Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Jan Bajt

Detekcija političnih mnenj v slovenskih časopisih

diplomsko delo

univerzitetni študijski program
prve stopnje
računalništvo in informatika

Mentor: prof. dr. Marko Robnik Šikonja

Ljubljana, 2021

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Kandidat: Jan Bajt
Naslov: Detekcija političnih mnenj v slovenskih časopisih
Vrsta naloge: Diplomska naloga na univerzitetnem programu prve stopnje Računalništvo in informatika
Mentor: prof. dr. Marko Robnik Šikonja

Opis:
Besedilo teme diplomskega dela študent prepiše iz študijskega informacijskega sistema, kamor ga je vnesel mentor. V nekaj stavkih bo opisal, kaj pričakuje od kandidatovega diplomskega dela. Kaj so cilji, kakšne metode naj uporabi, morda bo zapisal tudi ključno literaturo. TO-DO

Title: Stance detection in Slovenian newspapers

Description:
opis diplome v angleščini

Zahvaljujem se mentorju prof. dr. Marku Robniku Šikonji za vse koristne nasvete in usmerite pri izdelavi diplomskega dela. V prvi vrsti pa se zahvaljujem staršem, saj so me tekom moje izobraževalne poti venomer spodbujali in podpirali.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
BERT	Bidirectional Encoder Representations from Transformers	predstavitev z dvosmernimi enkoderji arhitekture transformer
LDA	latent Dirichlet alocation	latentna Dirichletova alokacija

Povzetek

Naslov: Detekcija političnih mnenj v slovenskih časopisih

Avtor: Jan Bajt

Ključne besede: kontekstne vektorske vložitve, model BERT, LDA, detekcija tematik, detekcija sentimenta.

Abstract

Title: Stance detection in Slovenian newspapers

Author: Jan Bajt

Keywords: contextual word embeddings, model BERT, LDA, topic modeling, sentiment detection.

Poglavje 1

Uvod

Poglavje 2

Pregled področja

Poglavje 3

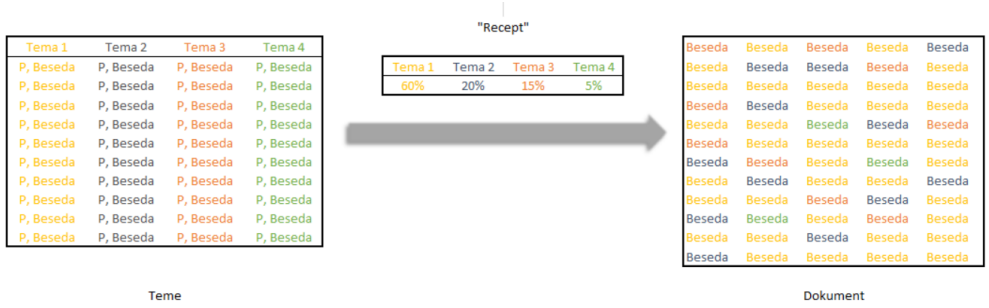
Pregled uporabljenih tehnologij

3.1 Latenta Dirichletova alokacija

Model latentne Dirichletove alokacije (LDA) je eden izmed najbolj uporabljenih modelov za modeliranje tem, ki so ga razvili David Blei, Andrew Ng in Michael Jordan leta 2003 in ga predstavili v članku *Latent Dirichlet Allocation* [2]. Modeliranje tem je metoda nenadzorovanega učenja, kjer iz korpusa besedil poiščemo skrite abstraktne teme, ki se pojavljajo v korpusu besedil [1]. Poleg modela LDA sta med najbolj uporabljenimi modeli še LSA (angl. Latent Semantic Analysis) in pLSA (angl. Probabilistic Latent Semantic Analysis).

Model LDA predpostavlja, da je v celotnem korpusu besedil določeno število tem (npr. 4 teme: Tema 1, Tema 2, Tema 3, Tema 4). Vsaka izmed tem vsebuje verjetnostno porazdelitev besed, ki se v temi nahajajo (npr. za Temo 1: 5% Beseda 1, 4% Beseda 2, 2,5% Beseda 3...). Vsak dokument je zgrajen iz naključne mešanice tem v korpusu (npr. 60% Tema 1, 20% Tema 2, 15% Tema 3 in 5% Tema 4). Iz vsake teme naključno izberemo določeno število besed (iz Teme 1 60% vseh besed v dokumentu, iz Teme 2 20% itd.), ki tvorijo nov dokument. Ta postopek je prikazan na sliki 3.1.

V dejanskem primeru je proces ravno obraten. Na sliki **??** je grafični



Slika 3.1: Primer generiranja dokumenta

prikaz delovanja modela LDA.

