



## AUTOMATSKI GENERISANJE KULINARSKIH RECEPTA OD DATIH SASTOJAKA

### AUTOMATIC GENERATION OF CULINARY RECIPES FROM A GIVEN SET OF INGREDIENTS

Branislav Anđelić, *Fakultet tehničkih nauka, Novi Sad*

#### Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

**Kratak sadržaj** – Sa porastom popularnosti online stranica za deljenje recepata, količina raspoloživih podataka iz oblasti kulinarstva je veća nego ikad. Ljudi su u konstantnoj potrazi za načinom da brzo pronađu i spreme obrok. U ovom radu predložen je sistem za automatsko generisanje recepata za sastojke koje korisnik ima na raspolaganju, upotrebom *sequence to sequence* modela, kao i odabir podskupa sastojaka koji najbolje idu jedni uz druge. Pokazano je da je moguće generisati smislene tekstove recepata iz bilo kojeg unetog skupa sastojaka, ali se dovodi u pitanje njihova upotrebljivost u praksi. Uvođenjem novih ideja obrade sastojaka, ovaj rad donosi dobru osnovu za dalja istraživanja i unapređenja u ovom polju.

**Gljučne reči:** generisanje recepata, generisanje prirodnog jezika, *sequence to sequence* modeli

**Abstract** – With the increasing popularity of online web pages for recipe sharing, the amount of widely available culinary data is larger than ever. People are constantly looking for a way to find a recipe and prepare their daily meals quickly. This paper presents a system of automatic recipe generation from the ingredients currently available to the user, using *sequence to sequence* models and extraction of a suitable subset of the given ingredients. This paper shows that generating a meaningful recipe text is feasible for any given set of ingredients but questions the practical use of such recipes. With the introduction of new ideas for culinary data processing, this paper represents a solid base for future work in the field.

**Keywords:** recipe generation, natural language generation, *sequence to sequence* models

#### 1. UVOD

U ovom radu opisan je rad na temu generisanja recepata iz skupa sastojaka. Cilj ovog sistema je da na osnovu raspoloživih kuhinjskih sastojaka predloži korisniku recept za pripremu jela, čime bi doprineo raznovrsnosti ishrane korisnika. Metodologija primenjena u ovom radu nije ograničena na domen recepata, odnosno, mogla bi pronaći primenu i u drugim oblastima. Osnovi problem podeljen je na dve celine: izdvajanje podskupa unetog skupa sastojaka za koje će se generisati recept i generisanje niza instrukcija za pripremu jela na osnovu datih sastojaka.

Ideja izdvajanja podskupa sastojaka iz datog skupa jeste da se izdvoje sastojci koji se često nalaze u kombinaciji u postojećim receptima. Ovaj problem moguće je rešiti jednostavnom pretragom recepata, kako bi se pronašao recept sa sastojcima sadržanim u datom skupu. Međutim, ovakav metod uvodi ograničenje na recepte iz postojećeg skupa podataka. Iz tog razloga, u ovom radu posmatrane su zavisnosti individualnih sastojaka, kao i grupe sličnih recepata, što omogućava pronalazak podskupa sastojaka koji je jedinstven u okviru dostupnih podataka. Sa stanovišta izdvajanja podskupa podataka, rešenja iz literature se u većinski fokusiraju na izdvajanje najčešćih elemenata. Kako se određeni sastojci nalaze u većini recepata, rešenja zasnovana samo na frekventnosti pojavljivanja elemenata su od ograničene upotrebe u domenu ovog problema.

Generisanje instrukcija za pripremu recepta na osnovu sastojaka je problem *Seq2seq* (*sequence to sequence*) transformacije. *Seq2seq* predstavlja grupu jezičkih modela za transformaciju niza reči u drugi niz reči, što je u kontekstu ovog problema niz sastojaka u tekst instrukcija za pripremu. U ovom radu upotrebljeni su encoder-decoder LSTM (*Long short-term memory*) [1] neuronske mreže sa i bez *attention* mehanizma [2], kao i *transformer* [3] model, što su ujedno i *state-of-the-art* rešenja.

