

Učenje programiranja alatima umjetne inteligencije

Mamula, Saša

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Pula / Sveučilište Jurja Dobrile u Puli**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:137:655432>

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

Download date / Datum preuzimanja: **2024-11-08**



Repository / Repozitorij:

[Digital Repository Juraj Dobrila University of Pula](#)



Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

Saša Mamula

Učenje programiranja alatima umjetne inteligencije

Diplomski rad

Pula, veljača 2024.

Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

Saša Mamula

Učenje programiranja alatima umjetne inteligencije
Diplomski rad

JMBAG: 0303085635, redovan student

Studijski smjer: Informatika

Kolegij: IT Management

Znanstveno područje: Društvene znanosti

Znanstveno polje: Informacijske i komunikacijske znanosti

Znanstvena grana: Informacijski sustavi i informatologija

Mentor: izv. prof. dr. sc. Darko Etinger

Komentor: : izv. prof. dr. sc. Snježana Babić

Pula, veljača 2024.



IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani **Saša Mamula**, kandidat za magistra Informatike ovime izjavljujem da je ovaj Diplomski rad rezultat isključivo mogega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio diplomskog rada nije napisan na nedozvoljeni način, odnosno da je prepisan iz kojega necitiranog rada te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

U Puli, _____



IZJAVA O KORIŠTENJU AUTORSKOG DJELA

Ja, **Saša Mamula** dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj diplomski rad pod nazivom „**Učenje programiranja alatima umjetne inteligencije**“ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, _____

Potpis

Sadržaj

1. Uvod.....	1
2. Programiranje.....	2
2.1 Opći pojam programiranja	2
2.2 Kategorije programskih jezika.....	3
2.2.1 Proceduralni programski jezik	3
2.2.2 Funkcijski programski jezik.....	3
2.2.3 Objektivno orijentirani programski jezik	4
2.2.4 Logički programski jezik	4
2.2.5 Skriptni programski jezik	5
3. Izazovi u učenju programiranja	6
4. Umjetna inteligencija	8
4.1 Povijesni razvoj umjetne inteligencije	8
4.2 Opći pojam i karakteristike umjetne inteligencije	9
4.3 Opći pojam i karakteristika chatbota	11
4.4 Veliki jezični modeli	13
5. Alati za učenje programiranja podržani umjetnom inteligencijom.....	15
5.1 ChatGPT.....	15
5.2 Copilot	18
5.3 Tabnine.....	21
6. Prednosti i nedostaci alata (ChatGPT, Copilot, Tabnine) za učenje programiranja podržani umjetnom inteligencijom	23
6.1 Prednosti ChatGPT-a, Copilota i Tabninea	23
6.2 Nedostaci ChatGPT-a, Copilota i Tabninea	29
7. Istraživanje percepcije studenata prema korištenju alata za učenje programiranja podržani umjetnom inteligencijom	35
7.1 Metodologija istraživanja	35
7.1.1 Cilj i metode istraživanja.....	35

7.1.2 Procedura i anketni upitnik u istraživanju	35
7.1.3 Uzorak ispitanika.....	37
8. Rezultati istraživanja	47
9. Zaključak.....	52
10. Literatura.....	53
11. Popis slika	60
12. Popis tablica.....	61
13. Prilozi	62
Sažetak	69
Abstract.....	70

1. Uvod

U današnjoj eri napredne tehnologije programiranje se pokazalo kao ključna vještina koja ima ogroman potencijal za osobni rast. Međutim, ovladavanje umijećem programiranja može biti veliki zadatak koji zahtijeva odlučnost, sustavan pristup i pristup visokokvalitetnim obrazovnim resursima. Srećom, pojava umjetne inteligencije (AI) revolucionirala je različite aspekte ljudskih nastojanja, uključujući obrazovanje. Alati pokrenuti umjetnom inteligencijom sada su pronašli svoj put u područje obrazovanja programiranja, predstavljajući nove puteve i vrijedne resurse za studente.

Cilj ovog rada je pobliže objasniti mogućnosti primjene alata za učenje programiranja podržanih umjetnom inteligencijom. S tim u vezi, na osnovu istražene literature, izdvojene su prednosti i nedostaci odabranih alata ove namjene te je provedeno kratko istraživanje mišljenja studenata o istima u procesu učenja programiranja.

Ovaj rad raspisan je u devet poglavlja. Početno poglavlje odnosi se na uvod u kojem se objašnjava temu ovoga rada, a to je učenje programiranja alatima umjetne inteligencije. Programiranje će biti objašnjeno u drugom poglavlju, njegovo opće značenje i kategorije programskih jezika. Treće poglavlje predstavlja izazove s kojima se programeri susreću tijekom programiranja. Četvrto poglavlje predstavlja umjetnu inteligenciju u kojoj će biti objašnjen razvoj umjetne inteligencije kroz povijest. Opći pojam i karakteristike vezane za umjetnu inteligenciju, opći pojam chatbotova te na kraju ukratko objašnjeni veliki jezični modeli. Peto poglavlje predstavlja istražene alate koji koriste umjetnu inteligenciju te su ondje ukratko objašnjeni. Šesto poglavlje vezano je za prednosti i nedostatke istraženih alata te njihovo objašnjenje vezano za istraženu literaturu. Sedmo poglavlje vezano je za provedenu anketu koja je bila vezana za studente i njihovo mišljenje o korištenju chatbotova za učenje programiranja. Osmo poglavlje objašnjava rezultate svih odgovora dobivenih iz ankete te kratko objašnjenje statistike svih odgovora. Na samome kraju, posljednje poglavlje vezano je za zaključak u kojem se opisuje unapređenje koje pruža učestalije korištenje alata umjetne inteligencije.

2. Programiranje

U okviru ovog poglavlja objasnit će se opći pojam programiranja i programski jezici.

2.1 Opći pojam programiranja

Programiranje je umijeće korištenja računalne tehnologije za vješto rješavanje složenih problema, uz istovremeno poticanje rasta analitičkih i računalnih sposobnosti. Obuhvaća korištenje inovativnih strategija rješavanja problema koji nadilaze područje samog programiranja, revolucionirajući tako nečije kognitivne procese (Resnick et al., 2009).

Konačni cilj programiranja je prenijeti studentima sposobnost algoritamskog razmišljanja i vještog rješavanja poteškoća u programiranju (Hercigonja, 2017).

Algoritam je niz naredbi koje, kad se izvode ispravnim redoslijedom, vode do rješenja za jasno definiran problem (Manger et al., 2003).

Učenje programiranja ne obuhvaća samo stjecanje novih koncepata, već i usavršavanje sposobnosti primjene ovih načela za rješavanje izazova na različitim programskim problemima. Unatoč stalnim istraživanjima u području obrazovanja programiranja, potraga za optimalnim pristupom ovladavanju ovom vještinom i dalje je u tijeku (Hassinen et al., 2006).

Razumijevanje programiranja zahtijeva vještu primjenu stečenog znanja za rješavanje novih problema. Bavljenje programiranjem tijekom nečijeg obrazovanja najvažnije je iz nekoliko uvjerljivih razloga: poticanje cjeloživotne žeđi za znanjem i znatiželje, usavršavanje organizacijske sposobnosti, stjecanje neophodnih kompetencija u tehnološkom napretku i poticanje šire zajednice pojedinaca koji razumiju tehnologiju (Passey, 2017).

2.2 Kategorije programskih jezika

Ovo poglavlje bit će posvećeno vrstama programskih jezika u odnosu na njihovu kategoriju. Postoji pet različitih kategorija koje korisnici mogu koristiti ovisno koju kategoriju zadatak zahtijeva. Kategorije su sljedeće: proceduralna, funkcijska, logička, skriptna i objektno orijentirana.

2.2.1 Proceduralni programski jezik

Proceduralni programski jezik vrsta je koja koristi skupove funkcija i naredbi pomoću kojih izvršava radnje vezane za zadatke. Postoji puno značajki ovog načina programiranja. Brojne od njih su jako važne u programiranju: predefinirane funkcije, lokalne i globalne varijable, modularnost i slično.

Predefinirane funkcije pomažu programerima kako bi riješili probleme pomoću korištenja postojećih funkcija bez potrebe za kreiranjem istih. Razlika između lokalne i globalne varijable je prilično jednostavna. Lokalna varijabla je namijenjena korištenju samo u funkciji koju je programer stvorio dok se globalna varijabla može koristiti i izvan te funkcije što omogućava bolju funkcionalnost cijeloga koda. Modularnost je jako važna stvar kod programiranja, a to je da programer može podijeliti funkcionalnost svojeg koda na manje nizove blokova koji se, na kraju, mogu povezati u jednu funkcionalnu cjelinu (Preston, 2023).

Primjeri proceduralnih programskih jezika su:

- Pascal
- C
- Basic
- Fortran

2.2.2 Funkcijski programski jezik

Funkcije služe za organizaciju programskog koda. Upotrebljavaju se u puno različitih računalnih jezika. Funkcionalno programiranje je način pisanja koda koji je organiziran i lako razumljiv – to je kao da se slijedi skup pravila za pisanje koda. Funkcionalno programiranje način je pisanja računalnih programa koji se razlikuje od objektno orijentiranog programiranja i proceduralnog programiranja. Određeni ljudi misle da su suprotni, ali zapravo, mnogi računalni sustavi koriste sva tri pristupa zajedno. U

funkcionalnom programiranju cilja se na to da funkcije budu stvarno jednostavne i da ne rade ništa neočekivano. Navedeno se naziva „čista funkcija“. Čista funkcija koristi samo informacije koje je dobila i ne radi ništa drugo. Ne mijenja ništa izvan sebe. Drugo važno pravilo je da se ne mijenjaju dobivene informacije. Također, postoje posebne funkcije koje se nazivaju „funkcije višeg reda“ koje mogu raditi s drugim funkcijama (Tyson, 2023).

Primjeri funkcijskih programskih jezika su:

- Haskell
- Lisp
- Erlang
- F#
- Scala

2.2.3 Objektno orijentirani programski jezik

Objektno orijentirano programiranje (OOP) način je pisanja računalnog koda koji koristi klase i objekte. Pomaže organizirati kod u manje dijelove koji se mogu koristiti iznova i iznova – to je kao korištenje nacрта za izgradnju različitih stvari. Postoje različiti jezici koji koriste OOP, kao što su JavaScript, C++, Java i Python. OOP jezici nisu samo korištenje objekata. Neki jezici kao što su JavaScript, Python i PHP dopuštaju korištenje i objekata i drugih načina pisanja koda (Doherty, 2020).

Primjeri objektno orijentiranih programskih jezika su:

- Java
- C#
- Ruby
- C++
- Python

2.2.4 Logički programski jezik

Logičko programiranje način je pisanja računalnih programa korištenjem logike. To je poput pisanja rečenica koje imaju smisla i slijede pravila. Navedeno se može koristiti kako bi se razumjele stvari i pronašli odgovori sagledavajući sve dostupne informacije. Kako bi se koristilo logičko programiranje, potrebno je imati neke osnovne činjenice koje

su istinite. Navedene se činjenice mogu koristiti za davanje izjava i postavljanje pitanja. Postoje različiti računalni jezici koji koriste logiku za izradu programa. Jedan od najčešćih naziva se Prolog koji može raditi s drugim jezicima kao što su Java i C. Prolog je bio jedan od prvih logičkih programskih jezika i nastao je 70-ih godina dvadesetog stoljeća. Koristi se vrstom logike koja se naziva predikatna logika koja omogućuje korištenje varijabli, umjesto samo istinitih ili lažnih izjava. Prolog, također, može brzo obraditi puno informacija te koristi umjetnu inteligenciju kako bi lakše shvatio stvari. Prolog se može programirati da funkcionira samostalno ili uz pomoć osobe (Virtusa, 2023).

Primjeri logičkih programskih jezika su:

- Alice
- ASP
- Datalog
- Prolog

2.2.5 Skriptni programski jezik

Skriptni jezici su posebne vrste računalnih jezika koji se koriste za pisanje kratkih programa. Ovi programi nisu poput običnih programa koje je potrebno sastaviti prije nego što se mogu koristiti. Umjesto toga, mogu se pokrenuti odmah, red po red. Upotrebljava se poseban program ili alat za čitanje i razumijevanje svake linije programa u hodu. Skripta je poput skupa uputa koje računalo može slijediti. Umjesto da se upisuje svaka uputa, moguće je napisati skriptu koja će to učiniti sama. Navedeno olakšava izvođenje velikog broja uputa odjednom. Skripte mogu pomoći računalima da rade stvari automatski i stvaraju mrežne stranice (Parker et al., 2023).

Primjeri skriptnih programskih jezika su:

- JavaScript
- Perl
- Groovy
- Bash

3. Izazovi u učenju programiranja

Umijeće programiranja predstavlja veliki izazov koji zahtijeva ne samo vrijeme i trud, već i strpljenje, predanost i um. Bez obzira na razinu stručnosti, bilo da se radi o početniku,iskusnom programeru iliiskusnom senioru, područje kodiranja postavlja izazove za sve, povremeno izazivajući frustracije kad put do rješenja izmiče razumijevanju. Ipak, u takvim trenucima nužno je prepustiti se osjećaju i ustrajno tražiti rješenje jer programiranje je uistinu delikatna ravnoteža kreativnosti i strukture (Rao et al., 2022).

Najčešći početnički problem prilikom rješavanja programskih zadataka je u tome što većina nema dovoljno dobru vještinu razumijevanja problema i zbog toga dolazi do problema prilikom rješavanja istih. Naime, ovo su neki od problema kroz koje početnici u programiranju prolaze (Gomes, 2007):

1. Razumijevanje problema

Studenti se često trude riješiti dileme a da ih ne shvate u cijelosti. Ponekad se to događa zbog njihove borbe s dešifriranjem iskaza problema, dok u drugim slučajevima proizlazi iz težnje da krenu s kodiranjem bez odgovarajućeg proučavanja i razumijevanja opisa problema.

2. Povezivanje znanja

Brojni studenti ne uspijevaju uspostaviti točne veze između prošlih problema i ne uspijevaju primijeniti svoje prethodno znanje na nove izazove. Umjesto identificiranja zajedničkih temeljnih načela, oni teže kategorizirati probleme isključivo na temelju površnih sličnosti. Posljedično, njihova rješenja često se oslanjaju na nepovezane presedane što rezultira pogrešnim ishodima.

Autorica Khimchenko (2023) tvrdi da postoje još neki važni čimbenici u izazovima programiranja, a to su:

1. Podcjenjivanje sigurnosti

Programeri koji su novi u izradi računalnih programa ponekad zaborave osigurati sigurnost od loših stvari. Više im je stalo do toga da program radi ispravno, nego da bude zaštićen od problema. Potrebno je koristiti posebne načine za zaštitu računala od loših ljudi koji žele ukrasti podatke. Jedan od načina je korištenje posebnih kodova kad se od

računala traži da pronađe ili spremi podatke. Drugi način je da se provjeri je li računalo sigurno postavljeno i ima li posebne programe za njegovu sigurnost.

2. Posuđivanje koda

Jedan od problema s kojim se programeri početnici često susreću je kad moraju raditi s kodom koji je netko drugi napisao. Navedeno može biti teško ako osobi koja je napisala kod ne mogu postaviti pitanja. Prije upotrebe računalnog programa, važno je provjeriti je li siguran i radi li dobro.

Potrebno je odvojiti vrijeme da se pažljivo pogleda i da se o njemu nauči prije njegovog korištenja. Navedeno će pomoći u sprječavanju problema i olakšati stvari u budućnosti. Korištenje istih dijelova vlastitog koda bez razmišljanja može uzrokovati probleme. Kopiranje i lijepljenje nije dopušteno u programiranju. Ponovno korištenje koda bez razmišljanja može se ponavljati i treba ga ažurirati.

Postoje, također, izazovi vezani i za iskusne programere koji se moraju prilagođavati novim uvjetima razvoja tehnologija i njihovom napretku. Neki od takvih izazova prema Krishni (2023) su:

- Brzo napredovanje tehnologije nameće pojačan osjećaj hitnosti stručnjacima za razvoj softvera da kapitaliziraju ove nove trendove dok stvaraju nove softverske proizvode koji se ističu i osiguravaju ogromnu prednost nad svojim konkurentima. Nedvojbeno, ovo predstavlja najveći izazov u području razvijanja softvera.

4. Umjetna inteligencija

Poglavlje umjetne inteligencije namijenjeno je pobjližem objašnjenju pojma umjetne inteligencije te pojašnjenju pojma chatbotova koji koriste umjetnu inteligenciju za svoj rad.

4.1 Povijesni razvoj umjetne inteligencije

Umjetna inteligencija podrazumijeva da računala i strojevi mogu raditi stvari koje rade ljudi, a da im se cijelo vrijeme ne govori što da rade. Mogu učiti iz onoga što su radili prije i shvatiti što učiniti u različitim situacijama. Jednostavan primjer je da se svjetla automatski pale kad netko uđe u prostoriju. Strojno učenje način je na koji se računala poboljšavaju i uče sama a da im ljudi ne govore što da rade. To je dio nečega što se zove umjetna inteligencija koja pomaže računalima da rade stvari poput ljudi, ali strojno učenje pomaže računalima da postanu još bolja u tim stvarima bez potrebe da ih ljudi više podučavaju. Neke od kvaliteta koje pruža umjetna inteligencija glede poboljšanja i ubrzanja cijelog razvoja strojnog učenja i učenja programiranja generalno su: planiranje projekata, analiza problema, upravljanje projektima, analiza kvalitete. Umjetna inteligencija pomaže programerima da naprave bolje programe tako što pronalaze pogreške i ispravljaju ih. Također, upotrebljava posebne računalne alate za testiranje programa i uvjeravanje da dobro funkcionira. Time je program općenito bolji (Majestic, 2023).

