Sumarizacija sudskih presuda na srpskom jeziku primenom NLP metoda za ekstrakciju i apstrakciju teksta

Nataša Ivanović, Vera Kovačević
Univerzitet u Novom Sadu, Fakultet tehničkih nauka, Novi Sad, Srbija
{natasaivanovic, kovacevic.r214.2021}@uns.ac.rs

Apstrakt – Sudske presude najčešće su veoma obimni dokumenti. Sudije i advokati neretko angažuju pravna lica čiji zadatak je da pripreme *skice*, odn. skraćene verzije sudskih presuda. Ovo je česta praksa jer i sudije i advokati žele da imaju uvid u što više sličnih slučajeva i njihovih presuda kako bi se što bolje pripremili za nova suđenja iz slične oblasti. Proces pisanja sižea je spor, skup i zahteva mnogo truda. Automatizacija ovog postupka značajno bi uštedela vreme i energiju *legal editor*-ima - iako mašina ne može sumirati sadržaj na podjednako visokom nivou kao čovek, ovo bi predstavljalo dobru polaznu tačku i donekle bi čovekov posao bio olakšan i ubrzan. Fokus ovog rada su rekurentne neuronske mreže jer se one najčešće koriste za rešavanje NLP problema (engl. *Natural Language Processing*). Predložena su dva modela za apstraktivnu sumarizaciju presuda – *Sequence2Sequence* model i prošireni *Sequence2Sequence* model sa *Attention* mehanizmom. U ove arhitekture uključeni su *LSTM*, *Bidirectional LSTM* i *Teacher forcing* mehanizam. Sumarizacija se vrši and podskupom teksta iz sudskih presuda dobijenog ekstrakcijom najznačajnih rečenica primenom *Text Rank* algoritma. Evaluacija rešenja rađena je korišćenjem *ROUGE* metrike.

Ključne reči – *NLP*, apstraktivna sumarizacija, ekstraktivna sumarizacija, *RNN*, *Seq2Seq* model, *Attention* mehanizam, *Text Rank* algoritam, *LSTM*, *BiLSTM*, *Teacher forcing*

I. UVOD

U oblasti prava često su u upotrebi veoma obimni dokumenti, te sudije i advokati koriste njihove skraćene verzije kako bi se za što kraće vreme upoznali sa što više dokumentovanih slučajeva. Primer takvih dokumenata su sudske presude, kojima se bavi ovaj rad. Pri sumarizaciji sudskih presuda potrebno je obratiti pažnju na njihove specifične karakteristike po kojima se razlikuju od ostalih tekstova:

- veličina – kao što je već navedeno, pravni dokumenti su često veoma obimni;
- struktura – pravni dokumenti sadrže strogo strukturirane elemente (npr. datum, mesto, ime sudije i sl.);
- terminologija – pravni dokumenti koriste domenski specifičnu terminologiju;
- višeznačnost – neke domenske fraze mogu da se interpretiraju različito u zavisnosti od konteksta, npr. vrste suda;
- citati – u pravnim dokumentima citati često imaju veći značaj nego u drugim vrstama dokumenata [1].

Na osnovu toga se može zaključiti da je pisanje skraćenih verzija sudskih presuda spor i skup posao za koji je potrebno angažovati osobe koje poznaju domen prava, odnosno *legal editor*-e.

Prema tome, zbog specifičnog domena postupak sumarizacije sudskih presuda još uvek ne može u potpunosti da se automatizuje, ali uz pomoć tehnika NLP-a (engl. *Natural Language Processing*) moguće je generisati sižee sudskih presuda koji bi mogli da olakšaju posao *legal editor*-a.

U ovom radu predstavljena je sumarizacija sudskih presuda na srpskom jeziku kombinovanjem dva tipa sumarizacije teksta:

- Ekstraktivna sumarizacija – siže se formira izdvajanjem i kopiranjem najbitnijih delova teksta, odnosno rečenica;
- Apstraktivna sumarizacija – siže se formira generisanjem novih fraza koje nisu nužno deo originalnog teksta.

