University of Ljubljana
Faculty of Computer and Information Science

Juš Hladnik

Topic analysis of Slovenian news and social media

master's thesis

the 2nd cycle master's study programme
computer and information science
Track: Data Science

Supervisor: prof. dr. Marko Robnik Šikonja

Ljubljana, 2023

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Juš Hladnik

Tematska analiza slovenskih novic in družbenih omrežij

magistrsko delo

magistrski študijski program druge stopnje
računalništvo in informatika
Smer: Podatkovne Vede

Mentor: prof. dr. Marko Robnik Šikonja

Ljubljana, 2023

Acknowledgments

I would like to express my gratitude to my supervisor, prof. dr. Marko Robnik Šikonja, for his great advice, responsiveness, and kindness.

I wish to thank Anže, Matic, and Miha, for making college a fun experience and for their help with course work.

Finally, I would like to thank my family who supported me in all ways possible.

Juš Hladnik, 2023

"All models are wrong, but some are useful."

— George E. P. Box

Contents

List of used acronmys

acronym	meaning
NLP	natural language processing
LDA	latent Dirichlet allocation
LSA	latent semantic analysis
PLSA	probabilistic latent semantic analysis
LSI	latent semantic indexing
NLP	natural language processing
NMF	non-negative matrix factorization
SVD	singular value decomposition
TF-IDF	term frequency-inverse document frequency
NPMI	normalized pointwise mutual information
BOW	bag-of-words
UMAP	Uniform Manifold Approximation and Projection
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise
PCA	principal component analysis
BERT	bidirectional encoder representations from transformers
SBERT	sentence bidirectional encoder representations from transformers
t-SNE	t-distributed stochastic neighbor embedding
ARI	Adjusted Rand Index
MBTS	maximum bipartite topic similarity
NPMI	normalized pointwise mutual information
GloVe	Global Vectors for Word Representation
PV-DM	paragraph vectors distributed memory
PV-DBOW	paragraph vectors distributed bag-of-words

Abstract

Title: Topic analysis of Slovenian news and social media

Topic modeling is an unsupervised machine learning technique that aims to discover hidden semantic structures within large collections of text documents, thus facilitating the exploration and understanding of vast textual data.

We conduct a comprehensive comparison of four popular topic modeling algorithms, namely LDA, NMF, Top2vec, and BERTopic, in the context of the Slovenian language. To assess the performance of these algorithms we use topic coherence and topic diversity quantitative evaluation and additionally manually interpret extracted topics. Our results demonstrate that all models achieve higher topic coherence on the news corpus compared to tweets. While BERTopic is the only algorithm to produce satisfactory results on the tweets corpus, all models perform well on the news corpus.

Furthermore, we introduce a novel method, MBTS (Maximum Bipartite Topic Similarity), for comparing the similarity of topic models and evaluating their stability. This method relies on semantic similarity and maximum bipartite matching. Our findings have important implications for the selection and application of topic modeling algorithms in the context of the Slovenian language and beyond.

Keywords

topic modeling, language models, Slovene language, topic model stability and similarity, natural language processing

Povzetek

Naslov: Tematska analiza slovenskih novic in družbenih omrežij

Modeliranje tem je nesupervizirana metoda strojnega učenja, ki si prizadeva odkriti skrite semantične strukture znotraj velikih zbirk dokumentov, s čimer omogoča raziskovanje in razumevanje obsežnih besedilnih podatkov.

Celovito primerjamo štiri priljubljene algoritme za modeliranje tem, in sicer LDA, NMF, Top2vec in BERTopic, v kontekstu slovenskega jezika. Modele kvantitativno ovrednotimo z metrikama koherentnost tem in raznolikost tem, poleg tega odkrite teme tudi ročno pregledamo in interpretiramo. Naši rezultati kažejo, da vsi modeli dosegajo višjo koherenco tem na korpusu novic v primerjavi s tviti. Medtem ko je BERTopic edini algoritem, ki na korpusu tvitov dosega zadovoljive rezultate, vsi modeli na korpusu novic dosegajo dobre rezultate.

Poleg tega predstavlimo novo metodo, MBTS (največja dvostranska podobnost tem), za primerjanje podobnosti modelov za modeliranje tem in ocenjevanje njihove stabilnosti. Ta metoda temelji na semantični podobnosti in maksimalnem dvostranskem ujemanju. Naše ugotovitve imajo pomembne posledice za izbiro in uporabo algoritmov za modeliranje tem v kontekstu slovenskega jezika in širše.

Ključne besede

modeliranje tem, jezikovni modeli, slovenščina, stabilnost in podobnost modelov za modeliranje tem, obdelava naravnega jezika

Razširjeni povzetek

V dobi velikih količin podatkov so učinkovito iskanje, organiziranje, združevanje in analiziranje nestrukturiranih besedil velik izziv. Modeliranje tem je nenadzorovana tehnika strojnega učenja, ki učinkovito odkriva latentne teme iz velikih korpusov nestrukturiranih besedil. Z odkrivanjem semantičnih struktur v dokumentih omogoča povzemanje, združevanje in iskanje ustreznih informacij. Pogosto se uporablja na različnih področjih, od analize mnenj v trženju do prepoznavja trendov na finančnih trgih.

Klasični modeli za modeliranje tem, kot sta latentna Dirichletova alokacija (LDA) [1] in ne-negativna matrična faktorizacija (NMF) [2], obstajajo že od začetka 2000-ih let, vendar imajo težave s krajšimi besedili. V zadnjem času so veliki jezikovni modeli in jezikovne vložitve privedli do izboljšanih metod za modeliranje tem, kot sta top2vec [3] in BERTopic [4]. Ti nevronski modeli za odkrivanje tem uporabljajo vektorske vložitve dokumentov in gručenje, vendar njihova učinkovitost na različnih področjih in v različnih jezikih še ni bila podrobno raziskana.

Še posebej malo raziskav je bilo narejenih na področju modeliranja tem v jezikih z malo viri, kot je slovenščina. V nalogi bomo primerjali klasične statistične modele z novejšmi modeli, ki temeljijo na nevronskih mrežah. Uporabili bomo dva korpusa slovenskih besedil, novice in tvite, ter ocenjevali njihovo uspešnost z metrikami kot sta koherentnost in raznolikost teme.

Poleg tega razvijemo tudi novo metodo za ocenjevanje podobnosti modelov za modeliranje tem, ki temelji na semantični podobnosti in maksimalnem dvostranskem ujemanju.

I Kratek pregled sorodnih del

Zgodnji modeli na področju obdelave naravnega jezika, kot sta BOW [5] in TF-IDF [6], so dokumente predstavljali kot števila posameznih besed v dokumentu. Osnovni algoritmi za modeliranje tem so vključevali metrike za podobnost besedila [7, 8] in metode gručenja [9, 10]. Modeliranje tem se je pozneje razvilo z bolj izpopolnjenimi pristopi, ki so teme predstavili z besedami in dokumentom določili teme. LSI [11] je bil med prvimi uspešnimi modeli, ki so uporabljali SVD za primerjavo dokumentov. PLSA [12] je uvedla generativni podatkovni model, ki namesto SVD uporablja statistično modeliranje na podlagi porazdelitve tem v dokumentih.

LDA [1] je Bayesovska izboljšava PLSA in odpravlja težave z linearno naraščajočimi parametri ocenjevanja in nezmožnostjo dodeljevanja verjetnosti novim dokumentom. NMF [2] se od verjetnostnih modelov razlikuje po aproksimaciji faktorizacije matrike beseda-dokument v matriko značilk in matriko skritih spremenljivk (tem).

Nevronski jezikovni modeli [13] so vpeljali porazdeljene besedne predstavitve, kot je word2vec [14], ki zajemajo sintaktične in semantične besedne povezave. Besede so predstavljene z gostimi vektorskimi vložitvami, ki so si podobne za besede, uporabljene v podobnih kontekstih. Drugi primeri, ki so skušali izboljšati word2vec vložitve, vključujejo GloVe [15], FastText [16] in ELMo [17]. Doc2vec [18] ustvari goste vektorje, ki predstavljajo dokumente. Ti vektorji se natrenirani za napovedovanje besed v dokumentih. Doc2vec je podoben word2vec, razen da med treniranjem modela za napovedovanje drugih besednih vektorjev doda še vektor dokumenta.

Pred kratkim sta top2vec [3] in BERTopic [4] dosegla izboljšave v primerjavi s tradicionalnimi metodami. Oba dosegata veliko boljše kvalitativne in kvantitativne rezultate, saj iz korpusa izluščita bolj informativne in reprezentativne teme.

Top2vec izkorišča skupno semantično vložitev dokumentov in besed za iskanje vektorskih vložitev tem. Pri tem predpostavlja, da se dokumenti, ki so si v vektorskem prostoru blizu, lahko interpretirajo kot dokumenti z isto

temo. Top2vec ne zahteva odstranjevanja zaustavitvenih besed, lematizacije in korenjenja besed ter sam najde število tem, kar je velik napredek v primerjavi s tradicionalnimi modeli. Model BERTopic pristopa k modeliranju tem kot nalogi gručenja in jo nadgradi. Privzeto uporablja model SentenceBERT [19] za generiranje vložitev dokumentov in jih združuje v gruče ter ustvarja predstavitve tem s postopkom, ki temelji na TF-IDF. Podobno kot top2vec, BERTopic uporablja UMAP [20] za redukcijo dimenzionalnosti in HDBSCAN [21] za gručenje.

Narejenih je bilo veliko raziskav na področju modeliranja tem za novice, predvsem ker so standardni korpusi, ki se najbolj pogosto uporabljajo za evalvacijo, sestavljeni iz novic [22, 23, 24, 25, 26, 27]. Prav tako je bilo narejeno dosti raziskav področju modeliranja tem tvitov [28, 29, 30, 31] in ugotavljajo pomankljivosti LDA za krajše dokumente, raziskujejo preference uporabnikov Twitterja ter dvome o cepivu za COVID.

Bajt in Robnik-Šikonja [32] primerjata slovenske medijske novice z analizo tem in sentimenta člankov. Za analizo tem uporabljata LDA na dveh ravneh. Prva raven izlušči splošne teme, nato pa izvedeta LDA na podmnožici člankov za vsako temo, da pridobita podteme člankov. Nato primerjata odnose različnih medijev do posameznih tem.

Berginc in Ljubešić [33] uporabljata LDA za modeliranje tem dveh velikih korpusov slovenskega jezika. Gigafida večinoma vsebuje tiskana besedila, slWaC pa vsebuje besedila iz spleta. Teme izluščita ločeno za vsak korpus in jih nato primerjata. Čeprav so nekatere teme najdene v obeh korpusih, trdita, da se korpusa precej razlikujeta glede na teme v njih.

Raznolikost tem, ki so jo predstavili Dieng et al. [34], in koherenca tem, zlasti NPMI [35], sta dve najbolj razširjeni metriki za avtomatizirano ocenjevanje modelov za modeliranje tem. Kljub temu pa Grootendorst [4] navaja, da lahko mere kakovosti tem, kot sta raznolikost tem in koherenca tem, služijo le kot približek uspešnosti modelov, ter da je velikokrat potrebna človeška interpretacija.

II Predlagana metoda

Za primerjavo modelov za modeliranje tem v slovenščini uporabimo dva korpusa besedil. Prvi zajema 2.9 miljona člankov iz slovenskih medijskih portalov med leti 2014 in 2021, drugi pa vsebuje 60 milijonov tvitov v slovenskem jeziku med letoma 2006 in 2021. V nalogi smo naključno izbrali 50.000 dokumentov iz vsakega korpusa iz leta 2020 za zmanjšanje časovne zahtevnosti. Predprocesiranje besedil vključuje odstranjevanje zaustavitvenih besed, posebnih znakov, številk in drugih nesmiselnih žetonov ter lematizacijo besedil z orodji CLASSLA-Stanza.

Za analizo tem v korpusih novic in tvitov predlagamo modele LDA, NMF, top2vec in BERTopic. LDA je generativni verjetnostni model, ki odkriva teme z iterativnim dodeljevanjem besed temam in tem dokumentom ter to ponavlja dokler ne doseže stabilne porazdelitve. NMF faktorizira matriko dokumentov in besed v dve matriki nižje dimenzije, ki predstavljata razmerja med dokumenti in temami ter med temami in besedami. Top2Vec prepoznava teme z vložitvami dokumentov in besed v isti vektorski prostor, pri čemer izkorišča strukturo vložitvenega prostora za iskanje gostih gruč podobnih dokumentov. BERTopic uporabi vložitve dokumentov generiranih z modelom BERT ter s pomočjo zmanjšanja dimenzionalnosti, gručenja podobnih dokumentov in uporabo prilagojenega TF-IDF odkrije teme v korpusu.

II.I Metode ocenjevanja

Za kvantitativno ocenjevanje modelov uporabimo metriki koherentnost tem in raznolikost tem. Najpogosteje uporabljena metrika za merjenje koherentnosti teme je NPMI, ki je pozitivno korelirana z ocenami ljudi pri nalogah prepoznavanja tujk v množici besed ter rangiranja kakovostnih tem. Raznolikost tem merimo z deležom unikatnih besed v prvih n besedah, ki predstavljajo temo.

Ker pa je kvantitativno ocenjevanje modelov za modeliranje tem pogosto nenatančno, odkrite teme tudi ročno pregledamo in interpretiramo ter modele primerjamo med sabo.

Dodatno razvijemo metodo za ocenjevanje podobnosti in stabilnosti modelov za modeliranje tem, ki jo poimenujemo največja dvostranska podobnost tem (eng. maximum bipartite topic similarity), saj temelji na največjem dvostranskem ujemanju in semantični podobnosti besed, ki predstavljajo teme.

Modeli izluščijo k tem, od katerih je vsaka predstavljena z n besedami. Ker vrstni red besed, ki predstavljajo temo, nima semantičnega pomena, uporabljamo maksimalno uteženo dvostransko ujemanje za izračun podobnosti dveh tem. Izračunamo podobnosti med vsemi pari besed, ki predstavljajo dve temi, in zgradimo maksimalno uteženo dvostransko ujemanje. Podobnost dveh tem je nato povprečje uteži v grafu. Enak postopek razširimo na izračun podobnosti dveh modelov za modeliranje tem. Vsak model odkrije k tem, zato izračunamo podobnosti za vse pare tem, kot že opisano. Ponovno uporabimo maksimalno uteženo dvostransko ujemanje, kjer so teme vozlišča v grafu, uteži pa podobnosti med temami. Podobnost dveh modelov je nato definirana kot povprečje uteži v maksimalnem uteženem dvostranskem grafu. Za izračun podobnosti dveh besed uporabimo kosinusno razdaljo med vektorskima vložitvama besed, pridobljenih z npr. Word2Vec modelom.

III Eksperimentalna evaluacija

V prvem poskusu smo analizirali vpliv števila dokumentov v korpusu na delovanje modelov. Standardna deviacija se ne zmanjšuje s povečevanjem števila dokumentov, zato smo nadaljevali s 50.000 dokumenti. Koherentnost tem se poveča z večjim številom dokumentov na korpusu tvitov, ne pa tudi na korpusu novic kar nakazuje, da potrebujemo večje število dokumentov, če so ti kratki. S povečevanjem števila besed, ki predstavljajo temo se zmanjša tako koherentnost kot raznolikost tem na obeh korpusih. Pri povečanju števila tem se izboljša koherentnost tem na korpusu novic, kar pomeni da je korpus novic bolj raznolik in vsebuje več specifičnih tem kot korpus tvitov.

Pri ročnem pregledu odkritih tem smo ugotovili, da BERTopic v obeh korpusih odkrije teme, ki so koherentne, raznolike in dovolj specifične, medtem ko top2vec, LDA in NMF odkrijejo kakovostne teme le na korpusu novic.