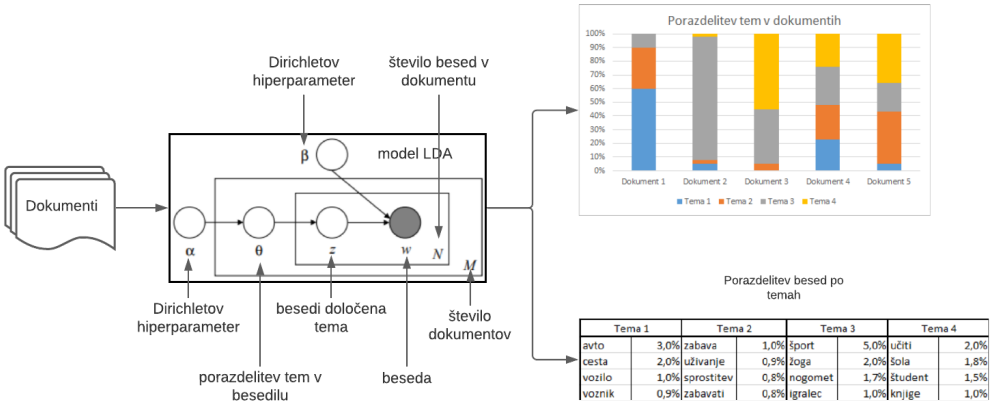
Model LDA na vhod prejme število tem K in korpus besedil, kjer je število M število dokumentov v korpusu, število N pa predstavlja število besed v posameznem besedilu. Na nivoju dokumenta (notranji okvir na sliki 3.2) model vsaki besedi w naključno določimo temo z s tem pa pridobimo porazdelitev tem θ v posameznem dokumentu.

Model ima še dva hiperparametra α in β, ki sta parametra Dirichletove porazdelitve. LDA namreč predvideva, da sta porazdelitvi tem v besedilu in besed v temah Dirichletovi. Hiperparameter α vpliva na porazdelitev tem v posameznih dokumentih, β pa na porazdelitev besed v posameznih temah. Večja vrednost parametra α pomeni, da bodo dokumenti mešanica večjega števila tem, večja vrednost parametra β pa pomeni, da bodo teme mešanica večjega števila besed. Obratno velja za majhne vrednosti obeh parametrov.

Iz korpusa dokumentov in števila tem, ki jih želimo na tem korpusu zaznati, model LDA izračuna dva tipa distribucij:

- distribucije besed za vsako izmed zaznanih tem in
- distribucije tem, ki se pojavljajo v posameznem dokumentu iz korpusa dokumentov.

Izračunan model lahko uporabimo za klasifikacijo novih dokumentov, ki niso del izbranega korpusa. Pri tem model uporabi zgolj besede, ki jih ima



Tema 1		Tema 2		Tema 3		Tema 4	
avto	3,0%	zabava	1,0%	šport	5,0%	učiti	2,0%
cesta	2,0%	uživanje	0,5%	žoga	2,0%	šola	1,8%
vozilo	1,0%	sprostitev	0,8%	nogomet	1,7%	študent	1,5%
voznik	0,9%	zabavati	0,8%	igralec	1,0%	knjige	1,0%

Slika 3.2: Grafični prikaz delovanja modela LDA

na voljo v korpusu besedil, s katerim smo naučili model. To pomeni, da besede iz nevidenega dokumenta, ki jih ni v naučenem modelu, model ne upošteva. Za vsak novi dokument dobimo verjetnostno porazdelitev tem, ki so del dokumenta.

3.2 LDAvis

LDAvis [9] je orodje, ki omogoča interaktivno vizualizacijo tem pridobljenih z modelom LDA. Z orodjem imamo pregled nad vsemi temami in razlikami med njimi, kot tudi pregled besed povezanih z izbrano temo (Slika 3.3). Poleg same vizualizacije pa vpelje še novo mero primernosti (angl. relevance) za vsako besedo znotraj teme.

Vizualizacija modela je razdeljena na dva dela:

- globalen pregled vseh tematik (leva polovica na sliki 3.3) in
- pregled najpogostejših besed znotraj izbrane teme (leva polovica na sliki 3.3).

Z analizo globalnega prikaza vseh tem lahko ugotovimo, kako pogosto se tema v besedilu nahaja in kako so teme med seboj povezane. Posamezne