Za apstraktivnu sumarizaciju presuda predložena je upotreba rekurentne neuronske mreže, odnosno *Sequence2Sequence* modela koji je baziran na *LSTM* (engl. *Long short-term memory*) mehanizmu. Takođe, dat je i predlog proširenog modela koji koristi *Attention* i *Teacher forcing* mehanizam. Zbog nedostataka izabrane metodologije, odnosno lošijih performansi i dugog trajanja treniranja rekurentnih mreža pri radu sa obimnim tekstovima, pre apstraktne sumarizacije odrađena je ekstraktivna sumarizacija primenom *Text Rank* algoritma kako bi se izdvojili najbitniji delovi presuda pre nego što se podaci proslede *Sequence2Sequence* modelu. Detalji implementacije biće opisani u narednim poglavljima.

Skup podataka preuzet je sa sajta [2] i sadrži 400 sudskih presuda na srpskom jeziku i njihove sažetke, koji će biti korišćeni kao referentni prilikom evaluacije. Za evaluaciju će biti korišćena *ROUGE* metrika.

II. PREGLED RELEVANTNE LITERATURE

Pri izradi projekta korišćeni su naučni radovi iz oblasti neuronskih mreža i mašinske obrade pravnih dokumenata. U nastavku će biti predstavljeni neki od relevantnih radova.

U radu [3] dat je uopšten uvod u sumarizaciju teksta, opisana je razlika između sumarizacije jednog i više dokumenata. Dat

je pregled metoda sumarizacije pravnih dokumenata, metrika i softverskih alata koji se koriste za tu svrhu.

Izdvojeno je nekoliko tehnika ekstraktivne sumarizacije dokumenta, od kojih je za ovaj rad relevantan pristup baziran na grafu na kojem se zasniva *TextRank* algoritam. Da bi se pronašle najrelevantnije rečenice u tekstu, formira se graf čije čvorove predstavljaju rečenice, dok je granama predstavljeno preklapanje reči, odnosno broj zajedničkih reči dve rečenice.

Takođe, opisana je *ROUGE-N* (*Recall-Oriented Understudy for Gisting Evaluation*) metrika za evaluaciju sumarizacije teksta, koja meri odziv, osnosno koliko reči (ili *n-gram*-a) u kolekciji referentnih sižea se pojavljuje u mašinski izgenerisanom sižeu.

U radu je predstavljeno i nekoliko pristupa sumarizacije teksta koji se odnose na pravne dokumenate:

- *Feature based approaches* – sumarizacija na osnovu baze znanja koja sadrži pravila koja opisuju selekciju najbitnijih fraza i rečenica;
- *Graph based approaches* – ekstraktivna sumarizacija bazirana na grafu prilagođena pravnim dokumentima;
- *Rhetorical role based approaches* – dodeljivanje labela rečenicama ili skupovima rečenica da bi se označila uloga te rečenice u dokumentu (npr. činjenica, postupak, odlaganje, itd.);
- *Classification based approaches* – proširenje prethodne metode korišćenjem klasifikatora koji određuje labelu rečenice.

U zaključku je navedeno da su metode ekstraktivne sumarizacije pravnih dokumenata u nekim slučajevima uspele da ostvare performanse slične metodama sumarizacije običnih tekstova, ali su ti rezultati većinom nekonzistentni na različitim skupovima podataka. Iako ne opisuje apstraktivnu sumarizaciju, rad predlaže da se i ova metoda isproba na pravnim dokumentima.