Za treniranje i evaluaciju modela upotrebljen je *recipe1m+* [4] skup podataka, koji sadrži oko milion strukturiranih recepata.

Kako su podaci iz ovog skupa podataka preuzeti iz više različitih izvora, sastojci su opisani u različitim formatima. Zbog velike varijacije u nazivima sastojaka, upotrebljen je *word2vec* [5] model zajedno sa metrikom za poređenje nizova karaktera, kako bi se grupisali nazivi koji predstavljaju isti sastojak, i time smanjio broj različitih obeležja u podacima.

Evaluacije odabira podskupa rađena je nad parovima skupova sastojaka i očekivanog podskupa, označenih od strane autora rada. Korišćene metrike evaluacije su preciznost i odziv.

U narednim poglavljima dat je pregled postojeće relevantne literature (poglavlje 2), specifikacije metodologije rešenja predstavljenog u ovom radu (poglavlje 3), rezultati predloženog rešenja (poglavlje 4) i zaključne reči (poglavlje 5)

#### 2. PRETHODNA REŠENJA

U ovom poglavlju dat je pregled postojeće literature relevantne za dati problem.

#### NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bila dr Jelena Slivka, vanr. prof.

RecipeGPT [6] je sistem za generisanje i evaluaciju recepata zasnovan na GPT-2 [7] pretreniranom transformer modelu. Treniranje modela odrađeno je nad *recipe1m* skupu podataka kao i u ovom radu. Podaci su preprocesirani izbacivanjem brojeva, reči koje predstavljaju količinu i komentara iz recepata, a sastojci su svedeni na imena sastojka upotrebom parsera za fraze sastojaka<sup>1</sup>. Takođe, upotrebljen je moduo za filtriranje sastojaka i recepata kreiran od strane autora, zadržavajući recepta sa barem dva sastojka. Slična procedura primenjena je i u ovom radu, gde su umesto parsera za sastojke iskorišćeni preprocesirani sastojci iz skupa podataka, i napravljen je sopstveni moduo za filtriranje. RecipeGPT [6] je sistem prilagođen da generiše naslov, sastojke ili instrukcije recepta, zbog čega se ulaz u transformer model sastoji iz ta tri navedena dela razdvojena tokenima za početak i kraj, za čime u ovom radu nije bilo potrebe. Kako bi se povećala raznovrstnost generisanog testka, u oba rada upotrebljen je *k-sampling* algoritam, koji bira reč iz skupa veličine *k* najviše rangiranih reči iz vokabulara. Model je treniran prilagođavanjem GPT-2 modela za dati problem, dok je za ovaj rad napravljen zaseban model. Kao i u ovom radu, za evaluaciju modela upotrebljene su BLEU [8] i ROUGE [9] metrike.

U radu *Recipe for Disaster* [10] predložen je sistem za generisanje tekstova upotrebom LSTM *encoder-decoder* modela. Kao i u radu [11] i u ovom radu, u radu [10] je upotrebljen *recipe1m* skup podataka. Podaci su filtrirani tako da ne sadrže nijedan sastojak duži od 10 reči niti tekst instrukcija duži od 50 reči, sa ciljem da se ograniči vokabular i smanji računarska moć potrebna za obradu skupa podataka. U ovom radu kod treniranja *encoder-decoder* modela ograničenja su postavljena na dužinu teksta recepta, sa maksimalnom dužinom od 100 reči, i na broj sastojaka u receptu, gde su odbačeni recepti sa manje od 5 ili više od 15 sastojaka. Reči iz sastojaka su predstavljene u vidu vektora dimenzije 50, a u ovom radu u vidu vektora dimenzije 256. Treniran je *encoder-decoder* model sa LSTM neuronskim mrežama i *Luong attention* [11] mehanizmom, dok je u ovom radu upotrebljen isti model sa *Bahdanau attention* [12]. Za evaluaciju upotrebljene su *perplexity* i BLEU metrike.

### 3. SPECIFIKACIJA I IMPLEMENTACIJA

U ovom poglavlju dati su detalji sistema za izdvajanje podskupa sastojaka i generisanje instrukcija recepta, što uključuje skup podataka (3.1), predprocesiranje podataka (3.2), izdvajanje podskupa sastojaka (3.3) i generisanje teksta instrukcija (3.4).