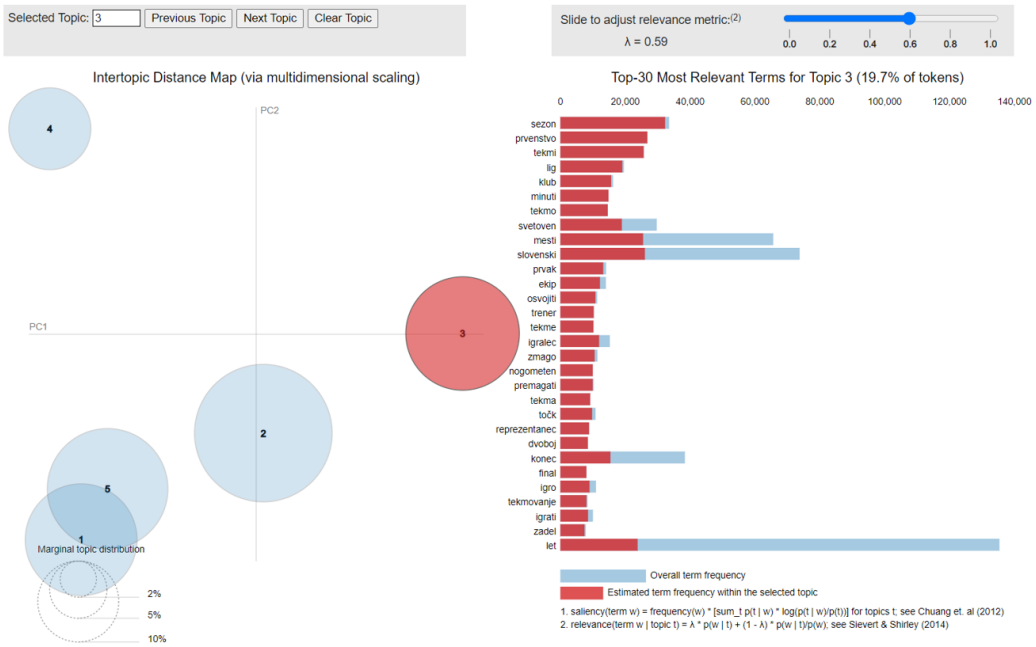
teme so predstavljene s krogi v dvodimenzionalnem prostoru. Večji krogi pomenijo, da je tema bolj razširjena med opazovanimi besedili, razdalja med krogi pa nam pove, kako močno so si teme različne. Dober model LDA ima velike kroge, ki se med sabo ne prekrivajo in so razpršeni po celotnem prostoru.

V desni polovici vizualizacije (Slika 3.3) so v obliki stolpičnega diagrama predstavljene najprimernejše besede za interpretacijo izbrane teme. Modri del posameznega stolpca predstavlja pogostost besede v celotnem korpusu besedil, rdeči del pa predstavlja pogostost besede v izbrani temi.

Pomemben del tega orodja je tudi vpeljava mere primernosti λ (angl. relevance) besede za določeno temo. Mera primernosti λ lahko zaseda vrednosti med 0 in 1. Vrednost $\lambda = 1$ pomeni, da so besede v desni polovici vizualizacije razvrščene po pogostosti besed znotraj izbrane teme (po velikosti rdečega dela stolpca). Nižja kot je vrednost λ, večjo prednost dajemo besedam, ki se bolj izključno pojavljajo v izbrani temi. Avtorji v članku [9] nastavljanje λ na vrednost 0.6. V vizualizaciji lahko to mero prilagajamo in s tem spreminjamo vrstni red besed v desni polovici vizualizacije, kar nam lahko pomaga pri interpretaciji modela.

3.3 BERT

3.3.1 SloBerta



Slika 3.3: Vizualizacija tem z LDAvis

Poglavje 4

Opis podatkov

V diplomski nalogi smo uporabili dve različni podatkovni množici slovenskih člankov. Prvo množico slovenskih člankov smo uporabili pri odkrivanju tematik, drugo pa pri odkrivanju sentimenta.

4.1 Event Registry

Podatkovna množica je bila pridobljena iz podatkovne množice Event registry [5]. Pridobljena podatkovna množica vsebuje članke objavljene med 1. 1. 2014 in 31. 12. 2020, ki so organizirani v arhive po posameznih letih. V posameznem arhivu je shranjenih nekaj tisoč datotek v formatu JSON, v celotni podatkovni množici je okrog 2.2 milijona člankov. Posamezen članek poleg samega besedila vsebuje še nekaj drugih meta podatkov kot so naslov, datum, čas, url in informacije o viru članka.

Za potrebe diplomske naloge smo za vsak članek izluščili njegovo vsebino (angl. body) in naslov (angl. title). Iz podatkovne množice člankov smo vzeli članke iz let 2018, 2019 in 2020 in jih razvrstili po posameznih medijih. V diplomski nalogi smo se omejili na nekatere najbolj popularne slovenske medije (Dnevnik, 24ur,com, RTV Slovenija in Siol.net), izbrali pa smo tudi nekaj medijev iz desnega političnega pola (Nova24TV, Tednik Demokracija

in Portal Politikis).

V tabeli 4.1 so prikazane povprečne dolžine člankov posameznih medijev pri izbranem letu. Iz člankov izbranih medijev smo odstranili tiste z manj kot 25 besedami in duplikate, ostale članke pa smo uporabili pri zaznavanju tematik. Končno število člankov izbranih medijev v posameznem letu je prikazano v tabeli 4.2.