Sličan pregled dat je i u novijem radu [4]. Izdvojeno je nekoliko algoritama koji se koriste za ekstraktivnu sumarizaciju pravnih dokumenata: *LexRank* (algoritam koji određuje relevantnost rečenice pomoću pomoću pristupa koji se bazira na grafu), *Latent Semantic Analysis (LSA)* (tehnika koja se bazira na traženju sličnosti između dokumenata pod pretpostavkom da slični dokumenti sadrže slične pojmove), *Reduction* (formiranje sižea uklanjanjem najmanje relevantnih rečenica) i dr. Zatim su opisani različiti pristupi ekstraktivnoj sumarizaciji pravnih dokumenata, od kojih je većina već spomenuta u prethodnom radu. Pored njih, navedena su još da pristupa:

- *Nature inspired* – klasifikacija rečenica pomoću algoritama optimizacije, npr. *Particle Swarm Optimization (PSO)*, genetski algoritam;
- *Machine learning based* – klasifikacija rečenica pomoću algoritama mašinskog učenja (npr. *Naive Bayes*) ili dubokih neuronskih mreža.

Ekstraktivna sumarizacija pomoću neuronskih mreža detaljnije je opisana u radu [5]. Ovaj rad predlaže korišćenje skupa podataka sa referentnim sižeima, na osnovu kojih se obučava neuronska mreža koja kalsifikuje rečenice u tekstu.

Navedeni radovi fokusirani su na ekstraktivnu sumarizaciju i daju zaključak da ne postoji dovoljno literature o apstraktivnoj sumarizaciji pravnih dokumenata, što znači da je ta oblast još uvek nedovoljno istražena.

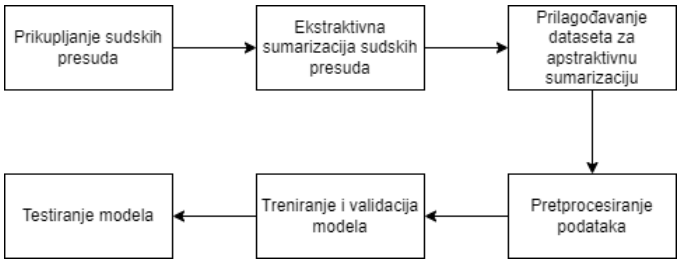
Model koja je korišćen za apstraktivnu sumarizaciju u ovom projektu opisan je u radovima [6] i [7]. Oba rada predstavljaju *Sequence2Sequence* arhitekturu sa dve rekurentne neuronske mreže – enkoder i dekoder. Enkoder pretvara sekvencu simbola u vektorsku reprezentaciju fiksne dužine, a dekoder na osnovu te reprezentacije generiše novu sekvencu. Ovaj model može da se primeni na različitim taskovima kao što su mašinsko prevođenje, sumarizacija teksta, automatsko odgovaranje na pitanja, *chatbot*. Pri evaluaciji je utvrđeno da model uspeva da uoči jezičke pravilnosti u tekstu i generiše smislene fraze.

Modeli neuronskih mreža za apstraktivnu sumarizaciju nisu prilagođeni obimnim i strukturiranim pravnim dokumentima, te je u radu [7] predloženo da se ovaj model unapredi upotrebom *Attention* mehanizma. Ovaj pristup bi omogućio da model ne tretira sve reči u rečenici jednako, nego da *obraća više pažnje* na relevantne fraze i samim tim *duže pamti*, što je korisno kada su u pitanju obimni dokumenti, kako bi model manje *zaboravljao* fraze s početka dokumenta.

U radu [7] je predložen metod koji kombinuje apstraktivnu i ekstraktivnu sumarizaciju – *Pointer Generator Network*. Ovaj metod predstavlja proširenje *Sequence2Sequence* modela određivanjem koje delove teksta kopirati (ekstraktivna sumarizacija), a za koje generisati novu sekvencu (apstraktivna sumarizacija).

III. METODOLOGIJA

Postupak sprovođenja sumarizacije sudskih presuda odvija se u nekoliko faza (Slika 1) – svaka od njih biće opisana u ovom poglavlju. Najpre je bilo neophodno formirati *dataset* prikupljanjem određenog skupa presuda, kao i njihovih sažetaka. Zatim je sprovedena ekstraktivna sumarizacija tokom koje su izvučene najznačajnije rečenice iz svake sudske presude. Izlaz iz ove faze je iskorišćen za ažuriranje *dataset*-a. Zatim je izvršeno pretprocesiranje prikupljenih podataka i tako obrađeni podaci prosleđeni su narednoj fazi – apstraktivnoj sumarizaciji koja generiše finalni sažetak sudske presude.



