Zbornik radova Fakulteta tehničkih nauka, Novi Sad

UDK: 004.9
DOI: https://doi.org/10.24867/28BE06Zivic

ČET-BOTOVI U DOMENU PUTOVANJA I TURIZMA

CHATBOTS IN TRAVEL AND TOURISM

Miloš Živić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *U radu je predstavljen čet-bot koji je sposoban da daje odgovore na pitanja iz domena putovanja i turizma. Pokušano je više pristupa, a dva glavna tipa modela su bazirana na odgovaranju putem izvačenja i odgovaranju putem generisanja.*

Ključne reči: *Čet-bot, generisanje, odgovori, putovanja, turizam*

Abstract – *The paper presents a chatbot capable of answering questions from the field of travel and tourism. Several approaches have been tried, and the two main model types were based on response by retrieval and response by generation.*

Keywords: *Chatbot, generation, answers, travel, tourism*

1. UVOD

Cilj ovog rada je predstavljanje rada čet-botova ali i samih ideja koje su dovele do njihovog nastanka. Prvo bih se zaustavio na prirodi komunikacije između ljudi i čet-botova, a dosta lepo poređenje je dato u radu *Real conversations with artificial intelligence*u u kom je upoređena konverzacija između ljudi sa konverzacijom između čoveka i čet-bota. Ovo istraživanje je dovelo do zanimljivih rezultata, a neki od njih su ti da imamo duplo manji broj reči po poruci kod konverzacije sa čet-botom u odnosu na konverzaciju sa čovekom, kao i duplo veći broj poruka.

Ovo ponašanje se može opisati ljudskom tendencijom da razmenjuju manje reči sa osobama koje ne poznaju, a u komunikaciji sa čet-botom se osećaju tako.

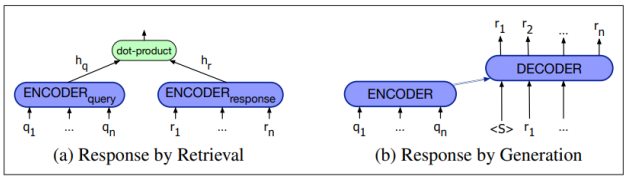
Kada pričamo o istoriji nastanka čet-botova važno je pomenuti prevaru zvanu Mehanički turčin. Ova mašina, za koju se kasnije ispostavilo da je pokretana od strane čoveka, naizgled je mogla da igra šah na visokom nivou krajem 18. veka. Iako je sve ovo bila prevara, sama ideja je zaitrigirala razne naučnike i propagirala se do Alana Tjuringa koji je napisao rad *Computing Machinery and Intelligence*, 1950. godine. U tom radu je predstavljena Igra imitacije koja je kasnije nazvana Tjuringov test. U igri postoje dva učesnika i sudija koji je čovek. Svaki od učesnika se nalazi u zasebnoj sobi, a na sudiji je zadatak da pogodi da li je u sobi mašina ili čovek. Ako sudija ne uspe da pogodi da se mašina nalazi u sobi to znači da je mašina prošla Tjuringov test.

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

Celo ovo prenošenje ideja sa generacije na generaciju naučnika dovelo je do prvog pravog čet-bota sposobnog za vođenje konverzacije. ELIZA čet-bot, razvijen 1966. godine spada u prvu od dve veće grupe čet-bototva, a to su čet-botovi bazirani na pravilima, a u ovu grupu takođe spada i PERRY, koji je razvjien šest godina kasnije. Ono što karakteriše čet-botove bazirane na pravilima je relativno mali broj unapred napisanih pravila po kojima ovi čet-botovi treba da se ponašaju. Oni rade tako što iz pitanja korisnika izvlače određene ključne reči, i na osnovu tih ključnih reči oni daju odgovor definisan unapred postavljenim pravilom. Na primer, ako korisnik u pitanju spomene majku ili oca, čet-bot tu reč prepozna kao ključnu i vrati odgovor koji može glasiti „Reci mi više o svojoj porodici“. Ovaj odgovor je dobijen direktnim korišćenjem pravila koje je asociralo odgovor sa ključnom reči.

U drugu grupu čet-botova spadaju čet-botovi bazirani na korpusu. U ovu grupu spadaju modeli korišćeni u aplikaciji koja je prethodila ovom radu. Ove čet-botove karakteriše veliki skup podataka koji varira od nekoliko stotina hiljada reči, ako govorimo o podskupu čet-botova koji je baziran na izvlačenju, do nekoliko stotina milijardi reči ako govorimo od podskupu koji je baziran na generisanju.

Razliku između ova dva pristupa možemo videti na sl. 1.