Pri zmanjšanju števila tem se koherentnost na tvitih rahlo poveča pri LDA, NMF in top2vec, vendar vseeno ne doseže koherentnosti tem modela BERTopic. Na korpusu novic se predvsem izkaže top2vec z zelo informativnimi in specifičnimi besedami, ki dobro predstavljajo teme.

S primerjanjem podobnosti in stabilnostni modelov z metriko MBTS ugotovimo, da top2vec ni podoben ostalim modelom tako na korpusu novic kot tvitov. Na korpusu novic sta si med sabo najbolj podobna LDA in NMF, BERTopic pa je rahlo bolj stabilen od ostalih. Prav tako je BERTopic najbolj stabilen na korpusu tvitov, sledi mu model NMF. Pri zmanjšanju števila tem se na korpusu novic vsem modelom izboljša stabilnost. Z MBTS primerjamo tudi podobnost tem med obema korpusoma. Za bolj obširno primerjavo bi potrebovali večje število korpusov.

IV Sklep

V nalogi smo raziskali uporabo štirih algoritmov za modeliranje tem (NMF, LDA, top2vec in BERTopic) za slovenski jezik z uporabo korpusa novic in korpusa tvitov. Modele smo kvantitativno ocenili z metrikama koherentnost tem in raznolikost tem. Rezultati so pokazali nižjo koherenco tem za vse modele na korpusu tvitov v primerjavi s korpusom novic. Kljub prekrivajočim se intervalom zaupanja je BERTopic v mnogih primerih prekašal druge modele. Koherenca tem in raznolikost sta se zmanjšali, ko smo povečali število besed, ki predstavljajo temo. Modeli so na korpusu novic odkrili bolj raznolike in specifične teme, pri čemer se je koherenca tem povečala za vse modele, ko smo povečali število tem.

Predstavili smo tudi novo metodo MBTS za ocenjevanje stabilnosti in podobnosti modelov za modeliranje tem, ki temelji na semantični podobnosti in maksimalnem dvostranskem ujemanju. Ugotovili smo, da je BERTopic najstabilnejši model, sledi mu NMF. LDA in NMF sta si najbolj podobna, top2Vec pa se je bistveno razlikoval od ostalih modelov. V prihodnje bi lahko eksperimentirali z alternativnimi prednatreniranimi kodirniki stavkov, natrenirali kodirnike stavkov samo na slovenščini ali uporabljali tehniko ansamblov

za modeliranje tem za izboljšanje zmogljivosti in stabilnosti. Metoda MBTS se lahko uporabi za ocenjevanje modelov tem na različnih naborih podatkov in jezikih, z možnostjo nadaljnjega raziskovanja z uporabo različnih vložitev besed ter metod za primerjanje besed in tem.

Chapter 1

Introduction

In the era of big data, there is a great challenge of being able to efficiently search, organize, cluster and analyze unstructured text. Topic modeling is an unsupervised machine learning technique that extracts latent topics from large text corpora. As the name suggests, it identifies latent semantic structure or topics present in a collection of documents. With identified topics, we can summarize a corpus, cluster documents and use them to retrieve relevant documents.

In the real world, topic modeling has found applications across various industries and sectors, such as finance, healthcare, marketing, and policy-making. For example, in finance, topic models have been used to analyze news articles and social media posts to identify market trends, assess investor sentiment, and predict stock prices. In marketing, topic modeling has been utilized to analyze customer reviews and feedback. In policy-making, topic models have been employed to analyze public opinion.

A topic is a thing being discussed and usually it is described with a set of words or a distribution over them. It can therefore be divided into subtopics and topics often overlap, so determining the right topic granularity and amount of words representing the topic can be challenging.

Classical topic models such as Latent Dirichlet Allocation (LDA) [1] and Non-negative matrix factorization (NMF) [2] have been known since early 2000's. Although LDA is a widely used topic model it has been shown that statistical models struggle with shorter texts and documents.

With the recent development of large language models and text embeddings there have also been improvements in topic modeling techniques. Two most notable approaches are top2vec [3] and BERTopic [4], which both embed documents in some way and use clustering for topic discovery. Because these neural topic models are novel there has not been a lot of research done on how they perform in various domains, languages and applications.

There is especially a lack of research on topic modeling for less-resourced languages such as Slovene. In this thesis we will compare classical statistical topic models with newer neural topic models on Slovene language, both on formal language documents (news) and short texts with informal language (tweets).

In the following sections we will introduce the related work in the areas of general topic modeling, topic modeling for news and tweets and topic modeling for Slovene language. After that we state the contributions of our work and provide the thesis structure.

1.1 Related work

Topic modelling

Topic modelling is an area of research in natural language processing (NLP) that aims to extract latent topics from a corpus of unlabeled documents. It is therefore an unsupervised machine learning method. Topic modeling first evolved from simple text representation models. One of these early models was the bag-of-words (BOW) model [5] which represents text as counts of each term present in the text. Another simple model is the term frequency-inverse document frequency (TF-IDF) [6]. This model weighs the number of occurrences of terms in a text against the frequency of their appearance in other documents within the corpus.

Basic algorithms for topic discovery range from simple text similarity metrics such as Jaccard similarity [7], Euclidean distance and Dice's coefficient [8] to clustering techniques that work with BOW and TF-IDF vectors such as hierarchical clustering [9] and k-means clustering [10].

Topic modeling then evolved as a separate research area with more elaborate approaches that not only cluster similar documents but also produce topic interpretation with clusters of words and n-grams that best represent them. On the other hand these approaches assign topics to documents in corpus.

One of the first successful topic models was Latent Semantic Indexing (LSI) introduced by Deerwester et al. [11] in 1990. LSI assumes that similar documents will have similar distribution of word frequencies for some or all words. It works by applying the singular value decomposition (SVD) to the term-document matrix. As a result we get matrix U, a document-topic matrix, matrix V, known as the term-topic matrix and a diagonal matrix S with singular values. By taking t largest singular values and multiplying the truncated U, S and V we get low dimensional latent vectors that are not sparse. With these we can compare documents even if they do not have any terms in common.

Hofmann [12] introduced Probabilistic Latent Semantic Analysis (PLSA) which is, in contrast to LSI, a generative model of the data which uses a probabilistic model instead of SVD to tackle the problem. It is based on an aspect model where latent variables (topics) are associated with observed variables (words and documents). The idea is that this model is able to generate the data that we observe in the term-document matrix, based on the topic distribution in the documents.

Latent Dirichlet Allocation (LDA) by Blei et al. [1] is a Bayesian improvement of PLSA as it introduces Dirichlet priors on document-topic and topic-word distributions which leads to better generalization. By using a Dirichlet prior distribution, LDA also overcomes the PLSA problems of linearly increasing number of estimation parameters and its inability to assign probabilities to previously unseen documents. Non-negative matrix factorization (NMF) [2] differs from the previously mentioned probabilistic models by finding the approximate factorization of term-document matrix $V \approx WH$ into a feature set W (visible variables documents and words) and hidden

variables (topics) H.

Srivastava & Sutton [36] introduced Autoencoded Variational Inference for Topic Models which trains an encoder neural network that directly maps a document to an approximate posterior distribution. They approximate the Dirichlet prior with logistic-normal distribution and use Adam optimizer to address component collapse (when encoder is stuck in local optimum).

Miao et al. [37] propose a range of topic models parametrised with neural networks and trained with variational inference used for constructing topic distributions, all of wich draw from a multivariate Gaussian distribution (instead of Dirichlet). In contrast to Srivastava & Sutton, their models directly parametrise the multinomial distribution with neural models, while Srivastava & Sutton follow LDA formulation by keeping Dirichlet-Multinomial parametrisation and appliy the Laplace approximation to allow for back-propagation.

Neural network language models [13] introduced distributed word representations such as word2vec [14] which capture syntactic and semantic word relationships. Words are represented by word embeddings which are dense vectors that are similar for words used in similar contexts. More examples that tried to improve the word2vec word embeddings include GloVe [15], FastText [16] and ELMo [17].

Doc2vec [18] creates dense vectors that are representations of documents. These vectors are trained to predict words in the documents. Doc2vec is similar to word2vec except that it ads another document vector during prediction of other word vectors when training the model.

Recently top2vec [3] and BERTopic [4] showed improvement over traditional methods. They both achieve much better qualitative and quantitative results, by extracting more informative and representative topics from the corpus.

Top2vec by Angelov [3] leverages joint document and word semantic embedding to find topic vectors, by assuming documents close to each other in vector space can be interpreted as documents with the same topic. Top2vec

does not require removal of stop words, lemmatization and stemming and finds the number of topics on its own which is a major improvement over the traditional models.

BERTopic model by Grootendorst [4] approaches topic modeling as a clustering task and extends on it. By default it uses SentenceBERT [19] to generate document embeddings, clusters them and generates topic representations with class-based TF-IDF procedure. Similarly to top2vec, BERTopic uses UMAP [20] for dimensionality reduction and HDBSCAN [21] for clustering.

Topic modelling for news and social media

There has been a lot of topic modeling research done the news domain, mostly because standard datasets containing news are commonly used to measure the performance of topic modelling approaches. The most popular news dataset is the 20 NewsGroups dataset [22] consisting of 16309 news articles across 20 categories. Another news dataset is the BBC News dataset [23] that contains 2225 articles between years 2004 and 2005.

Newman et al. [24] use a collection of 330,000 New York Times news articles. They use statistical language models and named-entity recognizers to analyze entities (persons, organizations, places) and topics in the collection of articles, as well as relations between them. They apply topic models to learn the latent structure behind named entities and show how the relative contributions of topics change over time, correlating with major news and events.

Tang et al. [25] use previously mentioned New York Times dataset along with tweets and Wikipedia articles to show the limitations of LDA. Some important findings are i) that once we have enough documents adding more will only yield diminishing improvements, ii) poor performance with short documents even if we have many of them and iii) LDA performs well when topics are well separated.

Hong & Davison [38] show that LDA yields better results when the tweets are aggregated into "pseudo-documents" for each user compared to single

tweets. Zhao et al. [28] compare contents of Twitter with a traditional news source (New York Times) and find that Twitter and traditional news media cover similar topics; although, the discovered topics have different distributions. Twitter users tweet more about personal life, celebrities and other entity-related topics that have less coverage in traditional media, but they also *retweet* news about world events, thus helping to spread important news.

Yang et al. [26] use topical modeling on a corpus of Texas newspapers from 1829 to 2008 to discover historical topics and trends. Hidayatullah et al. [29] use LDA to analyse tweets from Indonesian Twitter accounts that tweet about football and find some distinct topics such as pre-match analysis, live match update, football club achievements etc. Ma et al. [30] investigate COVID vaccine hesitancy by analyzing tweets with LDA and top2vec and find that top2vec discovers more topics than LDA and that they are more distinct.

Abuzayed & Al-Khalifa [27] compare LDA, NMF and BERTopic with different embeddings on the dataset composed of 108789 documents from three Arabic online newspapers: Assabah, Hespress and Akhbarona. They report much better performance of the BERTopic model with various embeddings.

Egger and Yu [31] try to bridge the gap between computational science and empirical social research by evaluating the performance of 4 topic modeling algorithms (LDA, NMF, Top2vec and BERTopic) on Twitter posts. They find that neural topic models work better although they have their limitations.

Slovene language

Bajt and Robnik-Šikonja [32] compare Slovenian news media by analyzing topics and article sentiment. For topic analysis, they use LDA on two levels. The first one extracts general topics and then they run LDA on the subset of articles for each topic to get the article subtopics. They use SloBERTa to analyze the sentiment of discovered subtopics for each covered media portal, and show the differences between Slovenian media in regards to topics that they cover and how they write about said topics. They use articles from

largest Slovenian online news outlets from years 2019 and 2020.

Berginc and Ljubešić [33] use LDA for topic modeling of two large Slovene language corpora. Gigafida mainly consists of printed texts and slWaC consists of online texts. They extract topics for each corpus separately and compare them afterwards. Although some topics are found in both corpora they argue that the corpora are quite different regarding the topics.

Škrjanec and Pollak [39] explore Slovene blogs and construct hierarchical topic ontologies. They place special focus on comparison of blogs by male and female bloggers.

Markoski et al. [40] preform topic modeling on Wikipedia articles from South Slavic languages and compare distributions of topics per language. They use LDA with 10 topics for topic modeling.

Miok et al. [41] analyze transcripts of parliamentary debates from Bulgarian, Czech, English, French, Slovene, Spanish and UK national parliament. They analyze and interpret the topics discussed and determine sentiment and emotions in the debates. For topic modeling they use LDA and visualise discovered topics.

Evaluation of topic modeling

Topic diversity, introduced by Dieng et al. [34], and topic coherence, particularly normalized pointwise mutual information (NPMI) [35], are two of the most widely used metrics for automated evaluation of topic models.

Röder et al. [42] conduct a systematic analysis of coherence measures and propose a framework for constructing new coherence measures by combining elementary components.

Newman et al. [43] introduce novel topic coherence evaluation task where they evaluate topic coherence based on word coocurrence over Wikipedia data, Google's search engine based similarity, and WordNet lexical word similarity.

Grootendorst [4] states that although topic quality measures such as topic diversity and topic coherence can be used as an indication of topic models performance, they are just that - an indication. Hoyle et al. [44] argues that

even though NPMI correlates with with human judgement, recent research suggests that this may only be true for classical models and not so much for modern neural approaches and suggest a needs-driven approach and human supervision where necessary and possible.

1.2 Contributions

The first main contribution of this thesis is analysis of 4 topic modeling algorithms, LDA, NMF, BERTopic and top2vec, and their performance on Slovene language. Quantitative analysis shows the behaviour of topic model performance with respect to key parameter values. Qualitative analysis identifies BERTopic as the best model in terms of performance on both datasets and usefulness of extracted topics for further use.

The second main contribution is the proposal of a novel method for evaluating topic model stability and topic model similarity based on semantic similarity and maximum bipartite matching called *Maximum bipartite topic similarity*. The proposed method is further applied to assess the stability of topic models for the Slovene language.

1.3 Thesis structure

The thesis is organised as follows. After the Introduction, Chapter 2 introduces methods and models we used. Chapter 3 describes evaluation framework and introduces our novel method for evaluating topic model stability. In Chapter 4, we present our datasets and describe the data preprocessing. In Chapter 5, we present and discuss results. In Chapter 6, we conclude and propose future work.

Chapter 2

Methods

In this chapter, we delve into the specifics of four topic modeling algorithms that we used in our work: Latent Dirichlet Allocation (LDA), Non-negative Matrix Factorization (NMF), BERTopic, and Top2vec. In the following sections, we provide a detailed exploration of the inner workings, strengths, and weaknesses of each algorithm, laying the groundwork for a comprehensive understanding of their applications in various contexts.

2.1 Latent Dirichlet allocation

Latent Dirichlet allocation (LDA) was first proposed by Blei et al. [1] as a model for topic discovery. LDA is a generative probabilistic model for text corpora where each item (document) is modeled as a finite mixture over a set of topics. Each topic is modeled as an infinite mixture over a set of topic probabilities [1]. Each document in a corpus can be represented by topic probabilities. In turn, each topic is characterized by a distribution over words.

LDA makes two key assumptions. One is that the number of topics k is predetermined and the other is that a document is just a collection of words so the order of words in the document does not matter (a "bag of words" model).

