$$MAE = \frac{\sum_{i=1}^n (y_i - x_i)}{n}$$

Kada se po ovoj formuli izračuna MAE, dobiju se iduće vrijednosti:

ChatGPT: 1.09

Gemini: 1.83

Dakle, prosječna apsolutna razlika između bodova koje asistenti dodijele za jedan zadatak i bodova koje dodijeli ChatGPT je 1.09 boda. Prosječna apsolutna pogreška Geminija u odnosu na asistente je nešto veća i iznosi 1.83 boda. Iz ovoga se može zaključiti da je ChatGPT griješio manje od Geminija što se tiče prosječne veličine pogreške.

4.1.2. Mean squared error

MSE (Mean Squared Error) predstavlja prosječnu kvadratnu razliku između bodova koju su dodijelili asistenti i bodova koje su dodijelili LLM-ovi. MSE daje veću težinu većim greškama zbog kvadriranja i zato je dobra metrika za vrednovanje jer je poželjno da uočimo

veće greške koje su radili LLM-ovi. Formula gdje je x vrijednost ocjene određenog LLM-a, y vrijednost ocjene asistenta, a n broj ispitanih studenata:

$$MSE = \frac{\sum_{i=1}^n (y_i - x_i)^2}{n}$$

Kada se po ovoj formuli izračuna MSE, dobiju se iduće vrijednosti:

ChatGPT: 3.77

Gemini: 7.22

Ovi rezultati ukazuju na to da su oba LLM znala imati prilično velika odstupanja u odnosu na asistente. Međutim, MSE Geminija je čak duplo veći nego ChatGPT-a pa ponovno zaključujem da je ChatGPT puno bolji izbor za ocjenjivanje studenata, iako je i on imao dosta odstupanja. Niti jedno od ovih odstupanja nije dovoljno malo da bi se moglo zanemariti i reći da su veliki jezični modeli uspješni u ocjenjivanju programskih rješenja studenata.

4.2. Pružanje povratnih informacija

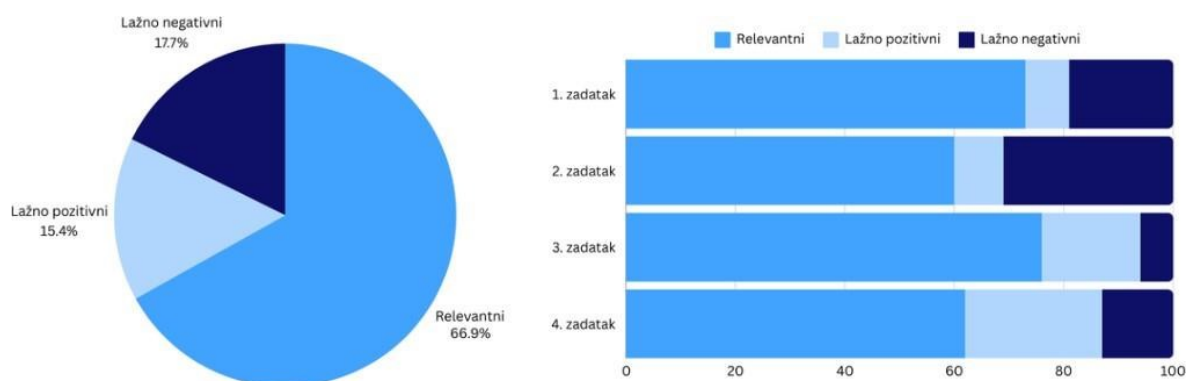
4.2.1. Prvi pristup

U prvom pristupu, početna pretpostavka bila je da su komentari asistenata relevantni. Metoda koju sam koristila je uspoređivanje komentara LLM-ova sa komentarima asistenata. Ukoliko su komentari asistenta i određenog LLM-a bili srodni, tada sam ih označila relevantnima. Komentari su smatrani srodnima ukoliko su upozoravali na iste pogreške u rješenjima, a ako asistenti nisu spomenuli niti jednu pogrešku, tada je provjereno je li LLM u svom komentaru također zaključio da nema značajnih pogrešaka osim u optimizaciji i čitkosti koda što su većinom bile njegove primjedbe. U slučaju kada su ChatGPT i Gemini u svom komentaru spominjali pogreške koje asistenti nisu primijetili i nije ih bilo u programskom rješenju, označeni su ih kao lažno negativni, dok su lažno pozitivnima označeni komentari koji nisu primijetili pogreške koje asistenti jesu i zbog kojih su bodovi trebali biti sniženi.