Slika 1. *Razlika između metode izvlačenja i metode generisanja*

Pravljenje ovako velikog skupa podataka definitivno nije lak posao. Neki od načina da se prikupe ovi podaci su upošljavanje velikog broja ljudi da imaju konverzacije na razne teme, često u određenim ulogama ili prikupljanje konverzacija sa društvenih mreža kao što su Facebook, Twitter (sada X), Reddit i druge. Još jedan način je prikupljanje mogućih odgovora sa platformi koje poseduju velike količine informacija i znanja kao što je na primer Wikipedia. Iz ovih podataka model može da nauči da prepriča priču ili vrati određene činjenice.

Nakon ove faze, najčešće sledi faza u kojoj ljudi testiraju ove modele, a konverzacije nastale iz ovog testiranja se kasnije koriste za dotreniravanje. Takođe, veoma je bitno da postoje precizne metrike koje mogu nedvosmisleno da nam kažu koji odgovori su dobri, a koji nisu.

2. PODACI

Kao što znamo, kvalitet modela mašinskog učenja zavisi od podataka koje taj model koristi. Što su kvalitetniji podaci to će model bolje raditi. Uzevši ovo u obzir, sastavili smo skup podataka od 5334 pojedinačnih pitanja i odgovora za trening i 60 pitanja i odgovora za testiranje.

Skup podataka za testiranje modela visokog odziva dobijen je prikupljanjem podataka pod nazivom web scraping, a korišćen je alat koji se zove Selenium. Web scraping se odnosi na prikupljanje podataka tako što se programski simulira korisnik i preuzima se sadržaj elemenata na veb stranicama.

Veb sajt koji je korišćen i ovu svrhu jeste Quora, a pristupano je raznim temama vezanim za putovanja i turizam na ovom sajtu.

Test skup podataka kreiran je ručno, biranjem nasumično 20 različitih pitanja i izmenom istih na 3 različita načina: malo (po jedno slovo ili kraća reč), srednje (po duža reč ili fraza) i puno (kompletno parafraziranje pitanja).

Podaci korišćeni za treniranje modela visokog odziva dobijeni su sa Kaggle-a, a tamo se nalaze pod nazivom Quora Question Pair. Skup podataka se sastoji, izuzimajući ID-jeve, od tri kolone: pitanje1, pitanje2 i duplikat. Kolona duplikat je binarana i kao vrednosti ima 0 ili 1, 0 označava da pitanje1 nije duplikat u odnosu na pitanje2, 1 govori da su ova dva pitanja ista, odnosno duplikati.

3. ARHITEKTURA

U ovom poglavlju ćemo obraditi izgled aplikacije i njenu arhitekturu, kao i arhitekturu modela korišćenih za predikciju odgovora na pitanje postavljeno od strane korisnika.

Tehnologije na bekendu i frontendu su iste za sve modele i ta strana aplikacije je fiksna za sve modele. Na bekendu imamo Python-ov radni okvir FastAPI za kreiranje endpointa, dok je na frontendu React.

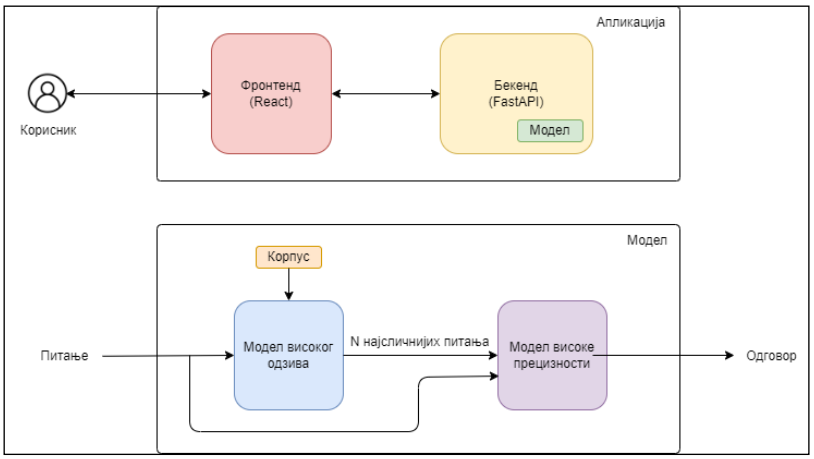
Glavni deo ovog rada je arhitektura samih modela i opis njihovog rada. Imamo dva glavna tipa modela čet-bota: čet-bot baziran na odgovaranju putem izvlačenja i čet-bot baziran na odgovaranju putem generisanja. Dalje, dva dela čet-bota koji daje odgovore uz pomoć izvlačenja su model visokog odziva i model visoke preciznosti. O svemu ovome se više može pročititati u nastavku.