We can imagine LDA as a machine that produces documents, depending on the settings (parameters) that we have. With better settings (parameters that we infer) we can produce real documents from our corpus with higher

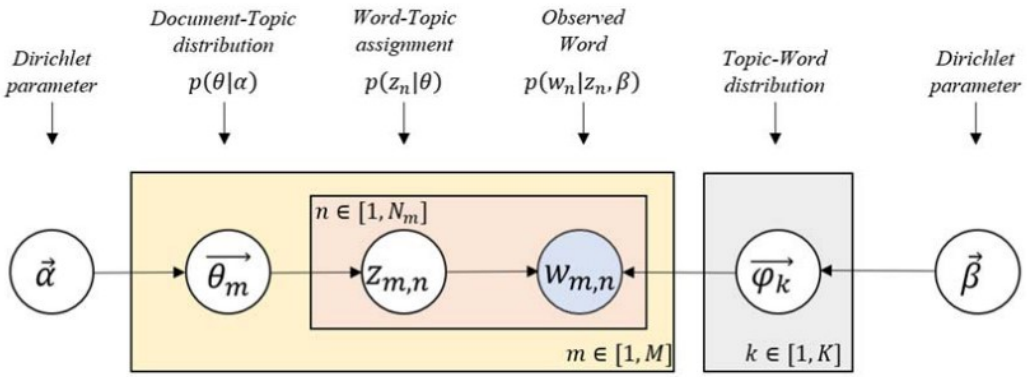

Figure 2.1: Graphical representation of LDA model. Source: [45]

probability. We can then infer topics from the settings of the machine.

LDA is a generative process that produces a document in the following way: first it chooses a topic mixture θ_m for a given document which follows a multinomial distribution whose hyperparameter α follows the Dirichlet distribution. The model picks a topic from this distribution. The distribution of words for topic k is denoted by φ_k which is also a multinomial distribution whose hyperparameter β follows the Dirichlet distribution [1]. Next the model chooses a word $w_{m,n}$ from $p(w_{m,n}|z_{m,n}, \beta)$, a multinomial probability conditioned on selected topic $z_{m,n}$. We repeat these steps N_m times to generate a document, where N_m is the number of words in the document. We can see the graphical representation of this process on Figure 2.1.

The total probability of the model is given by equation (2.1). In other words, this is the probability of all documents in our corpus being produced by the model. We want to maximize this probability meaning that we need to find the optimal document-topic distribution and topic-word distribution.

$$P(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{\varphi}; \alpha, \beta) = \prod_{j=1}^{M} P(\theta_m; \alpha) \prod_{k=1}^{K} P(\varphi_k; \beta) \prod_{n=1}^{N} P(z_{m,n} \mid \theta_j) P\left(W_{m,n} \mid \varphi_{z_{m,n}}\right) \tag{2.1}$$

In order to do that we need to work backwards. We start by randomly

assigning topics to each word in every document. Then we iterate over every word w in each document d and try to adjust the topic assignment k of the current word. The model assumes that all topic assignments are correct except for the current word. It calculates probability $p(k|d)$ which is a proportion of words in the document d that are assigned to the topic k, and probability $p(w|k)$, a proportion of assignments to topic k over all documents that come from the word w. We update the probability of a topic k belonging to the word w by multiplying said probabilities, i.e. $p(k|d) * p(w|k)$. According to this probability, we resample a new topic and assign it to the word w. We repeat the process to reach a steady state and optimal document-topic and topic-word distributions.

An example of a result of such process can be seen in Figure 2.2. Note that a Topic 4 that we get is not *"Politics"* but a distribution over words *elections*, *president*, *parliament*, *speech* and *protests*. From these words, we can then infer and name the topic.

Model hyperparameters α and β influence the document-topic and topic-word distributions. Larger α means that the document is a mixture of a larger number of topics and larger β means that a topic is represented by a larger number of words. [32]

Probabilistic latent semantic indexing (pLSI) is a similar method introduced by Hofmann [46], but there is no way to assign probabilities to new unseen documents. pLSI also has linear growth of parameters (in M) which is prone to overfitting. LDA has less parameters and better generalizes to new documents.

2.2 Non-negative matrix factorization

Basic idea of non-negative matrix factorization (NMF) is to factorize a matrix with non-negative elements $\boldsymbol{X}$ into matrices $\boldsymbol{W}$ and $\boldsymbol{H}$ which also have only non-negative elements, such that the product of $\boldsymbol{W}$ and $\boldsymbol{H}$ is an approximation of $\boldsymbol{X}$. Usually the $\boldsymbol{W}$ and $\boldsymbol{H}$ have much smaller dimension than $\boldsymbol{X}$.

Topic 1		Topic 2		Topic 3		Topic 4	
elections	0.2%	win	1%	physics	0.1%	movie	2%
president	0.1%	league	0.7%	planet	0.08%	book	0.5%
parliament	0.1%	ball	0.5%	Nobel	0.04%	actor	0.2%
speech	0.08%	tennis	0.02%	breakthrough	0.002%	Oscars	0.07%
protests	0.05%	Luka Dončić	0.003%	book	0.001%	star	0.02%

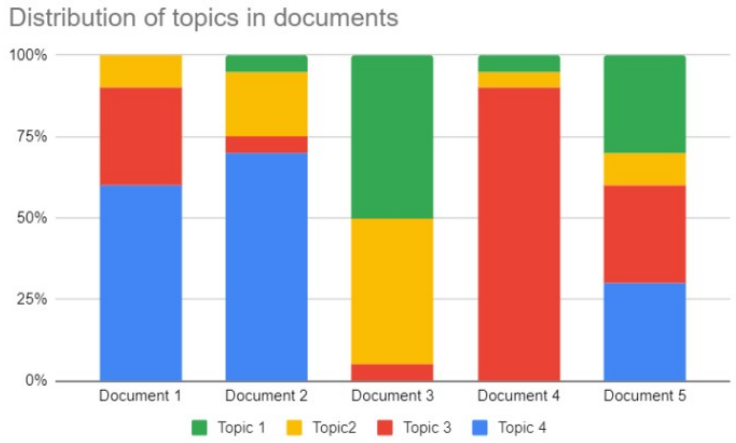

Figure 2.2: Example of a topic model. In the table, we see the topic distribution over words, and in the graph, we see the distribution of topics for each document.

Although there exist different algorithms and cost functions for NMF, the basic cost function is the Frobenius norm. Optimization problem is then defined as

$$min||\boldsymbol{X} - \boldsymbol{W}\boldsymbol{H}||_F^2 \; w.t. \; \boldsymbol{W}, \boldsymbol{H}, \; s.t. \; \boldsymbol{W}, \boldsymbol{H} > 0 \tag{2.2}$$

Lee and Senung [2] propose multiplicative update rule that minimizes Frobenius norm. First we initialize non-negative $\boldsymbol{W}$ and $\boldsymbol{H}$, then in each iteration, we update the values in them as follows

$$\boldsymbol{H}_{ij} \leftarrow \boldsymbol{H}_{ij} \frac{(\boldsymbol{W}^T\boldsymbol{X})_{ij}}{(\boldsymbol{W}^T\boldsymbol{W}\boldsymbol{H})_{ij}} \qquad \boldsymbol{W}_{ij} \leftarrow \boldsymbol{W}_{ij} \frac{(\boldsymbol{X}\boldsymbol{H}^T)_{ij}}{(\boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^T)_{ij}} \tag{2.3}$$

and we do this until $\boldsymbol{W}$ and $\boldsymbol{H}$ are stable or until we reach the maximum number of iterations.

In topic modelling, NMF is used on *document-term* matrix which is either a normalized bag-of-words or TF-IDF matrix. *Document-term* gets factorized into *document-feature* $\boldsymbol{W}$ and *feature-term* $\boldsymbol{H}$ matrices. Each row of $\boldsymbol{W}$ represents the weights of the topics that are present in this document. On the other hand, $\boldsymbol{H}$ tells us how much are terms representative for each topic.

2.3 BERTopic

Grootendorst introduced BERTopic [4], a novel topic model, in 2022 and quickly made strong impact. One of the biggest reasons for its popularity, besides achieving state-of-the-art results, is its easy to use Python package with the same name. Besides basic topic modeling, it also supports guided, supervised, semi-supervised, dynamic and hierarchical topic modeling. Besides topic model algorithms, it also provides various visualizations, which help immensely in qualitative evaluation.

We will describe each step of the algorithm in detail, but in, short BERTopic works in the following way. It first embeds documents to numerical vectors. Semantically similar documents get converted to similar vectors. This is useful for clustering similar documents in the same cluster. Next, BERTopic uses dimensionality reduction, as clustering does not work well on high-dimensional vectors because of curse of dimensionality. By default BERTopic, uses UMAP [20] projection to low-dimensional space as it preserves both local and global structure of the dataset. Then, BERTopic uses HDBSCAN [21] for clustering documents. HDBSCAN clusters can be of different shapes and sizes as HDBSCAN is a density-based clustering technique. Bag-of-words is used for representation of clusters after documents of the same clusters are combined into a single document. In the end, BERTopic uses class-based TF-IDF weighting for topic representations, and extracts the most representative words for each topic.

Because the algorithm's modular approach, we can choose different models in each step and avoid unwanted assumptions of models. For example we can use PCA instead of UMAP or we can choose sentence embedding model

that works best with our data or even one that we trained ourselves.

In the following subsections we will introduce the models that we used at each part of the BERTopic algorithm. Subsection 2.3.1 describes the Sentence BERT model, used to generate document embeddings, subsection 2.3.2 explains the UMAP algorithm, in 2.3.3 we describe the HDBSCAN algorithm and in subsection 2.3.4 we introduce the novel method for topic representation used by BERTopic.

2.3.1 Document embeddings

BERTopic uses Sentence BERT (SBERT) framework [19] to embed documents. The main goal of SBERT is to present sentences with semantically meaningful embeddings that map when semantically similar sentences close together in vector space. BERT [47] can be used to classify pairs of sentences as similar, but that can be computationally inefficient, when finding the most similar sentences as it has to check all possible combinations. SBERT is a modification of BERT which encodes a sentence with a fixed-sized vector. Similarity of two sentences can be then computed much faster with dot product of corresponding vectors.

SBERT creates Siamese and triplet networks and uses BERT as a base, to which it adds a pooling operation over time to derive a fixed size sentence embedding. Final network structure depends on the available data and objective function. One option is a classification objective function that concatenates both sentence embeddings u, v and their element-wise distance $|u-v|$. It then multiplies this vector with trainable weight and computes the softmax to predict similarity score. BERTopic uses the cross-entropy loss for optimization. Such structure can be seen on the left-hand part of Figure 2.3. Another option is a regression objective function which computes cosine similarity of sentence embeddings u and v, as can be seen on the right-hand part of Figure 2.3. The third option is triplet objective function which provides an anchor sentence a, positive sentence p, negative sentence n and calculates the triplet loss such that a is closer to p than to n.

SBERT is trained on SNLI [48] and Mulit-Genre NLI [49] datasets. They

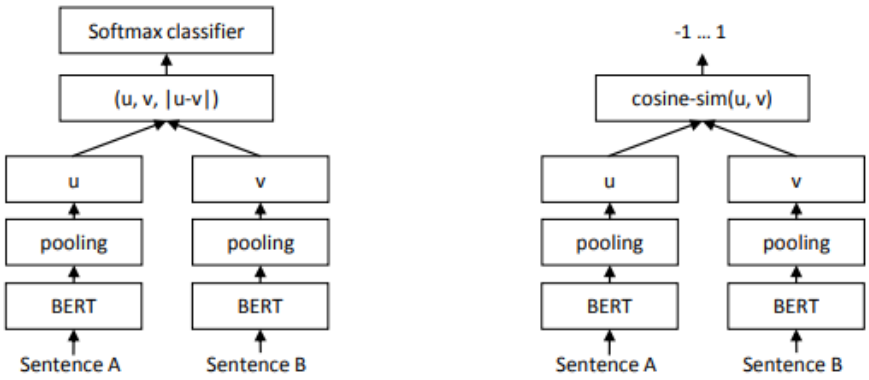

Figure 2.3: SBERT architectures with different objective functions. Classification objective function on the left and cosine-similarity on the right (which can be used for regression). Source: [19]

consist of sentence pairs annotated as *contradiction*, *entailment* and *neutral* for sentence pairs of both spoken and written text.

Reimers and Gurevych [50] present multilingual sentence embedding based on SBERT. The idea is that sentence embedding of a sentence in any language should be mapped to the same location in vector space. Multilingual SBERT uses the original monolingual SBERT model as a teacher that provides sentence embeddings for English. Then we train a student model on pairs of original and translated sentences so that it produces embeddings close to the original teacher sentence embedding. This process can be seen on Figure 2.4.

2.3.2 UMAP

UMAP [20] is a dimensionality reduction algorithm. Similarly to t-SNE [51], it is a neighbor-graphs based algorithm. In contrast, PCA is a matrix factorization dimensionality reduction technique. UMAP has strong theoretical foundations in manifold theory and topological data analysis. It first constructs a graph representation in a high-dimensional space and embeds it into a low-dimensional space while trying to preserve its structure as much as possible.

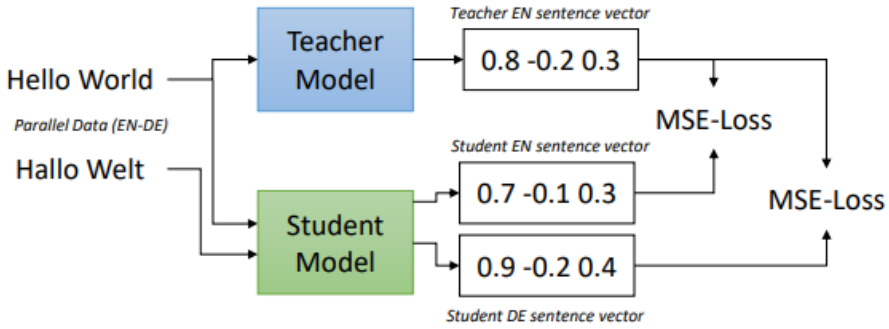

Figure 2.4: Multilingual SBERT training process. Source: [50]

The high-dimensional graph is a weighted graph where edge weights represent the probability of connections between two points. To determine a connection between two points, UMAP builds a radius for each point. If radii of two points overlap, the points are connected with some probability (based on the distance between points). If the radius is too small, the clusters are isolated, but if the radius is too big then everything is connected. So the radius of each point is determined locally for each point and it is the distance to the n-th nearest neighbor. Another constraint is that each point must be connected to at least its closest neighbor.

The second part of the algorithm is optimizing the low-dimensional representation to have similar topological structure as the high dimensional representation as measured by cross-entropy. The projection is done by a force-directed graph layout algorithm.

Two of the most important hyper-parameters in UMAP are n and $min-dist$. Parameter n determines the size of the local neighborhood (number of nearest neighbors) that UMAP uses to learn the manifold structure of high-dimensional data. It controls how UMAP balances local versus global structure in the data. A small n means that UMAP will focus more on local structure, while larger n forces UMAP to look at larger neighborhoods and thus preserve more global structure but lose the finer details.

The $min-dist$ parameter determines the minimum distance between the points in the low-dimensional projection. A small value will result in more

smooth embeddings, which can be useful for clustering.

Although UMAP is quite similar to t-SNE, it is better at balancing local and global structure and much faster.

2.3.3 HDBSCAN

HDBSCAN clustering algorithm by Campello, Moulavi, and Sander [21] is a hierarchical extension of DBSCAN [52], a density based clustering algorithm.

K-means clustering is one of the most widely used clustering algorithms. It assumes that clusters are round/spherical, equally sized, equally dense and not contaminated by noise. As clustering is used primarily for data exploration, we need as little assumptions as possible to get useful insights. HDBSCAN does not make many assumptions and performs well even for arbitrarily shaped clusters, clusters of different sizes and densities, and with noisy data, because it uses a density-based approach.

To form clusters, HDBSCAN finds areas with higher densities, separated from others with sparse areas or noise. HDBSCAN is a single linkage clustering which can be sensitive to noise when using Euclidean distances. HDBSCAN overcomes this problem with *mutual reachability distance*. Besides the distance between two points, it also estimates the density of each point as a distance to the k-th nearest neighbor called *core distance* and denoted as $core_k(x)$. Mutual reachability distance between point a and b is then defined as

$$d_{mreach-k}(a,b) = max\{core_k(a), core_k(b), d(a,b)\}. \tag{2.4}$$

This takes care of of the points that are close in a very sparse region due to randomness. By using mutual reachability distance such points get spread apart due to their core distance.