Na slikama su prikazani grafovi raspodjele komentara po navedenom kriteriju u tri skupine, posebno za ChatGPT i posebno za Gemini.

Na slici (Sl. 4.1) na lijevom grafu se vidi da su komentari ChatGPT-a dvotrećinskom većinom bili relevantni. Odstupanja su podjednako raspoređena u preostale dvije skupine lažno pozitivnih i lažno negativnih komentara.

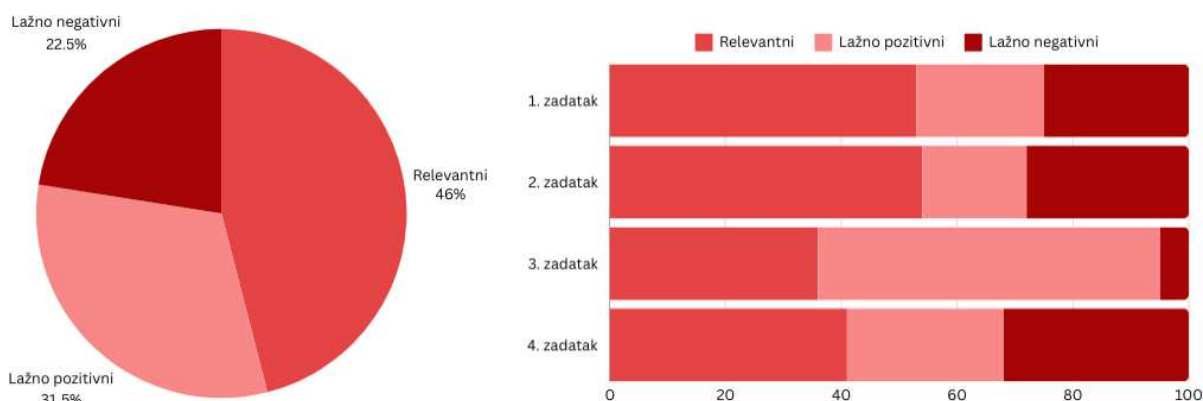
Na desnom grafu vidimo raspodjelu komentara u sva 4 zadatka. Ističe setreći zadatak, za koji su komentari ChatGPT-a u odnosu na komentare asistenata u najvećem broju bili relevantni, sa manje od 10% lažno negativnih i sa oko 20% lažno pozitivnih. U svakom zadatku je 60 ili više posto relevantnih, dok omjer lažno pozitivnih i negativnih varira. Drugi zadatak se ističe po najvećem broju lažno negativnih komentara jer je ChatGPT često ukazivao na pogreške u usporedbi brojeva kada ih zapravo nije bilo.



Sl. 4.1 Komentari ChatGPT-a

Na slici (Sl. 4.2) su prikazana dva grafa na kojima vidimo relevantnost njegovih komentara.

Što se tiče ukupnog broja komentara Geminija, njih manje od polovice su označeni relevantnima. Najuspješniji je bio u komentiranju prva dva zadatka, gdje je više od polovice komentara bilo relevantno. Upravo suprotno od ChatGPT-a, Gemini je u komentiranju programskih rješenja trećeg zadatka bio najlošiji. Kao što je vidljivo na grafu, imao je preko 60% lažno pozitivnih komentara jer nije znao prepoznati većinu pogrešaka na koje su asistenti upozoravali.