Slika 1 Pregled koraka sprovedenih tokom razvijanja rešenja

A. Prikupljanje podataka

Skup podataka preuzet je sa [2] i sadrži 400 sudskih presuda na srpskom jeziku i njihove sažetke, koji će kasnije biti korišćeni kao referentni sižei prilikom evaluacije. Kako je svaki od dostupnih dokumenata sadržao i prevod na engleski jezik, ti delovi dokumenata su uklanjani. Primećeno je da je 5% prikupljenih presuda napisano na ćirilici – ti dokumenti su konvertovani u latinicu korišćenjem javno dostupnih sajtova za ovu namenu. Na ovaj način je postignuta konzistentnost u skupu podataka i izbegnuto je da modeli iste reči tretiraju kao različite samo zbog različitog pisma.

B. Ekstraktivna sumarizacija

Ova metoda vrši sumarizaciju izvlačenjem najznačajnijeg podskupa rečenica iz originalnog teksta. Kako sam naziv sugeriše, ova metoda nema mogućnost generisanja novog teksta, već će izlaz iz ove faze uvek biti kopija dela originalnog teksta. U pravnom domenu, ovo može biti dobar pretkorak

apstraktnoj sumarizaciji jer može značajno da skrati veoma obiman tekst i izvuče najznačajnije delove – na ovaj način se smanjuje dimenzionalnost problema.

Ključna ideja iza ove metode je pronalaženje sličnosti između svih rečenica i vraćanje onih koje imaju maksimalni *similarity score*. Za formiranje matrice sličnosti primenjeno je računanje kosinusne sličnosti (*cosine similarity*). *TextRank* algoritam rangira rečenice na osnovu njihovog stepena važnosti – generiše se graf sačinjen od teksta napisanog na prirodnom jeziku. Zasniva se na principu *voting and recommendation*. Kada neki čvor ima vezu ka drugom, to generiše *vote* za taj čvor – što je veći broj glasova, veći je i stepen važnosti tog čvora.

Koraci implementirani za sprovođenje ekstraktivne sumarizacije dati su u nastavku:

1. Podela teksta sudske presude na rečenice i tokenizacija.
2. Kreiranje vektora za sve rečenice bazirano na tokenima koje sadrže.
3. Računanje kosinusne sličnosti između svih parova rečenica. Originalan algoritam u ovom koraku generiše N×N matricu (gde je N broj rečenica). Ovaj korak je u implementaciji optimizovan izbegavanjem suvišnih operacija zbog simetričnosti matrice, jer se očekuje da će sličnost između rečenice r_1 i r_2 biti ista kao između r_2 i r_1. Na ovaj način je smanjena dimenzionalnost problema, jer su pravni dokumenti veoma obimni.
4. Kreiranje grafa na osnovu matrice sličnosti gde svaki čvor predstavlja rečenicu, a grana predstavlja sličnost.
5. Rangiranje rečenica bazirano na *similarity score* vrednosti i vraćanje top N rečenica koje su uključene u sumarizovanu sudsku presudu. Kako u proseku sažeci korišćeni za evaluaciju modela imaju 10 rečenica, u implementaciji je za rangiranje odabrano N=10.

Ovim postupkom od originalnih sudskih presuda izgenerisane su ekstraktivno sumarizovane presude koje su iskorištene u *dataset*-u. Nad njima je vršeno treniranje modela.