HDBSCAN then uses the mutual reachability distance to build the minimum spanning tree. From minimum spanning tree HDBSCAN builds the cluster hierarchy by sorting the tree edges by distance in increasing order and splitting a cluster into two subclusters over that edge. Because a cluster is often split into one subcluster with only one or two points and the other with

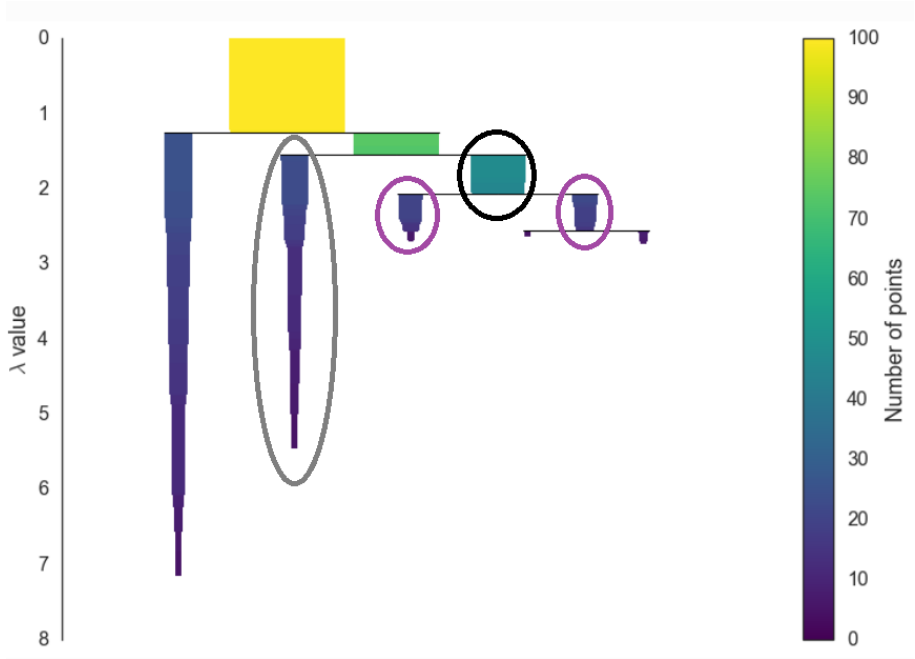

Figure 2.5: Condensed tree of clusters. Color indicates the number of points in the cluster, and $\lambda = \frac{1}{distance}$ is a measure of cluster persistence. By combining the two, we get the total colored area which is used for determining cluster splits. Adapted from [53].

the rest of points, a parameter *minimum cluster size* allows a split only if both of the subclusters contain at least the minimum cluster size number of points. We end up with a much more condensed cluster tree with less nodes and each of them containing information of how node decreases by changing the distance as seen on Figure 2.5.

The procedure has not yet determined the clusters but only a hierarchy of potential clusters and their splits. To produce actual clusters, we first consider all leaves as clusters and then move up the tree comparing if the cluster stability is larger than the sum of stabilities of subclusters. If yes, we merge them.

Simply stated, we merge clusters in the plot with the larger colored area compared to the sum of colored areas of their subclusters. We can see on Figure 2.5 that clusters encircled with blue have smaller sum of colored areas

than the cluster encircled with black, so we merge the clusters. On the other hand, clusters, encircled with black and grey stay separate.

2.3.4 Topic representation

The BERTopic interprets topics with a modified TF-IDF [6] approach. TF-IDF is a basic technique for word and document representation and it considers the importance of word t to document d. TF-IDF is a product of *term frequency* (TF) and *inverse document frequency* (IDF). Term frequency, $tf_{t,d}$ is the relative frequency of term t in document d. Inverse document frequency $idf_t = log(\frac{N}{df_t})$ measures the proportion of documents that contain term t and tells how much information the term provides. Terms that appear in a small number of documents have higher importance than terms that appear in many documents. TF-IDF is computed as a product of term frequency and inverse document frequency

$$w_{t,d} = tf_{t,d} \times idf_t \tag{2.5}$$

BERTopic uses a class-based variation, c-TF-IDF that treats identified clusters of documents as a single document (called class c) with documents in the cluster concatenated. c-TF-IDF for term t in class c is then defined as

$$w_{t,c} = tf_{t,c} \times log(1 + \frac{A}{tf_t}) \tag{2.6}$$

and we notice a small difference in the idf_t factor. It is calculated by taking the logarithm of the average number of words per class (A) divided by the frequency of term t across all classes [4] with added one to output only positive values. This modification allows us to model the importance of words in clusters instead of single documents.

2.4 Top2vec

Top2vec [3] is a topic modeling technique that leverages word and document embeddings obtained from doc2vec. It uses dimensionality reduction and clustering algorithm on document embeddings to determine the number and

sizes of topics as well as topic vectors and topic words that represent each topic.

In the following subsections we will describe the parts of the Top2vec algorithm. In subsection 2.4.1 we describe the word2vec algorithm used for learning word embeddings. In subsection 2.4.2 we introduce the doc2vec algorithm that learns vector embeddings of documents. Lastly in subsection 2.4.3 we explain how doc2vec algorithm extends to topic modeling.

2.4.1 Word2vec

Word2vec [14] is a technique for learning dense vector representations of words. It learns word embeddings that capture semantic and syntactic word properties which is not possible by one-hot encoded vectors. The idea is that other words in a sentence give meaning to the selected word, so two words are similar if they often appear in similar contexts, i.e. they are surrounded by similar words.

Based on this idea word2vec consists of two methods for computation of word vectors, continuous bag-of-words (CBOW) and skip-gram. Model schemata can be seen on Figure 2.6. CBOW tries to predict the missing word given other words in the context. If we have a sentence *"Alice likes beautiful red roses."*, CBOW tries to predict the word *beautiful* from other 4 context words. Word2vec uses gradient descent to update weight matrices $\mathbf{W}$ and $\mathbf{W}'$ until convergence and then uses rows from $\mathbf{W}$ as the learned word embeddings.

On the other, the hand skip-gram model uses the middle word and tries to predict the other four context words.

2.4.2 Doc2vec

Doc2vec [18] is an extension of word2vec that applies to documents, paragraphs and sentences. It aims to learn document embeddings as dense distributed vector representations of texts. They numerically represent semantic meaning of the document, no matter the length of the document.

Doc2vec treats a document like another word in the way that a document

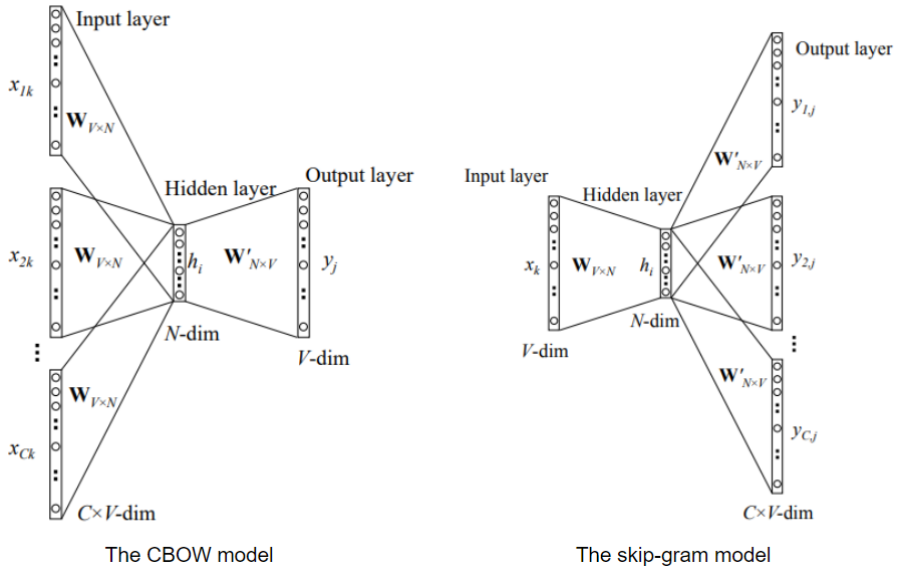

Figure 2.6: CBOW and skip-gram model schemata. Source: [54]

is represented by its own vector and used along with other context words when predicting a missing word. Besides **W**, a weight matrix for words, It learns **D**, a weight matrix for documents, which is then used to extract document embeddings.

Similarly to word2vec, doc2vec proposes two model architectures that build on CBOW and skip-gram. The first one is called Distributed Memory Model of Paragraph Vectors (PV-DM). Here the document token is represented as another word and acts as a memory that remembers what is missing from the current context [18]. The other model architecture is callled Distributed Bag of Words version of Paragraph Vector (PV-DBOW) and tries to predict words from a given document. Schemas for both architectures can be seen in Figure 2.7.

2.4.3 Doc2vec extension for topic modeling

Top2vec [3] is a direct extension of doc2vec method as it uses doc2vec document and word embeddings and their properties to extract topics.

PV-DBOW model works in a similar way as skip-gram doc2vec model. If

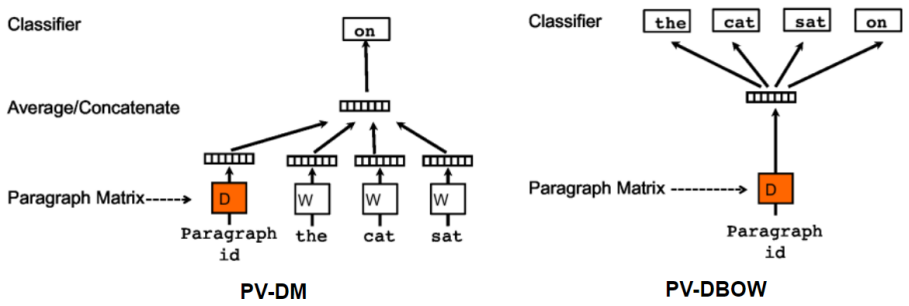

Figure 2.7: Distributed Memory version of Paragraph Vector (PV-DM) and Bag of Words version of Paragraph Vector (PV-DBOW) doc2vec model schemata. Source: [18]

we first train word embeddings with skip-gram and use them as pretrained context word matrix $\mathbf{W}'$ in doc2vec PV-DBOW model, then doc2vec learns document embeddings in the same semantic space as word embeddings. This means that semantically similar documents are close together as well as close to the words that best describe them. An example of such semantic space can be seen on Figure 2.8.

Authors of top2vec argue that this space is essentially a continuous representation of topics where each point represents its own topic and can be transformed into a probability distribution over the vocabulary of documents, so the closest word vectors are the topics' most representative words.

Just like in BERTopic, the idea of top2vec is to find documents that belong to the same dense area and consider them belonging to the same topic. Top2vec also uses UMAP for dimensionality reduction and HDBSCAN for discovering dense areas and clustering of documents. Then it finds a centroid of a cluster determined by HDBSCAN and calls it a topic vector as seen on Figure 2.9.

Word vectors that are the closest to the topic vector belong to words that represent the topic as seen on Figure 2.10. Number of topics in a corpus is determined by the number of dense areas in the semantic space. If needed, we can hierarchically group similar topics and reduce the number of topics

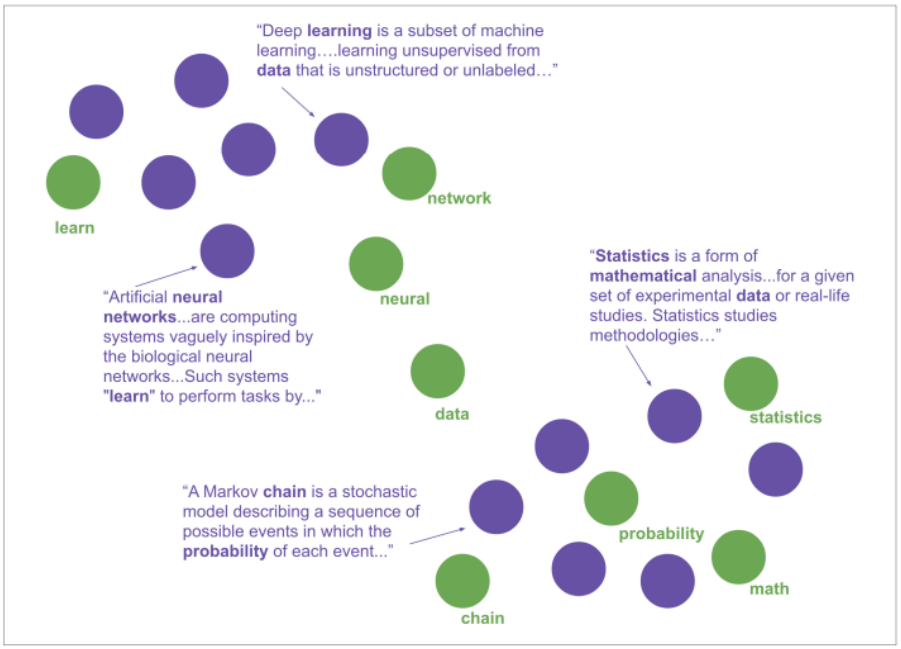

Figure 2.8: Jointly embedded semantic space of word and doument vectors, learned by doc2vec. Source: [18]

in a corpus.

Top2vec, like BERTopic, does not need removal of stop words, lemmatization or stemming to learn good topic vectors.

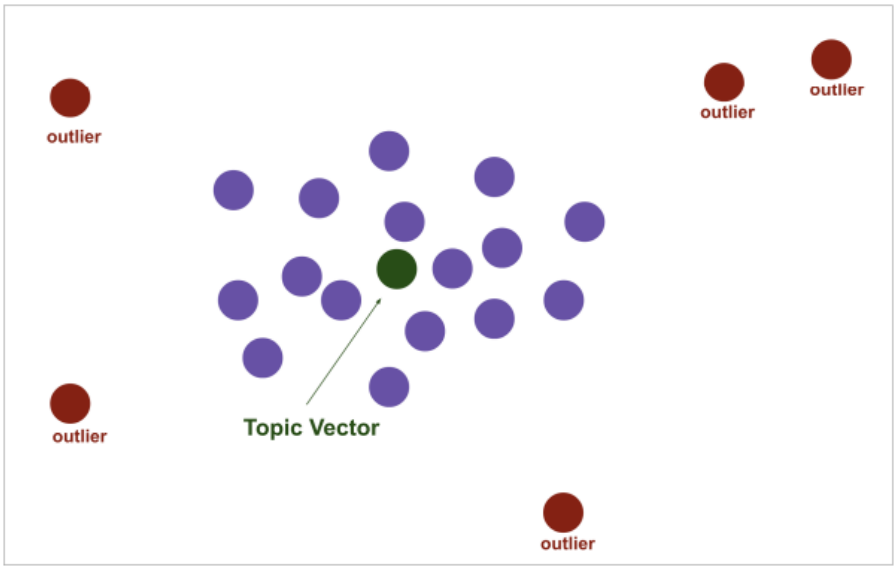

Figure 2.9: Dense area of documents that belong to the same topic and the corresponding topic vector. Source: [18]

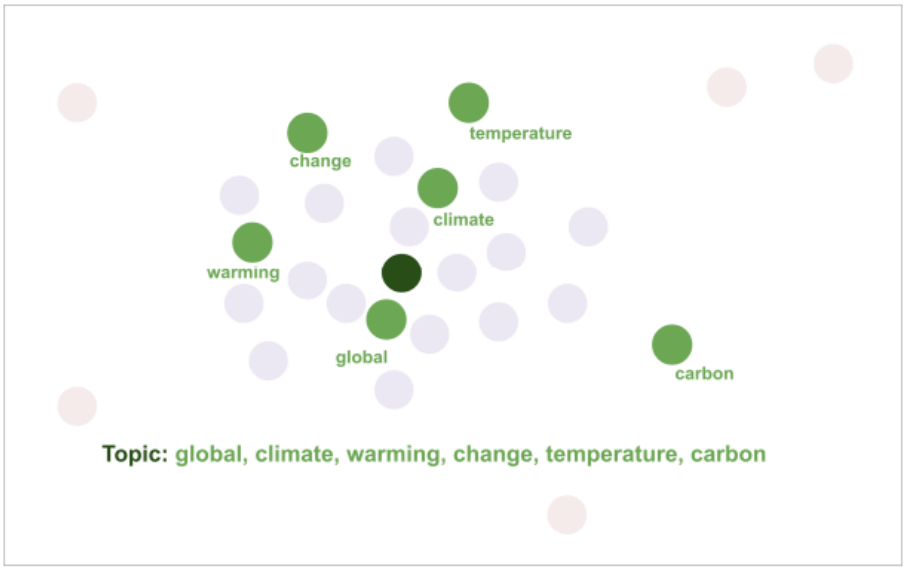

Figure 2.10: Words with embeddings closest to the topic vector are the most representative of said topic. Source: [18]

Chapter 3

Evaluation framework and metrics

There are two approaches to evaluation of topic models. First is human-centered where the evaluation is done by having people rate topics based on some criteria. Although this approach can be subjective and not directly comparable between different studies, it is useful because the outputs of topic models are most often used by humans for further analysis.