Medij	2018	2019	2020
RTV Slovenija	406	403	452
Siol.net	392	410	405
24ur.com	258	202	214
Svet24	326	340	346
Tednik Demokracija	381	424	463
Nova24TV	433	523	546
Portal Politikis	359	353	356
Dnevnik	214	253	253
Skupaj	345	351	373

Tabela 4.1: Povprečne dolžine (število besed) posameznih izbranih medijev pri določenem letu

4.2 SentiNews

Za učenje sentimenta v diplomski nalogi uporabimo podatkovno zbirko člankov slovenskih časopisov imenovano SentiNews [3, 4]. Celotna podatkovna zbirka je sestavljena iz več kot 250.000 dokumentov s politično, poslovno, ekonomično in finančno vsebino petih slovenskih medijskih virov na spletu (24ur, Dnevnik, Finance, RTV Slovenija in Žurnal24). Več kot 10.000 dokumentov je bilo ročno anotiranih z uporabo Likertove lestvice s petimi stopnjami (1 – zelo negativno, 2 – negativno, 3 – nevtralno 4 – pozitivno in 5 – zalo pozitivno). Te dokumente je anotiralo od 2 do 6 anotatorjev, povprečno vrednost

Medij	2018	2019	2020
RTV Slovenija	32.216	28.948	33.466
Siol.net	27.842	25.871	23.863
24ur.com	7.595	18.831	21.281
Tednik Demokracija	6.291	8.869	8.213
Nova24TV	6.277	6.524	7.170
Portal Politikis	3.915	6.142	5.321
Dnevnik	24.513	20.990	15.304
Skupaj	108.649	116.175	114.618

Tabela 4.2: Število člankov izbranih medijev pri določenem letu

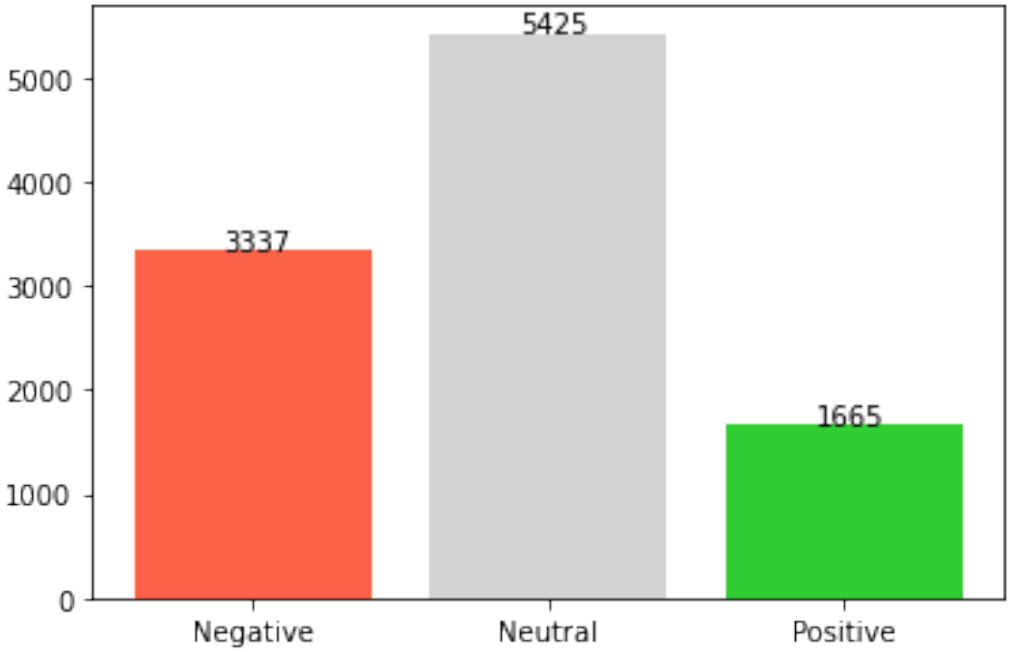
ocene pa so pretvorjene v eno izmed končnih oznak po naslednjih merilih [4]:

- "negative", če je povprečje ocen manj ali enako kot 2.4,
- "neutral", če je povprečje ocen med 2.4 in 3.6,
- "positive", če je povprečje ocen več ali enako kot 3.6.

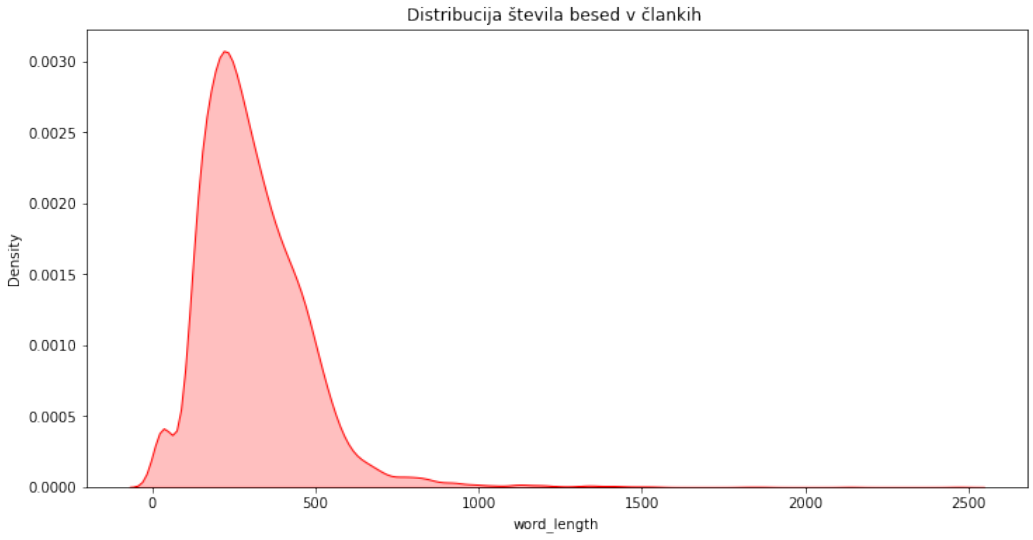
Članki so označeni s sentimentom na treh nivojih: nivo stavka, odstavka in dokumenta.