Prije nego što je tehnologija pomogla u izradi računalnih programa, bilo je jako teško pronaći greške u njima. Potrebno je bilo pogledati svaki redak i shvatiti što je pošlo po zlu. Umjetna inteligencija je doista poput pametnog pomagača računalnim programerima. Može pronaći i popraviti pogreške u programima koje naprave bez da im smeta. Također, pomaže da programi rade bolje i brže. Postoje novi računalni alati koji mogu sami pronaći i popraviti greške u programima koji će biti objašnjeni ispod, što znači da ljudi ne moraju trošiti toliko vremena na traženje i popravljjanje grešaka. Navedeno olakšava i brže završava projekte. Ponekad ljudi koji rade računalne programe moraju potrošiti puno vremena na čitanje i ispravljanje pogrešaka. Međutim, sada postoje posebni pomagači koji im mogu dati savjete i primjere kako bi im olakšali i ubrzali rad (Beladiya, 2022).

Programerska zajednica bez mladih i novih snaga koje sve brže i bolje usavršavaju nove tehnologije neće biti kompetentna. Oni koriste nešto što se zove umjetna inteligencija da im pomogne napraviti bolje programe i olakša programiranje. Isto tako, pomaže im da rade zajedno i dijele svoje znanje jedni s drugima, a čak i ljudi koji puno znaju mogu naučiti od ljudi koji tek počinju. Ovi alati mogu pomoći novim programerima da postanu bolji u izradi programa i napreduju u svom poslu. Umjetna inteligencija vezana za strojno učenje je otmjena tehnologija koja pomaže programerima softvera da svoj rad ubrzaju i poboljšaju. Nadaleko postoje velike mogućnosti da kod nije dovoljno zaštićen te je ranjiv. Kako bi se navedeno izbjeglo, postoje prilično dobri alati koji će spriječiti moguće posljedice. Kad se ažurira softver, postoji rizik da neće funkcionirati kako treba, ali AI može spriječiti da se to dogodi. Također, može pomoći programerima da provjere svoj rad i naprave bolji softver za projekte koji se traže od njih (Pallavi, 2018).

4.2 Opći pojam i karakteristike umjetne inteligencije

Umjetna inteligencija (AI) je vrsta tehnologije koja se može koristiti za učinkovitije funkcioniranje stvari. Još uvijek je u ranoj fazi, ali već mijenja mnoge dijelove ljudskih života. Ne postoji jedna definicija umjetne inteligencije, ali većina ljudi o njoj razmišlja kao o sustavu koji može raditi stvari koje se obično smatraju inteligentnima, poput pronalaženja kako se nositi s novim situacijama i učenja iz novih informacija. AI je grana računalne znanosti koja proučava kako stvoriti i koristiti umjetne sustave koji su slični ljudskom mozgu (Copeland, 2018).

Peterson (1990) je istaknuo kako je umjetna inteligencija grana računalne znanosti koja proučava i stvara računalne sustave s nekim oblikom inteligencije. Na primjer, to može biti sustav koji uči novi sustav. Koncepti i zadaci, sustavi koji mogu razmišljati o svijetu oko nas i donositi korisne zaključke, sustavi koji mogu razumjeti različite prirodne jezike te percipirati i razumjeti vizualne scene, sustavi koji izvode druge vrste pothvata koji su intrinzično potrebni tipu ljudi, inteligencija i slično. Prema ovoj definiciji, umjetna inteligencija sažima teoriju i razvoj računalnih sustava koji mogu pravilno obavljati zadatke koji zahtijevaju ljudsku inteligenciju, kao što su vizualna percepcija, prepoznavanje govora, donošenje odluka i prijevod između jezika (Peterson, 1990).

Općenito, postoje tri vrste umjetne inteligencije: jaka umjetna inteligencija, slaba umjetna inteligencija i opća umjetna inteligencija. Slabu umjetnu inteligenciju shvaća kao:

„Izgradite stroj koji je jednak ili premašuje naše mogućnosti za zadani cilj.“ S umjetnom inteligencijom dolazi moćna umjetna inteligencija. Ostale ljudske osobine koje se mogu povezati s jakom umjetnom inteligencijom jesu samosvijest, razboritost, emocija i moral. Nešto dalje od jakih umjetnih Inteligencija je opća umjetna inteligencija. Sposobna je izvršiti bilo koji zadatak koji čovjek može, ali na razini iznad ljudske sposobnosti (Walsh, 2017).

Alan Turing bio je poznata osoba na području umjetne inteligencije jer je prvi postavio pitanje mogu li strojevi misliti. Njegov rad pokazao je kako izgraditi inteligentne strojeve i kako testirati njihovu inteligenciju. Turing je bio važan i zbog svog rada po nazivom „igra oponašanja“ koja se i danas koristi za testiranje stupnja razvoja umjetne inteligencije koji se na kraju naziva Turingov test (Turing, 1950).

Wang je u svojoj knjizi „O definiranju umjetne inteligencije“, analizirao problem s definiranjem umjetne inteligencije, uz napomenu da ne postoji konsenzus oko definicije. Problem koji umjetna inteligencija ne može u potpunosti definirati zapravo je ona sama inteligencija. Inteligencija je složena i teško razumljiva zbog puno različitih razloga. Za razumijevanje inteligencije isti problem javlja se i kod definicije umjetne inteligencije. Zato veliki broj stručnjaka nije raspravljao o definicijama, već je odlučio nastaviti s vlastitim opcijama bez obzira na to spada li u područje istraživanja umjetne inteligencije (Wang, 2019).

Karakteristike umjetne inteligencije prema Josephine (2023) su:

- prilagodba i učenje
- rješavanje problema
- obrada prirodnog jezika
- percepcija
- automatizacija
- rukovanje podacima
- paralelna obrada
- kontinuirano učenje
- autonomno donošenje odluka
- interakcija čovjek-stroj

4.3 Opći pojam i karakteristika chatbota

Chatbotovi su poput robota koji govore i s kojima ljudi mogu razgovarati. Njih se prvi sjetio čovjek po imenu Alan Turing. Danas postoje napredniji roboti koji dobro razmišljaju, poput IBM Watsona. Svojevremeno, kad su ljudi pozvali tvrtku u pomoć, morali su pritisnuti brojne tipke na svom telefonu da bi razgovarali s robotom. Sada robot može razgovarati s njima na ekranu računala. To se dogodilo jer je tehnologija postala bolja, a robot pametniji (Oracle, 2023).

Chatbotovi su računalni programi koji razgovaraju s ljudima poput ljudi. Pokušavaju razgovarati razumljivim riječima. Chatbotovi su poput robota koji mogu razgovarati s ljudima. Oni koriste računalne programe kako bi razumjeli što ljudi govore i odgovorili na način koji ima smisla. Mogu razgovarati sami ili čekati da im se ljudi prvi obrate. Da bi to učinili, imaju posebne dijelove koji im pomažu razumjeti što ljudi govore i zatim smisliti što odgovoriti (Winkler et al., 2018).

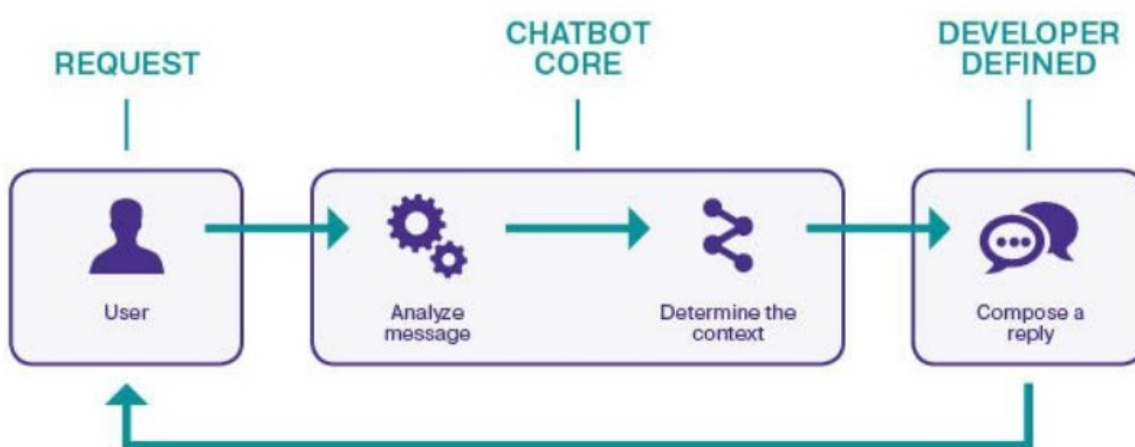
Chatbotovi usmjereni na zadatke (deklarativni) bazirani su na obavljanje jedne funkcije. Oni su poput pomagača koji jednu stvar adekvatno rade. Oni mogu odgovoriti na određena pitanja i pomoći s određenim zadacima, poput saznavanja radnog vremena trgovine ili dovršetka jednostavnih programskih zadataka. Često se koriste za korisničku službu i podršku. Postoje računalni pomoćnici koji se nazivaju chatbotovi i mogu razgovarati poput virtualnog pomoćnika. Neki chatbotovi mogu razumjeti riječi i učiti iz njih, a mogu čak zapamtiti što se ljudima sviđa i predložiti stvari koje bi mogli željeli. Za to koriste posebnu tehnologiju, a neki od primjera su Siri i Alexa (Oracle, 2023).

Hibridni chatbotovi su poput robota koji mogu razgovarati s ljudima. Oni mogu razumjeti što ljudi govore na različite načine, mogu slušati određene riječi ili fraze ili mogu dati ljudima popis opcija za odabir. To im pomaže da bolje odgovore na pitanja (Brush et al., 2021).

Chatbotovi mogu biti jako pametni i pomoći korisnicima da bolje uče. Koriste računala kako bi napravili plan koji korisnicima pomaže u učenju na dobar način te čini učenje lakšim i zabavnijim. Chatbotovi korisni su računalni programi koji mogu odgovarati na pitanja i razgovarati s korisnicima cijeli dan i noć. Rade vrlo brzo i čak se mogu koristiti u aplikacijama poput WhatsApp-a te ostalim sustavima u kojima je omogućeno proširenje

za uključivanje chatbota. Ako se na fakultetima koriste chatbotovi, studenti će biti motiviraniji za učenje i lakše će pronaći informacije o zadaćama i važnim fakultetskim događajima. Osim toga, profesori mogu lakše razgovarati sa studentima putem chatbota (Sydle, 2022).

Chatbotovi su poput korisnih robota koji vrlo brzo mogu odgovoriti na pitanja studenata. Umjesto da dugo čekaju da im netko odgovori e-poštom ili da čitaju dugačak popis pitanja i odgovora, mogu jednostavno postaviti pitanje chatbotu i odmah dobiti odgovor. Ovo studentima olakšava i ubrzava rad, što je dobro.



Slika 1. Prikaz rada chatbota i korisnika (Izvor: <https://www.elucian.com/insights/how-chatbots-benefit-higher-ed>)

Prikaz komunikacije korisnika s chatbotom vidljiv je na slici 1. koja objašnjava situaciju u kojoj chatbot prvo analizira postavljeni zahtjev (pitanje) korisnika te nakon toga radi upit vezan za postavljeni zahtjev. Na kraju odgovara te šalje odgovor krajnjem korisniku.

Chatbotovi razvijeni su na tri načina prema Izraylevych (2019), a to su:

- na temelju pravila – odnosi se na korisnikovo postavljanje pitanja chatbotu na koje kasnije chatbot daje striktne odgovore
- pod nadzorom umjetne inteligencije – chatbot ima puno informacija koje mu pomažu razumjeti što ljudi traže i kako odgovoriti – to mu je omogućeno preko podataka koje skuplja dok razgovara s korisnicima
- pod adaptivnom umjetnom inteligencijom – chatbot nastoji učiti pomoću podataka koje mu je dala umjetna inteligencija od prethodnih generacija.

4.4 Veliki jezični modeli

Veliki jezični model (LLM) je vrsta algoritma koji pruža izvanredan napredak u području umjetne inteligencije iskorištavajući moć tehnika dubokog učenja uz opsežne skupove podataka za razumijevanje, sažimanje, generiranje i predviđanje novog sadržaja. Naime, model jezika Eliza, predstavljen 1966. godine na MIT-u, predstavlja nesvakidašnji primjer takvog jezičnog modela umjetne inteligencije. Svaki jezični model prolazi početnu obuku korištenjem određenog skupa podataka, a zatim koristi različite tehnike za uočavanje veza i proizvodnju svježeg sadržaja na temelju stečenog znanja. LLM predstavlja transformativnu iteraciju koncepta jezičnog modela u umjetnoj inteligenciji, značajno proširujući širinu podataka korištenih tijekom obuke i zaključivanja, čime se modelima umjetne inteligencije daje neviđeni porast mogućnosti (Kerner, 2023).

Kad je riječ o njihovoj primjeni, veliki jezični modeli imaju potencijal za korištenje u raznim industrijama i područjima. Prvenstveno se povezuju s generativnom umjetnom inteligencijom. Djelujući kao vrsta modela transformatora, veliki jezični model ispituje odnose u sekvencijalnim skupovima podataka kako bi razumio značaj i kontekst pojedinačnih podatkovnih točaka. Vrste velikih jezičnih modela (Rebecca, 2023):

- zero-shot model
- fino podešeni modeli ili modeli specifični za domenu
- modeli Edge ili modeli na uređaju.

Prema autoru Aminu (2023), ovo su neki od primjera velikih jezičnih modela: Pathways Language Model (PaLM), XLNet, BERT, Generativni unaprijed obučeni

transformatori (GPT) i LLaMA. Iako je LLM još uvijek u razvoju, pomaže korisnicima s raznim zadacima i služi njihovim potrebama u raznim područjima kao što su obrazovanje, zdravstvo, korisnička služba i zabava. Uobičajeni ciljevi LLM-a uključuju:

- prijevod jezika
- generiranje koda i teksta
- odgovaranje na pitanje
- obrazovanje i osposobljavanje
- služba za korisnike
- pravna istraživanja i analize
- znanstvena istraživanja i otkrića.

Korištenje velikih jezičnih modela u području obrazovanja privuklo je značajnu pozornost zahvaljujući mnoštvu aplikacija koje ih nude. Veliki jezični modeli predstavljaju jedinstvenu priliku za poboljšanje iskustva učenja i poučavanja na svim razinama obrazovanja, bilo da se radi o osnovnom, srednjem, tercijarnom ili profesionalnom razvoju. Inherentna vrijednost leži u sposobnosti da se udovolji različitim preferencijama učenja, sposobnostima i zahtjevima pojedinačnih učenika čime se pružaju personalizirana i učinkovita obrazovna iskustva. Zanimljivo je da veliki jezični modeli mogu pomoći učenicima osnovnih škola u kultiviranju njihovih vještina čitanja i pisanja nudeći prijedloge za sintaktička i gramatička poboljšanja dok istovremeno potiču stil pisanja i sposobnosti kritičkog mišljenja. Slično tome, učenici osnovnih i srednjih škola mogu imati koristi od ovih modela jer mogu naučiti različite jezike i stilove pisanja koji se odnose na različite predmete kao što su matematika, fizika, jezik i književnost i druge discipline. Štoviše, studenti se mogu osloniti na velike jezične modele koji će im pomoći u njihovim istraživačkim i pisanim zadacima dok, istovremeno, usavršavaju svoje kritičko razmišljanje i vještine rješavanja problema (Kasneci, 2023).

5. Alati za učenje programiranja podržani umjetnom inteligencijom

Ovo poglavlje posvećeno je prikazu istraženih alata koji koriste umjetnu inteligenciju za učenje programiranja.

5.1 ChatGPT

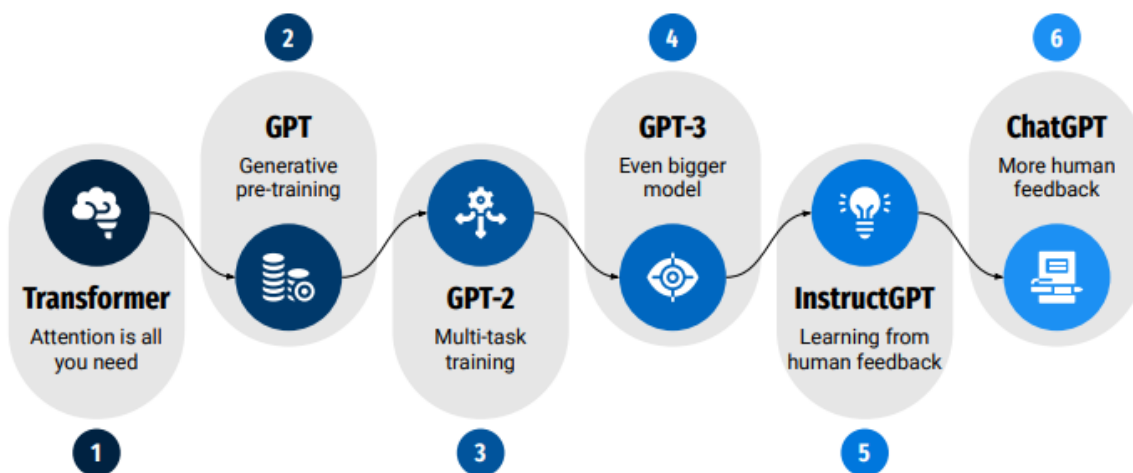
ChatGPT je napredan chatbot koji koristi umjetnu inteligenciju, a koji je razvio OpenAI. ChatGPT može razumjeti i pisati kao programeri jer je napravljen pomoću posebne vrste računalnog programa. ChatGPT temelji se na posebnoj vrsti računalnog programa pod nazivom GPT (Generative Pre-trained Transformer). Ovaj program je izuzetno pametan i ima neke važne značajke zbog kojih dobro funkcionira. Neke od temeljnih karakteristika o ChatGPT-iju su (Experts, 2023):

- Generativan – GPT modeli mogu napraviti nove stvari koristeći ono što su naučili iz primjera. Oni mogu pisati kao ljudi i imati smisla.
- Obučen unaprijed – može ga se zamisliti kao robota koji uči govoriti čitajući puno knjiga i članaka. Što više čita, bolje zna različite riječi, kako ih koristiti u rečenicama i što znače. Navedeno pomaže robotu da jako dobro priča i ostvaruje smislenost kad zapisuje stvari.
- Transformer – GPT modeli su poput stvarno pametnih robota koji su stvarno dobri u razumijevanju i razgovoru na ljudskom jeziku. Oni koriste poseban dizajn nazvan „Transformer“ koji im pomaže razumjeti velike jezične zadatke i reći stvari koje imaju smisla u različitim situacijama.