#### 3.1. Skup podataka

Za treniranje i evaluaciju podataka korišćen je *recipe1m* [4] skup podataka, koji se sastoji od preko milion recepata preuzetih sa različitih veb stranica. Podaci su organizovani u slojeve koji sadrže sastojke i instrukcije, slike i obrađene sastojke redom. Kako su recepti dobijeni automatskim pretraživanjem različitih izvora korisnički unetih recepata, podaci su dati u različitim oblicima, što otežava obradu podataka. Prvi sloj sadrži informacije o receptima koje uključuju naslov, listu neobrađenih tekstova sastojaka i listu instrukcija. U drugom sloju nalaze se slike pridružene receptima posredstvom identifikacionog obeležja, dok se u

trećem sloju nalaze obrađeni sastojci za svaki recept. Obradeni sastojci predstavljaju izdvojene nazive sastojka iz teksta sastojka, koji uobičajeno sadrži i količinu i mernu jedinicu.

#### 3.2. Pretprocesiranje podataka

Za različite algoritme unutar sistema primenjene su drugačije metode predprocesiranja podataka. U ovom delu biće opisan postupak predprocesiranja koji je univerzalan za čitav rad.

Iz teksta sastojaka i recepata izdvojeni su numerički karakteri i znakovi interpunkcije, i tekst je transformisan tako da sadrži samo mala slova. Obradeni sastojci iz skupa podataka su pridruženi tekstu recepata, gde su izbačeni svi nevalidni sastojci. Kako se među rezultujućim receptima i dalje nalazi značajan broj sastojaka koji su dati u obliku opisa ili komentara, koji se mogu posmatrati kao *outlier*-i, zadržani su samo sastojci koji se pojavljuju barem dva puta u okviru skupa.

Uočeno je da se među sastojcima nalazi značajan broj istih ili veoma sličnih sastojaka kojima je dodeljen različit naziv. U pokušaju da se smanji vokabular sastojaka, osmišljen je sistem za pridruživanje istih, odnosno, sličnih sastojaka u grupe, iz kojih je odabran najčešći kao reprezentativan za sve. Da bi se ovo postiglo, treniran je *word2vec* [5] model nad rečenica dobijenim spajanjem sastojaka iz pojedinačnih recepata. Prethodno je od svake liste sastojaka napravljeno deset sa nasumičnim redosledom sastojaka u nizu, zbog toga što *word2vec* posmatra susedne reči, u ovom slučaju po tri sa obe strane, a za svrhu ovog eksperimenta redosled sastojaka nije značajan. Dobijene vektorske reprezentacije, dužine 300, upotrebljene su kako bi se pomoću kosinusne sličnosti (eng. *cosine similarity*) zaključila međusobna sematička sličnost između sastojaka.

Pored prethodnog modela, za posmatranje leksičke sličnosti upotrebljena je funkcija<sup>2</sup> za poređenje sekvenci karaktera. Kombinovanjem ove dve metrike, za svaki jedinstveni sastojak pronađen je skup njemu sličnih sastojaka, gde svaki sastojak skupa ispunjava uslove da im je kosinusna sličnost veća od 0.6, a leksička veća od 0.75. Dobijeni sastojci i skupovi posmatrani su kao graf, gde su čvorovi grafa nazivi sastojaka, a grane sličnosti, tako što je za svaki čvor *b* iz skupa sličnih čvorova čvora *a* kreirana usmerena grana (*a*, *b*). Grane grafa su usmerene jer funkcija leksičke sličnosti nije simetrična. U ovako dobijenom grafu, jedno ostrvo grafa predstavlja grupu naziva koji predstavljaju isti sastojak, uz pretpostavku da ukoliko dva naziva ispunjavaju uslove sličnosti, oni predstavljaju isti sastojak. Iz svake grupe sastojak koji se najčešće pojavljuje u skupu podataka odabran je kao reprezentativan, i svi sastojci iz grupe su namapirani na njega. Navedenim postupkom, vokabular sastojaka smanjen je sa oko 18 hiljada reči na oko 11 hiljada.

