

**INTEGRACIJA CHAT-GPT-A SA TEHNOLOGIJAMA SEMANTIČKOG VEB****INTEGRATING CHAT-GPT WITH SEMANTIC WEB**Tara Pogančev, *Fakultet tehničkih nauka, Novi Sad***Oblast – SEMANTIČKI VEB**

**Kratak sadržaj** – Rad istražuje načine na koje se savremena tehnologija računarske inteligencije – ChatGPT, može integrisati sa tehnologijama semantičkog veba. Demonstrira kako se na osnovu podataka iz dve ontologije (ACM Curriculum i Bibo Ontology) mogu kreirati a potom i izvršiti SPARQL upiti na osnovu korisničkog zahteva.

**Ključne reči:** semantički veb, veštačka inteligencija, ChatGPT

**Abstract** – This work explores ways of integrating ChatGPT with the technologies of semantic web. It demonstrates how using ACM Curriculum and Bibo Ontologies we can generate and perform SPARQL queries using AI.

**Keywords:** semantic web, artificial intelligence, ChatGPT

**1. UVOD**

Doba rapidno rastućih informacionih tehnologija za sobom je ponelo i eksponencijalnu ekspanziju računarske inteligencije. Adaptacija novonastalih modela postepeno pronalazi svoj standard u industriji, kako bi se posredstvom računarske inteligencije olakšali brojni izazovi razvoja aplikacija.

Semantički web, poznat i kao Web 3.0 predstavlja ekstenziju *World Wide Web*-a kroz seriju standarda kako bi podaci bili mašinski čitljivi nezavisno od njihovog izvora. Fokus se svodi sa veb aplikacija na prezistirane podatke. Njihov oblik i korelacije detaljno su opisani putem **ontologija**. Podaci se čuvaju u **RDF** (*Resource Description Framework*) formatu. Ovakav način perzistencije podataka dozvoljava nam da koristimo upitni jezik specifično kreiran za RDF – **SPARQL** (*SPARQL Protocol and RDF Query Language*).

U ovom projektu, integracija sa tehnologijama računarske inteligencije svodi se na korišćenje ChatGPT-a, kako bi se na osnovu korisničkog zahteva pisanog u prirodnom jeziku za našu ontologiju izgenerisao validni SPARQL upit, čijim bi izvršavanjem bili vraćeni adekvatni podaci.

**NAPOMENA:**

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milan Segedinac, vanr. prof.

**2. ŠTA JE CHAT-GPT**

ChatGPT [1] je alat zasnovan na veštačkoj inteligenciji koji koristi procesiranje prirodnog jezika (NLP – *Natural language processing*) kako bi uz čovekovu interakciju kreirao realističnu i prirodnu konverzaciju. ChatGPT je forma generativnog AI – koji koristi ljudske unose za generisanje korespondentnih slika, video zapisa, ili u ovom slučaju, teksta.

GPT predstavlja *Generative Pre-trained Transformer* (generativni pretrenirani transformer). Model je obučen sa nadgledanjem kroz povratnu informaciju od strane čoveka koja bi nagrađivala tačnije i prirodnije odgovore. Korišćeno je duboko učenje kako bi odgovori rezultovali humanim tekstom kroz neuronske mreže transformera. Oni su specijalizovani za predikciju sekvence (generišu narednu reč, rečenicu, paragraf...).

Izuzetan alat kreiran je od strane kompanije OpenAI, u novembru 2022. godine. Samo nekoliko nedjelja po svom objavljivanju postao je nezamenjiva asistencija u raznim poslovnim branšama.