Tvrtka zadužena za razvoj ChatGPT-a jest Open AI koja je također napravila većinu vrhunskih rješenja koja se vežu za strojno učenje. Strojno učenje omogućilo je da se mogu generirati tekstovi i slike te pretvoriti govor u tekst. Godina 2015. smatra se početkom Open AI-a i svih novih tehnoloških otkrića koja su potaknuli razni istraživači i личности među kojima su: Elon Musk, John Schulman, Greg Brockman (Pop, 2023).

Godine 2018. napravljen je pametni računalni program nazvan GPT. Napravljen je pomoću novog načina učenja računala pod nazivom „Transformer“. Ashish Vaswani osmislio je cijeli program i pustio ga u rad. Program je imao oko 117 milijuna parametara što je bilo dovoljno za početak. Trenutno postoje četiri verzije ChatGPT-a. Najnovija je verzija ChatGPT 4 koji nije dostupan besplatno kao trenutni GPT – 3 model koji većina korisnika koristi. Najnovija inačica ChatGPT-a ima preko jedan trilijun parametara i smatra

se najpotpunijim chatbotom današnjice. ChatGPT i druga Generative AI rješenja jako će se promijeniti u sljedećih nekoliko godina. Neke promjene mogu se nagađati, međutim, ne zna se hoće li se dogoditi u budućnosti.

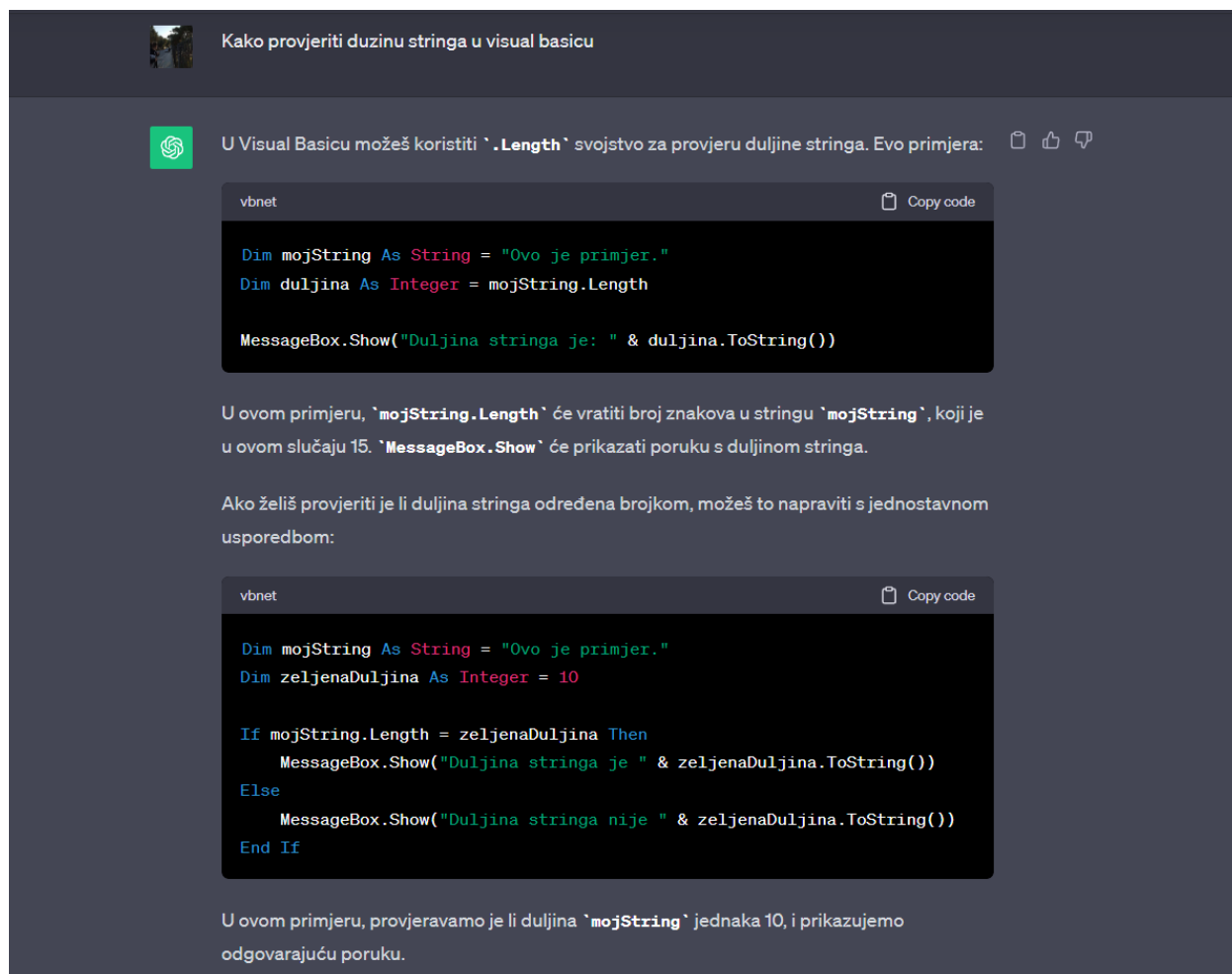


Slika 2. Prikaz razvitka transformer arhitekture ChatGPT-a (Izvor: <https://arxiv.org/pdf/2302.10724.pdf>)

Razvoj ChatGPT-a kroz vrijeme prikazan je na slici 2. Prvo se započelo osnovnim modelom koji je označen brojem 1. Brojem 2 prikazana je prva verzija GPT-a koji je omogućavao generativno treniranje. Pod brojem 3 prikazan je razvoj GPT-2 koji je unaprijeđen tako što mu je omogućen multitasking. GPT-3 koji je trenutno besplatan i koriste ga većinom svi, označen je brojem 4 te on započinje razvoj s puno dodatnih informacija. Učenje preko ljudskih odgovora označava verziju Instruct GPT-a koji je prikazan pod brojem 5. Najnovija verzija koja pruža iteraciju modela koji je temeljen na Instruct GPT-u te koji je trenutno najmoderniji chatbot prikazan je na kraju pod brojem 6.

ChatGPT poseban je pomoćnik za programere koji rade robote koji govore. Pomaže im da roboti zvuče kao ljudi kad razgovaraju s njima. ChatGPT kojima se služe programeri je alat koji mnogi ljudi koriste za lakše obavljanje određenih zadataka vezanih uz neki programski jezik te je navedeno postalo vrlo rašireno. ChatGPT vrlo je adekvatan u obavljanju puno različitih stvari i može odgovoriti na pitanja na pametan način jer je učio

iz puno različitih primjera. ChatGPT može raditi mnoge stvari poput pisanja riječi, odgovaranja na pitanja pa čak i stvaranja dijelova koda. Spomenuto može pomoći programerima koji izrađuju računalne programe da svoj rad učine bržim i boljim (SJ, 2023).



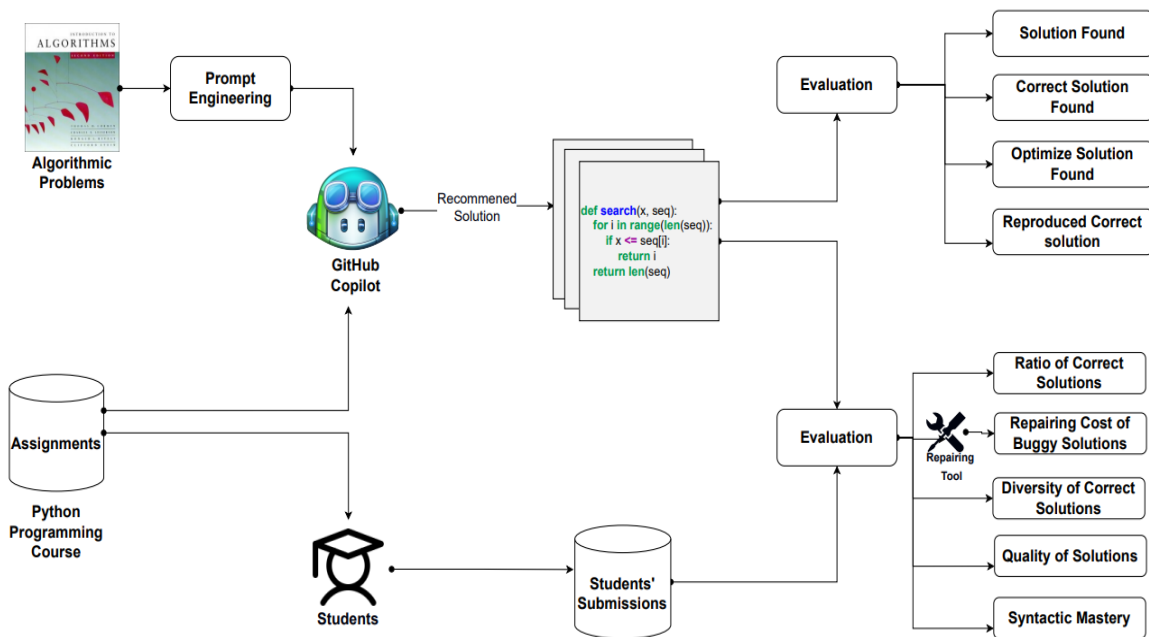
Slika 3. Prikaz sučelja ChatGPT-a (Izvor: obrada autora)

Način rada i komunikacije s chatbotom koji odgovara na određeno pitanje programeru prikazano je na slici 3. Napravljen je upit za chatbota koji treba napraviti upit koji će provjeriti dužinu nekog podatka koji mu je pohranjen u određenoj varijabli za programski jezik C#.

5.2 Copilot

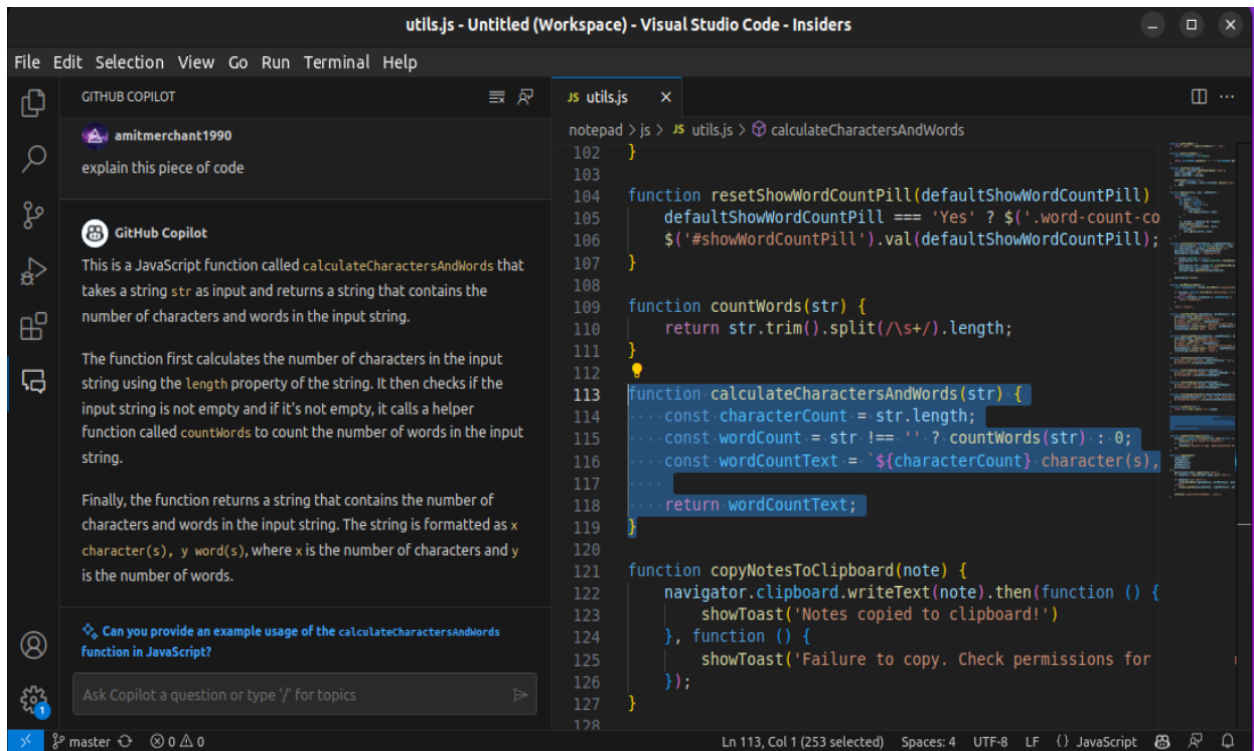
GitHub Copilot pomno je obučen za opskrbu raznolikim rasponom jezika koji se nalaze u javnim spremištima. Kalibar prijedloga koji se dobivaju od ovog izvanrednog alata ovisi o obilju i raznolikosti podataka o obuci dostupnih za određeni jezik. Vrijedno je napomenuti da je JavaScript, obilno prisutan u javnim spremištima, jedan od najuspješnije podržanih jezika GitHub Copilota. Međutim, vjerojatno je da jezici s manjom prevalencijom u javnim repozitorijima mogu dati skromnije ili manje opsežne prijedloge. GitHub Copilot predstavlja preporuke izvedene iz opsežnog modela koji je pažljivo izradio OpenAI, crpeći iz golemog repozitorija otvorenog koda. Posljedično, moguće je da skup za obuku GitHub Copilot može uključivati neke uzorke kodiranja koji su potencijalno opasni, pogreške ili čak reference na zastarjela sučelja ili konvencije za programiranje aplikacija. Kao rezultat toga, prijedlozi koje generira GitHub Copilot mogu nenamjerno sadržavati neželjene uzorke. Međutim, GitHub pruža mnoštvo izvanrednih značajki usmjerenih na poboljšanje kvalitete koda, uključujući, ali ne i ograničavajući se na GitHub Actions, Dependabot, CodeQL i skeniranje koda (GitHub, 2023).

GitHub Copilot je tehnologija koja koristi ogromnu snagu GPT-3 AI modela koji je pomno izradio OpenAI s nevjerojatnih 175 milijardi parametara jezične obrade. Ova priča započela je s pojedincima koji su pokušavali upotrijebiti GPT-3 za dovršavanje koda, koristeći običan engleski unos. Iako su uspjeli postići neke rudimentarne formacije koda, rezultatima je, nažalost, nedostajala praktičnost. Međutim, OpenAI, očaran idejom GPT-3 pisanja koda, krenuo je na put k modi sofisticiranih motora za pisanje koda koji su se, na kraju, materijalizirali u njihovu cijenjenu liniju Codex. Inspiriran ovim revolucionarnim razvojem, GitHub je krenuo u razvoj vlastite Copilot tehnologije. Finim podešavanjem Codexovih motora s opsežnim nizom javno dostupnog koda na GitHubu, konstruiran je temeljni okvir GitHub Copilota. Svako postavljanje pitanja vezanih za pomoć pri pisanju koda bit će odrađeno na najvišem nivou jer GitHub Copilot marljivo nadograđuje kod s najvećom preciznošću, oslanjajući se na sveobuhvatno znanje koje je stekao iz svog ogromnog korpusa za obuku (Tilgner, 2022).



Slika 4. Tijek rada predloženih metoda pomoću kojih radi copilot (Izvor: <https://arxiv.org/pdf/2206.15331.pdf>)

Ova sveobuhvatna studija obuhvaća dva različita pristupa za procjenu učinkovitosti Copilota u pružanju preporuka koda za rješavanje programskih problema. Početni projekt usredotočen je na algoritamske izazove izvedene iz poznatog djela o dizajnu algoritama. Nasuprot tome, kasniji niz se vrti oko zadataka tečaja programiranja u Pythonu, dopuštajući detaljnu usporedbu između Copilota i učenika u njihovoj vještini rješavanja poteškoća u programiranju na različitim aspektima, sve vidljivo na slici 4.



Slika 5. Prikaz sučelja copilot bota u Visual Studio Code programu
(Izvor:<https://www.amitmerchant.com/github-copilot-chat/>)

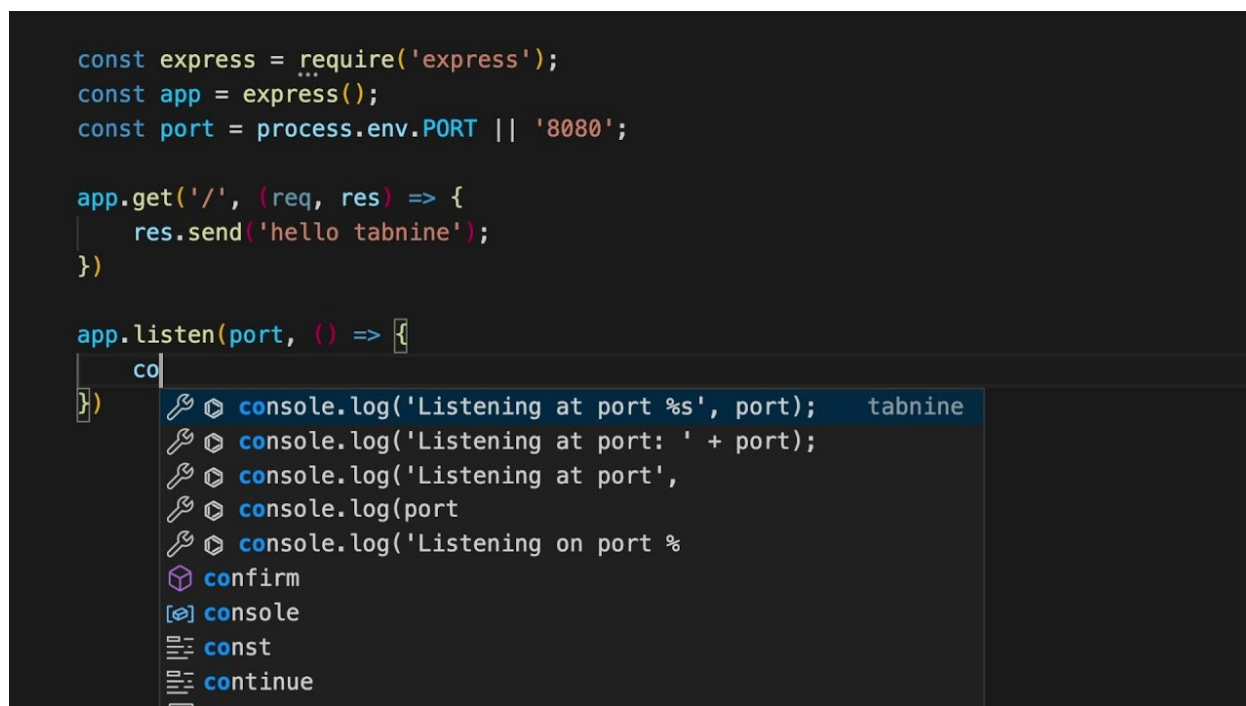
Uvid u rad s copilot botom, koji se dodaje kao ekstenzija na određenu aplikaciju, u ovom slučaju je Visual Studio Code, vidljivo na slici 5. Kad se otvori ekstenzija copilot, može se vidjeti da je postavljeno pitanje da objasni poplavljeni dio koda koji se nalazi s desne strane koji je napisan u JavaScriptu. Copilot objašnjava korisniku da je kod napisan u JavaScriptu te da je ovo funkcija (calculateCharactersAndWords) koja uzima određene parametre dužine riječi i na kraju prikazuje kolika je duljina te riječi. Navedeni alat jako je koristan za početnike ako ne razumiju kod koji su pronašli na internetu i zalijepili.