Za potrebe dilomske naloge smo uporabili članke, ki so s sentimentom označeni na nivoju celotnega dokumenta. Izmed skupno 10.427 anotiranih člankov je 5.425 člankov označenih s pozitivnim , 3.337 z negativnim in 1665 z nevtralnim sentimentom (Slika 4.1

Povprečno število besed v člankih je okrog 309 besed. Na sliki 4.2 je prikazana distribucija števila besed v anotiranih člankih.



Slika 4.1: Število člankov za vsako oznako sentimenta

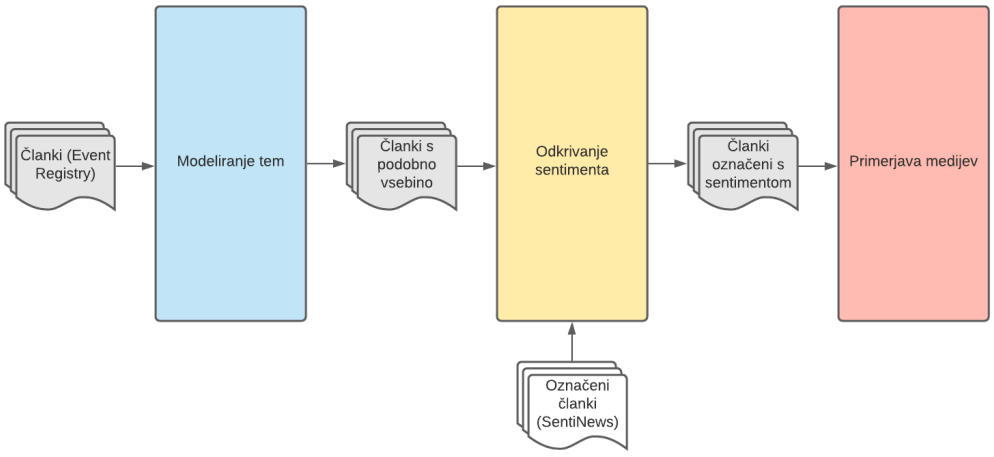


Slika 4.2: Distribucija števila besed v člankih

Poglavje 5

Metodologija

Diplomsko delo lahko razdelimo na tri večje sklope. Prvi sklop je modeliranje tem, kjer želimo iz množice člankov odkriti teme, o katerih pišejo članki (5.1). Drugi sklop predstavlja odkrivanje sentimenta, najprej naučimo model SloBERTa za odkrivanje sentimenta, nato pa s tem modelom klasificiramo izbrane članke (5.2). Zadnji sklop pa je primerjava medijev (5.3). V tem sklopu želimo s pomočjo zaznanih sentimentov na izbranih člankih odkriti razlike v poročanju slovenskih medijev o določeni (politični) tematiki.



Slika 5.1: Splošni proces diplomskega dela

5.1 Modeliranje tem

Z modeliranjem tem želimo v množici slovenskih člankov odkriti različne teme, o katerih pišejo slovenski mediji. Za te teme želimo, da so čimbolj podrobne, da bomo lahko na njih odkrivali sentiment in opravili primerjavo (npr. ena izmed tem bi lahko bila cepljenje za COVID-19 ali menjava vlade v letu 2020 ipd.). Za pridobivanje tako podrobnih tem bi lahko izračunali model LDA za veliko število tem (400 in več), vendar bi si s tem otežili interpretacijo pridobljenih tem. Namesto tega smo najprej izračunali model LDA na celotnem korpusu besedil za majhno število tem (5-15), ki jih hitreje in lažje interpretiramo.