**3. SPECIFIKACIJA ZAHTEVA**

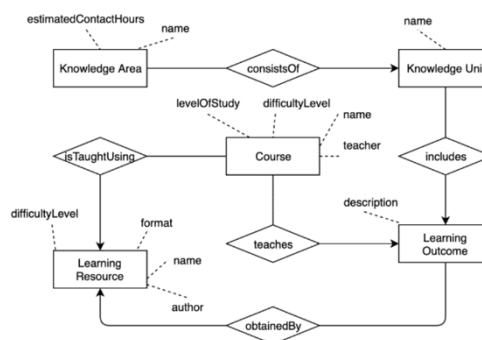
Specifikacija prototipske aplikacije koju ovaj rad istražuje za cilj ima da pokaže jednostavan primer integracije dve tehnologije – **semantički web i ChatGPT**.

Domen podataka kojima se aplikacija bavi jesu informacije o studijskim programima, kursovima, oblastima znanja, naučnim materijalima sa svojim bibliografskim podacima...

Podaci su predstavljeni u **RDF** formatu, opisani pomoću dve ontologije:

- *ACM Curriculum Ontology*
- *Bibo Ontology (The Bibliographic Ontology)* [2]

Primarni fokus jeste ACM Curriculum ontologija, čija je šema prikazana na slici 1.



Slika 1. Šema ACM Curriculum ontologije

Pored demonstracije ručno napisanih upita, glavni zahtev aplikacije jeste integracija našeg sistema sa ChatGPT modelom. Na korisnikov zahtev pisan prirodnim jezikom model treba da izgeneriše adekvatan SPARQL upit koji se izvršava na serveru. Rezultat istog se potom vraća korisniku. Pre generisanja zahteva, ChatGPT model potrebno je obučiti znanjem o ontologiji, relacijama između entiteta kojima se barata, kao i snabdeti adekvatnim primerima SPARQL upita.

#### 4. ARHITEKTURA SISTEMA

Arhitektura sistema sačini se od 3 ključne komponente:

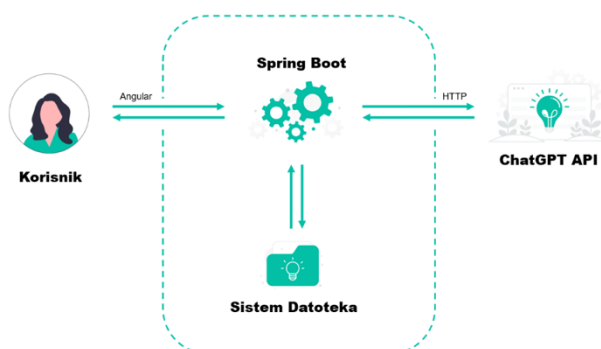
- Klijentska aplikacija
- Serverska aplikacija
- ChatGPT API

Kao razvojno okruženje za klijentsku aplikaciju korišćen je Angular 2. **Angular** je razvijen od strane kompanije Gugl, u septembru 2016. godine, i od tada je doživeo rapidan razvoj šestomesečnim nadogradnjama i novim verzijama. Arhitektonski obrazac koji Angular diriguje jeste MCV – Model-View-Controller.

Na serverskoj strani je korišćen **Spring Boot** – besplatno okruženje za razvoj Java aplikacija. Za cilj ima da pojednostavi rad programera uvodeći razne konvencije, automatske konfiguracije i integracije sa brojnim tehnologijama.

Kao eksterni servis korišćen je **API ChatGPT-a** za generisanje SPARQL upita. Visoki nivo određivanja ponašanja API-ja pomaže programerima da odgovore bolje prilagode svojim potrebama. Pored odabira konkretnog modela koji žele da koriste, programeri mogu da specificiraju domen u kome će model davati odgovore. Moguće je navesti stil generisanog teksta, kao i dati potrebne kontekstualne informacije pre konkretnog korisnikovog upita.

Arhitektura sistema definiše korišćenje sistema datoteka za perzistenciju podataka. Kako se koriste tehnologije semantičkog veća, u .owl fajlovima definisane su arhitekture ontologija, dok se sami podaci čuvaju u .rdf fajlovima. Nacrt arhitekture sistema prikazan je na slici 2.