5.3 Tabnine

Umjetnost programiranja uvelike se oslanja na vještinu pisanja koda. U svjetlu nedavnog napretka u strojnom učenju i obilja otvorenog izvornog koda, pojavila se izvanredna prilika za pojednostavljenje procesa kodiranja. Prepoznavši ovaj potencijal, vizionari Eran Yahav i Dror Weiss, umovi koji stoje iza Tabnine-a, izradili su alat pokrenut umjetnom inteligencijom za podizanje produktivnosti svakog programera od samog početka. Integracijom ovih najsuvremenijih modela u privatne baze kodova organizacija, Tabnine osigurava pristupačnost dok daje prioritet najvećoj sigurnosti i sigurnosti koda. Tabnine je, svojim nesvakidašnjim modelima strojnog učenja osmislio rješenje koje programerima daje besprijekorne preporuke koda, omogućujući im da bez napora izvrše gotovo trećinu svojih zadataka kodiranja. Iskorištavajući snagu i model otvorenog koda te prilagođenih modela umjetne inteligencije, ova inovacija premošćuje sve postojeće praznine u znanju, dajući tvrtkama i programerima neprocjenjiv resurs koji ne samo da štedi veliku količinu vremena, već i besprijekorno pojednostavljuje njihove svakodnevne profesionalne napore (Google, 2022).

Ono što izdvaja Tabnine je njegova platforma sljedeće generacije koja uključuje vrhunske modele umjetne inteligencije posebno obučene za kod. Za razliku od drugih modela koji su samo prilagođeni tekstualnim modelima, Tabnine-ovi modeli pomno su razvijeni od nule. Svaki je model pomno optimiziran za određeni jezik ili domenu što Tabnine-u omogućuje kontekstualno točno dovršavanje koda. Navedeno znači da Tabnine može bez napora dovršiti cijele retke koda ili predložiti relevantne isječke koda s neusporedivom preciznošću. Ukratko, sofisticirani softver Tabnine nudi programerima elegantno i učinkovito rješenje za pronalaženje koda za višekratnu upotrebu. Njegovi napredni AI modeli, posebno obučeni za kod, osiguravaju iznimnu razinu točnosti i produktivnosti. Besprijekornom integracijom u popularne uređivače koda i podržavanjem širokog raspona programskih jezika, Tabnine revolucionira proces kodiranja, čineći ga jednostavnijim i lakšim. Tabnine, softverska tvrtka sa sjedištem u Tel Avivu, razvila je napredni softver koji se ističe u identificiranju višekratnog koda u širokom rasponu programskih jezika. Uz podršku za 31 jezik, uključujući Python, JavaScript, Java, C# i TypeScript, Tabnine-ov softver se neprimjetno integrira s popularnim uređivačima koda kao što su Visual Studio Code i IntelliJ IDE. Ovo rješenje poboljšava iskustvo kodiranja

tako što bez napora predlaže relevantne isječke koda, bez ometanja standardnog tijeka rada programera (Preimesberger, 2022).



```
const express = require('express');
const app = express();
const port = process.env.PORT || '8080';

app.get('/', (req, res) => {
  res.send('hello tabnine');
})

app.listen(port, () => {
  console.log('Listening at port %s', port);
  console.log('Listening at port: ' + port);
  console.log('Listening at port',
  console.log(port
  console.log('Listening on port %
  confirm
  console
  const
  continue
```

Slika 6. Prikaz izgleda rada Tabnina u pomoći pri kodiranju (Izvor:<https://i.ytimg.com/vi/twPtvZuBrAg/maxresdefault.jpg>)

Na slici 6. može se uočiti prikaz rada Tabnine ekstenzije u Visual Studio Codu pomoću koje programeri izrađuju aplikacije za svoje projekte. Uočivši sliku, može se uvidjeti da kada korisnik krene pisati kod, sama ekstenzija mu dozvoljava da preko tipke Tab na tipkovnici ispiše cijelu liniju koda koju alat nudi. Tabnine alat trenutno je najbolji i najtočniji alat za pomoć pri programiranju aplikacija i zahtjevnijih sustava koji je vidljiv na slici 6.

6. Prednosti i nedostaci alata (ChatGPT, Copilot, Tabnine) za učenje programiranja podržani umjetnom inteligencijom

Ovo poglavlje namijenjeno je prikazu prednosti i nedostataka istraženih alata za učenje programiranja podržanih umjetnom inteligencijom.

6.1 Prednosti ChatGPT-a, Copilota i Tabninea

1. ChatGPT

Optimizacija koda koju pruža ChatGPT nudi preporuke za dovršetak koda i sveobuhvatne provjere i otklanjanje pogrešaka. Ovaj pristup poboljšava učinkovitost programiranja te jamči najvišu kvalitetu rezultirajućeg koda. Nadalje, njihov alat omogućuje personalizirano učenje koje je sve istražilo (Rahman et al., 2023).

Prema istraživanju Stojkovskog (2023) predstavlja izvanredno rješenje koje povećava učinkovitost programera i značajno pojednostavljuje proces učenja programskih jezika, posebno za pojedince koji se prvi put upuštaju u ovo područje. Ovaj genijalni alat utire put brzom razumijevanju osnovnih principa programiranja što se pokazalo nezamjenjivim za razvojne programere. S obzirom na pionirsku prirodu ovog pristupa, posjeduje ogroman potencijal da se razvije u vodeći alat za umjetnu inteligenciju u području programiranja.

Važnost poboljšane produktivnosti i olakšavanje usvajanja programskog jezika pomoću ChatGPT-a objašnjavaju Konstantinos et al. (2023). Dodatno se naglašava jednostavnost alata što omogućuje širem spektru korisnika da u potpunosti iskoriste njegove prednosti, bez obzira na njihovu razinu tehničke stručnosti.

U području razvoja softvera, Johnson (2023) stavlja najveći naglasak na usavršavanje jasnoće i razumljivosti koda. Korištenjem opsežnog skupa podataka, njihov inovativni alat ne samo da potiče stvaranje koda koji je vizualno privlačan, već također promiče duboko razumijevanje njegovih zamršenosti. Posljedično, ovo olakšava besprijekornu suradnju i skladnu sinkronizaciju među programerima.

Brzina i jednostavnost rješavanja programskih zadataka predstavljaju neprocjenjiv alat koji pomaže u prepoznavanju i ispravljanju pogrešaka. ChatGPT ima potencijal za revoluciju u razvoju softvera omogućavanjem bržih i učinkovitijih rješenja složenih izazova, objašnjava u svom radu Jeevanandam (2023).

Sofisticirano i uvjerljivo rješavanje zadataka pomoću ChatGPT-a koji pruža 24-satnu dostupnost i potiče motivaciju studenata svojom dinamičnom i zavidljivošću interakcijom. Njihov inovativni alat rješava izazove te obogaćuje obrazovno putovanje zanimljivim primjerima i sveobuhvatnim objašnjenjima (Siregar, 2023).

Neprocjenjiv je alat koji korisnicima daje neograničen pristup nizu rješenja, dostupnih kad im to odgovara. Svojom sučeljem prilagođenim korisniku i izvanrednim kapacitetom za povećanje učinkovitosti, ovaj alat ima potencijal značajno ublažiti opterećujuće radno opterećenje koje programeri podnose, oslobađajući njihovo dragocjeno vrijeme i energiju za poticanje njihove maštovite sposobnosti u području razvoja softvera u svojem istraživanju, govori Bendell (2022).

Prema istraživanjima koja su proveli navedeni autori iz tablice 1., može se uvidjeti da ChatGPT uvelike pomaže u produktivnosti, optimizaciji i kreativnosti prilikom rješavanja zadataka.

Tablica 1. Prednosti upotrebe ChatGPT-a prema autorima i istraženoj relevantnoj literaturi

AUTOR:	REZULTAT ISTRAŽIVANJA - Prednosti:
Rahman et al. (2023)	<ul style="list-style-type: none"> - optimizacija koda rješenja - pruža prijedloge za izvršavanje koda - provjerava greške i otklanja iste - omogućava personaliziranost u učenju
Stojkovski, B. (2023)	<ul style="list-style-type: none"> - bolja produktivnost - olakšavanje učenja programskog jezika novim programerima - veliki potencijal da postane jedan od najboljih AI-a
Konstantinos, I. et al. (2023)	<ul style="list-style-type: none"> - olakšavanje učenja programskog jezika novim programerima - jednostavan za korištenje
Johnson, H. (2023)	<ul style="list-style-type: none"> - omogućava pisanje koda koji je jasan svim programerima - izučen je na velikom skupu podataka
Jeevanandam, N. (2023)	<ul style="list-style-type: none"> - ubrzava i olakšava zadatke programiranja - mogućnost pomoći pri pronalasku problema i otklanjanju istog
Siregar et al. (2023)	<ul style="list-style-type: none"> - alat koji je dostupan 24/7 - poboljšava motivaciju kod studenata jer pruža dinamičnu i zanimljiviju interakciju - omogućava objašnjavanje pomoću primjera te pruža dodatna objašnjenja za sve probleme
Bendell, L. (2022)	<ul style="list-style-type: none"> - besplatan alat - ChatGPT dostupan je korisniku u bilo kojem vremenu - jednostavan za korištenje - ubrzava i smanjuje opterećenje korisnika

(Izvor: obrada autora)

2. Copilot

Alat Copilot pojavljuje se kao katalizator za povećanje produktivnosti programera nudeći iznimnu kontekstualiziranu interaktivnu pomoć tijekom cijelog razvojnog puta. Isto tako, Copilot se neprimjetno integrira u istaknuta razvojna okruženja, otvarajući neusporedivu eru učinkovitosti programiranja prema Reini (2021).

Prvobitan kod koji Copilot razvije može se koristiti u daljnjim zadacima kao šablona. Također, Arsovski (2022) objašnjava da je Copilot izvrstan u pisanju standardnog koda koji svaki početnik zahtijeva da mu se olakša pisanje i razumijevanje logike na koji način radi određeni programski jezik, a uz to je jednostavan za korištenje.

Revolucionarna inovacija Copilot, prema istraživanju Harvarda (2023), ističe se svojom izvanrednom sposobnošću brzog rješavanja programskih prepreka. Značajka Copilota je ne samo da pomaže programerima, već služi i kao dragocjeno pedagoško sredstvo za ambiciozne programere, pružajući im impresivna i interaktivna iskustva učenja uz praktične ilustracije.

Izvanredan alat prema Wermelinger (2023) ističe da je svojom pristupačnošću bez premca, opskrbljujući širok niz uređivača koda. Njegova iznimna konfigurabilnost i prilagodljive značajke omogućuju svakom korisniku da otkrije svoj idealan način angažmana, stvarajući tako istinski personalizirano iskustvo programiranja.

Tablica 2. Prednosti upotrebe GitHub Copilota prema autorima i istraženom relevantnoj literaturi

AUTOR:	REZULTAT ISTRAŽIVANJA - PREDNOSTI:
Wermelinger, M. (2023)	<ul style="list-style-type: none"> - pristupačnost, dostupan za razne uređivače koda - mogućnost konfiguracije prilagodbe za korisnika
Arsovski, A. (2022)	<ul style="list-style-type: none"> - izvrstan u pisanju standardnog koda - šablonski kod se može ponovno upotrijebiti gdje je to potrebno - relativno jednostavan za korištenje
Prather, J. et al., (2023)	<ul style="list-style-type: none"> - poboljšava produktivnost - pruža kontekstualiziranu interaktivnu podršku - Copilot radi kao dodatak unutar popularnih razvojnih okruženja
Harvard et al., (2023)	<ul style="list-style-type: none"> - brzo izdaje odgovore - prijedlozi Copilota mogu poslužiti kao obrazovni resursi za studente - različiti načini interakcije

(Izvor: obrada autora)

Prema istraživanjima koja su proveli navedeni autori iz tablice 2. može se vidjeti da GitHub Copilot omogućava izvrsno pisanje koda te poboljšava produktivnost zbog brzog rješavanja problema i davanja odgovora na iste.

3. Tabnine

Ramachandran (2023) govori kako Tabnine posjeduje veliku brzinu prilikom rješavanja zadataka koja uvelike pomaže programerima da određene zadatke brže i točnije riješe.

Jedna od teškoća koje proizlaze iz ovog okvira uključuje mogućnost davanja netočnih odgovora. Ako alat naiđe na kod koji značajno odstupa od primjera na kojima je obučan, postoji opasnost da generiranim odgovorima nedostaje preciznosti ili da budu potpuno pogrešni. Ključno je to imati na umu, osobito u situacijama u kojima se alati poput Copilota koriste za rješavanje jedinstvenih i nekonvencionalnih izazova programiranja. Jedna potencijalna prepreka na koju bi korisnici mogli naići odnosi se na ograničenu pomoć dostupnu za novi ili nekonvencionalni kod. Alati kao što su Reini ili Copilot mogu imati problema s točnim razumijevanjem ili ponuditi precizne preporuke za kod koji bitno odstupa od standardnih primjera. Kao posljedicu, to može stvoriti komplikacije pri radu s iznimnim tehnologijama, jezicima ili konceptima. Ovo postaje osobito ključno kad se suoči s novim, zamršenim ili nekonvencionalnim programskim zahtjevima, prema istraživanju Reini (2021).

Tablica 3. Prednosti upotrebe Tabnine prema autorima i istraženom relevantnoj literaturi

AUTOR:	REZULTAT ISTRAŽIVANJA - PREDNOSTI:
Ramachandran, A. (2023)	- vrlo brz
Reini, N. (2021)	- poboljšava produktivnost - poboljšava kvalitetu koda - pruža podršku za različite stilove kodiranja

(Izvor: obrada autora)

Prema istraživanjima koja su proveli navedeni autori iz tablice 3., može se uvidjeti da Tabnine poboljšava produktivnost i kvalitetu koda, pritom radeći velikom brzinom.

6.2 Nedostaci ChatGPT-a, Copilota i Tabninea

1. ChatGPT

Inovativni pristup koji ima potencijalne izazove, ponajprije pojavu halucinacija podataka, pri čemu umjetna inteligencija ima sposobnost izmišljanja nepostojećih informacija (Rahman et al., 2023). Nadalje, pretjerano oslanjanje na ovu metodu rješavanja problema može nenamjerno umanjiti sposobnost učenika za samostalno razmišljanje i rješavanje problema.

Iako ChatGPT posjeduje sposobnost generiranja rješenja, neophodno je priznati inherentna ograničenja u spektru dostupnih rješenja, u svojem istraživanju ističe Stojkovski (2023). Nadalje, postoji potencijalna opasnost od pogrešaka unutar generiranog koda koje može biti teško identificirati i ispraviti. Kako bi se osigurali optimalni rezultati, važno je da korisnici pažljivo formiraju i definiraju svoje upite s najvećom preciznošću.

Na mogućnost pogrešnih davanja odgovora ChatGPT-a, upozorava Johnson (2023). Štoviše, postoji opasnost od nenamjernog repliciranja postojećih rješenja čime se ugrožava originalnost generiranog koda.

Konstantinos et al. (2023) oštroumno priznaju potencijalne izazove koji se mogu pojaviti pri korištenju ovog alata za rješavanje upita. Nadalje, strastveno naglašavaju neizostavnu ulogu ljudskog elementa u programiranju jer sama umjetna inteligencija nije u stanju u potpunosti zamijeniti urođene kvalitete ljudske intuicije, kreativnosti i razumijevanja zamršenih konteksta i koncepata.

Korištenje ChatGPT-a ima potencijal zadirati u urođenu kreativnost korisnika jer uvelike ovisi o unaprijed postavljenim predlošcima (Bendell, 2022). Nadalje, nedostatak sposobnosti stjecanja i prilagođavanja novim okolnostima može predstavljati značajnu prepreku u području učinkovitog rješavanja problema.

Privatnost podataka koji se mogu izgubiti korištenjem Chat GPT-a objašnjava Siregar (2023) dok izvanredne sposobnosti umjetne inteligencije imaju sposobnost suptilno utjecati ili čak oblikovati kognitivne procese učenika. Imperativ je priznati potencijalne strahove koji se mogu pojaviti kad je u pitanju točno razumijevanje uvida koje donose ovi napredni sustavi.