Sl. 4.2 Komentari Geminija

4.2.2. Drugi pristup

U drugom pristupu, korišten je skup podataka od 10 nasumično izabranih studenata iz početnog skupa. Za te studente su pregledana njihova programska rješenja za sva 4 zadatka i zamišljen je „idealni“ komentar koji bi studentu pružio sve potrebne povratne informacije. U usporedbi s tim komentarom, komentari sva tri ocjenjivača označeni su relevantnima ili nerelevantnima. Na slici (Sl. 4.3 Relevantni komentari ocjenjivača) je prikazan stupčasti graf koji prikazuje broj relevantnih komentara koje je za određeni zadatak dodijelio svaki od ocjenjivača.

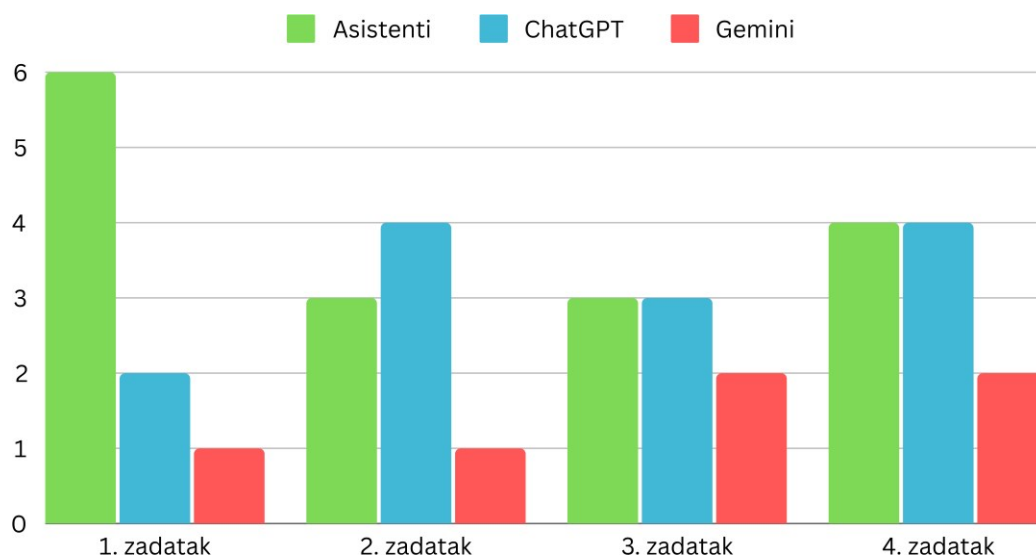
U prvom zadatku vidimo najveću razliku između asistenata, koji su imali 60% uspješnosti komentiranja, i LLM-ova. Unatoč tome, u svim ostalim zadacima, ChatGPT i asistenti su imali vrlo sličan broj relevantnih komentara. Gemini je u svim zadacima uvjerljivo najlošiji i još je jednom pokazano da je on najmanje relevantan izbor za ocjenjivača.

Što se tiče uspješnosti u komentiranju na cijelom proučavanom skupu, asistenti su imali 40%, ChatGPT 32,5%, a Gemini tek 15% relevantnih komentara. Niti jedan ocjenjivač nije imao više od polovice relevantnih komentara što je vrlo loš rezultat.

Komentari asistenata su bili vrlo kratki, većinom bez dovoljno informacija za studente koje bi im koristile za bolje razumijevanje svojih pogrešaka i dobivenog broja bodova.

ChatGPT imao je dovoljno opširne komentare, ali postojale su greške koje nije primjećivao pa je zato njegov postotak nešto niži nego od asistenata.

Gemini je najblaži ocjenjivač koji je primjećivao najmanje grešaka i imao najmanju varijabilnost u svojim komentarima.