3.1 Čet-bot baziran na odgovarnju putem izvlačenja

Sada ćemo preći na prvi tip modela koji je baziran na odgovaranju putem izvlačenja i sastoji se od modela visokog odziva i modela visoke preciznosti.

U model visokog odziva ulazi pitanje zajedno sa celim korpusom pitanja, a cilj mu je da brzo nadje N najsličnijih pitanja. N najsličnijih pitanja idu dalje u kompleksniji model visoke preciznosti gde se nalazi najsličnije pitanje.

Na slici 2 vidimo kako ova arhitektura izgleda u celosti, a u nastavku ćemo ući u detalje i objasniti kako ona u stvari radi.



Slika 2. *Arhitektura aplikacije*

3.1.1 Model visokog odziva

Model visokog odziva je deo čet-bota baziranog na odgovaranju putem izvlačenja koji daje N najsličnijih pitanja, ne nužno najpreciznije, ali relativno brzo. Cilj je da se najsličnije pitanje nađe bilo gde unutar N najsličnijih pitanja.

Svi tipovi modela visokog odziva koriste k-NN nakon vektorizacije za izvlačenje N najsličnijih pitanja. Pošto odradimo vektorizaciju pitanja treba nam metod kojim bismo odredili koja pitanja su najsličnija. Najjednostavniji, ali ne i najbolji pristup je da nakon vektorizacije uporedimo vektor ulaznog pitanja sa svakim vektorom pitanja iz baze pojedinačno. Upoređivanje vektora bi bilo vršeno nekom metrikom sličnosti kao što je euklidska ili kosinusna, i bili bi vraćeni vektori sa sličnišću najbližom jedinici. Ovo bi za velike baze verovatno trajalo predugo. Zato smo u ovoj aplikaciji primenili k-NN algoritam, koji kao izlaz iz algoritma daje najbliže vektore po određenoj metrici.

Da bismo dobili N mogućih kandidata koji bi kasnije ušli u model visoke preciznosti moramo prvo vektorizovati ulazne dokumente. Koristili smo tri različita pristupa: vektorizacija na nivou celog dokumenta, vektorizacija na nivou reči uz pomoć Word2vec algoritma i vektorizacija uz pomoć BERT-a.

Za vektorizaciju na nivou dokumenta smo koristili TF/TF-IDF algoritam. Ovaj algoritam radi tako što gleda koliko često se pojavljuju određene reči u dokumentu, odnosno pitanju. Na kraju dobijemo vektor koji predstavlja broj pojavljivanja svake reči iz vokabulara.

Kada pričamo o vektorizaciji na nivou reči, tu smo probali Word2vec algoritam. Word2vec radi tako što je treniran da predvidi koje reči se nalaze zajedno, to jest u istom kontekstu. Iz ovog treninga dobijamo guste vektore reči, kod kojih se reči koje imaju slično značenje nalaze blizu u vektorskom prostoru.

Poslednji algoritam za vektorizaciju je BERT. BERT je treniran na raznim taskovima iz NLP-ja, tako da nakon ovog treninga sadrži neku predstavu jezika. Ovo mu omogućava da napravi dobre embedinge, odnosno vektore, reči. Verzija BERT-a koja je ovde korišćena se zove TourBERT, a razlika je ta da je ovaj model dotreniran na podacima iz sveta turizma i putovanja, tako da bi trebalo da može da napravi bolju reprezentaciju reči koje su vezane za turizam.

3.1.2 Model visoke preciznosti

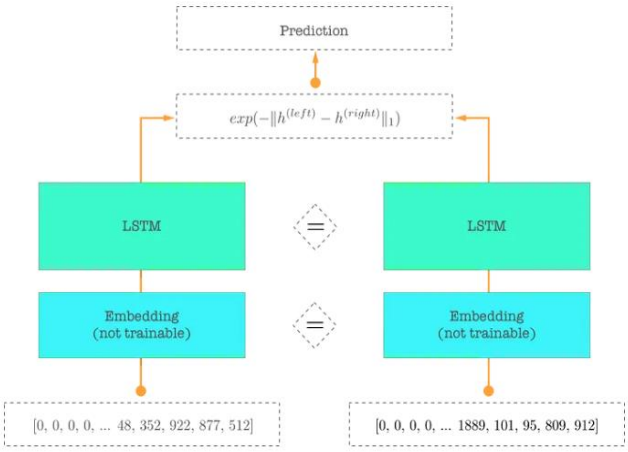
Model visoke preciznosti je drugi deo čet-bota, a služi da od N pitanja dobijenih od modela visokog odziva odluči koje je najsličnije ulaznom pitanju postavljenom od strane korisnika. Ovaj model je po pravilu komplikovaniji i radi sporije, ali bi u teoriji bolje mogao da nađe najsličnije pitanje.