Two common human judgement tasks are topic rating [55] and word intrusion task [56] measuring topic coherence. In topic rating, annotators are given a topic and they assign it a quality score usually on a three-point ordinal scale. In word intrusion task, humans are given top words in a topic and an additional "intruder" word which has low probability of belonging to that topic and the task is to identify the intrusion word. In topics with high coherence such words are easy to identify. One of the major drawbacks of the human-centered approach to evaluating topic models is the need for a large number of people willing to evaluate the results.

The second evaluation approach is an automated evaluation using various metrics such as topic diversity and topic coherence (e.g. normalized pointwise mutual information).

The automatic approach can provide reliable and directly comparable results, but it is not able to capture the nuances of human interpretation. Some common objective metrics for evaluating topic models include topic coherence and topic diversity. Each of these metrics has its own strengths and limitations, and they are often used in combination to provide a com-

prehensive picture of a topic model's performance.

In section 3.1 we describe the topic coherence measure, specifically normalized pointwise mutual information and in section 3.2 we describe the topic diversity. In section 3.3 we introduce and describe a novel method *maximum bipartite topic similarity* used for evaluation of topic model stability.

3.1 Topic coherence

Topic coherence metrics measure how the top-n words of each topic relate to each other. The most commonly used metric for topic coherence is normalized pointwise mutual information (NPMI). This metric has been shown to positively correlate with human scores on word intrusion and rating tasks [35]. A topic has a high NPMI score if its top-n words (summed over all pairs) have higher joint probabilities compared to their marginal probabilities [44]. NPMI of a topic with top-n words is defined as:

$$NPMI(t) = \sum_{j=2}^{n} \sum_{i=1}^{j-1} \frac{\log \frac{P(w_j, w_i)}{P(w_i)P(w_j)}}{-\log P(w_j, w_i)} \tag{3.1}$$

Hoyle et al. [44] also argue that this measure might not be suitable for neural topic models as neural word embeddings directly optimise NPMI. In other words, a neural topic model with high NPMI scores may not necessarily produce topics that are coherent to humans. Therefore, human interpretation is still needed when evaluating the performance of neural topic models.

Other measures for topic coherence have been proposed, but they are all very similar to NPMI, as they are based on joint probabilities of words [57].

3.2 Topic diversity

Topic diversity metrics measure how diverse the top-n words of a topic are to each other. Topic diversity, as introduced by Dieng et al. [34], is defined as percentage of unique words in the top 25 words of all topics recognised by the topics model.

3.3 Maximum bipartite topic similarity

All existing metrics measure the quality of a single topic and the quality of a topic model by averaging the scores of its topics. We can then take scores of multiple topic models and compare them between each other to try and identify the best topic model. But there is a huge caveat that with this approach we cannot directly compare discovered topics between each other and determine if the topics are semantically similar or if the two topic models discovered some totally different topics from the same text.

There also exist metrics for evaluating the clustering quality such as silhouette [58], Rand index [59] and Adjusted Rand Index (ARI) [60]. They work on documents level and require a partition of documents while some topic modeling methods output just the distribution over topics (such as LDA). We would like a metric that works on the topic representation level. An option would be to use the Adjusted Rand Index on the topic level by treating topic words as instances and then comparing the assignment of words to topics as clusters. But more often than not two topic models do not use the same topic words (let alone topics themselves) and assign them differently, but use many different words altogether, which we would need to handle in some unnatural way, for example assigning all the words used in the other topic model to a separate cluster. Although this is an idea worth exploring further, we opt for a different approach.

In this work we introduce a new metric for measuring topic model similarity called *maximum bipartite topic similarity* or *MBTS* which is based on maximum bipartite matching algorithm.

Bhagwani et al. [61] first introduced maximal weighted bipartite matching for measuring similarity of two texts. They represent a sentence by tokens, calculate token-token similarity of all pairs of tokens from two sentences and then compute maximal weighted bipartite match where tokens represent vertices and similarites are weighted edges between them. They use WordNet based word similarities and statistical word similarities.

3.3.1 Algorithm

We use the maximum weighted bipartite matching for measuring topic model similarity. Topic models extract k topics where each is represented by n words. Because words that represent a topic do not have any semantically meaningful order using maximum weighted bipartite matching is natural, so we calculate similarity of two topics similarly as Bhagwani. We calculate similarities between all pairs of words representing two topics and construct a maximal weighted bipartite match as seen on Figure 3.1. Similarity of two topics is then the average of weights in a graph.

We extend the same procedure to calculation of similarity for two topic models. Each topic model discoveres k topics so we calculate similarities for all pairs of topics as described before. We again use the maximum weighted bipartite matching where topics are vertices in a graph and weights are similarities between topics. Similarity of two topic models is then defined as an average of weights in a maximal weighted bipartite graph as seen on Figure 3.2. The pseudocode of the algorithm is described in Algorithm 1.

Maximum bipartite matching provides an overall the best topic matching, while using a greedy approach, where we would choose the maximum from each node, would prefer stronger matches. This metric would be asymmetric, but we would a prefer a symmetric metric and an overall best topic matching.

For measuring the word similarities, we can use any similarity metric. We chose cosine similarity on word embeddings. For word embeddings, we take the pretrained Word2Vec [62] or fastText [63] embeddings for Slovenian language. In theory the cosine similarity always belongs to the interval $[-1, 1]$, but in practice when we are computing the cosine similarity of Word2vec vectors the value is between 0 and 1. This is because word representations have non-zero mean, meaning word vectors share a large common vector (with norm up to a half of the average norm of word vector), which was shown by Mu et al. [64]. Cosine similarity close to 1 means that two words are very similar and similarity close to 0 means words are not similar. The cosine similarity takes negative values when we are computing similarity of 2 very rare

tokens from the vocabulary, e.g. some large numbers, technical abbreviations or personal names. Because topic models represent topics with words that are common and representative of said topics, the MBTS will most probably belong to the interval $[0, 1]$, while in theory it can be negative as well. MBTS for two topic models with value 0 means the topic models are not similar at all and MBTS value 1 means that the topic models are a perfect match.

In Tables 3.1 and 3.2 we can see two toy examples of calculating MBTS with Word2vec embeddings. We compare a topic model M_1 to a very similar topic model M_2 and get a significantly higher MBTS score than when we compare M_1 with a topic model M_3 that has semantically very different topics. With these toy examples we can get a feeling of how high the MBTS is when two topic models are very similar and when the topic models are very different.

Although we showed that MBTS can be used for measuring topic model similarity, we should point out its shortcomings as well. MBTS assumes that both topic models use the same top-n words and number of topics parameters. If they would not, there would be either too many words or topics to match, so we would not be able to use maximum bipartite matching. An option would be to discard the topic with the smallest average similarity to the topics in the other topic model. Another thing to consider is different word forms of a certain word. For example words *otrok* and *otroci*, a singular and a plural form of the word *otrok*, have essentially the same semantic meaning but cosine similarity of their word2vec embeddings is just 0.66. This quickly lowers the MBTS score of topic. Even when topics are very similar, they might get lower scores because of different word forms. A possible improvement would be to stem the words before calculating similarities, but we would lose some possibly important information with that as well so we have to consider the trade-off.

3.3.2 Additional use cases

Additionaly, we can use MBTS for evaluating stability of topic model algorithms. By running the same topic model algorithm with different random

Algorithm 1 Maximum bipartite topic similarity

Input: M_1, M_2 - topic models with discovered topics, represented by topic words

$TopicSimilarityMatrix \leftarrow$ init 2d array
for $T_i \in topics(M_1)$ **do**
  for $T_j \in topics(M_2)$ **do**
    $WordSimilarityMatrix \leftarrow$ init 2d array
    for $w_a \in words(T_i)$ **do**
      for $w_b \in words(T_j)$ **do**
        $WordSimilarityMatrix[w_a, w_b] \leftarrow similarityMetric(embedding(w_a), embedding(w_b))$
      end for
    end for
    $TopicSimilarityMatrix[T_i, T_j] \leftarrow maxBipartiteMatchingMean(WordSimilarityMatrix)$
  end for
end for
$TopicModelSimilarity \leftarrow maxBipartiteMatchingMean(TopicSimilarityMatrix)$
return $TopicModelSimilarity$

seeds and evaluating MBTS on topics discovered in different runs, we can measure how much do the discovered topic models vary between each run.

Furthermore, we can also use this metric not to measure topic model algorithm similarities but similarities of extracted topics between different corpora of documents.

M_1	M_2	topic_similarity
(znanost, raziskovati, analiza, fizika, laboratorijski)	(znanost, raziskave, analiza, kemija, laboratorijski)	0.86
(rezultat, košarka, športen, ekipa, točka)	(zmagovalec, nogomet, športnik, ekipa, točka)	0.81
(otrok, starši, družinski, vrtec, vzgoja)	(otroci, starši, družina, vrtec, vzgojitelj)	0.73
(pandemija, cepivo, zdravstvo, virus, bolnik)	(kriza, okužba, zdravje, cepljen, bolnik)	0.69
(gospodarstvo, kredit, ekonomija, posel, podjetje)	(evro, denar, ekonomski, posel, podjetnik)	0.57
		MBTS=0.73

Table 3.1: Toy example of Maximimum bipartite topic matching of two very similar topic models.

M_1	M_3	topic_similarity
(gospodarstvo, kredit, ekonomija, posel, podjetje)	(sodišče, ustaven, sodnik, vrhoven, zakon)	0.36
(otrok, starši, družinski, vrtec, vzgoja)	(samozaposlen, umetnost, gledališče, igralec, filmski)	0.35
(pandemija, cepivo, zdravstvo, virus, bolnik)	(požar, gasilec, kras, helikopter, podpora)	0.34
(rezultat, košarka, športen, ekipa, točka)	(intervju, ura, oddaja, televizija, gost)	0.32
(znanost, raziskovati, analiza, fizika, laboratorijski)	(potovati, turizem, gostilna, obiskovalec, plaža)	0.30
		MBTS=0.34

Table 3.2: Toy example of Maximimum bipartite topic matching of two very different topic models.

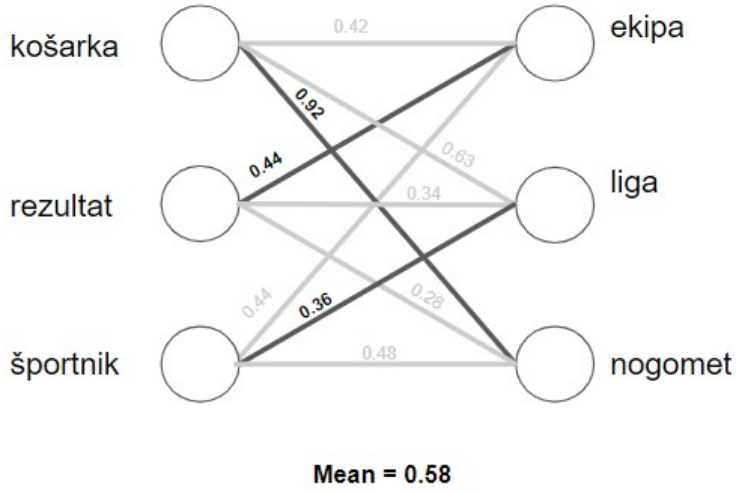

Figure 3.1: An example of maximum bipartite word matching with the word2vec embeddings and cosine similarity. Connections with numbers represent the cosine similarity between words and bold connections are the output of maximum bipartite matching of words. Mean is the mean of similarities in the word matching.

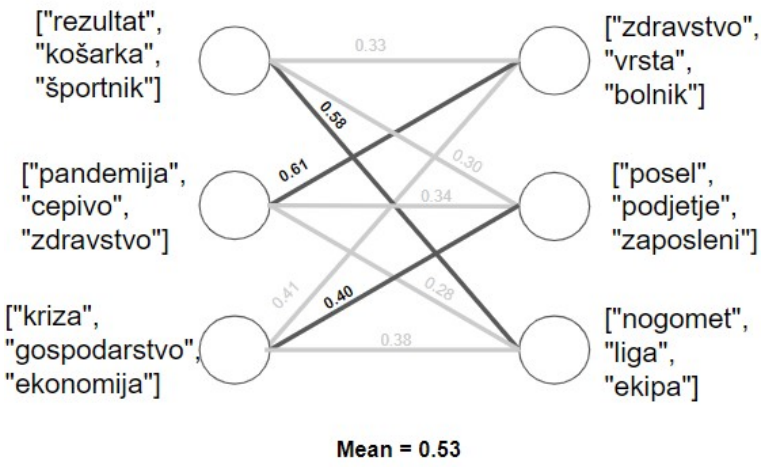

Figure 3.2: An example of the maximum bipartite topic matching with word2vec embeddings and cosine similarity. MBTS is the mean of the similarities of matched topics, in this case $MBTS = 0.53$.

Chapter 4

Datasets and data preprocessing

In this master thesis, we use two datasets for the evaluation of our methods and to analyze the Slovenian media concerning covered topics.

News dataset: The first dataset consists of Slovenian news articles published on the internet between 2014 and 2021. The dataset consists of articles from both the largest media portals such as rtvslo.si and 24ur as well as from small and local outlets (eg. prelkija-on.net). Altogether, there are about 2.9 million articles. Because most of the articles are from reputable news sources, the language is formal and without many grammatical mistakes or typos. The dataset was acquired from Eventregistry.org.

Twitter dataset: The second dataset [65] consists of tweets from 2006 to 2021 that are labeled as being written in Slovenian language. The dataset contains 60 million tweets but according to the authors of the dataset only 34 million tweets are very likely to be Slovenian (by using the fastText based model). Because of the tweet limitation to 280 characters, the documents in this dataset are much shorter than in the news dataset and often contain informal language, slang, typos and grammatical mistakes.

4.1 Data processing

We limited our study to documents from the year 2020 and used 50,000 randomly sampled documents (unless specified otherwise) in our experiments to reduce the time complexity. We later show that using larger number of documents does not reduce variance of the results.

We preprocessed the datasets by removing stopwords, special characters, numbers, twitter handles and other meaningless tokens (such as 'RT' for retweets). We lemmatized the texts with CLASSLA-Stanza tools [66]. We filtered out documents containing less than three tokens.

Chapter 5

Evaluation

In this chapter, we describe our experimental setup and the obtained results. We train four different topic model algorithms with different key parameter values. We quantitatively evaluate the results with previously described metrics and analyse the results. Additionally, we present and qualitatively evaluate the discovered topics and their representations. We argue which models provide the most useful topics for further use.

5.1 Quantitative evaluation

In this section, we show and compare results of different topic models with regards to different values of the most important parameters.