Sl. 4.3 Relevantni komentari ocjenjivača

Zaključak

U ovom radu istraženo je koliko su veliki jezični modeli uspješni u ocjenjivanju i ostavljanju povratnih informacija programskih rješenja studenata na međuispitu predmeta Uvod u programiranje. Cilj je bio utvrditi bi li LLM-ovi mogli zamijeniti nastavno osoblje u ocjenjivanju kako bi se ubrzao proces, uskladili kriteriji, te studentima pružile kvalitetne povratne informacije o njihovim rješenjima. LLM-ovi koji su korišteni u istraživanju su ChatGPT i Gemini.

Istraživanje i analiza provedena je na uzorku od 100 studenata. Pomoću različitih grafova i statističkih podataka prikazan je odnos bodova između nastavnog osoblja koje je prethodno ocijenilo studente i bodova koje su im dodijelili LLM-ovi. Osim analize bodova, provedena je i analiza komentara odnosno povratnih informacija koje su dodijelili različiti ocjenjivači.

Nakon provedene analize, predložene su mjere za ocjenjivanje uspješnosti velikih jezičnih modela u ocjenjivanju programskih rješenja. Pomoću prosječne kvadratne razlike i prosječne apsolutne razlike ocijenjena je uspješnost LLM-ova u dodjeljivanju bodova, a za ostavljanje povratnih informacija izračunat je postotak relevantnih komentara koje su dodijelili LLM-ovi.

Prethodno navedenim metodama pokazano je da niti jedan od korištenih modela umjetne inteligencije ne bi mogao zamijeniti nastavno osoblje na predmetu u ocjenjivanju ispita. Iako je ChatGPT pokazao bolje rezultate od Geminija, u usporedbi s ocjenama i povratnim informacijama nastavnog osoblja, i dalje nije dovoljno uspješan i relevantan.

Literatura

- [1] *What is a large language model (LLM)?*, Cloudfare Poveznica: <https://www.cloudflare.com/learning/ai/what-is-large-language-model/>; pristupljeno 8. lipnja 2024.
- [2] OpenAI, Poveznica: <https://openai.com/>; pristupljeno 8. lipnja 2024.
- [3] *What does GPT stand for?*, ZDNET, Poveznica: <https://www.zdnet.com/article/what-does-gpt-stand-for-understanding-gpt-3-5-gpt-4-and-more/>; pristupljeno 8. lipnja 2024.
- [4] *What is Google Gemini?*, ZDNET, Poveznica: <https://www.zdnet.com/article/what-is-google-gemini/>; pristupljeno 9. lipnja 2024.
- [5] *Prompt engineering*, Promediagroup, Poveznica: <https://promediagroup.eu/novosti/prompt-engineering/>; pristupljeno 9. lipnja 2024.

Sažetak

Naslov: **Korištenje velikih jezičnih modela u ocjenjivanju programskog koda studenata**

Ovim radom je cilj bio istražiti uspješnost velikih jezičnih modela u ocjenjivanju programskog koda studenata na rješenjima zadataka iz ispita. ChatGPT i Gemini su LLM-ovi korišteni u istraživanju. Opisan je skup podataka za analizu, veliki jezični modeli i korištena metodologija. Provedene su analize bodova i povratnih informacija koje su studentima dodijelili veliki jezični modeli u usporedbi s bodovima prethodno dodijeljenim od strane nastavnog osoblja s predmeta. Predložene su mjere kojima se ocijenila uspješnost LLM-ova u ocjenjivanju.

Ključne riječi:

- Veliki jezični modeli
- ChatGPT
- Gemini
- Analiza podataka
- Bodovi
- Povratne informacije

Summary

Title: Using Large Language Models in evaluating student programming code

The aim of this work was to investigate the success of large language models in evaluating the students' program code on the solutions to the exam tasks. The LLMs used in the research are ChatGPT and Gemini. The work described the data set for the analysis, the large language models and the used methodology. Analyzes of the scores and feedback awarded to students by the large language models were compared to the scores previously awarded by the subject teaching staff. Measures were proposed to evaluate the success of LLMs in evaluation.

Keywords:

- Large language models
- ChatGPT
- Gemini
- Data analysis
- Score
- Feedback