Model koji je korišćen kao model visoke preciznosti je sijamska LSTM mreža. Ulazi u mrežu su vektori prvog i drugog pitanja, vektorizovani uz pomoć pretreniranog Word2vec modela, a postoji mogućnost korišćenja modela koji je treniran na našim podacima.

Tako vektorizovana pitanja prolaze kroz LSTM mrežu koju možemo zamisliti kao da ima dve putanje, za svako pitanje po jednu, ali realno postoji samo jedna putanja jer LSTM mreža deli težine i upravo zbog toga se i zove sijamska.

Na kraju, uz pomoć Menhetn distance mreža predviđa koliko su ova dva pitanja slična, a labela iz trenning skupa podataka, koja govori da li su ta dva pitanja duplikati se koristi za podešavanje težina mreže.

Na slici 3 možemo videti kako izgleda arhitektura sijamske LSTM neuronske mreže. Kao ulaz imamo vektore koji predstavljaju pitanja, dobijamo embedinge pitanja i šaljemo ih u sijamsku LSTM neuronsku mrežu. Dalje se u LSTM mreži proverava da li su ta dva pitanja prafrazirana verzija istog pitanja, tako što na izlazu iz LSTM-a imamo Menhetn distancu koja nam govori da li su ta dva vektora, koji su izlaz iz sijamske LSTM neuronske mreže, dovoljno slični.



Slika 3. *Arhitektura sijamske LSTM neuronske mreže*

Takođe, treba napomenuti da na slici izgleda kao da imamo dva LSTM-a, ali je to u stvari jedan koji deli težine, kao što je pričano ranije.

3.2 Čet-bot baziran na odgovarnju putem generisanja

Do sada smo pričali o modelu koji je pisan od nule, a model je spadao u grupu modela baziranih na odgovranju putem izvlačenja. U nastavku ćemo pričati o modelima koji su bazirani na generisanju isključivao i koriste se već pretrenirani pozivanjem njihovog API-ja.

Koristili smo generativne modele koji su preuzeti sa Hugging Face platforme. Ova platforma sadrži ogromnu bazu raznih tipova pretreniranih transformer modela i trenutno je najpopularnija u svetu transformera.

Neki od modela koji su korišćeni pretrenirani sa Hugging Face-a su GPT2 i Bloom, ali je korišćen i ChatGPT gađanjem direktno OpenAI API-ja.

4. REZULTATI

Probali smo razne modele visokog odziva na testnom skupu podataka, a u tabeli 1 su navedene performanse nekih od njih. Kao što se na tabeli vidi, najbolje performanse je postigao model koji je koristio Stemming i Stop reči u kombinaciji.

Naravno, performanse ovih modela u velikoj meri zavise od testnog skupa podataka. Ovde je korišćen testni skup od 60 pitanja o kom je bilo priče u poglavlju 2. Podaci. Moguće unapređenje ovog skupa podataka bi bilo njegovo proširenje i pažljiviji izbor pitanja, a ovom aspektu će više reči biti u poglavlju 6. Dalji rad.

Tabela 1. *Poređenje performansi različitih modela*

Model	Efficiency percent
Stemming	97.8%
Lemmatization	96.4%
N-grams	96.3%
Stemming + Stop words	98.3%
Custom word vectors with IDF	98.0%
Custom word vectors with POS+NER	97.4%
Pretrained word vectors	95.8%
TourBERT word vectors	91.5%

5. ZAKLJUČAK

U aplikaciji je primenjeno više pristupa, kao što sy pravljenje od nule čet-bota baziranog na odgovaranju putem izvlačenja i korišćenje pretreniranih čet-botova koji su bazirani na odgovaranju putem generisanja.

Kada pričamo o prvom tipu čet-bota on radi tako što pokušava da nađe najsličnije pitanje iz baze pitanja i odgovora, pitanju koje je postavio korisnik, i da vrati odgovor asociran sa tim pitanjem. Čet-bot se sastoji od dva glavna dela, modela visokog odziva i modela visoke preciznosti. Uloga modela visokog odziva je ta da vrati skup od par desetina ili stotina pitanja za relativno kratko vreme, i time značajno smanji broj pitanja sa kojima model visoke preciznosti treba da radi. Sa druge strane, model visoke preciznosti troši mnogo veće resurse, pa tako i vreme, da bi našao pravo pitanje među onih par desetina ili stotina dobijenih iz modela visokog odziva.