C. *Pretprocesiranje podataka*

Pretprocesiranje podataka odrađeno je primenom *Tokenizer*-a iz *keras* biblioteke. Kako neuronske mreže ne primaju reči, nego samo numeričke vrednosti, sprovedeni su koraci transformacije stringova u brojeve. Kreiran je rečnik primenom *fit_on_texts* metode gde ključ predstavlja reč iz teksta, a vrednost njena frekvencija. Zatim se svaka reč iz teksta transformisala u numeričku vrednost koja predstavlja indeks iz rečnika – na osnovu njega će kasnije biti izvršeno dekodiranje, odn. vraćanje rečenica iz numeričkog oblika u tekstualnu reprezentaciju, razumljivu čoveku. Kako su rečenice različite dužine, primenjen je *padding* na osnovu dužine najduže rečenice u dokumentu, jer *keras* biblioteka zahteva da dužina ulaza ima konstantnu vrednost.

Ovi koraci pretprocesiranja su sprovedeni kako bi se prilikom implementacije modela mogao koristiti *embedding* sloj. On zahteva da ulazne vrednosti budu *integers*, gde svaka reč ima jedinstvenu *integer* vrednost. Benefiti *embedding* sloja jesu ti što nudi mogućnost konvertovanja svake reči u vektore fiksne dužne (engl. *feature vectors*) i doprinosi boljoj numeričkoj reprezentaciji reči, jer će semantički slične reči biti pozicionirane bliže jedna drugoj u *embedding* prostoru.

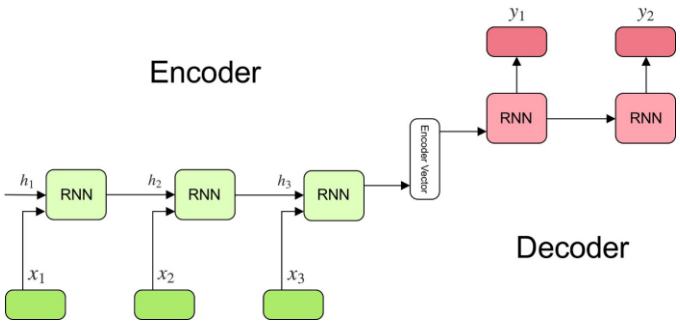
D. *Apstraktivna sumarizacija*

Osnovna ideja ovog pristupa je obučavanje modela tako da za ulaznu sudsku presudu generiše sažetak parafraziranjem originalnog teksta. Za potrebe rešavanja ovog problema implementirani su *vanilla Sequence-To-Sequence* model, kao i njegovo proširenje *Attention* mehanizmom.

a. *Sequence-To-Sequence-Model*

Sumarizacija sudskih presuda je kompleksan problem jer dužina ulaznog teksta, ali i sažetka varira. Za istu dužinu ulaza, ne postoji garancija da će se dobiti izlaz iste dužine. Kao odgovor na ovaj problem predstavljena je *Sequence-To-Sequence* arhitektura – dualni RNN (engl. *Recurrent Neural Network*) sistem koji se sastoji iz enkodera i dekodera (Slika 2). Osnovna ideja ovog modela je preuzimanje ulazne sekvence reči i kreiranje izlaza koji predstavlja novu sekvencu reči.

Enkoder procesira svaki element iz ulazne sekvence. Upotrebom informacija enkapsuliranih u sekvenci generiše *context vector* koji predstavlja kompaktnu, informativnu reprezentaciju ulaza (na slici je to *encoder vector*). On se dalje prosleđuje dekoderu, čija uloga je da generiše izlaznu sekvencu. Enkoder i dekoder su dve rekurentne neuronske mreže – za potrebe sumarizacije sudskih presuda kao finalan izbor odabrana su dva LSTM-a.



Slika 2 Prikaz Sequence-To-Sequence arhitekture

Dekoder je inicijalizovan finalnim stanjima enkodera – u kontekstu ovog rada to su finalni *hidden* i *cell* vektori dobijeni iz LSTM-a. Korišćenjem ovih inicijalnih vrednosti dekoder započinje generisanje izlaza, tako da je svaki izlaz uzet u obzir za narednu predikciju reči.