Slika 2. Nacrt arhitekture

#### 5. PROTOTIPSKA IMPLEMENTACIJA

Prototipska implementacija aplikacije zasnovana je na demonstraciji osnovnih funkcionalnosti opisanih tehnologijama semantičkog veća.

Komunikacija između klijentske Angular i serverske Spring Boot aplikacije rađena je pomoću HTTP poziva. Takođe, putem istih poziva server komunicira sa ChatGPT API-em.

##### 5.1 Predefinisani upiti

Prvi način na koji korisnik može da interaguje sa aplikacijom jeste pozivom predefinisanih SPARQL upita. Definisani od strane programera, ovi upiti primaju ulazne parametre pomoću kojih se definiše pitanje koje korisnik želi da postavi. Definisana pitanja su:

1. Koji **resursi** su potrebni za učenje specifičnog **kursa**?
2. Koji **resursi** su potrebni za učenje specifične **jedinice znanja**?
3. Za koje **kurseve** su potrebni **materijali** od određenog **autora**?
4. Koji **kursevi** daju znanje za specifičan **ishod učenja**? Koji **autori** su kreatori za **materijale** tih **kurseva**?
5. Koji **dokumenti** su citirani od strane određenog **autora** u specificiranim **formatima materijala**?

##### 5.2 ChatGPT integracija

Kako bi OpenAI omogućio integraciju sa svojim alatima, potrebno je aktivirati API ključ na njihovom sajtu. Kao pomoć programerima u korišćenju ovog servisa dostupna je i zvanična dokumentacija [3].

Pre slanja samog zahteva, neophodno je definisati telo istog. Osnovne informacije koje su definisane u opisanom slučaju su:

- Maksimalan broj tokena odgovora: **200**
- Temperatura (nasumičnost modela): **0.7**
- Model: **text-davinci-003**
- Ulazni tekst

Najbitniji od navedenih parametara je ulazni tekst. Nivo njegove detaljnosti pomoći će modelu da razume širu sliku konteksta u kome korisnik postavlja pitanje. Početak ovog ulaza jeste detaljna definicija ontologije. Potom, na osnovu prethodno definisanih SPARQL upita iste prosleđujemo modelu kao primere. Na samom kraju, dodajemo korisnikov teksta na osnovu koga generisanje počinje.

Opisani deo ontologije ulaznog teksta je sledeći:

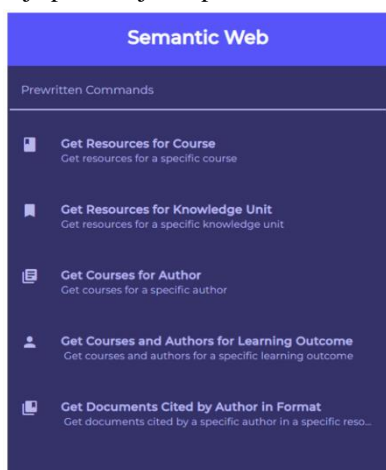
„I have a Semantic web project and it uses a custom ACM Curriculum Ontology. Its entities are: KnowledgeArea (described with estimatedContactHours and name). KnowledgeArea consists of KnowledgeUnit which has a property name. KnowledgeUnit includes LearningOutcome, which has a property description. LearningOutcome is obtainedBy a LearningResource which has difficultyLevel, format, name and author. A Course isTaughtUsing LearningResource. A course has leveOfStudy, difficultyLevel, name and teacher. Course teaches one or more LearningOutcome. The namespace prefix is acm:, and the namespace url is: [http://www.semanticweb.org/sasaboros/ontologies/2020/11/se\\_c\\_ontology#](http://www.semanticweb.org/sasaboros/ontologies/2020/11/se_c_ontology#). If asked to connect the Course with the learning Outcome it teaches, please use the 'teaches' connector between them. I will give you

some example queries for this Ontology. (Query examples...) I will provide you with a question, and your job is to provide me with the SPARQL query for getting said data in described ontology. Do not include any properties the user does not mention himself. The question is: ...“