We trained four different topic models. Non-negative matrix factorization (NMF) and Latent Dirichlet Allocation (LDA) are traditional statistical approaches and we used Gensim librarys [67] implementation of these algorithms. For Top2vec and BERTopic, the modern neural approaches, we use their accompanying libraries Top2vec [3] and BERTopic [4].

We evaluate the performance of the models with topic diversity and topic coherence. We show how the performance of the model changes with different values of their important parameters. Unless specified otherwise, the default number of documents used is 50,000, the number of top-n words is 10 and the number of topics is 15.

For every combination of parameters, we run each model 3 times with a different random seed in order to compute the standard deviation of models

performance.

5.1.1 Number of documents

In the first experiment, we analyze how the number of documents from the same corpus affects the performance of selected topic models. We also analyze the standard deviation based on the number of documents. In Figure 5.1, we see that in the majority of cases for both topic coherence and topic diversity, the standard deviation does not decrease with more documents. Because of this, we decided to keep 50k random documents from each corpus for further analysis to reduce the time complexity.

With the exception of BERTopic on the news dataset, the topic diversity does not change much when increasing the number of documents. It is interesting to see the drop of topic diversity of BERTopic on the news dataset with larger dataset. Apparently it identifies words that are representative of multiple topics which is also the case for NMF and LDA.

Inversely, BERTopic improves the topic coherence on news dataset when increasing the number of documents and also seems the best choice with a larger number of documents which is also the case on the Twitter dataset. For BERTopic, LDA and NMF, we can see an asymptotical increase of topic coherence on the Twitter dataset with an increasing number of documents. This means that larger number of documents in a corpus with short documents produces better performance although it does not decrease variance of the results. The Top2vec algorithm looks very unstable on the Twitter dataset which shows that it may not be very suitable for short documents.

5.1.2 Number of top-n words

In the Figure 5.2, we can see the results of different values of top-n words parameter for topic diversity and topic coherence on both datsets. Both topic diversity and topic coherence mostly decrease when increasing the number of top-n words. This makes sense as adding more words to the representation of a topic increases the probability that some words are already in other topics, therefore decreasing topic diversity. Similarly, each new word included in a

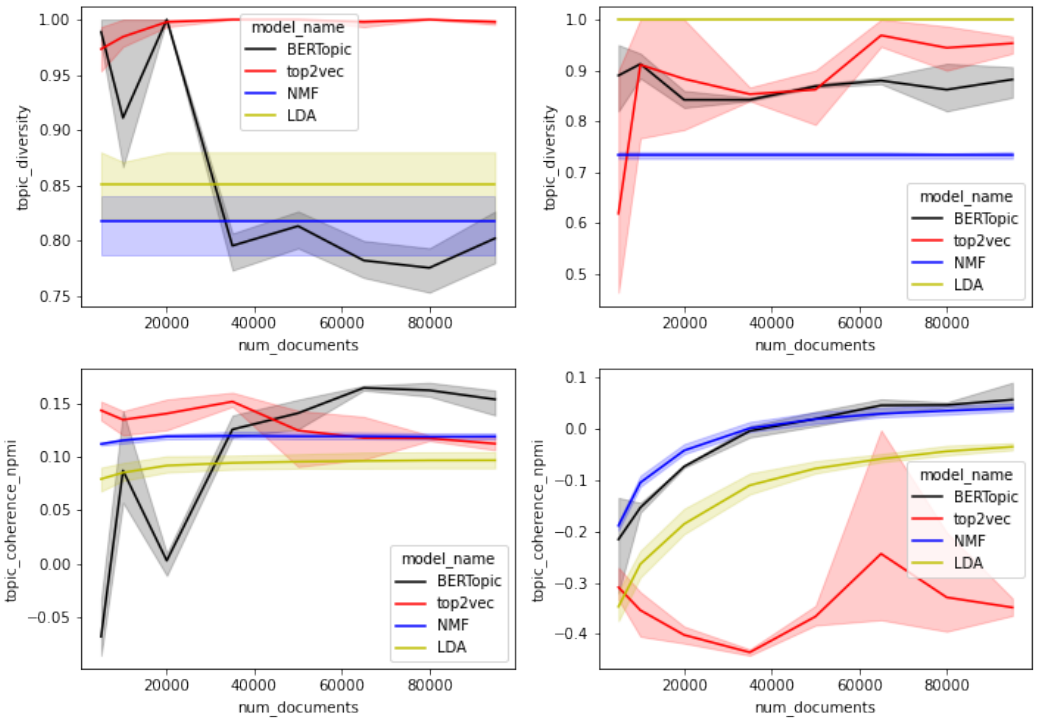

Figure 5.1: Topic diversity and topic coherence measures with respect to the number of documents. On the left-hand, the results are for the news dataset and on the right-hand the results show the Twitter dataset.

topic representation is less informative and representative of the said topic and therefore reduces the topic coherence. On the other hand, having more words representing the topic gives a more comprehensive idea about the topic and may help to distinguish two similar topics in the case of a large number of topics in the topic model.

Another interesting thing is a very high topic diversity of top2vec model for news dataset and a very high topic diversity of LDA on the Twitter dataset. The same happens in the experiments with different number of documents as well as with different values of number of topic parameters. From this we can conclude that top2vec does indeed preserve good topic diversity on long documents while LDA achieves good topic diversity on shorter documents regardless of different parameter values.

It is hard to draw any conclusions about which topic model is best across all values of top-n words parameter as the confidence intervals overlap and the order of the models (with regard to their performance) is changing with different parameter values. But all things considered if we would choose a model for further use we would decide for BERTopic as it is consistenly at the top according to topic coherence for both dataset and does not lag behind too much when considering topic diversity

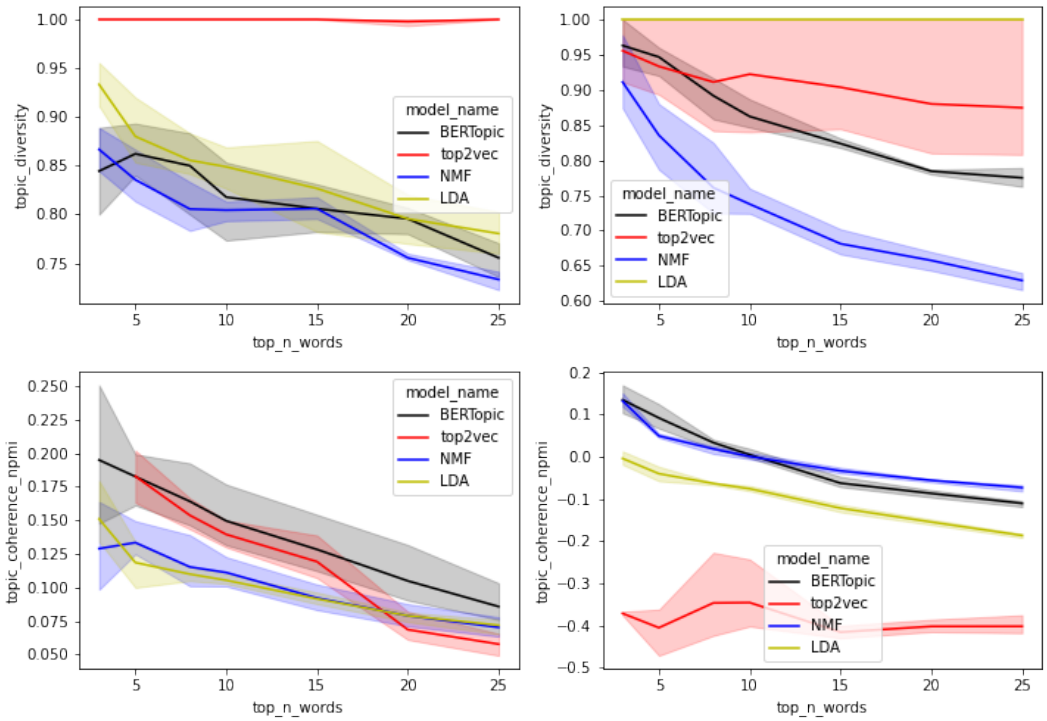

Figure 5.2: The topic diversity and topic coherence measures with respect to the number of top-n words parameter. On the left-hand, the results are for the news dataset and on the right-hand, the results are for the Twitter dataset.

5.1.3 Number of topics

On Figure 5.3, we can see that the topic diversity does not change much when increasing the number of topics except for BERTopic and top2vec on

the Twitter dataset.

The topic coherence also does not change with a larger number of topics on the Twitter dataset but we can see a consistent improvement in the topic coherence for all four models on the news dataset when we increase the number of topics. This means that the topics in the news dataset tend to be more specific and diverse so if we choose a low number of topics all models find some general topics from the corpus but by increasing the number of topics, the models can split the general topics into more specific ones. It is expected that all models achieve much higher topic coherence on the news dataset than on the Twitter dataset as documents are longer and contain formal language so the models can easier recognize topics. BERTopic consistently achieves the highest topic coherence. On the other hand top2vec achieves much lower topic coherence than other models on the Twitter dataset, which consistently happened in the previous experiments as well. This means that top2vec model is not suitable for corpora with short documents

5.2 Qualitative evaluation

In this section, we show the actual representations of the extracted topics and discuss pros and cons of each topic model.

As we have seen in section 5.1 on the quantitative evaluation, the performance of topic models varies with regard to different parameter values and it is hard to point out a single model as universally the best. As mentioned, the most important evaluation is qualitative, so we inspect the actual outputs of topic models, the topic representations, discuss differences and try to identify topic representations most useful for human users.

5.2.1 Default parameter value outputs

In Table 5.1, we can see the outputs of the models with default parameter values on the Twitter dataset. With the BERTopic model, we do not have much problems understanding which topics are represented by the selected words. Topics are granular and most of them really represent a single topic. For the topics 1, 2, and 6 we could say that they belong to the same topic

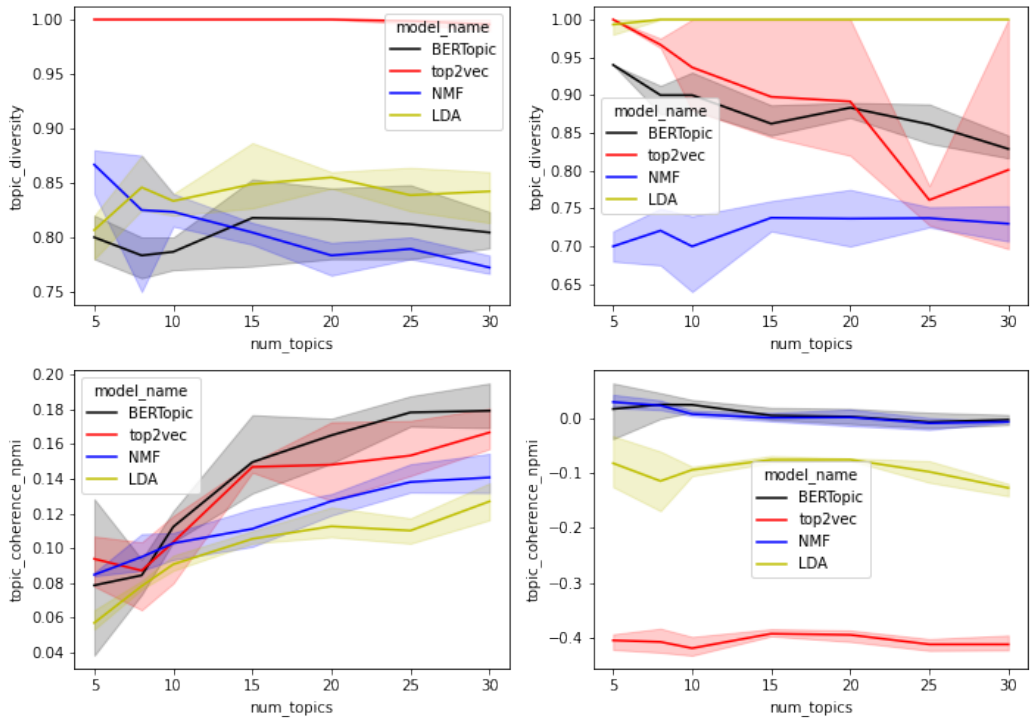

Figure 5.3: The topic diversity and topic coherence measures with respect to the number of topics parameter. On the left-hand, the results are for the news dataset and on the right-hand, the results are for the Twitter dataset.

of health and COVID-19. It is no surprise that this topic is prevalent, as the year 2020 (which we analyze) was the start of the COVID and therefore media were reporting about COVID and general health situation regularly and almost the entire year. But even in this case, the 3 topics are different enough that we could say that topic 1 represents the general covid topic, topic 2 is about general health situation, and topic 6 is more about COVID regulations.

In contrast to BERTopic, topic representations found by top2vec make much less sense. For a majority of words, we have trouble understanding what they really represent. For example, topic 1 to includes words sleep, corona virus, gun, sky and collaborate in one topic even though these words do not make much sense together and people would have trouble answering

a question whether some article belongs to this "topic".

Similarly to top2vec, LDA also yields much less coherent topic representations then BERTopic. Sometimes the topics look better than in top2vec with just a few words sticking out, but othertimes there are words that really do not belong together or some topics are too general.

The topic representations of NMF seem to be more informative than top2vec and LDA but are still nowhere near BERTopic. It is interesting that NMF is the only method that uses a lot of names as words representing the topics, such as RTV, Ljubljana, Hojs, Bojan.

Words in Table 5.2 represent the topics in the news dataset.

BERTopic discovers very clear and informative topics. While significant amount of discovered topics in the Twitter dataset were about politics and COVID, the topics detected in news are more diverse. Some of those include movies, music, and American politics.

Top2vec performs much better on the news dataset compared to the Twitter dataset. The represented topics are clear, concise and understandable for the reader. Some topics are very similar to the ones discovered by BERTopic (such as topics 6, 8, 10) and can be easily paired.

LDA and NMF perform much better than on the Twitter dataset, as well, and some topics are found by all four models. It would be hard to conclude that any model preforms much better than the rest, as all produce satisfiable results on the news dataset.

5.2.2 Lower number of topics

Using 15 topics for further analysis might be too much and sometimes the models make two topics that would make more sense if they would be merged. If we only want the most prevalent topics in a corpus and a short summary of documents in a corpus, we might want to lower the number of topics.