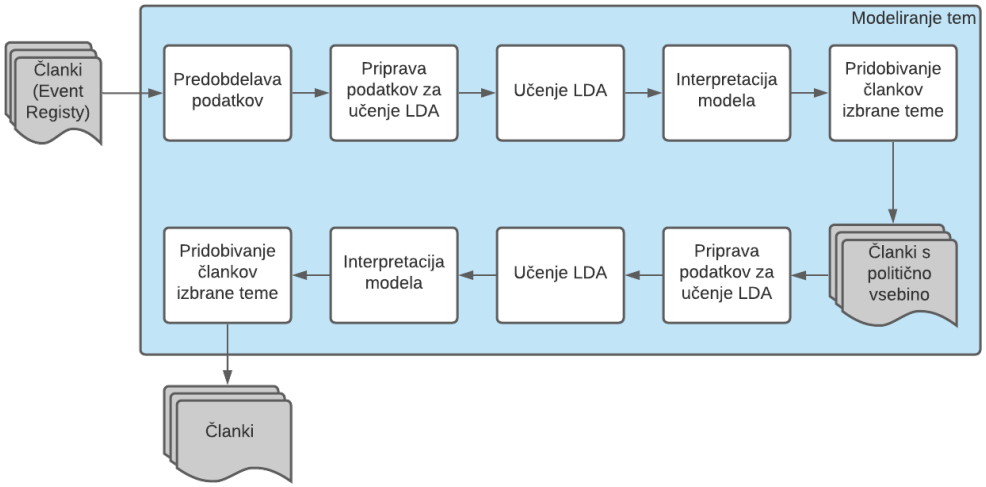
Po opravljeni interpretaciji izbranega modela se odločimo, katere teme nas zanimajo za nadaljno obravnavo. Za namene diplomskega dela se omejimo na politične teme. Iz množice člankov izberemo tiste z najvišjimi verjetnostmi, da pripadajo politični tematiki (verjetnosti so ¿ 0.8). Tako dobimo novo množico člankov, s katerimi zgradimo nov model LDA. Postopek modeliranja tem je identičen zgoraj opisanem, le da nam tokrat ni potrebno predprocesirati člankov. Pripraviti moramo zgolj slovar besed in korpus člankov predstaviti kot vreče besed, postopek učenja in interpretacije pa je enak opisanemu.

Na koncu iz nekaj tem izluščimo najbolj verjetne članke in jih uporabimo pri odkrivanju sentimenta in primerjavi medijev.

Modeliranje tem sestavljajo naslednji koraki (Slika 5.2):

1. Predobdelava člankov (podpoglavje 5.1.1)
2. Priprava podatkov za računanje modela LDA (podpoglavje 5.1.2)
3. Učenje modela LDA (podpoglavje 5.1.3)
4. Interpretacija modela (podpoglavje 5.1.4)
5. Izbor teme in člankov za računanje podrobnejšega modela

6. Ponovimo 2., 3. in 4. korak
7. Pridobivanje člankov za nadaljno analizo.



Slika 5.2: Postopek pridobivanja člankov določene teme

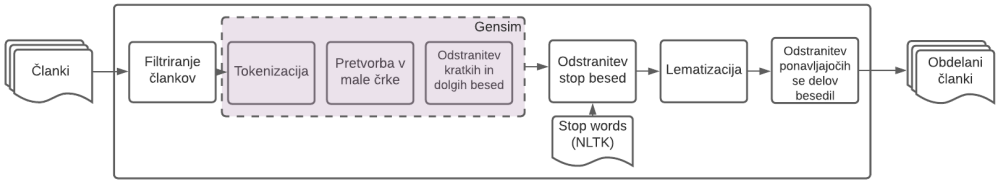
5.1.1 Predobdelava člankov

Postopek predobdelave podatkov igra eno izmed ključnih vlog v odkrivanju tematik. V tem postopku iz besedil izluščimo tiste dele, ki so za nas pomembni. Postopek predobdelave besedil člankov sestavljajo naslednji koraki (Slika 5.3):

1. Filtriranje člankov
2. Tokenizacija
3. Pretvorba besed v male črke
4. Odstranitev besed krajših oz. daljših določene dolžine
5. Odstranitev t.i. stop besed (angl. stop words)

6. Lematizacija

7. Odstranitev ponavljajočih se delov besedil



Slika 5.3: Proces predobdelave podatkov

V prvem koraku smo članke filtrirali. Najprej smo odstranili vse članke, ki so krajši od določenega števila besed. Odstranili smo tudi vse podvojene članke, t.j. članke z identičnimi naslovi. Pri odstranjevanju duplikatov je potrebno poudariti, da smo odstranili zgolj članke s popolnoma identičnimi naslovi, ne pa tudi člankov, kjer je spremenjena zgolj določena beseda, je pa očitno, da gre za enako vsebino članka.

Naslednje tri korake predobdelave smo opravili s pomočjo metode *simple_preprocess* iz knjižnice *Gensim* [8]. Metoda besedilo razdeli na posamezne besede, jih pretvori v male črke in odstrani vse besede, ki so krajše oz. daljše od določene dolžine.

Iz preostalih besed odstranimo še t.i. *stop besede* (angl. stopwords). Stop besede so besede, ki nimajo posebnega pomena v povedih (npr. vezniki, zaimki, imena mesecev...) Seznam stop besed za slovenski jezik smo pridobili iz knjižnice *NLTK* [7].

Vse preostale besede smo pretvorili v osnovno obliko (lematizacija). Za lematizacijo slovenskih besed smo uporabili orodje *Classla* [6].