#### 3.3. Izdvajanje podskupa sastojaka

Nakon što je korisnik označio koje sastojke poseduje, bira se smislen podskup tih sastojaka za korišćenje u receptu. Dati skup sastojaka transformisan je mapiranjem sastojaka na njihove reprezentativne iz grupe. Zatim je izdvojen podskup sastojaka pronalaženjem *K* najbližih recepata po

<sup>1</sup> <https://github.com/nytimes/ingredient-phrasetag>

<sup>2</sup> *SequenceMatcher.ratio()* - <https://docs.python.org/3/library/difflib.html>

skupu sastojaka algoritmom *K-nearest-neighbors*. Konkretno, sastojci iz svakog recepta iz skupa podataka i sastojci koje korisnik unese se predstavljaju kao vektori pomoću *bag-of-words* modela, a zatim se nađe K vektora iz skupa podataka koji su najbliži vektoru korisnikovih sastojaka. Na kraju, bira se presek između korisnikovih sastojaka i sastojaka iz najbližijih recepata. Na taj način, biraju se sastojci koji generalno idu jedni uz druge.

Jedan izazov u ovom pristupu bio je pronaći najbolje K. Uzimanjem većeg K posmatra se veći broj sličnih recepata, i uzima se veći podskup korisnikovih sastojaka i obrnuto. Kako bi se odredilo koje K najbolje rešava problem, označen je deo skupa podataka tako što su se uzimali nasumični recepti i njihovi sastojci, a zatim im se dodao šum u vidu sastojaka koji se obično ne nalaze u sličnim receptima. Zatim se pokretao algoritam za različito K i merile su se performanse. Za merenje performansi poređeni su skupovi algoritamski izabranih sastojaka i označeni sastojci i računala se F-mera. Najbolji rezultat postignut je sa K=25.

### 3.4. Generisanje teksta instrukcija

Poslednji zadatak programa je generisanje teksta recepta. Kao ulaz u ovaj deo dolazi skup sastojaka u obliku teksta, a kao izlaz očekuje se generisani tekst recepta. U tu svrhu isprobano je nekoliko rešenja, a najbolje su se pokazali:

- *Encoder-decoder* [13] model sa *attention* [12] mehanizmom
- *Transformer* [3] neuronska mreža.

#### 3.4.1. Encoder-decoder model sa attention mehanizmom

Iz skupa podataka su, radi smanjenja dimenzija ulaza i izlaza mreže, izbačeni svi recepti sa više od 15 sastojaka i sa više od 100 reči u instrukcijama. Osim toga su izbačeni svi recepti koji imaju manje od 5 sastojaka, jer je to premali broj ulaza za mrežu. Što se tiče pretprocesiranja ulaza, svi sastojci se prvo predstavljaju sa malim slovima, a zatim se tokenizuju. Tokenizacija je proces u kom se za svaki sastojak iz skupa podataka dodeli jedinstven broj, a zatim se svaka lista sastojaka predstavi kao vektor brojeva. Na kraju se radi dopuna tih vektora nulama (*padding*), kako bi svaki bio iste dužine. Što se tiče teksta recepata, koji u ovom slučaju predstavlja ciljane labele, radi se isto pretprocesiranje kao i kod ulaza. Osim toga, na početak i kraj svakog se dodaju takozvani *start* i *end* tokeni, koji predstavljaju početak i kraj recepta. Početni token se dodaje da bi *decoder* imao ulaz pre nego što generiše prvu reč, a krajnji token da bi se znalo kad je generisani tekst završen, jer se uvek generiše isti broj tokena.

*Encoder* mreža se sastoji iz dva sloja. Prvi je *embedding* sloj, koji pretvara ulazne reči (identifikacione brojeve reči) u vektore, a drugi je rekurentni sloj, u ovom slučaju GRU. Kao izlaz *encoder* sloja vraćaju se izlazi iz svakog koraka rekurentnog sloja, kao i poslednje skriveno stanje. *Decoder* mreža se sastoji iz *embedding* sloja, koja pretvara prethodno generisane reči u vektore, *attention* sloja, koji za određeni korak računa koji deo ulaza je relevantan za generisanje sledeće reči, rekurentnog sloja (GRU) i na kraju potpuno povezanog sloja, koji ima čvorova koliko je dužina vokabulara tekstova recepata, i on služi da bi se dodelile verovatnoće svakoj reči da budu izabrane kao naredna generisana reč.