In Table 5.3, we can see five extracted topics from the Twitter dataset with each model. BERTopic again extracts very clear and coherent topics with COVID and health topics prevaling. On one hand, it might be more appropriate to join these two topics into one. On the other hand, the new

	BERTopic	top2vec	LDA	NMF
0	(imeti, slovenija, vlada, iti, človek, nov, čas, slovenski, država, vedeti)	(tovaris, igralec, firma, navodilo, mariborski, ustanova, edino, doseci, preteklost, rtvslo)	(človek, vedeti, Janša, delati, dati, videti, povedati, začeti, Janez, volitev)	(iti, RTV, reči, spati, priti, Rtv, dati, nebo, naprej, povedati)
1	(koronavirus, okužba, italija, nov, okužen, slovenija, test, epidemija, testiranje, covid)	(kam, spati, delez, uporabiti, coronavirus, puska, sodelovati, nebo, dvakrat, iranski)	(imeti, Slovenija, nov, čas, covid, ukrep, zato, epidemija, dr., koronavirus)	(nov, okužba, koronavirus, mesto, demokracija, star, stranka, medij, revija, okužen)
2	(zdravnik, zdravstven, zdravstvo, dr, zdravje, bolnišnica, bolnik, medicinski, pacient, covid)	(izgledati, motiti, zivec, strokovnjak, kupovati, tocka, zaposlitev, vplivati, vladen, skusati)	(zdravstven, ZDA, doma, gasilec, požar, oprema, ulica, zaščiten, začetek, letos)	(ukrep, covid, Janša, epidemija, delo, Janez, koronavirus, okužba, virus, število)
3	(sodišče, sodnik, ustaven, pravnik, tožilstvo, sodnica, sodstvo, sodba, vrhoven, zapor)	(droga, nauciti, poteza, vrhunski, dnevno, negativen, domoljub, skupnost, vrhoven, vladati)	(dobiti, reči, sodišče, meja, vojna, treba, rdeč, ustaven, pomeniti, mlad)	(svet, državen, medij, predsednik, priti, videti, seja, evropski, RTV, stranka)
4	(protest, protestnik, smrt, protestirati, petkov, ljubljana, grožnja, novica, janšizem, umor)	(ubiti, umrl, polovica, posten, jesti, spominjati, rtvslo, beovic, korupcija, ukrepati)	(praviti, gledati, življenje, prositi, jasen, smrt, zanimiv, zgoditi, novica, poročati)	(vlada, predsednik, lev, Janšev, prejšnji, Janez, Šarec, Janša, opozicija, podpora)
5	(tekma, šport, zmaga, prvak, olimpija, liga, športen, prvenstvo, nogomet, slovenski)	(zaliti, majica, ponoviti, navodilo, zbor, igralec, top, naloziti, jasa, prvak)	(demokracija, zdravnik, napad, ostal, hrvaški, naslednji, pokazati, sprejeti, nujen, uspeti)	(Slovenija, RTV, republika, koronavirus, predsednik, vojna, samostojen, Slovenec, evropski, zveza)
6	(maska, zaščiten, nositi, oprema, nošenje, imeti, nos, obraz, človek, obvezen)	(spomniti, merilo, kritizirati, nehati, jopic, fasizem, odlicno, sram, spodaj, branje)	(maska, ostati, podpora, potrebovati, policist, pot, zdeti, najprej, Janšev, nesreča)	(čas, vedeti, maska, dati, nositi, epidemija, star, večen, zaščiten, videti)
7	(otrok, zastava, starš, družina, mama, mati, oče, imeti, mlad, vzgojitelj)	(prenasati, zgaga, rezerva, veja, spati, dvakrat, srecati, vest, rumen, praznik)	(delo, otrok, policija, šola, čakati, služba, video, ime, odličen, slika)	(človek, delati, umreti, življenje, razumeti, dati, zdravje, videti, denar, misliti)
8	(komunist, komunističen, komunizem, narod, totalitarizem, imeti, slovenec, človek, partija, naslednik)	(obsoditi, teroristicen, narociti, bitka, noc, izgledati, jajce, desetletje, tvit, motiti)	(slovenski, predsednik, Slovenec, narediti, govoriti, del, narod, roka, postati, državljan)	(imeti, pravica, težava, zato, delati, pojem, povedati, reči, stranka, glas)
9	(vlada, tehničen, pomoč, oseba, gasilec, prijava, reševalec, obolel, helikopter, vrata)	(eksplozija, prevoz, pozar, gasilec, prihod, vikend, voznja, preprost, kamera, pgd)	(Ljubljana, pomoč, občina, pomagati, skupina, trg, poročilo, tehničen, poslati, prejeti)	(Ljubljana, gasilec, pomoč, protest, požar, občina, sodišče, prijava, začeti, vozilo)
10	(ljubljana, slovenija, slovenec, slovenski, stanovanje, kranj, ulica, imeti, janez, čas)	(eksplozija, gasilec, pgd, pozar, odpeljati, brezplacen, voznik, sarcev, cesta, resevalec)	(državen, konec, intervju, živeti, družba, prispevek, voditi, Šarec, laž, zgodovina)	(dr., intervju, ura, oddaja, čas, epidemija, Prof, Petrič, Jože, Bojan)
11	(policija, policist, policijski, sindikat, policaj, slovenski, minister, ura, slovenija, imeti)	(izgledati, prof, programski, udba, prodati, matjaz, corrupt, galet, strosek, ubiti)	(protest, ura, oddaja, vprašanje, verjeti, protestnik, prebrati, slišati, družina, policijski)	(slovenski, vojska, narod, politika, Slovenec, medij, predsednik, meja, državljan, vojak)
12	(volitev, demokracija, volilen, glas, slovenski, slovenija, stranka, glasovati, demokratičen, belorusija)	(slovenscina, stalisce, balkanski, trump, posameznik, obnasati, davkoplacevalec, peticija, neumen, ideologija)	(vlada, iti, država, svet, medij, javen, stranka, priti, političen, RTV)	(država, globok, praven, predsednik, evropski, demokracija, zato, živeti, normalen, demokratičen)
13	(minister, ministrica, notranji, zunanji, premier, vlada, ministrstvo, sekretar, obisk, delo)	(kolumna, galet, ustanova, sram, posel, zidan, razkritje, korupcija, pocitnice, odlicno)	(minister, misliti, hvala, današnji, zmaga, končno, delavec, knjiga, korona, premier)	(minister, javen, političen, stranka, zunanji, protest, delo, notranji, medij, Hojs)
14	(evro, milijon, proračun, eur, milijarda, denar, kredit, banka, mio, dobiti)	(nauciti, uspesen, dnevno, vrhunski, uciti, ukvarjati, okrevanje, trend, nasvet, desetletje)	(denar, pravica, levičar, problem, vojska, milijon, evro, podpirati, mnenje, kriza)	(dobiti, otrok, denar, volitev, covid, glas, dati, javen, šola, videti)

Table 5.1: Topic representations of the Twitter dataset with default parameters.

	BERTopic	top2vec	LDA	NMF
0	(imeti, nov, slovenija, čas, država, slovenski, človek, delo, vlada, ukrep)	(posvariti, bruselj, tiskoven, kanclerka, fonet, parlament, unija, premier, borrell, oblast)	(izdelek, sistem, nov, imeti, hrana, model, razvoj, uporaba, električen, različen)	(delo, otrok, dom, šola, čas, delavec, zaposlen, deloven, starš, delodajalec)
1	(tekma, točka, sezona, zmaga, liga, minuta, ekipa, mesto, igralec, turnir)	(koncnica, kosarkar, clippers, dvoboj, soigralec, priigrati, ljubljancan, podaja, poraz, tekma)	(tekma, liga, klub, sezona, točka, minuta, igrati, igralec, ekipa, nogometen)	(tekma, točka, mesto, slovenski, zmaga, minuta, konec, ekipa, dirka, sezona)
2	(občina, stanovanje, objekt, prostor, požar, gradnja, projekt, odpadek, nov, cesta)	(kanalizacijski, vgradnja, gradnja, gradben, novogradnja, izgradnja, razsvetljava, eko, toploten, vodovoden)	(občina, delo, šola, dom, otrok, javen, Ljubljana, zaposlen, center, zavod)	(občina, ura, javen, mesten, številka, občinski, center, občan, telefonski, pošta)
3	(okužba, nov, koronavirus, kitajski, država, človek, število, potrditi, umreti, virus)	(policist, kazniv, osumljenec, osumljen, povzrocitev, pridrzati, oskodovanec, prostost, dejanje, storilec)	(policija, evropski, sodišče, država, policist, dejanje, napad, protest, preiskava, vojska)	(okužba, koronavirus, covid, število, nov, ukrep, človek, bolnik, država, okužen)
4	(film, filmski, imeti, življenje, igralec, meghan, družina, vloga, režiser, igralka)	(nogometas, vezist, zabiti, zadetek, nogometen, prvoligas, branilec, enajstmetrovka, derbi, vratar)	(ženska, otrok, moški, družina, oče, leten, znan, fotografija, povedati, imeti)	(človek, država, slovenski, svet, policija, življenje, ameriški, vojna, iti, meja)
5	(okužba, nov, dom, cepivo, covid, okužen, bolnik, stanovalec, koronavirus, bolnišnica)	(bolnik, bolnisnicen, pacient, obolenje, obolel, hospitaliziran, infekcijski, prebolevati, dihalo, infektologinja)	(okužba, nov, koronavirus, covid, število, zdravstven, okužen, človek, bolnik, bolnišnica)	(nov, koronavirus, blago, ura, mesto, polje, potrditi, umreti, prejemnik, testiranje)
6	(stranka, vlada, desus, predsednik, koalicija, poslanec, volitev, smc, janša, poslanski)	(smc, desus, poslanski, lms, koalicija, sds, koalicijski, poslanec, nsi, sab)	(vlada, Slovenija, predsednik, stranka, minister, zakon, predlog, slovenski, državen, Janša)	(vlada, stranka, predsednik, Janša, poslanec, volitev, minister, političen, medij, koalicija)
7	(slovenija, ukrep, vlada, meja, država, odlok, hrvaški, trgovina, nov, javen)	(odlok, razkuzevanje, nijz, prehajanje, omejitev, izjema, ponujanje, upostevanje, obratovanje, higienski)	(ukrep, država, hrvaški, maska, Slovenija, nov, meja, vlada, koronavirus, veljati)	(ukrep, Slovenija, zakon, država, javen, vlada, epidemija, blago, člen, predlog)
8	(trump, biden, zda, ameriški, predsednik, volitev, trumpov, donald, država, demokrat)	(demokrat, republikanski, trump, republikanec, trumpov, biden, demokratski, donald, bidnov, clinton)	(ameriški, ZDA, država, kitajski, predsednik, Trump, agencija, poročati, tiskoven, svet)	(Ljubljana, Slovenija, slovenski, Nbsp, minister, evropski, predsednik, mednaroden, konferenca, ministrstvo)
9	(vozilo, cesta, nesreča, voznik, prometen, avtocesta, policist, ura, promet, prehod)	(padavina, vremenoslovec, veter, jugozahodnik, vremenski, nizina, burja, nevihta, dez, naliv)	(vozilo, cesta, območje, ura, nesreča, voznik, policist, kraj, prometen, hiša)	(vozilo, spleten, podatek, voznik, avtomobil, piškotek, uporabnik, policist, uporaba, vožnja)
10	(dirka, etapa, kolesar, roglič, pogačar, tour, zmaga, ekipa, majica, kolesarski)	(stopnicka, dirka, karavana, kronometer, etapen, tour, sprint, start, dirkati, roglic)	(sezona, mesto, svetoven, slovenski, dirka, ekipa, tekmovanje, športen, imeti, prvenstvo)	(liga, klub, sezona, tekma, prvak, evropski, igralec, igrati, ekipa, nogometen)
11	(ljubezen, življenje, imeti, partner, odnos, otrok, zato, čas, služba, človek)	(obozevalec, pevka, seksi, zaljubiti, obozevalka, obozevati, ljubezenski, zaljubljen, zmenek, postaven)	(imeti, čas, človek, zato, iti, življenje, vedeti, videti, svet, priti)	(čas, podjetje, ljubezen, zato, življenje, svet, pomemben, projekt, področje, partner)
12	(odstotek, indeks, evro, cena, rast, nafta, dolar, odstoten, borza, delnica)	(obresten, likvidnost, bruto, makroekonomski, rast, kreditiranje, medletno, prihodek, bdp, potrosnja)	(evro, odstotek, podjetje, milijon, družba, Slovenija, sredstvo, država, finančen, ukrep)	(odstotek, evro, milijon, podjetje, milijarda, cena, država, odstoten, lani, družba)
13	(glasba, glasben, koncert, pesem, album, skladba, festival, glasbenik, slovenski, skupina)	(literaren, opus, gledaliski, umetniski, pisatelj, dramski, umetnost, umetnik, razstava, pripoved)	(slovenski, film, Slovenija, knjiga, nagrada, Ljubljana, dogodek, nov, kulturen, projekt)	(župnija, služba, ljubezen, župnik, imenovan, Ljubljana, zdravje, razrešen, kaplan, splošno)
14	(piškotek, uporabnik, telefon, spleten, pameten, huawei, naprava, aplikacija, omogočati, mobilen)	(vmesnik, pameten, android, digitalen, googlov, windows, tehnologija, platforma, omrezen, operacijski)	(spleten, uporabnik, aplikacija, telefon, omrežje, podatek, naprava, pameten, mobilen, nov)	(imeti, človek, iti, vedeti, priti, reči, videti, čas, zato, delati)

Table 5.2: The topic representations of the news dataset with default parameters.

coronavirus and general health situation were reported very much and apparently BERTopic learned to tell them apart.

Topics extracted by LDA, NMF and top2vec look a bit more coherent than with 15 extracted topics, but they are still nowhere near BERTopic and do not produce satisfiable topic representations, useful for further analysis or downstream tasks.

	BERTopic	top2vec	LDA	NMF
0	(imeti, vlada, slovenija, iti, slovenski, človek, nov, država, čas, vedeti)	(srecati, lastnik, upanje, spregledati, skoda, opraviciti, cca, prenasati, kongres, pogovor)	(imeti, iti, človek, država, čas, vedeti, covid, ukrep, svet, maska)	(imeti, vedeti, država, pravica, težava, delati, zato, povedati, čas, maska)
1	(koronavirus, okužba, nov, italija, okužen, slovenija, epidemija, test, testiranje, covid)	(odlicno, obramben, kupovati, plebiscit, alenka, balkanski, dvigniti, cepivo, jopic, potres)	(Slovenija, nov, slovenski, predsednik, dr., Slovenec, koronavirus, mesto, evropski, politika)	(nov, čas, covid, ukrep, slovenski, okužba, koronavirus, minister, država, epidemija)
2	(zdravnik, zdravstven, zdravstvo, dr, zdravje, bolnišnica, bolnik, covid, medicinski, pacient)	(aja, poraz, zgaga, nor, veselje, krsiti, kolesariti, teroristicen, moder, vreme)	(Ljubljana, pomoč, policija, ura, sodišče, meja, oddaja, vojna, občina, beseda)	(Slovenija, država, RTV, slovenski, republika, predsednik, svet, Janša, koronavirus, evropski)
3	(sodišče, sodnik, ustaven, pravnik, tožilstvo, sodnica, sodstvo, sodba, vrhoven, zapor)	(najprej, preziveti, firma, opozorilo, nemski, polovica, milan, udba, policaj, ponavljati)	(vlada, minister, Janša, medij, javen, stranka, političen, državen, RTV, poslanec)	(vlada, predsednik, slovenski, Janša, Janez, minister, lev, Janšev, ukrep, prejšnji)
4	(protest, protestnik, smrt, protestirati, petkov, ljubljana, grožnja, novica, janšizem, umor)	(zjutraj, spustiti, resevalec, izpad, odpeljati, dneven, voznja, polozaj, pozar, gasilec)	(pravica, služba, koalicija, direktor, član, slišati, ime, laž, vesel, ženska)	(iti, človek, vedeti, država, maska, RTV, delati, dati, javen, videti)

Table 5.3: Topic representations of the Twitter dataset with 5 topics.

In Table 5.4, we can see five topics from the news dataset extracted by each model. We could say that BERTopic and top2vec extract a bit more coherent topics than LDA and NMF, but even LDA and NMF perform well enough. A very interesting thing to see is that BERTopic is the only model that does not extract a topic that represents covid-19 situation although there were 3 topics related to it when we extracted 15 topics. We can also see one caveat in topics 1 and 2 as they are very similar and could be easily

represented by a single topic.

In this case, top2vec looks like a clear winner (in contrast to the Twitter dataset). All its topics are very coherent and informative. They are also very distinct and well separated, and no two topics could be easily merged.