Postoje velike mogućnosti u nerazumijevanju zahtjeva koji studenti postavljaju ChatGPT-u jer ne može u potpunosti razumjeti različite jezike kojima korisnici pišu. Alat, iako izvanredan, može naići na određena ograničenja prilikom dešifriranja zamršenih upita i pružanja rješenja s najvećom preciznošću (Bansal, 2023).

Ograničenja ChatGPT-a, naglašava Harve (2023) bi bila da se nepostojanje najboljeg rješenja za jedan određeni zadatak ne bi trebalo pogrešno tumačiti kao jamstvo učinkovitosti u rješavanju drugih dodijeljenih zadataka. Ograničena univerzalnost ovog alata ima potencijal smanjiti njegovu korisnost u različitim scenarijima.

Tablica 4. Nedostaci upotrebe ChatGPT-a prema autorima i istraženju relevantnoj literaturi

AUTOR:	REZULTAT ISTRAŽIVANJA - Nedostaci:
Rahman et al. (2023)	<ul style="list-style-type: none"> - oslanjanje na chat u rješavanju zadataka - haluciniranje podatka - oslabljuje način rješavanja problema kod studenata
Stojkovski, B. (2023)	<ul style="list-style-type: none"> - malen raspon dostupnih rješenja - mogućnost nastanka greške tijekom generiranja koda koju je teško pronaći i ukloniti - pitanja moraju biti točno definirana za dobivanje najboljeg odgovora
Johnson, H. (2023)	<ul style="list-style-type: none"> - netočnost u rješavanju problema koje zadaju korisnici - kod može biti plagiran
Konstantinos, I. et al. (2023)	<ul style="list-style-type: none"> - poteškoće u odgovoru na pitanje - ne može zamijeniti ljudski faktor u programiranju
Bendell, L. (2022)	<ul style="list-style-type: none"> - prijetnja kreativnosti korisnika - nedostatak sposobnosti učenja
Siregar et al. (2023)	<ul style="list-style-type: none"> - problem s privatnošću podataka - mogućnost da umjetna inteligencija manipulira ili utječe na mišljenje studenata
Bansal, Y. (2023)	<ul style="list-style-type: none"> - programu je otežano razumijevanje korisnikova prirodnoga jezika pomoću kojega mu postavlja pitanje te ne može odgovoriti točno
Harve, A. (2023)	<ul style="list-style-type: none"> - nedostatak pri obavljanju jednog zadatka ne može rješavati druge zadatke koji su mu postavljeni

(Izvor: obrada autora)

Prema istraživanjima koja su proveli navedeni autori, iz tablice 4. može se vidjeti da postoje greške koje se mogu dogoditi tijekom dobivanja odgovora ChatGPT-a. Postoji i veliki problem haluciniranja podatka te narušavanja privatnosti korisnika.

2. Copilot

Prema stručnjaku Wermelingeru (2023), prepoznao je brojne potencijalne prepreke koje se mogu pojaviti pri korištenju ovih alata. Jedan takav izazov leži u ograničenoj kompatibilnosti alata s drugim razvojnim okruženjima što može spriječiti njegovu integraciju u trenutne tijekove rada programera. Budući da pitanje transparentnosti u vezi s potencijalnim slučajevima plagijata predstavlja veliku zabrinutost, programeri ne mogu znati može li generirani kod doista biti originalan ili je plagijat.

Jedna od početnih prepreka s kojom se susreću korisnici alata Copilot leži u nepreciznim prijedlozima koje on generira. Unatoč predviđenoj sposobnosti alata da precizno razazna namjere razvojnog programera i pruži relevantne prijedloge, on često daje preporuke koje nisu u potpunosti točne ili prikladne. Posljedično, ova pojava može rezultirati gubitkom neprocjenjivog vremena jer razvojni programer nastoji ispraviti pogrešne prijedloge i postići željeni ishod. Alat Copilot, iako učinkovit u generiranju koda na temelju specifičnih inputa i zahtjeva dovodi do gubitka izvornog iskustva rješavanja problema. Njegova upotreba često udaljava programere od zamršenosti procesa rješavanja problema, ometajući njihovo razumijevanje temeljne složenosti. Posljedično, to otežava napredak razvoja koda i održavanja, čineći ih teškim zadacima (Prather et al., 2023).

Paralelno, Copilot nailazi na vlastiti niz prepreka unatoč svojim izvanrednim prednostima. Mogućnost ograničenog razumijevanja zamršenog koda predstavlja rizik od stvaranja pogrešnih ili neprikladnih preporuka koje bi potencijalno mogle programere dovesti u zabludu i ugroziti integritet koda. Budući da postupak provjere generiranog koda predstavlja dodatnu složenost, alat povremeno može generirati netočnosti koje zahtijevaju precizno ručno ispitivanje i ispravljanje. Mora se priznati zabrinjavajući aspekt koji se odnosi na korištenje Copilota jer posjeduje sposobnost nenamjernog promicanja plagiranja. Dok se ovaj alat oslanja na velika spremišta postojećeg koda, postoji

mogućnost da programeri nenamjerno ili namjerno koriste generirani kod bez odgovarajuće potvrde ili razumijevanja. Takva okolnost stvara etičke nedoumice i pravne komplikacije što dovodi do bojazni u pogledu izvornosti i autentičnosti kodeksa govori Harvard (2023).

Ako se nije adekvatno objasnio problem za koji se želi da ga alat ispravi ili poboljša, može se očekivati zbrkan i nečitljiv kod. Teži zadaci, kompleksniji od postavljanja početka projekta, mogu rezultirati time da Copilot nema dovoljno znanja za rješenje određenog problema (Arsovski, 2022).

Prema istraživanjima koja su proveli navedeni autori iz tablice 5., može se uočiti da GitHub Copilot ima probleme vezane za davanje točnih odgovora zbog toga što nije učen na dovoljno velikom skupu podataka. Također, često se može kopirati tuđe rješenje koje je pronašao i to može studentima jako loše izgledati.

Tablica 5. Nedostaci upotrebe GitHub Copilota prema autorima i istraženju relevantnoj literaturi

AUTOR:	REZULTAT ISTRAŽIVANJA - NEDOSTACI:
Wermelinger, M. (2023)	<ul style="list-style-type: none"> - potencijalna ograničena kompatibilnost s ostalim IDE-ovima - nedostatak transparentnosti ne daje do znanja korisniku je li kod možda plagijat
Arsovski, A. (2022)	<ul style="list-style-type: none"> - kod koji generira često može postati zbrkan - ponekad ne razumije kontekst vašeg upita - ograničen samo na prijedloge kodova i ne može raditi izvan granica kreacije svog kreatora (OpenAI)
Prather, J. et al. (2023)	<ul style="list-style-type: none"> - netočni prijedlozi - gubitak izvornog rješavanja problema - učinkovitost Copilota može biti ograničena na određene vrste programskih problema
Harvard et al., (2023)	<ul style="list-style-type: none"> - potencijalni nedostatak u razumijevanju koda - izazovi validacije (netočan kod) - može izazvati zabrinutost u vezi s plagijatom

3. Tabnine

Loša kvaliteta koda u većini zadataka prema Ramachandranu (2023) može imati poteškoća u generiranju visokokvalitetnog koda za većinu zadataka. To znači da generirani kodovi mogu biti nepotpuni, neoptimizirani ili čak pogrešni, stoga je potrebno dodatno provjeriti i prilagoditi generirani kod prije konačne primjene. Jedna od teškoća koje proizlaze iz ovog okvira uključuje mogućnost davanja netočnih odgovora.

Ako alat naiđe na kod koji značajno odstupa od primjera na kojima je obučan, postoji opasnost da generiranim odgovorima nedostaje preciznosti ili da budu potpuno pogrešni. Ključno je to imati na umu, osobito u situacijama u kojima se alati poput Copilota koriste za rješavanje jedinstvenih i nekonvencionalnih izazova programiranja. Jedna potencijalna prepreka na koju bi korisnici mogli naići odnosi se na ograničenu pomoć dostupnu za novi ili nekonvencionalni kod. Alat kao što je Copilot može imati problema s točnim razumijevanjem ili ponuditi precizne preporuke za kod koji bitno odstupa od standardnih primjera. Kako bi se u potpunosti iskoristio potencijal ovih alata, programerima je imperativ priznati njihova ograničenja i marljivo proučiti kod koji se generira. Ovo postaje osobito ključno kada se netko suoči s novim, zamršenim ili nekonvencionalnim programskim zahtjevima (Reini, 2021).

Tablica 6. Nedostaci upotrebe Tabninea prema autorima i istraženju relevantnoj literaturi

AUTOR:	REZULTAT ISTRAŽIVANJA - NEDOSTACI:
Ramachandran, A. (2023)	- loša kvaliteta koda u većini zadataka
Reini, N. (2021)	- ovisi o podacima na kojima je obučan - potencijalno može pružiti netočne odgovore - ograničena podrška za novi ili neuobičajeni kod

(Izvor: obrada autora)

Prema istraživanjima koja su proveli navedeni autori iz tablice 6., može se uvidjeti da problemi oko kvalitete i točnosti koda kod Tabnine alata ovise o veličini podataka koje sam alat posjeduje.

7. Istraživanje percepcije studenata prema korištenju alata za učenje programiranja podržani umjetnom inteligencijom

7.1 Metodologija istraživanja

U okviru ovog poglavlja definirani su cilj i metode empirijskog istraživanja provedenog za potrebe ovog diplomskog rada. Zatim je opisana procedura i korišten anketni upitnik te opće karakteristike ispitanika.

7.1.1 Cilj i metode istraživanja

U ovom radu glavni cilj istraživanja bio je ispitati percepciju studenata na Fakultetu informatike u Puli te njihovo stajalište prema korištenju alata za učenje programiranja podržani umjetnom inteligencijom. Ovim istraživanjem nastojat će se uvidjeti stajalište studenata prema alatima za učenje programiranja. Prema istraženju literaturi, studentima će biti postavljeno par pitanja prednosti i mana alata za učenje programiranja i prema tim podacima će se vidjeti cijela korisnost istih. Korištena je metoda anketiranja za prikupljanje svih podataka. Metoda anketiranja je postupak kojim se na temelju anketnog upitnika istražuju i prikupljaju podaci, informacije, stavovi i mišljenja o predmetu istraživanja (Radeka, 2018).

7.1.2 Procedura i anketni upitnik u istraživanju

Istraživanje se provodilo u periodu od 7. rujna 2023. god. do 22. rujna 2023. god. pomoću online anketnog obrasca. Istraživanje je provedeno putem ankete koja je napravljena u poznatom Google alatu (Google obrasci) za izradu obrazaca. Podatci su prikupljeni pomoću društvenih medija povezanih s Fakultetom informatike u Puli. Ispitanici su upitnik mogli rješavati putem poveznice koja je bila dostupna putem Facebooka u grupi Informatičara. Također, isti upitnik je poslan putem WhatsApp-a, Vibera, Slacka. Excel datoteka je preuzeta nakon što se provelo istraživanje u kojem su svi podatci uređeni i grafički prikazani pomoću alata Microsoft Excel. Tvrdnje za prednosti prilagođene su na osnovi istražene literature prema autorima Stojkovski (2023), Bendell (2022), Siregar et al. (2023). i Jeevanandam (2023). Tvrdnje vezane za nedostatke navedene su prema Rahman et al (2023), Bansal (2023), Siregar et al. (2023).

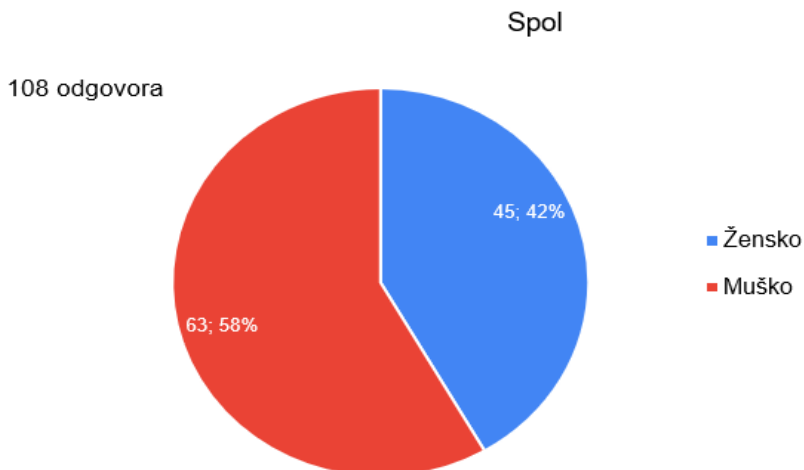
Anketni upitnik sastoji se od par pitanja zatvorenog tipa koja su imala ponuđene odgovore za odabir. Postoji pet sociodemografskih pitanja u ovom anketnom upitniku

(spol, dob, smjer, godina studija, status studenta). Nakon ispunjavanja osnovnih sociodemografskih pitanja postavljeno je još pet osnovnih pitanja vezanih za upotrebu alata za učenje programiranja podržane umjetnom inteligencijom. Pitanja su bila vezana za korištenje takvih alata, koliko često se isti koriste, u koju svrhu se koriste, u kojoj mjeri se oslanjanju na pomoć od strane istih alata te na kraju koliko se smatraju vještima u korištenju istih alata.

Upitnik je sadržavao Likertovu ljestvicu s pet stupnjeva (1 – potpuno netočno, 2 – uglavnom netočno, 3 – niti netočno, niti točno, 4 – uglavnom točno, 5 – potpuno točno) na temelju koje su ispitanici dali svoje mišljenje o prednostima i nedostacima korištenja alata za učenje programiranja. Na kraju su dodane tri tvrdnje koje ispituju studente namjeravaju li u budućnosti koristiti alate za učenje programiranja.

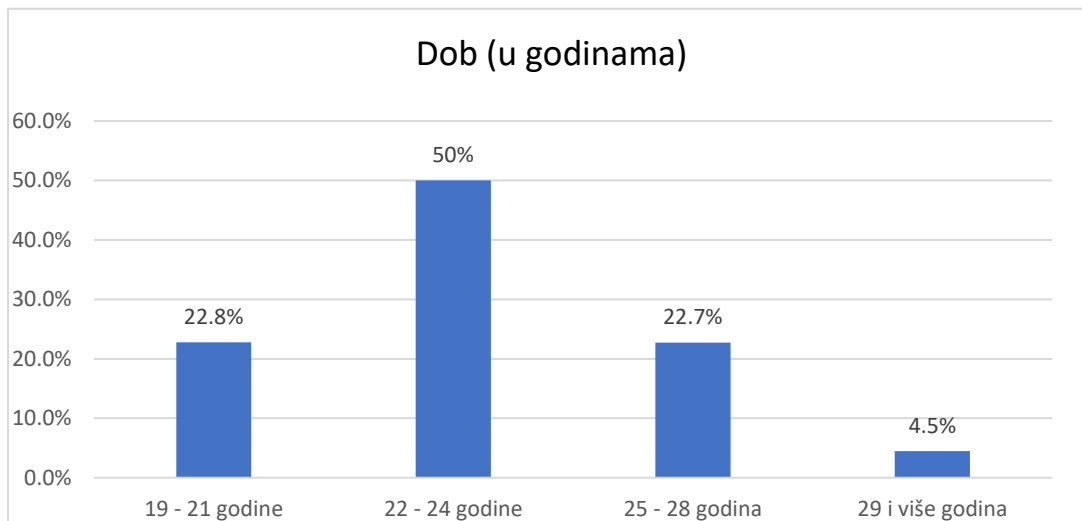
7.1.3 Uzorak ispitanika

Provedeno istraživanje sadržavalo je 108 ispitanika studenata na fakultetu. Istraživanje je bilo provedeno na Fakultetu informatike u Puli te su svi sudionici u anketi bili studenti. Prema rezultatima vidljivih iz grafikona na slici 7. može se uvidjeti da je veći postotak muških ispitanika 63 (58%) riješilo anketu te 45 (42%) ženskih ispitanika.



Slika 7. Broj ispitanika obzirom na spol (izvor: Obrada autora)

Prema rezultatima koji prikazuju dob ispitanika (108), iz grafikona na slici 8 može se primijetiti da je sudjelovalo najviše ispitanika u dobi od 22 do 24 godine; njih 55 (50%). Zatim, postoje dvije skupine koji imaju podjednak broj ispitanika, a to su u rasponu od 19 do 21 godinu i 25 do 28 godina, svatko od njih ima 24 (22.8%) ispitanika. Na kraju, 5 (4.5%) ispitanika pripalo je u najmanju skupinu ispitanika od 29 i više godina.

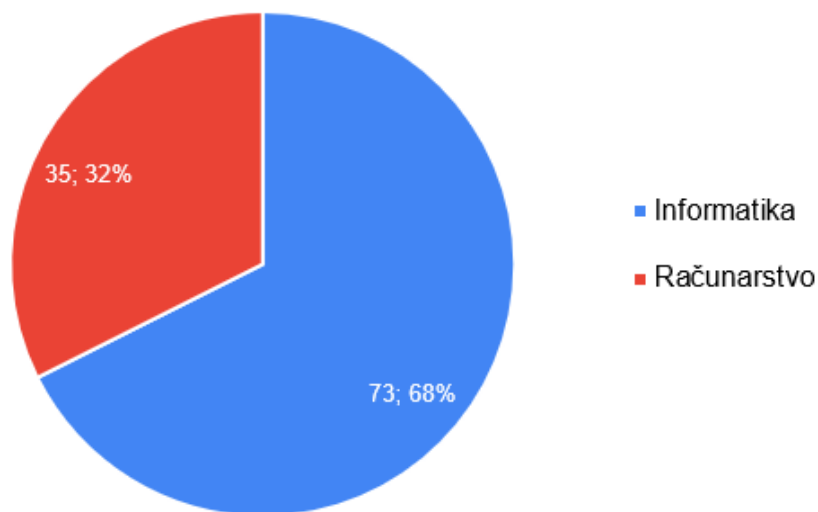


Slika 8. Postotak ispitanika obzirom na dob (izvor: obrada autora)

Grafikon na slici 9 prikazuje omjer studenata s različitih smjerova, u ovom slučaju smjer Informatika i Računarstvo. Prema grafikonu, može se uvidjeti da 73 (68%) ispitanika pohađa smjer Informatika te 35 (32%) ispitanika pohađa smjer Računarstvo.