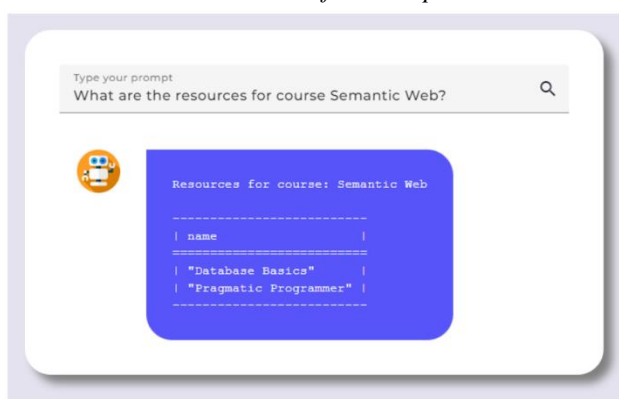
Nakon definisanja ulaznog teksta, preostalo je samo da pozovemo model putem API-a. Vraćen rezultat je u obliku String-a, i kao takav se proseđuje servisnoj kasi *SparqlService*. Nakon jednostavnih obrada objekta upita generisanog od strane modela, izvršava se poziv istog nad ontologijom. Rezultati se korisniku vraćaju na isti način kao i kod predefinisanih SPARQL upita.

### 5.3 Demonstracija klijentske aplikacije

Korisničko iskustvo definisano je lakoćom korišćenja klijentskog interfejsa. Na slikama 3 i 4 prikazan je izgled istog. Sa leve strane dostupni su predefinisani upiti, dok se u desnom odeljku ekrana nalazi konverzacioni prostor za postavljanje proizvoljnih upita.



Slika 3. Predefinisani upiti



Slika 4. Konverzacioni prikaz

## 6. VERIFIKACIJA I VALIDACIJA

Rezultati ChatGPT modela većinski su se pokazali tačnim. Zbog sohastičke prirode istog, unošenje identičnog ulaznog teksta više puta daće različite rezultate. U testiranim primerima, rezultati ovih upita nisu se vidno razlikovali, iako su sami upiti bili drugačiji.

Model je pokazao dobru mogućnost razumevanja ontologije, i generisanja takvih upita čijim će se izvršavanjem izvući stvarni rezultati. Pored toga, model

se uspešno snalazi i sa jednostavnim matematičkim apstrakcijama SPARQL upita, poput direktive za prebrojavanje - COUNT.

U početnim iteracijama rešenja, upiti koji su iziskivali upravo COUNT direktivu često su bili netačni, kako bi generisan kod imao sintaksne greške. Ovaj problem rešen je dodavanjem primera pravilno napisanih upita u ulazni tekst modela.

Prilikom testiranja modela primećeno je da je veoma bitan način na koji se upit formira. Kako bi generisani upiti bili tačniji, potrebno je koristiti što više pojmova iz same ontologije (nazive klasa i poveznika).

Na primer, ukoliko bismo pitanje „How many knowledge units does each of the knowledge areas consist of?“ pokušali da svedemo na skraćenu verziju oblika „How many knowledge units does each of the areas have?“, sa pretpostavkom da se *knowledge area* može skratiti na *area*, kao i da se *consist of* podrazumeva sa *have*, rezultati upita ne bi bili tačni.

Jedan od primera na kojima model ima predvidivo, ali nepoželjno ponašanje jeste davanje upita koji su nevezani za ontologiju. Na primer, na pitanje „What are the most popular breeds of cats?“, model će izgenerisati sledeći upit:

```
PREFIX acm:
<http://www.semanticweb.org/sasaboros/ontology/2020/1/1/sec_ontology#> SELECT ?breed WHERE {?cat acm:breed ?breed . ?cat acm:popularity ?popularity . FILTER (?popularity > 0) ORDER BY DESC(?popularity) LIMIT 10}
```

Iako je u pitanju sintaksno ispravan SPARQL upit, vidimo da se model prilagodio postavljenom zahtevu iako je znao da pomenute klase ne postoje u definisanoj ontologiji. Sa trenutnom implementacijom, ChatGPT će krenuti sa pretpostavkom da sve navedene klase u ulaznom pitanju postoje, te će na osnovu toga generisati upit čije će pokretanje izazvati grešku na serverskoj aplikaciji.