U cilju unapređenja trening faze implementiran je *Teacher forcing* mehanizam. Umesto prosleđivanja prethodno generisanog izlaza u dekoderu, prosleđuje se tačna reč, odn. *target word* – čak i ukoliko model nije predvideo dobru reč, *Teacher forcing* mehanizam će ga ispraviti, tako da model sledeću reč može da predvidi na osnovu ispravne reči. Ovaj pristup pomaže prilikom trening faze, jer može biti zahtevno za model da nauči celu rečenicu odjednom. Implementacija ovog mehanizma je sprovedena tako što se dekoderu prosleđivala ispravna rečenica sa *offsetom* 1, što je postignuto dodavanjem <SOS> (*start-of-sentence*) tokena.

b. *Sequence-To-Sequence-Model sa Attentiom mehanizmom*

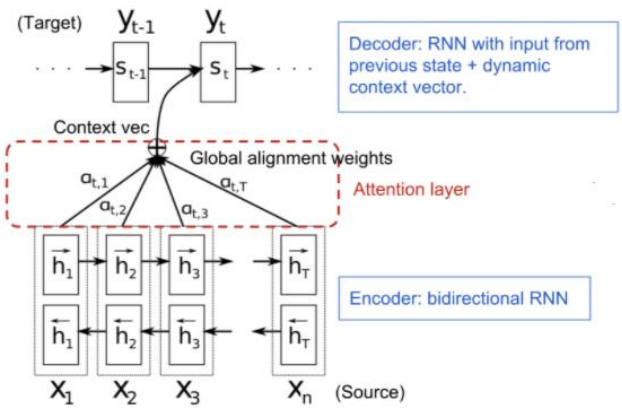
U slučaju *vanilla Seq2Seq* modela enkoder generiše *context vector* – u jednom vektoru enkapsulirane su informacije o celoj ulaznoj sekvenci. S obzirom na to da su rečenice u okviru pravnih dokumenata najčešće dugačke, postavlja se pitanje da li jedan vektor može da zadrži sve relevantne informacije potrebne dekoderu za generisanje izlaza. Ovo takođe predstavlja otežavajuću okolnost za dekoder, jer na raspolaganju ima samo jednu fiksnu reprezentaciju ulaza, na osnovu koje mora da generiše izlaz.

Attention mehanizam predstavlja rešenje za ove probleme. Osnovna ideja je izbegavanje učenja *single* vektor reprezentacije za svaku ulaznu rečenicu – umesto da se prosledi

finalni *hidden* vector, prosleđuje se lista. Prilikom svakog dekoder koraka dešavaju se sledeće faze:

1. Dekoder prima trenutno dekoder stanje s_t i sva enkoder stanja h_1, h_2, …, h_m.
2. Računanje *attention* skorova – za svako enkoder stanje h_k *attention* mehanizam računa stepen relevantnosti za dekoder stanje s_t. U kontekstu implementacije sumarizacije sudskih presuda, primenjuje se funkcija *one_step_attention* koja kao parametre prima po jedno enkoder i dekoder stanje i vraća skalarnu vrednost. *Attention* skor je računat *dot* operacijom.
3. Računanje *attention* težina primenom *softmax*-a nad *attention* skorovima.
4. Računanje *attention* izlaza.

Za potrebe rada enkoder je implementiran kao *BiLSTM* – čitanje ulazne sekvence se odvija u oba smera. Na taj način mreža stiče bolju sliku o kontekstu rečenice jer za svaki token ima svest o prethodnoj i sledećoj reči. Sva *hidden* stanja prosleđuju se dekoderu, odn. LSTM-u. Pregled arhitekture se može videti u nastavku (Slika 3).



Slika 3 Prikaz Sequence-to-Sequence arhitekture sa Attention mehanizmom

U odnosu na *vanilla Seq2Seq* model, proširenje *Attention* mehanizmom donosi benefite u kontekstu tumačenja ulaznih sekvenci i određivanja stepena važnosti određenih tokena u okviru rečenica, ali s druge strane donosi i računsku kompleksnost – kako bi se uspešno odradilo treniranje ovog modela nad datasetom koji se sastoji od sudskih presuda potrebno je imati mašinu sa dobrim performansama.