Koji studijski smjer pohađate ?

108 odgovora

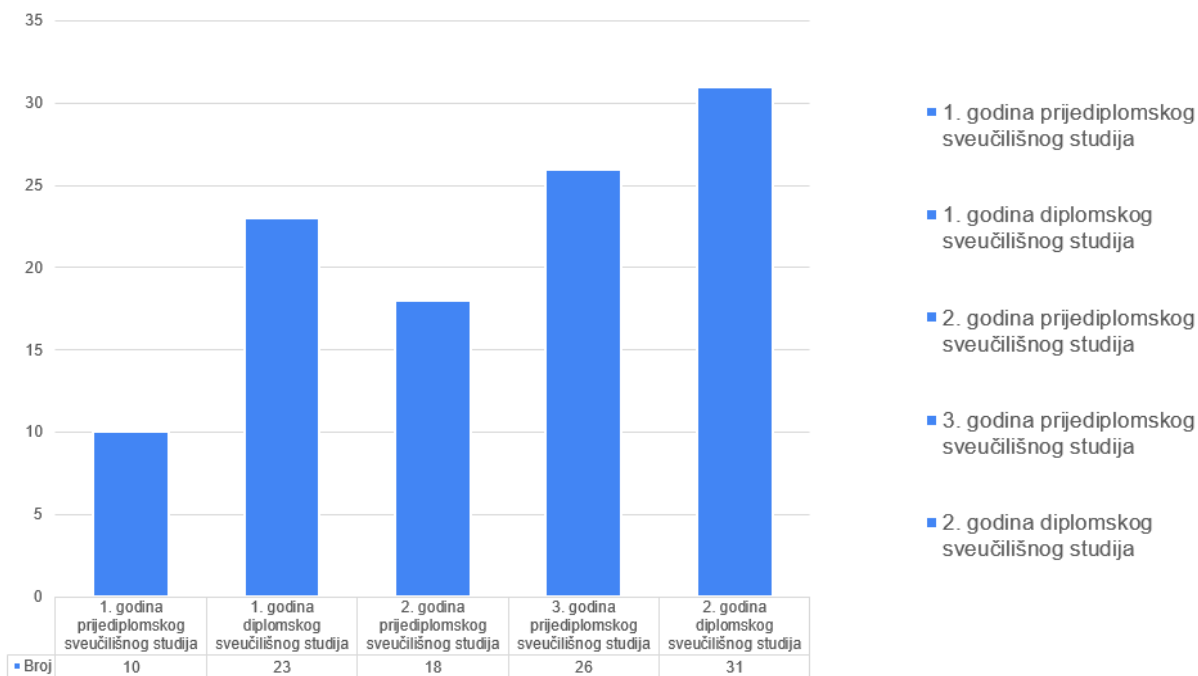


Slika 9. Broj ispitanika obzirom na studijski smjer koji pohađaju (izvor: obrada autora)

Grafikon na slici 10 služi za prikaz godine studija koje pohađaju studenti. Može se uvidjeti da su rezultati podjednaki, što znači da je polovica od ukupnih ispitanika na diplomskom sveučilišnom studiju njih 54 (50%), a ostala polovica na prijediplomskom sveučilišnom studiju njih 54 (50%).

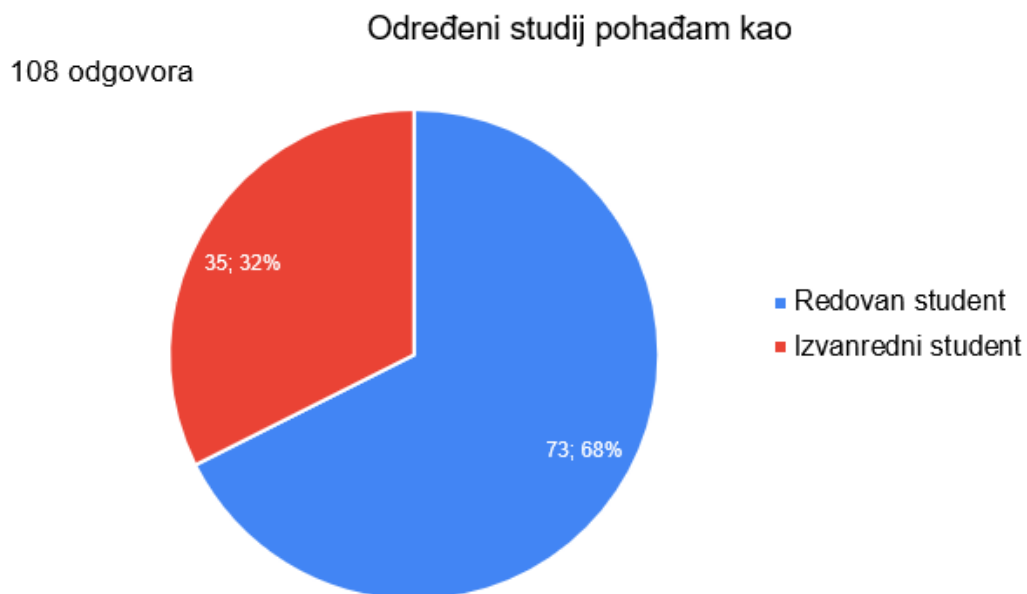
Koja ste godina studija?

108 odgovora



Slika 10. Postotak ispitanika s obzirom na godinu studija (izvor: obrada autora)

Određeni studij koji pohađaju studenti prikazan je na slici 11. Grafikon prikazuje da su 73 (68%) ispitanika redovni studenti, dok 35 (32%) ispitanika ima status izvanrednih studenata.



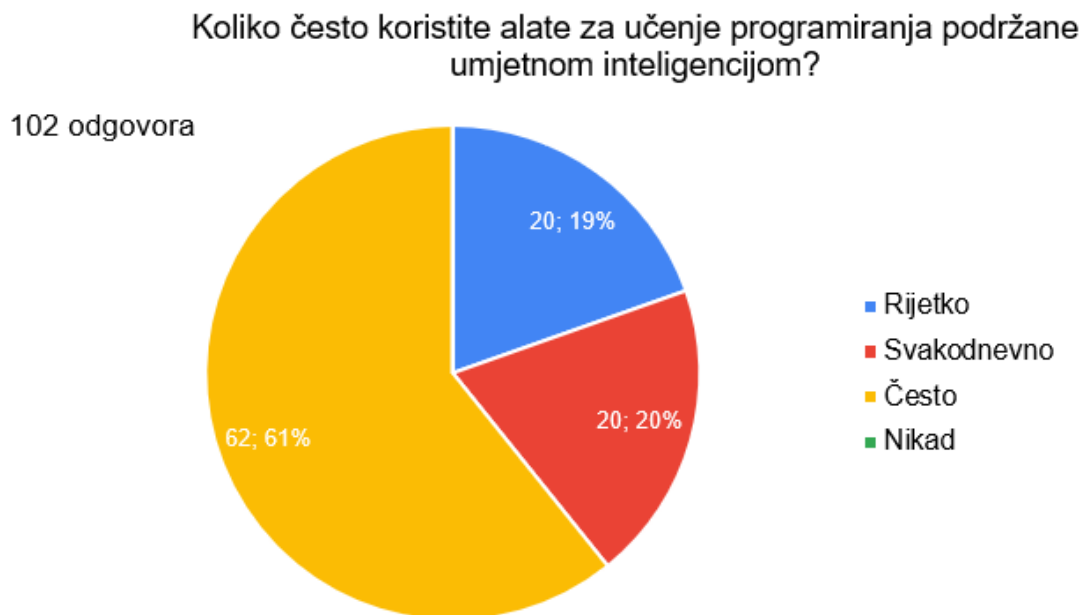
Slika 11. Postotak ispitanika prema vrsti studija kojeg pohađaju (izvor: obrada autora)

Korištenje alata podržani umjetnom inteligencijom za učenje programiranja prikazani na slici 12. pružaju uvid da veliki broj ispitanika, čak 102 (94%) koristi takve alate dok 6 (6%) ne koristi takve alate. Prema dobivenim odgovorima isključujemo studente točnije njih 6 (6%) koji ne koriste alate podržane umjetnom inteligencijom u daljnjem istraživanju te se fokusiramo na sve one koji koriste alate za učenje programiranja dakle njih 102 (94%).



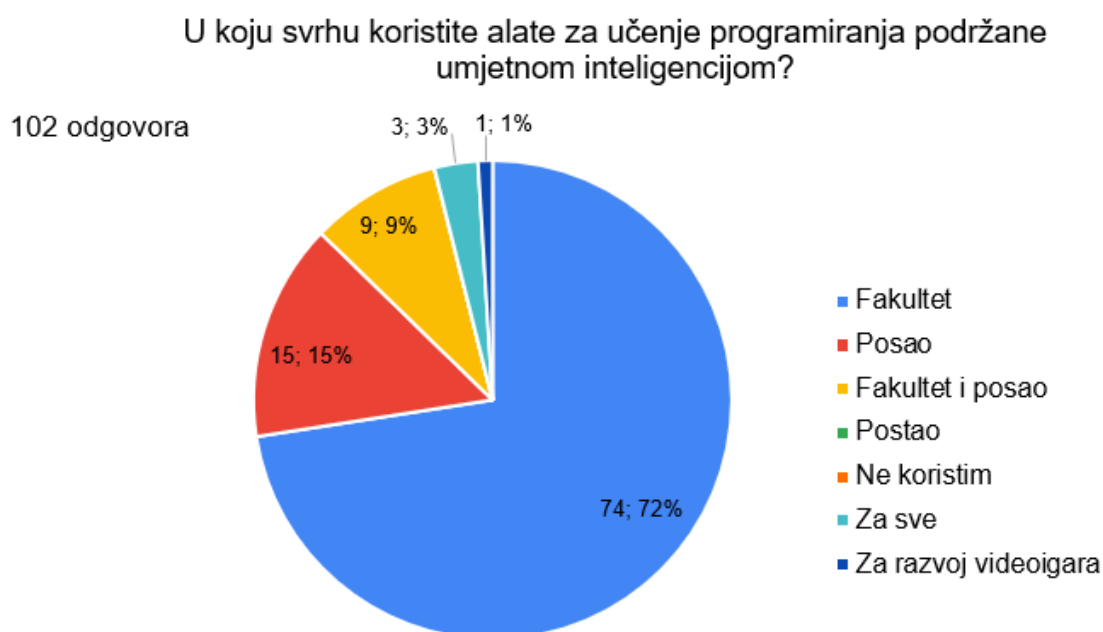
Slika 12. Postotak ispitanika koji koriste alate podržane umjetnom inteligencijom za učenje programiranja (izvor: obrada autora)

Koliko često se koriste alati za učenje programiranja podržani umjetnom inteligencijom prikazuje grafikon na slici 13. Prema grafu se može uvidjeti da većina ispitanika, njih 62 (61%) česti su korisnici alata za pomoć pri programiranju. Rijetko korištenje alata koristi njih 20 (19%). Svakodnevno korištenje alata za učenje programiranja vidljivo je kod 20 (12%) ispitanika.



Slika 13. Postotak ispitanika s obzirom na korištenje alata za učenje programiranja podržane umjetnom inteligencijom (izvor: obrada autora)

Postotak podataka vezanih za svrhu korištenja alata za učenje programiranja podržane umjetnom inteligencijom prikazani su na slici 14. Najveći broj ispitanika, njih 74 (72%) koristi alate za učenje programiranja u svrhu pomoći na fakultetu. Podatak koji je vezan u svrhu posla napisalo je 15 (15%) ispitanika, dok je za kombinaciju fakulteta i posla odgovorilo je 9 (9%). Ostali ispitanici su odgovorili da koriste alate za sve navedeno 3 (3%) te samo 1 (1%) ispitanik koristi alate u svrhu razvoja videoigara.

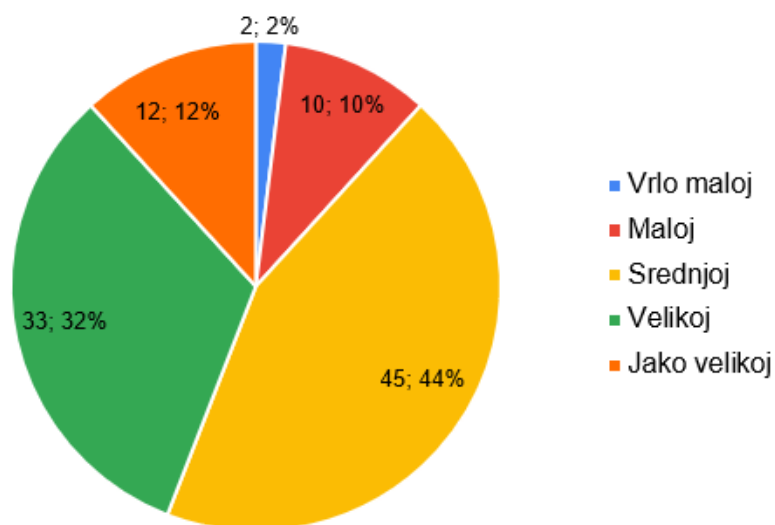


Slika 14. Postotak ispitanika za svrhu korištenja alata za učenje programiranja podržane umjetnom inteligencijom (izvor: obrada autora)

Oslanjanje na pomoć od strane alata za učenje programiranja podržani umjetnom inteligencijom vidljivo je iz grafa na slici 15. Može se uvidjeti da se 45 (44%) ispitanika oslanja u prosječnoj mjeri na pomoć alata. Nadalje, 33 (32%) njih smatra da se u velikoj mjeri oslanjaju na pomoć alata, 12 (12%) ispitanika smatra da se u jako velikoj mjeri oslanja na pomoć od strane alata. Ostali broj ispitanika, njih 10 (10%) smatra da se oslanjaju u maloj mjeri te ostalih 2 (2%) da se malo oslanja na pomoć od strane umjetne inteligencije.

U kojoj mjeri se oslanjate na pomoć od strane alata za učenje programiranja podržane umjetnom inteligencijom?

102 odgovora



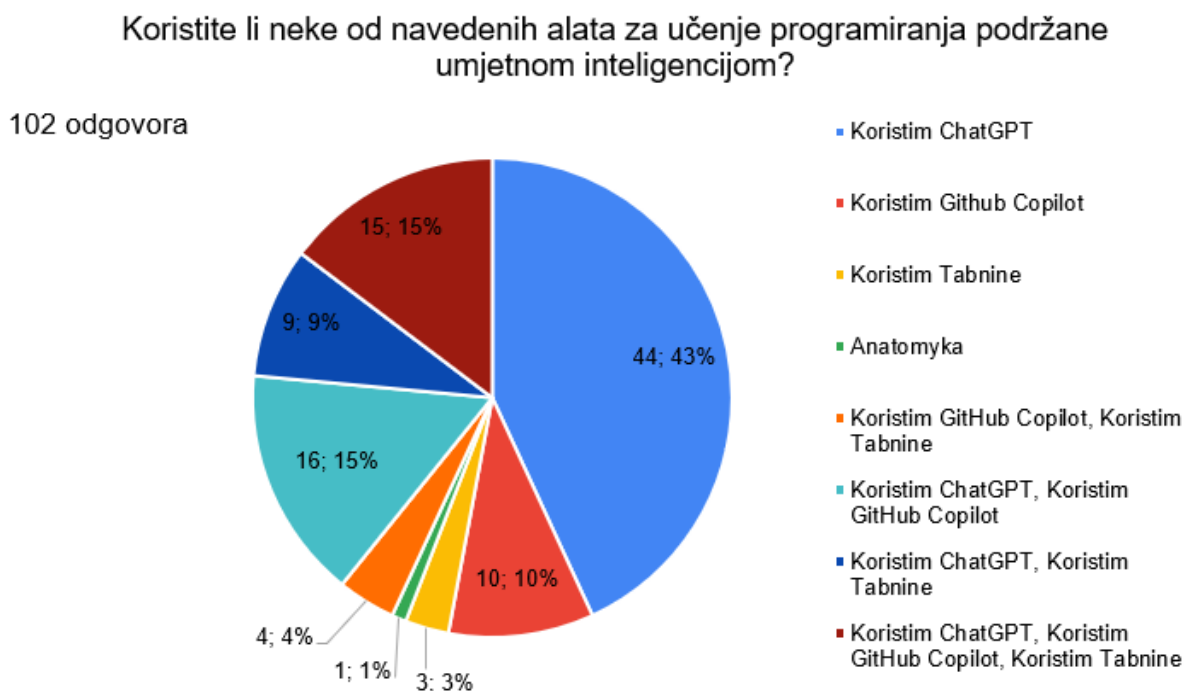
Slika 15. Postotak ispitanika u svrhu oslanjanja na pomoć od strane alata za učenje programiranja podržani umjetnom inteligencijom (izvor: obrada autora)

Vještina korištenja alata za učenje programiranja podržane umjetnom inteligencijom vidljivo je iz grafikona na slici 16. Većina ispitanika, njih 62 (61%), smatra da ima veću vještinu znanja prilikom korištenja alata za pomoć pri programiranju, nego 24 (23%) ispitanika koji se smatraju dobro vještima. Dovoljno vještima odgovorilo je 14 (14%), a samo 2 (2%) ispitanika smatra da su nedovoljno vješti u korištenju alata za pomoć pri programiranju.



Slika 16. Postotak ispitanika obzirom na vještinu korištenja alata za učenje programiranja podržani umjetnom inteligencijom (izvor: obrada autora)

Rezultate vezane za korištenje istraženih alata za učenje programiranja možemo vidjeti iz grafikona na slici 17. ChatGPT se koristi uvelike te je priličan broj ispitanika postavilo da ga koristi, njih 44 (43 %). GitHub Copilot koristi 10 (10 %) studenata te na alat Tabnine koristi 3 (3%) ispitanika. Kombinacija korištenja GitHub Copilota i Tabnina koristi 4 (4 %) ispitanika, korištenje ChatGPT i GitHub Copilota iskazalo je 16 (15 %) ispitanika. Korištenje ChatGPT i Tabnina iskazalo je njih 9 (9 %) ispitanika te na kraju korištenje svih alata iskazalo je njih 15 (15 %) ispitanika.