V zadnjem koraku smo pregledali predobdelane članke posameznih medijev in pri določenih medijih opazili ponavljajoče dele besedil, katere smo odstranili. Pri večih medijih se je v člankih velikokrat pojavljala beseda "foto", ki je bila v spletni obliki članka del podnapisa priloženim fotografi-

jam. Poleg omenjene besede so se ponavljali tudi določene besede virov kot na primer "reuters", "getty images", "urbanec"in športid", ki predstavljajo vir fotografije in smo jih prav tako odstranili iz seznama besed. Pri mediju 24ur.com smo opazili ponavljajoč začetni del besedila, ki od uporabnika zahteva omogočenje piškotkov spletne strani. Članki medija Siol.net Novice so imeli prav tako ponavljajoč začetni del besedila, ki se je nanašal na t.i. *termometer*, ki bralcu članka razloži vlogo le-tega pri poročanju o popularnosti članka. Pri ostalih medijih večjih ponavljajočih se delov na začetku člankov nismo opazili.

5.1.2 Priprava podatkov za model LDA

S postopkom predobdelave podatkov smo iz člankov izluščili posamezne besede, ki nam lahko nekaj povejo o temah člankov. V člankih se določene besede večkrat pojavljajo skupaj (npr. Marjan Šarec, Janez Janša, državni zbor itd.), to pa nam lahko pomaga pri sami interpretaciji tem. Zato v naslednjem koraku v člankih zaznamo pogoste dvojice (angl. bigrams) in trojice (angl. trigrams) besed. Knjižnica *Gensim* [8] ponuja model za avtomatsko zaznavanje pogostih besednih zvez imenovan *Phrases* (Primer 1. Zaznane besedne zveze smo nato pretvorili v en sam niz besed ločenih s podčrtajem in jih dodali v seznam besed preprocesiranih člankov.

```
from gensim import models

# documents predstavlja seznam predobdelanih dokumentov

bigram = models.Phrases(documents, min_count=10, threshold=80)
bigram_mod = models.phrases.Phraser(bigram)
texts = documents.to_list()
bigrams = [bigram_mod[doc] for doc in texts]
```

Listing 1: Primer iskanja dvojic besed v besedilih

Iz predobledanih člankov je potrebno pridobiti podatke, ki jih potrebujemo za učenje modela LDA. V diplomskem delu za učenje modela LDA uporabimo knjižnico *Gensim* [8], ki na vhodu sprejme t.i. vrečo besed (angl. bag of words) za vsak članek in slovar besed (angl. dictionary) v celotnem korpusu besedila. Slovar besed vsebuje vse unikatne besede iz korpusa predobdelanih besedil, za vsako od teh besed pa določi unikatno identifikacijsko število (id). S pomočjo slovarja knjižnice *Gensim* tvorimo vrečo besed za vsak članek v korpusu z uporabo metode slovarja *doc2bow*, kot je prikazano v delu kode 2.

S tem imamo pripravljen slovar in korpus člankov predstavljenih z vrečami besed in lahko začnemo z učenjem modela LDA.

```
from gensim.corpora import Dictionary

clanki = [['pes', 'človek', 'prijatelj', 'pes'], ['avto', 'cesta', 'voziti']]
slovar = Dictionary(clanki)
korpus = [slovar.doc2bow(clanek) for clanek in clanki]
```

Listing 2: Enostaven primer pridobivanja slovarja in korpusa člankov predstavljenih z vrečami besed

5.1.3 Učenje LDA modela

V poglavju 5.1.1 smo pripravili podatke, ki jih potrebujemo za učenje modela LDA, poglavju 3.1 pa razložili delovanje samega modela LDA. V tem delu pa bomo razložili proces pridobivanja tem iz množice člankov slovenskih medijev.

Implementacijo modela LDA nam ponuja knjižnica *Gensim*. Na vhod prejme število tem, ki jih želimo odkriti v besedilu, slovar besed in korpus člankov predstavljeni z vrečami besed. Postopek pridobitve slovarja besed in korpusa člankov smo opisali v poglavju 5.1.1, določiti moramo še število tematik.

Glavno merilo evalvacije modela LDA je smiselnost in interpretabilnost tem, zato smo izračunali več modelov z različnim številom tematik. Te modele smo poizkusili interpretirati, za nadaljevanje naše naloge pa smo uporabili tisti model, ki se je nam zdel najbolj interpretabilen in smiseln.

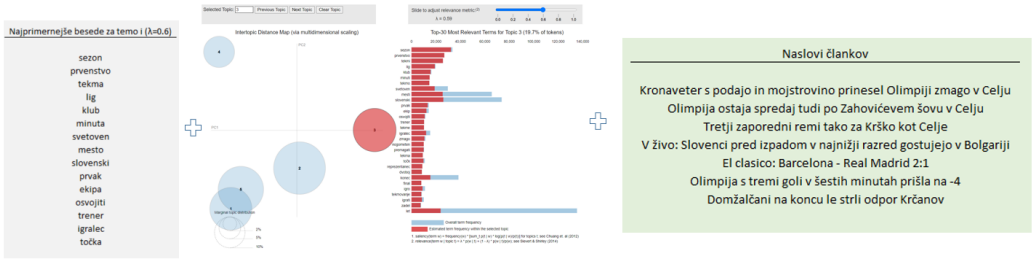
5.1.4 Interpretacija tem

Pri postopku interpretacije modelov smo si v glavnem pomagali z najpogostejšimi besedami posameznih tem. Poleg teh besed smo si pomagali tudi s knjižnico *pyLDAvis* [9]. S pomočjo te knjižnice lahko vizualiziramo model oz. teme modela in primerjamo podobnosti med temami. Pomembna je tudi mera primernosti (angl. relevance), s katero damo prednost besedam, ki pripadajo opazovani temi v večji meri kot v drugih temah. Tako namesto najpogostejših tem uporabimo besede, ki so najbolj primerne (visoka vrednost mere primernosti) za opazovano temo.