Kod ovakve mreže procesi treniranja i upotrebe se delimično razlikuju. Kad se mreža trenira, dekodirer nikad ne uzima u obzir reči koje je on prethodno generisao, već reči iz ciljnog teksta, kako bi se olakšalo treniranje. Taj proces se naziva *teacher forcing*. Kasnije kad se mreža koristi, dekodirer se na ulaz daju reči koje je prethodno sam generisao.

#### 3.4.2. Transformer neuronska mreža

Velika prednost korišćenja *transformer* modela u odnosu na modele sa rekurentnim mrežama je što ne postoji deo koji mora da se izvršava sekvencijalno. Samim tim, trening može da se paralelizuje i da se izvršava mnogo brže. Stoga, mogu da se kreiraju mreže sa mnogo većim brojem parametara, koje će postizati bolje rezultate.

U ovom radu, korišćena je pretrenirana BERT [14] mreža uz dodatno treniranje za problem generisanja recepata. Sva implementacija se oslanja na biblioteku *simpletransformers*<sup>3</sup>, koja uz trening modela radi potrebno pretprocesiranje podataka.

## 4. REZULTATI

U ovom poglavlju dati su rezultati evaluacija korišćenih metoda, kao i diskusija o mogućim poboljšanjima priloženog sistema.

### 4.1. Evaluacija algoritma za izdvajanje podskupa sastojaka

Za evaluaciju algoritma za izdvajanje podskupa sastojaka označen je mali skup podataka od 10 primera. Svaki od primera se sastoji od sastojaka koji se koriste u nekom receptu, uz nekoliko dodatnih sastojaka koji predstavljaju šum. Za izbor dodatnih sastojaka uzeti su sastojci koji se ne pojavljuju u receptima sličnim izabranom receptu.

Evaluacija je vršena poređenjem podskupa sastojaka koji je izabrao algoritam sa podskupom koji je označen. Korišćena je F-mera kao dobar odnos između preciznosti i odziva. Jedini hiper-parametar u algoritmu je K za *K-nearest-neighbors*. Stoga je mereno kako menjanje K utiče na izmerenu F-meru. Rezultati su dati u tabeli 1. Na osnovu ovih merenja, zaključeno je da K=25 daje najbolje rezultate.

Tabela 1. Zavisnost F-mere od K.

K	10	15	20	25	30	35	40	45
F-mera	0.87	0.88	0.89	0.91	0.9	0.9	0.89	0.9

### 4.2. Evaluacija algoritma za generisanje teksta

Za evaluaciju metoda za generisanje teksta korišćene su mere evaluacije *ROUGE* i *BLEU* sa  $N=1$  i  $N=2$  vrednostima za posmatranje *n-gram*-a. Obe su zastupljene u zadacima za mašinsko prevođenje ili generalno prevođenje jedne sekvence teksta u drugu.

Rezultati *BLEU* mere i *ROUGE* F-mere za *encoder-decoder* model su dati u tabeli 2. Rezultati *BLEU* mere i *ROUGE* F-mere za *transformer* model su dati u tabeli 3.

<sup>3</sup> <https://simpletransformers.ai/>

Tabela 2. BLEU i ROUGE metrike za enkoder-dekoder model za različite N-grame

N	BLEU	ROGUE
1	0.00	0.07
2	0.00	0.001

Tabela 3. BLEU i ROUGE metrike za transformer model za različite N-grame

N	BLEU	ROGUE
1	0.30	0.33
2	0.09	0.10

Može se zaključiti da transformer model daleko bolje radi. Osim toga, vidi se da se modeli bolje snalaze sa 1-gram-ima nego sa 2-gram-ima, odnosno, da bolje predviđaju pojedinačne reči koje bi trebalo da se nađu u dokumentu nego parove reči. To je i smisleno jer je sam zadatak generisanja pojedinačnih reči lakši.