Slika 17. Postotak ispitanika vezanih za korištenje određenih alata za učenje programiranja podržane umjetnom inteligencijom (izvor: obrada autora)

8. Rezultati istraživanja

Tablice u nastavku služe za prikazivanje postotka ispitanika te uz njih broj na tvrdnje koje su odgovorili na mjernoj skali od 1 – potpuno netočno do 5 – potpuno točno.

Tablica 7. Percepcija ispitanika vezanih uz odgovore na odabrane tvrdnje u anketnom upitniku uz prednosti upotrebe alata za učenje programiranja podržane umjetnom inteligencijom (frekvencija, postotci (%) ispitanika na skali od 1 - potpuno netočno do 5 - potpuno točno, M - arit. sredina, σ - st.dev.

TVRDNJA - PREDNOSTI	FREKVENCIJA I POSTOTCI (%) ISPITANIKA (N = 102)						
	1	2	3	4	5	M	σ
P1. Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) omogućavaju bolju produktivnost. (Stojkovski,2023)	0 0%	2 2%	26 25%	56 55%	18 18%	3.88	0.71
P2. Smatram da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) omogućavaju brz pristup informacijama. (Jeveendam)	0 0%	3 3%	17 17%	51 50%	31 30%	4.08	0.77
P3. Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) povećavaju motivaciju za učenje programiranja. (Siregar et al. 2023)	0 0%	8 8%	39 38%	41 40%	14 14%	3.60	0.82
P4. Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) pružaju korisniku potporu u bilo koje doba dana. (Bendel 2023)	0 0%	0 0%	17 17%	48 47%	37 37%	4.20	0.70
P5. Smatram da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) olakšavaju razumijevanje koda putem jednostavnih primjera. (Siregar et al. 2023)	0 0%	5 5%	26 25%	46 45%	25 25%	3.89	0.83

Izvor: obrada autora rada (27.09.2023)

Početna tvrdnja (P1) glasi: „Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) omogućavaju bolju produktivnost.“ Prema provedenom istraživanju većina ispitanika se slaže s ovom tvrdnjom, njih 55% smatra da je tvrdnja uglavnom točna i 18% smatra da je tvrdnja u potpunosti točna. Nitko se ne slaže da je tvrdnja u potpunosti netočna, dok samo 2% ispitanika smatra tvrdnju uglavnom netočnom. Na kraju, 25% ispitanika je neodlučno u odgovoru. Prema provedenom istraživanju, za ovo pitanje prosjek iznosi 3.88, a standardna devijacija 0.71.

Sličan slučaj se može vidjeti i na tvrdnji (P2) koja glasi: „Smatram da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) omogućavaju brz pristup informacijama. “Također, u ovoj tvrdnji se većina uglavnom slaže s ovom tvrdnjom, njih 50% te s njima potpunu cjelinu čine i oni koji su postavili da se u potpunosti slažu, a to je 30%. Postoji manji broj ispitanika koji je suzdržan, njih 17%. Na kraju samo 3% smatra da je tvrdnja uglavnom netočna. Prema prosjeku ovog pitanja, ono iznosi 4.08, dok je standardna devijacija 0.77.

Sljedeća tvrdnja (P3) glasi: „Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) povećavaju motivaciju za učenje programiranja“. Većina je pokazala da je ova tvrdnja točna, točnije da potiče motivaciju za učenje, njih 40% reklo je da tvrdnja uglavnom točna uz 14% njih koji se slažu u potpunosti. Također, oko 38% nalazi se između, dok 8% smatra da je tvrdnja uglavnom netočna. Prema ukupnom istraživanju prosjek je 3.60, a standardna devijacija iznosi 0.82.

Najveći prosjek, 4.20 vezan za tvrdnje o prednosti ima tvrdnja (P4) koja glasi: „Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) pružaju korisniku potporu u bilo koje doba dana.“ Ovdje se velika većina uglavnom složila da je tvrdnja točna, njih 47% uz 37% njih koji smatraju tvrdnju potpuno točnom. Kao iz tvrdnje (P2), ima skoro pa identičan postotak onih koji se nalaze između, a to je njih 17%. Na kraju, standardna devijacija iznosi 0.70.

Zadnja tvrdnja prednosti (P5) glasi: „Smatram da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) olakšavaju razumijevanje koda putem jednostavnih primjera.“ Ovdje možemo uvidjeti da je opet

većina njih, 45% smatra tvrdnju uglavnom točnu i uz njih se vežu i 25% koji smatraju da je tvrdnja također u potpunosti točna. Ovdje imamo 25% njih koji nisu ni za jednu stranu, dok 5% njih smatra da je tvrdnja uglavnom netočna. Prosjek prema ovoj tvrdnji iznosi 3.89, a standardna devijacija je 0.83.

Nastavlja se druga tablica koja služi za prikazivanje nedostataka na koje su ispitanici odgovarali na mjernoj skali od potpuno netočno – 1 do potpuno točno – 5.

Tablica 8. Percepcija ispitanika vezani uz odgovore na odabrane tvrdnje u anketnom upitniku uz prepreke upotrebe alata za učenje programiranja podržane umjetnom inteligencijom (frekvencija, postotci (%) ispitanika na skali od 1 - potpuno netočno do 5 - potpuno točno, M - arit. sredina, σ - std.

TVRDNJA - NEDOSTACI	FREKVENCIJA I POSTOTCI (%) ISPITANIKA (N = 102)						
	1	2	3	4	5	M	σ
P1. Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) potiču ovisnost o potrebi korištenja istih. (Rahman et al. 2023).							
	2 2%	9 9%	33 32%	40 39%	18 18%	3.62	0.94
P2. Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) ne mogu odgovoriti točno na postavljeno pitanje ako mu se ne postavi cjelokupan kontekst. (Yash Bansal)							
	0 0%	6 6%	30 30%	37 36%	29 28%	3.87	0.90
P3. Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) mogu narušiti privatnost putem neovlaštenog prikupljanja osobnih podataka. Siregar et al. (2023)							
	2 2%	11 11%	37 36%	31 30%	21 21%	3.57	1.00
P4. Smatram da alate za učenje programiranja podržane umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) namjeravam tijekom učenja programiranja koristiti u budućnosti. (Autor)							
	1 1%	4 4%	11 11%	38 37%	48 47%	4.25	0.88

P5. Smatram da alate za učenje programiranja podržane umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) namjeravam više koristiti u budućnosti. (Autor)	2 2%	2 2%	24 24%	40 39%	34 33%	4.00	0.91
P6. Pomisao na korištenje alata podržane umjetnom inteligencijom u meni potiče negativne osjećaje. (Autor)	33 32%	18 18%	14 14%	12 12%	25 24%	2.78	1.60

Izvor: obrada autora rada (27.09.2023)

Započinje se s (P1) tvrdnjom koja glasi: „Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) potiču ovisnost o potrebi korištenja istih.“ Također, ovdje postoji većina koja priliježe na stranu da je uglavnom točna tvrdnja njih 39% uz 18% ispitanika koji se slažu u potpunosti čine većinu. 32% njih se niti slaže niti ne slaže s tvrdnjom, dok oni koji smatraju tvrdnju uglavnom netočnom, njih 9% uz 2% onih koji smatraju da je tvrdnja u potpunosti netočna. Prosjek na ovoj tvrdnji iznosi 3.62, a standardna devijacija je 0.94.

Točnost odgovora na koje se postavljaju pitanja поближе objašnjava (P2) tvrdnja koja glasi: „Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) ne mogu odgovoriti točno na postavljeno pitanje ako mu se ne postavi cjelokupan kontekst.“ Velika većina smatra da je ova tvrdnja točna u potpunosti i uglavnom njih čak 64%. Također 30% njih nije dobivalo češće lošije odgovore, te njih 6% smatra da je uglavnom netočna tvrdnja. Prosjek ove tvrdnje iznosi 3.87, a standardna devijacija je 0.90.

Tvrdnja (P3) koja glasi: „Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) mogu narušiti privatnost putem neovlaštenog prikupljanja osobnih podataka.“ Ovdje se može uvidjeti da su rezultati podjednaki, njih 21% smatra da je tvrdnja u potpunosti točna te njih 30% se slaže da je tvrdnja uglavnom točna. 36% ispitanika ne može se složiti niti da je tvrdnja točna niti da je netočna. Njih 11% tvrdi da je tvrdnja netočna. Za kraj, 2% smatra da je

tvrdnja u potpunosti netočna. Prosjek ove tvrdnje iznosi 3.57, a standardna devijacija je 1.00.

Zadnje tri tvrdnje odnose se na korištenje alata za učenje programiranja u budućnosti. Tvrdnja (P4) glasi: „Smatram da alate za učenje programiranja podržane umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) namjeravam tijekom učenja programiranja koristiti u budućnosti.“ Ovdje se može uvidjeti da će većina koristiti ove alate u budućnosti, njih 84% smatra da će sigurno koristiti u budućnosti, dok 11% nije još sigurna hoće li ih koristiti toliko. Mala količina, njih 5% smatra da ih neće koristiti u budućnosti. Trenutno najveći prosjek koji iznosi 4.25 uz standardnu devijaciju koja je 0.88.

Slična tvrdnja (P5) koja glasi: „Smatram da alate za učenje programiranja podržane umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) namjeravam više koristiti u budućnosti.“ Ova tvrdnja je u većini generalni odgovor na sve alate za pomoć pri programiranju. Opet, postoji većina koja se slaže da će ih koristiti u budućnosti, njih 72%, dok 24% još nije sigurna. Mali postotak, njih 4% smatra da ih neće koristiti. Prosjek ove tvrdnje iznosi 4.00 uz standardnu devijaciju koja iznosi 0.91.

Zadnja tvrdnja (P6) koja se odnosi na utjecaj korištenja alata za učenje programiranja glasi: „Pomisao na korištenje alata podržane umjetnom inteligencijom u meni potiče negativne osjećaje.“ Većina smatra da ova tvrdnja ne potiče negativne osjećaje prilikom korištenja, njih 50% te 14% nalazi se između određivanja kako se ono odnosi na njih. Također, njih 36% slaže se da potiče negativne osjećaje prilikom njihovog korištenja alata za pomoć pri programiranju. Prosjek na ovoj tvrdnji je 2.78 uz standardnu devijaciju koja iznosi 1.60.

9. Zaključak

Budućnost programiranja donosi obećavajuće inovacije i transformacije. Primjećuje se trend prema sveprisutnoj automatizaciji, gdje će programi postati još intuitivniji i sposobniji za učenje iz iskustava. Ova evolucija oblikovat će našu digitalnu stvarnost na načine koje možda još nismo mogli ni zamisliti. Učenje programiranja alatima podržanim umjetnom inteligencijom pruža izuzetne prednosti za brže i efikasnije usvajanje programerskih vještina. Ova inovativna metoda omogućava personalizirani pristup učenju, prilagođen individualnim potrebama i tempu svakog učenika. Kombinacija AI tehnologija i obrazovnih resursa obećava svijetlu budućnost za one koji žele postati stručnjaci u programiranju. Osim što omogućava učenje na vlastitim uvjetima, učenje programiranja uz pomoć umjetne inteligencije otvara prostor za kreativnost i inovacije. Simulacije i praktične primjene olakšavaju apstraktne koncepte i pružaju studentima konkretne alate za rješavanje stvarnih izazova u programiranju.

Empirijsko istraživanje provedeno u svrhu ovog diplomskog rada potvrdilo je mnoge tvrdnje koje podržavaju umjetnu inteligenciju, ali isto tako ukazuje da se ljudi previše oslanjaju na njezinu pomoć. Zaključak je da se utjecaj korištenja alata na negativne emocije mijenja s obzirom na period namjere korištenja.

Uz ove prednosti, ne smijemo zanemariti da učenje programiranja uz pomoć umjetne inteligencije doprinosi razvoju kritičkog razmišljanja, rješavanju problema i timskom radu. Sposobnost razumijevanja i primjene programerskih koncepata postaje ključna kompetencija u suvremenom profesionalnom okruženju, te je nužno razmotriti načine razvoja programerskih kompetencija kod studenata uz primjenu ChatGPT-a s tehničkog i pedagoškog aspekta te provesti dodatna istraživanja u različitim okruženjima.

10. Literatura

1. Resnick, M., Silverman, B., Kafai, Y., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E. and Silver, J. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), p.60. doi:<https://doi.org/10.1145/1592761.1592779>.
2. Hassinen, M. and Mäyrä, H. (2006). Learning programming by programming. *Proceedings of the 6th Baltic Sea conference on Computing education research Koli Calling 2006 - Baltic Sea '06*. doi:<https://doi.org/10.1145/1315803.1315824>.
3. Copeland, B.J. (2018). Artificial intelligence. *Encyclopædia Britannica*. URL: <https://www.britannica.com/technology/artificial-intelligence> (Pristupljeno: 14.04.2023.)
4. Passey, D. (2017). Computer science (CS) in the compulsory education curriculum: Implications for future research. *Education and Information Technologies*, 22(2), 421-443.
5. Friedman, J. (2021) *When was C# created? A brief history, C# Station*. Dostupno na: <https://csharp-station.com/when-was-c-sharp-created-a-brief-history/> (Pristupljeno : 29.10.2023).
6. Preston, R. (2023). A Complete Guide to Procedural Programming Language. [online] Indeed Career Guide. Available at: <https://www.indeed.com/career-advice/career-development/procedural-programming-language>.
7. Rao, V.S. and Technologies, M. (2022) Challenges that every programmer face while programming, *Top Challenges That Every Programmer Face (With Solution)*. Dostupno na: <https://www.technotification.com/2018/09/common-programming-challenges.html> (Pristupljeno: 14.04.2023.).
8. Gomes, A. and Mendes, A. (2007). Learning to program -difficulties and solutions. [online] Dostupno na : <https://www.ineer.org/Events/ICEE2007/papers/411.pdf>.
9. Peterson, W. D. (1990). *Introduction to Artificial Intelligence and Expert Systems*. New Jersey: Prentice Hall (Pristupljeno : 14.04.2023).
10. Walsh, Toby. (2017). *Android Dreams: The Past, Present and Future of Artificial Intelligence*. London: Hurst and company (Pristupljeno: 14.04.2023).

11. Wang P. (2019) On Defining Artificial Intelligence. Journal of Artificial General Intelligence, Vol.10 (Issue 2), pp. 1-37. (Pristupljeno: 14.04.2023)
12. Turing A. M. (1950). I.—COMPUTING MACHINERY AND INTELLIGENCE, <https://academic.oup.com/mind/article/LIX/236/433/986238> (pristupljeno 12.04.2023.)
13. Tyson, M. (2021). What is functional programming? A practical guide. [online] InfoWorld. Dostupno na: <https://www.infoworld.com/article/3613715/what-is-functional-programming-a-practical-guide.html>. (Pristupljeno: 14.04.2023).
14. Virtusa (2023). Logic Programming. [online] www.virtusa.com. Dostupno na: <https://www.virtusa.com/digital-themes/logic-programming>. (Pristupljeno: 14.04.2023).
15. Bethanny Parker, Nate Delesline III (2021). The Ultimate Guide to Scripting Languages | BestColleges. [online] Dostupno na: <https://www.bestcolleges.com/bootcamps/guides/ultimate-guide-to-scripting-languages/>. (Pristupljeno: 14.04.2023).
16. Doherty, E. (2020). What is Object Oriented Programming? OOP Explained in Depth. [online] Educative: Interactive Courses for Software Developers. Dostupno na: <https://www.educative.io/blog/object-oriented-programming>. (Pristupljeno: 14.04.2023).
17. Lilya Khimchenko (2023). 10 Most Common Problems New Programmers Face. [online] arounda.agency. Dostupno na: <https://arounda.agency/blog/10-most-common-problems-new-programmers-face>. (Pristupljeno: 14.04.2023)
18. Schroer, A. (2023) *Artificial Intelligence.*, BuiltIn. BuiltIn. Available at: <https://builtin.com/artificial-intelligence> (Pristupljeno: 14.04.2023).
19. Middleton, M. (2021) Deep Learning vs. machine learning, Flatiron School. Flatiron School. Dostupno na: <https://flatironschool.com/blog/deep-learning-vs-machine-learning/> (Pristupljeno: 14.04.2023).
20. Josephine Hart (2023). Top Characteristics of Artificial Intelligence You Must Know. [online] www.theknowledgeacademy.com. Dostupno na: <https://www.theknowledgeacademy.com/blog/characteristics-of-artificial-intelligence/> (Pristupljeno: 18.04.2023).

21. Majestic, S.G.S. (2023) Advantages of AI & Machine Learning in Software Development, Advantages of AI & Machine Learning in Software Development | SGS Technologie. SGS. Available at: https://www.sgstechnologies.net/blog/Advantages_of_AI_and_Machine_Learning_in_Software_Development (Pristupljeno: 18.04.2023).
22. Bansal, Y. (2023). The Limitations of Chat-GPT and the Need for Further Research and Development. [online] Medium. Dostupno na: <https://medium.com/@yashbansal42/the-limitations-of-chat-gpt-and-the-need-for-further-research-and-development-91bc6caae533> [Pristupljeno: 18.04.2023].
23. Stojkovski, B. (2023). Can ChatGPT Replace the Art of Programming? The Pros and Cons of Using AI as a Developer. [online] TheRecursive.com. Available at: <https://therecursive.com/chatgpt-in-programming-pros-and-cons/>.
24. Beladiya, K. (2022) Advantages: AI & Machine Learning in Software Development, Kovair Blog. Kovair. Available at: <https://www.kovair.com/blog/advantages-of-ai-and-machine-learning-in-software-development/> (Pristupljeno: 18.04.2023).
25. Pallavi, M. (2018) How developers can benefit from AI and machine learning, LinkedIn. LinkedIn. Dostupno na: <https://www.linkedin.com/pulse/how-developers-can-benefit-from-ai-machine-learning-pallavi> (Pristupljeno: 18.04.2023).
26. Brush, K. and Scardina, J. (2021) What is a chatbot and why is it important?, Customer Experience. Dostupno na: <https://www.techtarget.com/searchcustomerexperience/definition/chatbot> (Pristupljeno: 18.04.2023).
27. Sydle (2022) Chatbots for Education: What Are the Benefits for Your Institution? SYDLE, 14 November. Dostupno na: <https://www.sydle.com/blog/chatbots-for-education-what-are-the-benefits-634d4f67491b8e20c4b01486> (Pristupljeno: 18.04.2023).
28. Rebecca Lake (2023). What Is a Large Language Model (LLM)? [online] Dostupno na: <https://www.investopedia.com/large-language-model-7563532>. (Pristupljeno: 18.04.2023).