Poleg omenjenega si pomagamo še s samimi članki. Vsakemu članku smo najprej določili temo, kateri pripada v največji meri (največja verjetnost). Članke smo nato združili po temah in za vsako temo izbrali nekaj člankov, ki najbolje predstavljajo posamezno temo (imajo najvišjo verjetnost, da pripadajo temi). Iz teh člankov smo izluščili naslove in jih uporabili pri interpretaciji.

V postopku interpretacije tem določimo temam naslove s pomočjo omenjenih treh elementov (Slika 5.4):

- vizualizacija s pyLDAvis,
- najpogostejše oz. najprimernejše besede določene teme,
- naslovi člankov.



Slika 5.4: Elementi interpretacije

5.2 Odkrivanje sentimenta

5.3 Primerjava medijev

Poglavje 6

Evalvacija

Poglavje 7

Rezultati

Poglavje 8

Zaključek

Članki v revijah

[1] David Blei, Lawrence Carin in David Dunson. "Probabilistic Topic Models: A focus on graphical model design and applications to document and image analysis". V: *IEEE signal processing magazine* 27 (nov. 2010), str. 55–65. DOI: 10.1109/MSP.2010.938079.

[2] David M Blei, Andrew Y Ng in Michael I Jordan. "Latent dirichlet allocation". V: *the Journal of machine Learning research* 3 (2003), str. 993–1022.

[4] Jože Bučar, Martin Žnidaršič in Janez Povh. "Annotated news corpora and a lexicon for sentiment analysis in Slovene". V: *Language Resources and Evaluation* 52.3 (2018), str. 895–919.

[8] Radim Rehurek in Petr Sojka. "Gensim–python framework for vector space modelling". V: *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic* 3.2 (2011).

Članki v zbornikih

[5] Gregor Leban in sod. "Event registry: learning about world events from news". V: apr. 2014, str. 107–110. DOI: 10.1145/2567948.2577024.

[6] Nikola Ljubešić in Kaja Dobrovoljc. "What does Neural Bring? Analysing Improvements in Morphosyntactic Annotation and Lemmatisation of Slovenian, Croatian and Serbian". V: *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*. Florence, Italy: Association for Computational Linguistics, avg. 2019, str. 29–34. DOI: 10.18653/v1/W19-3704. URL: https://www.aclweb.org/anthology/W19-3704.

[7] Edward Loper in Steven Bird. "NLTK: The Natural Language Toolkit". V: *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics*. 2002.

[9] Carson Sievert in Kenneth Shirley. "LDAvis: A method for visualizing and interpreting topics". V: *Proceedings of the workshop on interactive language learning, visualization, and interfaces*. 2014, str. 63–70.

Celotna literatura

[1] David Blei, Lawrence Carin in David Dunson. "Probabilistic Topic Models: A focus on graphical model design and applications to document and image analysis". V: *IEEE signal processing magazine* 27 (nov. 2010), str. 55–65. DOI: 10.1109/MSP.2010.938079.

[2] David M Blei, Andrew Y Ng in Michael I Jordan. "Latent dirichlet allocation". V: *the Journal of machine Learning research* 3 (2003), str. 993–1022.

[3] Jože Bučar. *Manually sentiment annotated Slovenian news corpus SentiNews 1.0*. Slovenian language resource repository CLARIN.SI. 2017. URL: http://hdl.handle.net/11356/1110.

[4] Jože Bučar, Martin Žnidaršič in Janez Povh. "Annotated news corpora and a lexicon for sentiment analysis in Slovene". V: *Language Resources and Evaluation* 52.3 (2018), str. 895–919.

[5] Gregor Leban in sod. "Event registry: learning about world events from news". V: apr. 2014, str. 107–110. DOI: 10.1145/2567948.2577024.

[6] Nikola Ljubešić in Kaja Dobrovoljc. "What does Neural Bring? Analysing Improvements in Morphosyntactic Annotation and Lemmatisation of Slovenian, Croatian and Serbian". V: *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*. Florence, Italy: Association for Computational Linguistics, avg. 2019, str. 29–34. DOI: 10.18653/v1/W19-3704. URL: https://www.aclweb.org/anthology/W19-3704.

[7] Edward Loper in Steven Bird. "NLTK: The Natural Language Toolkit". V: *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics*. 2002.

[8] Radim Rehurek in Petr Sojka. "Gensim–python framework for vector space modelling". V: *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic* 3.2 (2011).

[9] Carson Sievert in Kenneth Shirley. "LDAvis: A method for visualizing and interpreting topics". V: *Proceedings of the workshop on interactive language learning, visualization, and interfaces*. 2014, str. 63–70.