Ovakvo ponašanje nije neočekivano od strane ChatGPT v3.5. I kompletan model često je (pogotovo na programerskim temama) sklon davanju netačnih odgovora, kako bi zadovoljio nameru korisnika. Pojednostavljeno, model će se potruditi da korisniku da potvrđan odgovor čak i ako on ne postoji. Primer za ovo bilo bi generisanje koda sa metodama koje nisu definisane u specifikaciji, ali čije bi ime sugerisalo rešenje za problem koji korisnik navodi.

I ovako nepovoljno ponašanje modela moglo bi se popraviti dodatnim proverama, treniranjem, ili strožijim pravilima uz manji stepen kreativnosti ChatGPT-a.

## 7. ZAKLJUČAK

Rapidan razvoj veštačke inteligencije na svim poljima pruža veoma dobar potencijal za implementaciju semantičkog veća. AI ne samo da može da poboljša načine na koje menjamo podatke, već i iz korena promeni tehnike kojima im pristupamo.

Posredstvom savremenih modela, informacijama već danas možemo da pristupamo ne na nivou aplikacija ili

dokumenata, već na nivou znanja koje postoji u širem kontekstu.

Rad istražuje načine na koji se jedan od trenutnih *cutting-edge* generativnih modela – ChatGPT može iskoristiti sa tehnologijama semantičkog veća. Konkretno, istraženo je kako ChatGPT može da generiše SPARQL upite za ontologije definisane pomoću OWL-a, nad podacima čuvanim u RDF formatu.

Rezultati ove integracije su vrlo uspešni, međutim, bitno je naglasiti da kvalitet generisanog upita veoma zavisi od teksta unesenog od samog korisnika. Pravilno korišćenje imena klasa, atributa i veza dovešće do generisanja boljih upita koji se mogu pravilno izvršavati nad odabranom ontologijom.

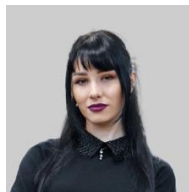
Prostori za poboljšanje ovog prototipskog rešenja primarno se zasnivaju na detaljnijem treniranju modela, i mogućnosti da razume korisničke namere čak i ako prosleđen zahtev semantički odstupa od pojmova definisanih ontologijom. Dodatno, rešenje bi se moglo poboljšati integracijom dodatnih ontologija, i temeljnije uvezanih podataka.

Iako je ovo samo jedan od primera moguće integracije modela veštačke inteligencije sa semantičkim većom, neosporno je da će u skoro dolazećem periodu AI igrati ključnu ulogu u načinu na koji pristupamo podacima na internetu. Svedoci smo stupanja u jednu novu, inteligentnu eru za tehnološki svet, čije su mogućnosti neiscrpna tema za istraživanje i napredak.

## 8. LITERATURA

- [1] Definition: ChatGPT, <https://www.techtarget.com/whatis/definition/ChatGPT> (pristupljeno u maju 2023.)
- [2] Bibo Ontology, <https://github.com/structuredynamics/Bibliographic-Ontology-BIBO> (pristupljeno u julu 2023.)
- [3] ChatGPT documentation, <https://platform.openai.com/docs/api-reference/chat> (pristupljeno u maju 2023.)

### Kratka biografija:



**Tara Pogančev** rođena je 3. februara 2000. u Novom Sadu. Školske 2018/2019. godine se upisuje na Fakultet tehničkih nauka u Novom Sadu, na studijskom programu računarstvo i automatika. Nakon diplomiranja u julu 2022. godine, upisuje master studijski program Inteligentni sistemi na istom fakultetu.

kontakt: [tarapogancev@gmail.com](mailto:tarapogancev@gmail.com)