Drugi tip čet-bota koji je korišćen radi na u potpunosti drugačiji način. Treniran je na velikom skupu podataka, i uz pomoć toga je uspeo da stvori neku reprezentaciju znanja, koja je kasnije korišćena za generisanje odgovora.

Oba ova pristupa imaju svoje veoma važne prednosti. Čet-botovi bazirani na odgovaranju putem izvačenja su odlični kada je potrebna preciznost prilikom odgovaranja, ali i kada podaci koji treba da se koriste prilikom odgovaranja na pitanje nisu javno dostupni ili su veoma novi.

U poređenju sa ovim pristupom, čet-botovi bazirani na odgovaranju putem generisanja, daju novu dimenziju odgovaranja. Sposobni su da daju odgovore na pitanja na kojim nisu eksplicitno trenirani gde imaju i određenu dozu kreativnosti i fleksibilnosti koja manjka u prethodnom pristupu.

Naravno, svaki od ovih tipova modela ima i svoje mane. Recimo, kod odgovaranja putem izvlačenja postoji ograničenje kod dobijanja odgovora na pitanja koja ne postoje u skupu podataka. Isto tako, kod odgovaranja putem generisanja vrlo lako može da se desi da model da pogrešan odgovor iz pokušaja da odgovori na pitanje na koje ne zna odgovor. Sve ovo gorenavedeno nam govori da ovi modeli nisu savršeni, i imaju svoje prednosti i mane koje je potrebno pravilno adresirati.

6. DALJI RAD

Svakako, postoji dosta prostora kako bi se aplikacija i rad mogli unaprediti. Pre svega, kod modela baziranih na odgovaranju putem izvlačenja bismo mogli prikupiti bolje podatke koji bi bili korišćeni prilikom izvlačenja odgovora. Takođe, postoji prostor za optimizaciju izvršavanja koda, kao i za optimizaciju samih modela u smislu poboljšavanja pronalaska najsličnijeg pitanja. Kao poboljšanje na model koji je baziran na odgovaranju putem generisanja bismo mogli koristiti neki od naprednijih modela za koje su potrebni jači računarski resursi.

Pored ovoga, moglo bi se probati dotreniravanje postojećih modela na manjem skupu podataka, što bi dodatno unapredilo odgovaranje modela na pitanja iz specifičnog domena kao što je turizam. Konačno, poboljšanje bi moglo da ide u smeru kombinacije postojeća dva pristupa, gde bismo koristili RAG.

Treba pomenuti da bi se ovaj čet-bot, kao deo unapređenja, mogao integrisati u aplikaciju razvijanu kao deo diplomskog rada „Sistem za preporučivanje turističkih destinacija“ istog autora i mentora, gde bi doneo dodatnu vrednost i popravio korisnički doživljaj, ali i unapredio samu upotrebljivost obe aplikacije.

7. LITERATURA

[1] Daniel Jurafsky & James H. Martin, Speech and Language Processing, Chapter 15: Chatbots & Dialogue Systems, Stanford, January 2023

[2] A. M. TURING, I.—COMPUTING MACHINERY AND INTELLIGENCE, *Mind*, Volume LIX, Issue 236, October 1950, Pages 433–460, https://doi.org/10.1093/mind/LIX.236.433

[3] Joseph Weizenbaum. 1966. ELIZA—a computer program for the study of natural language communication between man and machine. Commun. ACM 9, 1 (Jan. 1966), 36–45. https://doi.org/10.1145/365153.365168

[4] Jennifer Hill, W. Randolph Ford, Ingrid G. Farreras, Real conversations with artificial intelligence: A comparison between human–human online conversations and human–chatbot conversations, Computers in Human Behavior, Volume 49, 2015, Pages 245-250, ISSN 0747-5632, https://doi.org/10.1016/j.chb.2015.02.026

[5] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Efficient Estimation of Word Representations in Vector Space, 2013, https://doi.org/10.48550/arXiv.1301.3781

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, Attention Is All You Need, 2017, https://doi.org/10.48550/arXiv.1706.03762

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018, https://doi.org/10.48550/arXiv.1810.04805

[8] Veronika Arefieva, Roman Egger, TourBERT: A pretrained language model for the tourism industry, 2022, https://doi.org/10.48550/arXiv.2201.07449

[9] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. Neural Comput. 9, 8 (November 15, 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Kratka biografija:

Miloš Živić rođen je u Novom Sadu 1999. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva pod nazivom Sistem za preporučivanje turističkih destinacija odbranio je 2022.god.
kontakt: miloszivic99@gmail.com