## 5. ZAKLJUČAK

U ovom radu opisano je nekoliko pristupa problemu generisanja recepata iz korisnički unetih sastojaka. Ovakav sistem doprineo bi raznovrsnosti ljudske ishrane, posebno u situacijama u kojima je pristup različitim namirnicama ograničen. Takođe, tokom ovog rada razmatrane su različite metode obrade i generisanje teksta, polja u kojima je postignut značajan napredak tokom skorašnjih godina.

Problem je razdvojen na dva glavna dela, odabir podskupa sastojaka i generisanje teksta recepta. Pored toga, analiza i procesiranje samih podataka igralo je značajnu ulogu u razvoju predloženog rešenja. Stečen je uvid u prirodu skupa podataka recepata i isprobano je nekoliko novih metoda njihove obrade, od kojih je najznačajnije pronalaženje različitih naziva za iste sastojke i njihovo spajanje. Za odabir sastojaka upotrebljen je KNN algoritam sa pronalaženjem preseka najbližnjih recepata, dok su za generisanje teksta razrađena dva *seq2seq* modela: *encoder-decoded* sa *attention* mehanizmom i *transformer*.

Sistem se pokazao sposoban da odabere odgovarajuće sastojke i generiše u velikoj meri smislen tekst recepta. Rezultati predloženog transformer modela uporedivi su sa trenutno najuspešnijim modelima za generisanje teksta po posmatranim metrikama, sa nešto slabijim rezultatima. Uzimajući u obzir raspoloživ hardver i vreme treniranja modela, pristup iz ovog rada pokazuje potencijal da dostigne rezultate koje postižu *state-of-the-art* rešenja u ovom polju.

## 6. LITERATURA

- [1] Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. "Sequence to sequence learning with neural networks."
- [2] Ramachandran, Prajit, et al. "Stand-alone self-attention in vision models." *arXiv preprint arXiv:1906.05909* (2019).
- [3] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems*. 2017.
- [4] Marin, Javier, et al. "Recipe1m+: A dataset for learning cross-modal embeddings for cooking recipes and food images." *IEEE transactions on pattern analysis and machine intelligence* 43.1 (2019): 187-203.
- [5] Goldberg, Yoav, and Omer Levy. "word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method." *arXiv preprint arXiv:1402.3722* (2014).
- [6] H. Lee, Helena, et al. "RecipeGPT: Generative pre-training based cooking recipe generation and evaluation system." *Companion Proceedings of the Web Conference 2020*. 2020.
- [7] Radford, Alec, et al. "Language models are unsupervised multitask learners." *OpenAI blog* 1.8 (2019): 9.
- [8] Papineni, Kishore, et al. "Bleu: a method for automatic evaluation of machine translation." *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 2002.
- [9] Lin, Chin-Yew. "Rouge: A package for automatic evaluation of summaries." *Text summarization branches out*. 2004.
- [10] Teisberg, Dev Bhargava Thomas. "Recipe for Disaster: A Seq2Seq Model for Recipe."
- [11] Luong, Minh-Thang, Hieu Pham, and Christopher D. Manning. "Effective approaches to attention-based neural machine translation." *arXiv preprint arXiv:1508.04025* (2015).
- [12] Chorowski, Jan, et al. "Attention-based models for speech recognition." *arXiv preprint arXiv:1506.07503* (2015).
- [13] Cho, Kyunghyun, et al. "Learning phrase representations using RNN encoder-decoder for statistical machine translation." *arXiv preprint arXiv:1406.1078* (2014).
- [14] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018).

### Kratka biografija:



**Branislav Andelić** rođen je 1997. godine. Osnovne akademske studije završio je 2020. godine na Fakultetu tehničkih nauka, na kom brani i master rad 2021. godine iz oblasti Elektrotehnike i računarstva – Softversko inženjerstvo i informacione tehnologije.

Kontakt: [branislav.andjelic@uns.ac.rs](mailto:branislav.andjelic@uns.ac.rs)