E. Implementacija Inference modela

Kao što je već u ovom poglavlju navedeno, prilikom treniranja modela primenjen je *Teacher Forcing* mehanizam – odn. dekoderu se kao ulaz prosleđuje i ispravna (*target*) sekvenca. Tokom test faze to nije moguće – umesto toga, kao svaki sledeći ulaz prosleđuje se prethodno prediktovana reč. Posledično, dolazi do razlike u dimenzionalnosti trening i *inference* (test) modela – odn. za testiranje *input* sekvenca će uvek imati dimenziju 1. *Keras* biblioteka zahteva da dužina ulaza ima konstantnu vrednost.

Ovaj problem je rešen implementacijom dva odvojena modela za trening i testiranje. *Inference* model je konstruisan od istog skupa slojeva kao i trening model. Jedina razlika je u *input* sloju, gde je *input_length=1*. Iz istreniranih modela preuzete su vrednosti *hidden* vektora i to je prosleđeno kao ulaz *inference* modelu – na ovaj način postignuta je ekvivalencija između trening i test modela.

IV. REZULTATI I DISKUSIJA

Za planirano treniranje *Sequence2Sequence* modela na skupu podataka od 400 presuda, kao i istog modela sa *Attention* mehanizmom, potrebni su hardverski resursi koji nisu bili dostupni za ovaj projekat. Za potrebe treniranja korišćen je računar sa sledećom konfiguracijom: 16GB RAM memorije i procesor AMD Ryzen 7 5800x.

Nije bilo moguće odjednom učitati trening dataset od 380 sudskih presuda [8]. Kao alternativa, korišćen je dataset od 25 sudskih presuda, a testni skup je brojao 17. *Vanilla Seq2Seq* model je treniran na 3900 epoha (koliko je hardver izdržao – inicijalna ideja je bila 5000 epoha). Postignuti rezultati i izgenerisani sižei mogu da se vide na github-u. U nastavku (Tabela 1) je prikazan jedan primer sumarizacije sa *ROUGE precision* metrikom. Iako je ovo daleko od željenih rezultata, može se primetiti da model izvlači značajne reči iz sudske presude. Originalni output je duži, radi preglednosti i donekle uočavanja smisla generisanog izlaza, prikazane su jedinstvene reči.

Epoha	Sumarizacija	ROUGE-p
3000.	nameri izdržava presude bolnicama lične ljudskom novog bliske zadržavanjem	0.00
3300.	dobije izvršeno predao 2005 godine	0.17
3700.	zahtev zemljište alkoholičara zatvorenike to mesta	0.29

Tabela 1 Prikaz rezultata sumarizacije

U cilju treniranja modela nad većim delom dataseta, isproban je *transfer learning* pristup. Ulazni podaci se treniraju u grupama od 20, nakon čega se trenirani model sačuva. Težine sačuvanog modela se zatim koriste za narednu grupu podataka – ideja je bila da se tako mreža vremenom trenira nad celim skupom. Međutim, nakon druge grupe podataka trening je prekinut sa *Memory Allocation* greškom. Rezultujući model je treniran nad 40 sudskih presuda – po 500 epoha za svaku grupu od 20. U nastavku (Tabela 2) su prikazani neki od rezultata.

Epoha	Sumarizacija	ROUGE-p
1-400	kažnjavanje zatvorski odeće bugarskom došlo redarstvenoj	0.17
2-400	2009 godine zatvora pojedinačno zagreb	0.33
2-500	gutsanovi uživanje zna uverljive nesreću zaključak	0.14

Tabela 2 Prikaz rezultata sumarizacije nakon transfer learning-a

Svi rezultati dostupni su na [9].