29. Winkler, R., and Söllner, M. 2018. "Unleashing the Potential of Chatbots in Education: A State-Of-The-Art Analysis," in Academy of Management Annual Meeting (AOM), Chicago, USA (Pristupljeno: 18.04.2023).
30. Oracle (2023) What is a chatbot?, Oracle. Oracle. Available at: <https://www.oracle.com/chatbots/what-is-a-chatbot/> (Pristupljeno: 18.04.2023).
31. Kerner, S.M. (2023). What is a large language model (LLM)? – TechTarget Definition. [online] WhatIs.com. Dostupno na: <https://www.techtarget.com/whatis/definition/large-language-model-LLM>. (Pristupljeno: 18.04.2023).
32. Bala Krishna Ragala. (2023). Software Engineer Challenges and Solutions to Overcome. [online] www.knowledgehut.com. Dostupno na: <https://www.knowledgehut.com/blog/web-development/software-engineer-challenges>. (Pristupljeno: 18.04.2023).
33. Izraylevych, I. (2019). What do you need to know about chatbot development. Dostupno na: <https://chatbotlife.com/what-you-need-to-know-about-chatbot-development4900e9fbf702> (Pristupljeno: 18.04.2023).
34. Manger, R. and Marušić, M. (2003). Sveučilište u Zagrebu Prirodoslovno Matematički Fakultet -Matematički odjel STRUKTURE PODATAKA I ALGORITMI skripta. [online] Dostupno na: <http://lnr.irb.hr/soya/nastava/spa-skripta.pdf> (Pristupljeno: 18.04.2023).
35. Experts, E.D. (2023) What is chat GPT? - everything you need to know, Unlock the Power of Data. Dostupno na: <https://blog.enterprisedna.co/what-is-chat-gpt-everything-you-need-to-know/> (Pristupljeno: 18.04.2023)..
36. Pop, A. (2023) History and future impact of chat GPT, Pop! Automation. Dostupno na: <https://www.popautomation.com/post/history-and-impact-of-chat-gpt> (Pristupljeno: 18.04.2023).
37. Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., ... (2023). ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103(102274). doi:<https://doi.org/10.1016/j.lindif.2023.102274>. (Pristupljeno: 18.04.2023).

38. Zoran Hercigonja (2017). Učenje programiranja korištenjem interaktivnog dijagrama tijeka. [online] Pogled kroz prozor. Dostupno na: <https://pogledkrozprozor.wordpress.com/2017/01/31/ucenje-programiranja-koritenjem-interaktivnog-dijagrama-tijeka/> (Pristupljeno: 26.06.2023).
39. Bansal, Y. (2023) The advantages and limitations of chat-GPT: What it can and cannot do, Medium. Dostupno na: <https://medium.com/@yashbansal42/the-advantages-and-limitations-of-chat-gpt-what-it-can-and-cannot-do-119def54a264> (Pristupljeno: 26.06.2023).
40. Harve, A. (2023) Chatgpt in higher education: Pros and cons, Hurix Digital. Dostupno na: <https://www.hurix.com/chat-gpt-pros-and-cons-of-using-chatgpt-in-higher-education/> (Pristupljeno: 26.06.2023).
41. Bendell, L. (2022) CHATGPT: The Pros and the cons, TheWebGate. Dostupno na: <https://thewebgate.net/chatgpt-pros-and-cons/> (Pristupljeno: 26.06.2023).
42. Docs, G. (2023) About github copilot for individuals, GitHub Docs. Dostupno na: <https://docs.github.com/en/copilot/overview-of-github-copilot/about-github-copilot-for-individuals> (Pristupljeno: 26.06.2023).
43. Tilgner, A. (2022) A deep dive into github copilot, Medium. Dostupno na: <https://betterprogramming.pub/ai-review-github-copilot-d43afde51a5a> (Pristupljeno: 26.06.2023).
44. Arsovski, A. (2022) The purpose, benefits, and downsides of github copilot, proxify. Dostupno na: <https://proxify.io/articles/what-is-github-copilot> (Pristupljeno: 26.06.2023).
45. Aminu Abdullahi (2023). What Is a Large Language Model? | Guide to LLMs. [online] eWEEK. Dostupno na: <https://www.eweek.com/artificial-intelligence/large-language-model/>. (Pristupljeno: 27.06.2023).
46. Google (2021) Tabnine case study, google cloud, Google. Dostupno na: <https://cloud.google.com/customers/tabnine> (Pristupljeno: 27.06.2023).
47. Preimesberger, C.J. (2022) Tabnine updates AI-based code-writing assistant for developers, VentureBeat. Dostupno na: <https://venturebeat.com/ai/tabnine-updates-ai-based-code-writing-assistant-for-developers/> (Pristupljeno: 27.06.2023).

48. Ramachandran, A. (2023) Ai code assistants: Head to head, Codeium · Free AI Code Completion & Chat. Dostupno na: <https://codeium.com/blog/code-assistant-comparison-copilot-tabnine-ghostwriter-codeium> (Pristupljeno: 27.06.2023).
49. SJ Inovation (2023) Boosting efficiency for developers & coders with the help of chatgpt!, SJ Innovation LLC. Dostupno na: <https://sjinnovation.com/boosting-efficiency-developers-coders-help-chatgpt> (Pristupljeno: 27.07.2023).
50. Rahman, Md.M. and Watanobe, Y. (2023) Chatgpt for Education and research: Opportunities, threats, and strategies, MDPI. Dostupno na: <https://doi.org/10.3390/app13095783> (Pristupljeno: 02 August 2023).
51. Konstantinos I. Roumeliotis, Nikolaos D. Tselikas (2023) ChatGPT and Open-AI Models: A Preliminary Review, Future Internet, vol.15, no.6, pp.192, Dostupno na: <https://ieeexplore.ieee.org/document/10105236> (Pristupljeno: 27.07.2023).
52. Siregar, F.H., Hasmayni, B. and Lubis, A.H. (2023) The Analysis of Chat GPT Usage Impact on Learning Motivation among Scout Students. Dostupno na: https://www.researchgate.net/profile/Andre-Lubis/publication/372448822_The_Analysis_of_Chat_GPT_Usage_Impact_on_Learning_Motivation_among_Scout_Students/links/64b7bc12b9ed6874a52e29ed/The-Analysis-of-Chat-GPT-Usage-Impact-on-Learning-Motivation-among-Scout-Students.pdf (Pristupljeno: 27.07.2023).
53. Reini, N. (2021) The impact of AI powered code completion in the software engineering field. Dostupno na: <https://www.theseus.fi/bitstream/handle/10024/781700/The%20Impact%20of%20AI%20powered%20code%20completion%20in%20the%20software%20engineering%20field.pdf?sequence=2> (Pristupljeno: 27.07.2023).
54. Prather, J. et al. (2023) 'it's weird that it knows what I want': Usability and interactions with copilot for novice programmers, arXiv.org. Dostupno na: <https://arxiv.org/abs/2304.02491> (Pristupljeno: 27.07.2023).
55. Harvard, P.V. et al. (2023) Expectation vs. experience: Evaluating the usability of code generation tools powered by large language models: Extended abstracts of the 2022 CHI Conference on Human Factors in Computing Systems,

ACM Conferences. Dostupno na: <https://dl.acm.org/doi/pdf/10.1145/3586030>
(Pristupljeno: 27.07.2023).

57. Wermelinger, M. (2023) 'Using github copilot to solve simple programming problems', Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 [Preprint]. doi:10.1145/3545945.3569830.
58. INDIAai. (n.d.). Is ChatGPT good at coding? Can it solve coding problems? [online] Dostupno na: <https://indiaai.gov.in/article/is-chatgpt-good-at-coding-can-it-solve-coding-problems> [Pristupljeno: 27.07.2023].

11. Popis slika

Slika 1. Prikaz rada chatbota i korisnika (Izvor: https://www.ellucian.com/insights/how-chatbots-benefit-higher-ed)	12
Slika 2. Prikaz razvitka transformer arhitekture ChatGPT-a (Izvor: https://arxiv.org/pdf/2302.10724.pdf).....	16
Slika 3. Prikaz sučelja ChatGPT-a (Izvor: obrada autora).....	17
Slika 4. Tijek rada predloženih metoda pomoću kojih radi copilot (Izvor: https://arxiv.org/pdf/2206.15331.pdf).....	19
Slika 5. Prikaz sučelja copilot bota u Visual Studio Code programu (Izvor: https://www.amitmerchant.com/github-copilot-chat/).....	20
Slika 6. Prikaz izgleda rada Tabnina u pomoći pri kodiranju (Izvor: https://i.ytimg.com/vi/twPtvZuBrAg/maxresdefault.jpg).....	22
Slika 7. Broj ispitanika obzirom na spol (izvor: Obrada autora)	37
Slika 8. Postotak ispitanika obzirom na dob (izvor: obrada autora).....	37
Slika 9. Broj ispitanika obzirom na studijski smjer koji pohađaju (izvor: obrada autora)	38
Slika 10. Postotak ispitanika s obzirom na godinu studija (izvor: obrada autora)	39
Slika 11. Postotak ispitanika prema vrsti studija kojeg pohađaju (izvor: obrada autora)	40
Slika 12. Postotak ispitanika koji koriste alate podržane umjetnom inteligencijom za učenje programiranja (izvor: obrada autora)	41
Slika 13. Postotak ispitanika s obzirom na korištenje alata za učenje programiranja podržane umjetnom inteligencijom (izvor: obrada autora).....	42
Slika 14. Postotak ispitanika za svrhu korištenja alata za učenje programiranja podržane umjetnom inteligencijom (izvor: obrada autora).....	43
Slika 15. Postotak ispitanika u svrhu oslanjanja na pomoć od strane alata za učenje programiranja podržani umjetnom inteligencijom (izvor: obrada autora)	44
Slika 16. Postotak ispitanika obzirom na vještinu korištenja alata za učenje programiranja podržani umjetnom inteligencijom (izvor: obrada autora).....	45
Slika 17. Postotak ispitanika vezanih za korištenje određenih alata za učenje programiranja podržane umjetnom inteligencijom (izvor: obrada autora).....	46

12. Popis tablica

Tablica 1. Prednosti upotrebe ChatGPT-a prema autorima i istraženom relevantnoj literaturi	25
Tablica 2. Prednosti upotrebe GitHub Copilota prema autorima i istraženom relevantnoj literaturi	27
Tablica 3. Prednosti upotrebe Tabninea prema autorima i istraženom relevantnoj literaturi	28
Tablica 4. Nedostaci upotrebe ChatGPT-a prema autorima i istraženom relevantnoj literaturi	31
Tablica 5. Nedostaci upotrebe GitHub Copilota prema autorima i istraženom relevantnoj literaturi	33
Tablica 6. Nedostaci upotrebe Tabninea prema autorima i istraženom relevantnoj literaturi	34
Tablica 7. Percepcija ispitanika vezanih uz odgovore na odabrane tvrdnje u anketnom upitniku uz prednosti upotrebe alata za učenje programiranja podržane umjetnom inteligencijom (frekvencija, postotci (%) ispitanika na skali od 1 - potpuno netočno do 5 - potpuno točno, M - arit. sredina, σ - st.dev.....	47
Tablica 8. Percepcija ispitanika vezanih uz odgovore na odabrane tvrdnje u anketnom upitniku uz prepreke upotrebe alata za učenje programiranja podržane umjetnom inteligencijom (frekvencija, postotci (%) ispitanika na skali od 1 - potpuno netočno do 5 - potpuno točno, M - arit. sredina, σ - std.....	49

13. Prilozi

Istraživanje mišljenja o alatima za učenje programiranja podržana umjetnom inteligencijom

Ovo istraživanje se provodi za potrebe pisanja diplomskog rada studenta Fakulteta informatike Sveučilišta Jurja Dobrile u Puli. Ovaj upitnik je anoniman i dobrovoljan. Nakon što ga popunite i predate, smatramo da ste suglasni s korištenjem Vaših odgovora za potrebe znanstvenog istraživanja i statističke obrade podataka. Predviđeno vrijeme za popunjavanje upitnika je oko 15 minuta, a pitanja su uglavnom koncipirana tako da treba označiti u kojoj mjeri se na Vas odnose pojedine tvrdnje.

Spol

- Muško
- Žensko

Dob (u godinama):

Vaš odgovor _____

Koji studijski smjer pohađate ?

- Informatika
- Računarstvo

Koja ste godina studija?

- 1. godina prijediplomskog sveučilišnog studija
- 2. godina prijediplomskog sveučilišnog studija
- 3. godina prijediplomskog sveučilišnog studija
- 1. godina diplomskog sveučilišnog studija
- 2. godina diplomskog sveučilišnog studija

Određeni studij pohađam kao

- Redovan student
- Izvanredni student

Koristite li alate podržane umjetnom inteligencijom za učenje programiranja?

- Da
- Ne

Koliko često koristite alate za učenje programiranja podržane umjetnom inteligencijom?

- Nikad
- Rijetko
- Često
- Svakodnevno

U koju svrhu koristite alate za učenje programiranja podržane umjetnom inteligencijom?

- Fakultet
- Ostalo: _____

U kojoj mjeri se oslanjate na pomoć od strane alata za učenje programiranja podržane umjetnom inteligencijom?

- Vrlo malo
- Malo
- Srednjoj
- Velikoj
- Jako velikoj

Koliko se smatrate vještima u korištenju alata za učenje programiranja podržane umjetnom inteligencijom?

- Nedovoljno
- Dovoljno
- Dobro
- Vrlo dobro
- Izvrsno

Koristite li neke od navedenih alata za učenje programiranja podržane umjetnom inteligencijom? *

- Koristim ChatGPT
- Koristim GitHub Copilot
- Koristim Tabnine
- Ostalo: _____

Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) omogućavaju bolju produktivnost.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Smatram da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) omogućavaju brz pristup informacijama.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) povećavaju motivaciju za učenje programiranja.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) pružaju korisniku potporu u bilo koje doba dana.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Smatram da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) olakšavaju razumijevanje koda putem jednostavnih primjera.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) potiču ovisnost o potrebi korištenja istih.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) nisu u mogućnosti odgovoriti točno na postavljeno pitanje ukoliko mu se ne postavi cijelokupan kontekst.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Mišljenja sam da alati za učenje programiranja podržani umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) mogu narušiti privatnost putem neovlaštenog prikupljanja osobnih podataka.

- Potpuno netočno
- Uglavnom netočno
- Niti netočno, niti točno
- Uglavnom točno
- Potpuno točno

Smatram da alate za učenje programiranja podržane umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) namjeravam tijekom učenja programiranja koristiti u budućnosti.

- 1 2 3 4 5
- U potpunosti se ne slažem U potpunosti se slažem

Smatram da alate za učenje programiranja podržane umjetnom inteligencijom (npr. ChatGPT, Copilot, Tabnine) namjeravam više koristiti u budućnosti.

1 2 3 4 5

U potpunosti se ne slažem U potpunosti se slažem

Pomisao na korištenje alata podržane umjetnom inteligencijom u meni potiče negativne osjećaje.

1 2 3 4 5

U potpunosti se ne slažem U potpunosti se slažem

Sažetak

Stalni napredak umjetne inteligencije rezultirao je velikim tehnološkim otkrićima i unapređivanju rada svih alata za programiranje koji imaju mogućnost korištenja umjetne inteligencije. Korištenje alata podržanih umjetnom inteligencijom uvelike ubrzava učenje i otkrivanje grešaka kod studenata što poboljšava i unapređuje njihovu želju za učenjem i stalnim razvijanjem.

Empirijsko istraživanje vezano za korištenje alata podržanih umjetnom inteligencijom za pomoć pri učenju programiranja provedeno je u sklopu ovog rada. Metodologija obuhvaća provođenje anketnog istraživanja među studentima kako bi se prikupile relevantne informacije o njihovoj percepciji i iskustvima u vezi s korištenjem alata podržanih umjetnom inteligencijom. Anketni upitnik prilagođen je studentima vezan za ovo istraživanje te je distribuiran među 102 studenata na Fakultetu informatike u Puli.

Rezultati istraživanja pružit će uvid u trenutno stanje i prihvaćenost alata podržanih umjetnom inteligencijom među studentima, kao i identifikaciju eventualnih izazova s kojima se susreću prilikom korištenja istih.

Ključne riječi: Programiranje, umjetna inteligencija, chatbot, prednosti i nedostaci chatbotova

Abstract

The continuous advancement of artificial intelligence has led to significant technological discoveries and the improvement of all programming tools with the capability of utilizing artificial intelligence. The use of AI-supported tools greatly accelerates learning and error detection among students, thereby enhancing and fostering their desire for learning and continuous development.

Empirical research related to the use of AI-supported tools for programming learning assistance was conducted as part of this study. The methodology involves conducting a survey among students to collect relevant information about their perception and experiences regarding the use of AI-supported tools. The survey questionnaire was tailored to students involved in this research and was distributed among 102 students at the Faculty of Computer Science in Pula.

The research results will provide insight into the current state and acceptance of AI-supported tools among students, as well as identifying any challenges they may encounter in their use.

Keywords: Programming, artificial intelligence, chatbot, advantages and disadvantages of chatbots.