S obzirom na računsku kompleksnost *attention* mehanizma, prošireni *Seq2Seq* model nije istreniran nad skupom sudskih presuda zbog hardverskih ograničenja. Čak i sa samo jednom presudom, treniranje zahteva mnogo vremena zbog velike dimenzionalnosti problema. Validacija ovog modela odrađena je na drugom, pomoćnom *datasetu*, koji nije iz pravnog domena, ali sadrži dosta kraći tekst – uočeno je da nad takvim skupom podataka treniranje na našoj mašini može da se izvrši.

Pored nedostatka hardverskih resursa, prepreka za ostvarivanje dobrih rezultata navedenog modela je i sama priroda srpskog jezika. U srpskom jeziku postoji mnogo više različitih oblika jedne reči nego u engleskom jeziku, što značajno povećava dimenzionalnost problema i otežava generisanje smislenih rečenica.

Ekstraktivna sumarizacija je donela benefit kraćeg teksta, ali i ograničenje u kontekstu sažetaka nad kojima je vršeno treniranje – zbog dimenzionalnosti problema finalni vrednost N je setovana na top 5 rečenica. Ovo je značajno ubrzalo trening fazu, jer je bilo manje veoma dugačkih rečenica koje su diktirale dimenziju ulaznih slojeva i tako povećavale računsku kompleksnost množenja velikih matrica.

V. ZAKLJUČAK

U ovom radu predložena su *vanilla Seq2Seq* i *Seq2Seq* sa *attention* mehanizmom za apstraktivnu sumarizaciju sudskih presuda na srpskom jeziku. Kako su pravni dokumenti obimniji od nekih drugih tekstova koji se sumarizuju i imaju definisanu strukturu, pre toga je odrađena i ekstraktivna sumarizacija dokumenata da bi se izdvojile najrelevantnije rečenice i samim tim smanjio obim teksta i dimenzionalnost problema.

Pripremljen je skup podataka od 400 presuda i njihovih referentnih sižea na srpskom jeziku, međutim zbog hardverskih ograničenja osnovni model nije istreniran na celom skupu podataka, već je iskorišćeno 25 presuda. Model sa attention mehanizmom nije istreniran zbog računske kompleksnosti. U predlogu projekta navedena je i *Pointer Generator Network* – kako je suština ove neuronske mreže nadogradnja attention mehanizma iz prethodno navedenih razloga zaključilo se da nije odgovarajuća za rešavanje ovog problema.

Dobijeni rezultati nemaju formu rečenice, ali se može uočiti da model bira reči koje imaju semantičku vrednost u dokumentu. Pretpostavlja se da bi rezultati bili značajno bolji da je dataset bio veći i da je trening vršen nad Seq2Seq Attention modelu. Iako rezultati nisu na visokom nivou, predstavljeni modeli se mogu smatrati dobrom polaznom osnovom za dalja istraživanja u pravnom domenu.

LITERATURA

[1] Ambedkar Kanapala, Sukomal Pal, Rajendra Pamula (2017). Text summarization from legal documents: a survey. Springer Science+Business Media B.V.

[2] Pravosudna akademija *eCase*, https://e-case.eakademija.com/, preuzeto marta 2022. godine

[3] Deepali Jain, Malaya Dutta Borah, Anupam Biswas (2021). Summarization of legal documents: Where are we now and the way forward. Department of Computer Science and Engineering, National Institute of Technology Silchar, Assam, 788010, India

[4] Deepa Anand, Rupali Wagh (2019). Effective deep learning approaches for summarization of legal texts. Journal of King Saud University – Computer and Information Sciencs, India

[5] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares Holger Schwenk, Yoshua Bengio (2014). Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

[6] Ilya Sutskever, Oriol Vinyals, Quoc V. Le (2014). Sequence to Sequence Learning with Neural Networks

[7] Dong Quiu, Bing Yang (2021). Text summarization based on multi-head self-attention mechanism and poiner network

[8] https://github.com/verak13/pravna-informatikann/tree/main/LegalCasesSummarization/image-error

[9] https://github.com/verak13/pravna-informatikann/tree/main/LegalCasesSummarization/results