	BERTopic	top2vec	LDA	NMF
0	(imeti, nov, slovenija, čas, država, človek, slovenski, vlada, delo, ukrep)	(demokraticen, politicen, demokracija, opozicija, jansev, koalicija, jansa, koalicijski, ideoloski, parlamentaren)	(Slovenija, vlada, država, evro, predsednik, odstotek, delo, slovenski, evropski, podjetje)	(nov, okužba, koronavirus, država, človek, število, covid, potrditi, ukrep, okužen)
1	(tekma, klub, liga, sezona, nogometen, prvenstvo, prvak, nogometaš, imeti, ekipa)	(branilec, tekma, kapetan, soigralec, trener, zadetek, derbi, reprezentancen, reprezentant, dres)	(nov, okužba, koronavirus, država, človek, ukrep, covid, število, zdravstven, imeti)	(Slovenija, slovenski, vlada, država, Ljubljana, predsednik, evropski, evro, minister, stranka)
2	(tekma, točka, sezona, zmaga, liga, minuta, mesto, ekipa, imeti, igralec)	(okuzba, bolnik, obolel, covid, bolnisnicen, okuzen, hospitaliziran, sirjenje, beovicev, dihalo)	(tekma, sezona, liga, ekipa, klub, imeti, mesto, zmaga, slovenski, točka)	(tekma, sezona, liga, ekipa, imeti, klub, mesto, slovenski, točka, igrati)
3	(občina, stanovanje, objekt, prostor, nov, projekt, požar, gradnja, delo, cesta)	(obresten, depozit, dobickonosnost, donosnost, portfelj, likvidnost, digitalizacija, makroekonomski, kreditiranje, investiranje)	(občina, nov, vozilo, ura, policist, imeti, cesta, spleten, prostor, območje)	(občina, ukrep, javen, delo, ura, čas, mesten, dom, epidemija, center)
4	(film, imeti, filmski, življenje, igralec, družina, vloga, meghan, festival, režiser)	(romanticen, ljubezenski, obozevalka, pevka, ljubiteljica, seksi, pesem, videospot, ljubimec, zaljubiti)	(imeti, čas, človek, življenje, otrok, iti, zato, svet, nov, slovenski)	(imeti, človek, čas, iti, otrok, zato, življenje, vedeti, priti, videti)

Table 5.4: Topic representations of the news dataset with 5 topics.

5.2.3 Shorter topic representations

In the Table 5.5, we can see the topics extracted by all models with 5 words representing the topics in the Twitter dataset. BERTopic and LDA just use the same top words as with more top-n words while top2vec and NMF use different words representing the topics and therefore we get altogether different topics. Unfortunately, this does not mean that the topics are more coherent

or informative than with more words. Using less words with BERTopic gives more clear topics as taking away less informative words reduces the chances of having a word in the representation that does not belong to the topic. This is in line with what we found with quantitative analysis.

In Table 5.6, we can see the topics extracted by all models with 5 words representing the topics in the news dataset. As mentioned before BERTopic and LDA just use the same top-n words, which does not affect the quality of topics much as they were good before as well. Top2vec extracts quality topics with very descriptive words while NMF does achieve the same performance compared to using more top-n words. Here the words are too general and with a lot of topics we do not really know what they represent.

5.3 Similarities of discovered topics

In this section, we use our newly defined MBTS metric to analyze the similarity of discovered topics between different topic models and also within the same topic model, depending on the random seed. Using this method, we open a new aspect of topic mode evaluation and comparison.

High value of MBTS for the same model with two different random seeds means that a model is stable as it discovers similar topics regardless of random seed which is a highly desirable trait. Models with low values for different seed extract semantically different topics with each run so we cannot trust the actual topic representations in further analyses.

5.3.1 Default parameters similarity matrices

On Figure 5.4, we can see topic model similarities between different and same models for different initializations.

All models find topics that are more similar to each other on the news dataset compared to the Twitter dataset. This is expected as we already saw previously that on this dataset NMF, LDA and top2vec extract unclear topics with very random representations. BERTopic is the most stable model as it has the highest self-similaritiy on both the news and Twitter datasets. Besides BERTopic, NMF also achieves relatively high self-similarities although

	BERTopic	top2vec	LDA	NMF
0	(imeti, slovenija, vlada, iti, človek)	(pustiti, kupiti, svoboden, spomenka, vecer)	(človek, vedeti, Janša, delati, dati)	(imeti, pravica, težava, zato, delati)
1	(koronavirus, okužba, italija, nov, okužen)	(tocno, hkrati, coronavirus, prispevati, polovica)	(imeti, Slovenija, nov, čas, covid)	(nov, okužba, koronavirus, mesto, demokracija)
2	(zdravnik, zdravstven, zdravstvo, dr, zdravje)	(spati, edin, delez, zacenjati, nebo)	(zdravstven, ZDA, doma, gasilec, požar)	(ukrep, covid, Janša, epidemija, delo)
3	(sodišče, sodnik, ustaven, pravnik, tožilstvo)	(mafija, razkriti, ukrepati, preganjati, beovic)	(dobiti, reči, sodišče, meja, vojna)	(vlada, predsednik, lev, Janšev, prejšnji)
4	(protest, protestnik, smrt, protestirati, petkov)	(napacen, umreti, dejati, erika, nauciti)	(protest, ura, oddaja, vprašanje, verjeti)	(iti, RTV, reči, spati, priti)
5	(tekma, šport, zmaga, prvak, olimpija)	(zmagovalec, srecati, biden, pustiti, jokati)	(minister, misliti, hvala, današnji, zmaga)	(dr., intervju, ura, oddaja, čas)
6	(maska, zaščiten, nositi, oprema, nošenje)	(spati, rezerva, slucajno, veter, poletje)	(maska, ostati, podpora, potrebovati, policist)	(čas, vedeti, maska, dati, nositi)
7	(otrok, zastava, starš, družina, mama)	(spati, nebo, skrajnez, kritizirati, desnicar)	(praviti, gledati, življenje, prositi, jasen)	(dobiti, otrok, denar, volitev, covid)
8	(komunist, komunističen, komunizem, narod, totalitarizem)	(peticija, obnasati, sedanji, pamet, davkoplacevalec)	(denar, pravica, levičar, problem, vojska)	(človek, delati, umreti, življenje, razumeti)
9	(vlada, tehničen, pomoč, oseba, gasilec)	(zaupanje, opozicijski, naloga, prispevati, stanje)	(Ljubljana, pomoč, občina, pomagati, skupina)	(Ljubljana, gasilec, pomoč, protest, požar)
10	(ljubljana, slovenija, slovenec, slovenski, stanovanje)	(kam, naselje, matjaz, blagoven, antifa)	(slovenski, predsednik, Slovenec, narediti, govoriti)	(slovenski, vojska, narod, politika, Slovenec)
11	(policija, policist, policijski, sindikat, policaj)	(pgd, eksplozija, pozar, voznik, naselje)	(delo, otrok, policija, šola, čakati)	(svet, državen, medij, predsednik, priti)
12	(volitev, demokracija, volilen, glas, slovenski)	(plebiscit, spodaj, letnica, nos, merilo)	(demokracija, zdravnik, napad, ostal, hrvaški)	(država, globok, praven, predsednik, evropski)
13	(minister, ministrica, notranji, zunanji, premier)	(blagoven, potreba, franc, odhod, prebivalec)	(vlada, iti, država, svet, medij)	(minister, javen, političen, stranka, zunanji)
14	(evro, milijon, proračun, eur, milijarda)	(metati, enoten, premalo, opozicijski, spraviti)	(državen, konec, intervju, živeti, družba)	(Slovenija, RTV, republika, koronavirus, predsednik)

Table 5.5: Topic representations of the Twitter dataset with 5 top-n words.

	BERTopic	top2vec	LDA	NMF
0	(imeti, nov, slovenija, čas, država)	(okusen, solata, zacimba, slasten, kis)	(imeti, čas, človek, zato, iti)	(človek, država, slovenski, svet, policija)
1	(tekma, točka, sezona, zmaga, liga)	(koncnica, kosarkar, dvoboj, ljubljancan, soigralec)	(tekma, liga, klub, sezona, točka)	(liga, klub, sezona, tekma, prvak)
2	(občina, stanovanje, objekt, prostor, požar)	(gradnja, direkcija, izgradnja, kanalizacija, gradben)	(občina, delo, šola, dom, otrok)	(delo, otrok, dom, šola, čas)
3	(okužba, nov, koronavirus, kitajski, država)	(okuziti, koronavirus, okuzba, smrten, ozdraveti)	(ukrep, država, hrvaški, maska, Slovenija)	(nov, koronavirus, blago, ura, mesto)
4	(film, filmski, imeti, življenje, igralec)	(nogometas, vezist, nogometen, zabiti, zadetek)	(izdelek, sistem, nov, imeti, hrana)	(imeti, človek, iti, vedeti, priti)
5	(okužba, nov, dom, cepivo, covid)	(bolnisnicen, nijz, bolnik, obolel, beovicev)	(okužba, nov, koronavirus, covid, število)	(okužba, koronavirus, covid, število, nov)
6	(stranka, vlada, desus, predsednik, koalicija)	(sds, jansev, koalicija, jans, levica)	(vlada, Slovenija, predsednik, stranka, minister)	(vlada, stranka, predsednik, Janša, poslanec)
7	(slovenija, ukrep, vlada, meja, država)	(vremenoslovec, padavina, veter, nevihta, vremenski)	(policija, evropski, sodišče, država, policist)	(ukrep, Slovenija, zakon, država, javen)
8	(trump, biden, zda, ameriški, predsednik)	(unija, bruselj, zaveznica, miroven, borrell)	(ameriški, ZDA, država, kitajski, predsednik)	(Ljubljana, Slovenija, slovenski, Nbsp, minister)
9	(vozilo, cesta, nesreča, voznik, prometen)	(osumljenec, kazniv, policist, osumljen, storilec)	(vozilo, cesta, območje, ura, nesreča)	(vozilo, spleten, podatek, voznik, avtomobil)
10	(dirka, etapa, kolesar, roglič, pogačar)	(karavana, dirka, stopnicka, sprint, kronometer)	(sezona, mesto, svetoven, slovenski, dirka)	(tekma, točka, mesto, slovenski, zmaga)
11	(ljubezen, življenje, imeti, partner, odnos)	(ljubezen, seksi, ljubiti, partnerjev, prijateljica)	(ženska, otrok, moški, družina, oče)	(čas, podjetje, ljubezen, zato, življenje)
12	(odstotek, indeks, evro, cena, rast)	(obresten, bruto, kreditiranje, rast, medletno)	(evro, odstotek, podjetje, milijon, družba)	(odstotek, evro, milijon, podjetje, milijarda)
13	(glasba, glasben, koncert, pesem, album)	(priredba, glasben, glasba, obcinstvo, pesem)	(slovenski, film, Slovenija, knjiga, nagrada)	(župnija, služba, ljubezen, župnik, imenovan)
14	(piškotek, uporabnik, telefon, spleten, pameten)	(vmesnik, android, snapdragon, pameten, procesor)	(spleten, uporabnik, aplikacija, telefon, omrežje)	(občina, ura, javen, mesten, številka)

Table 5.6: Topic representations of the news dataset with 5 top-n words.

that does not help much as the topics are unclear and incoherent.

NMF and LDA are the most similar to each other on the news dataset which is especially evident with the fasttext embeddings. This is expected, as they are both statistical methods. Top2vec extracts topics that are very different from other topic models, even from BERTopic, although they are both neural models.

5.3.2 Lower number of topics

The results are a bit different when we reduce the number of topics as we can see on Figure 5.5. We see an improvement in LDA and NMF stability as they even achieve higher self-similarity than BERTopic on the news dataset, while the results for the Twitter dataset are similar to results with more topics. Similarities between LDA and NMF are even more evident in this case.

This means that when we have a corpus with long documents and we want only a small number of topics, LDA and NMF might be more suitable than BERTopic as they manage to generalize well and find the most important topics in each run while BERTopic seems to vary more.

5.3.3 Shorter topic representations

Similarities of topics represented with less words are quite similar to those with more words as seen on Figure 5.6. On one hand, we would expect improvement in self-similarities as we exclude less-representative words that are more prone to random selection. On the other hand, we could expect lower self-similarities as using less words introduces more randomness to selection, so obtaining rather similar results makes sense.

5.3.4 More documents

Using more documents, we can see a slight improvement in stability in almost all models, as shown in Figure 5.7. This is expected as having more documents makes it easier for the models to extract the prevalent topics as we reduce randomness in the dataset.

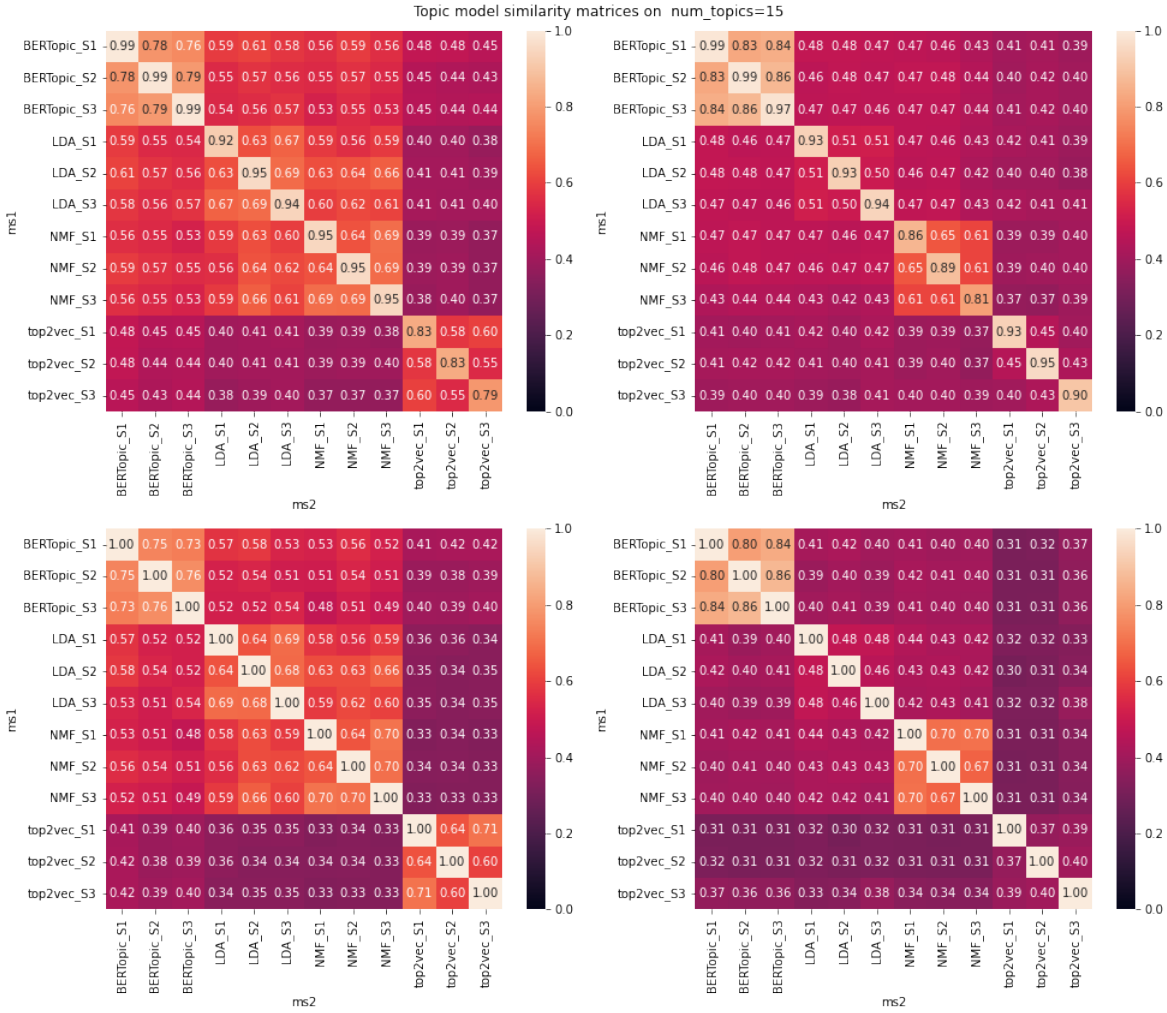

Figure 5.4: The topic model similarity matrix of models with default parameters with different seeds. In the left column, the results show the news dataset, and in the right the Twitter dataset. In the first row, we use word2vec embeddings and in the second row, we use fasttext embeddings. Each matrix shows MBTS values between topic models in rows and columns. Values are between 0 (no similarity) and 1 (perfect match). Each used initialization seed is denoted with _S.

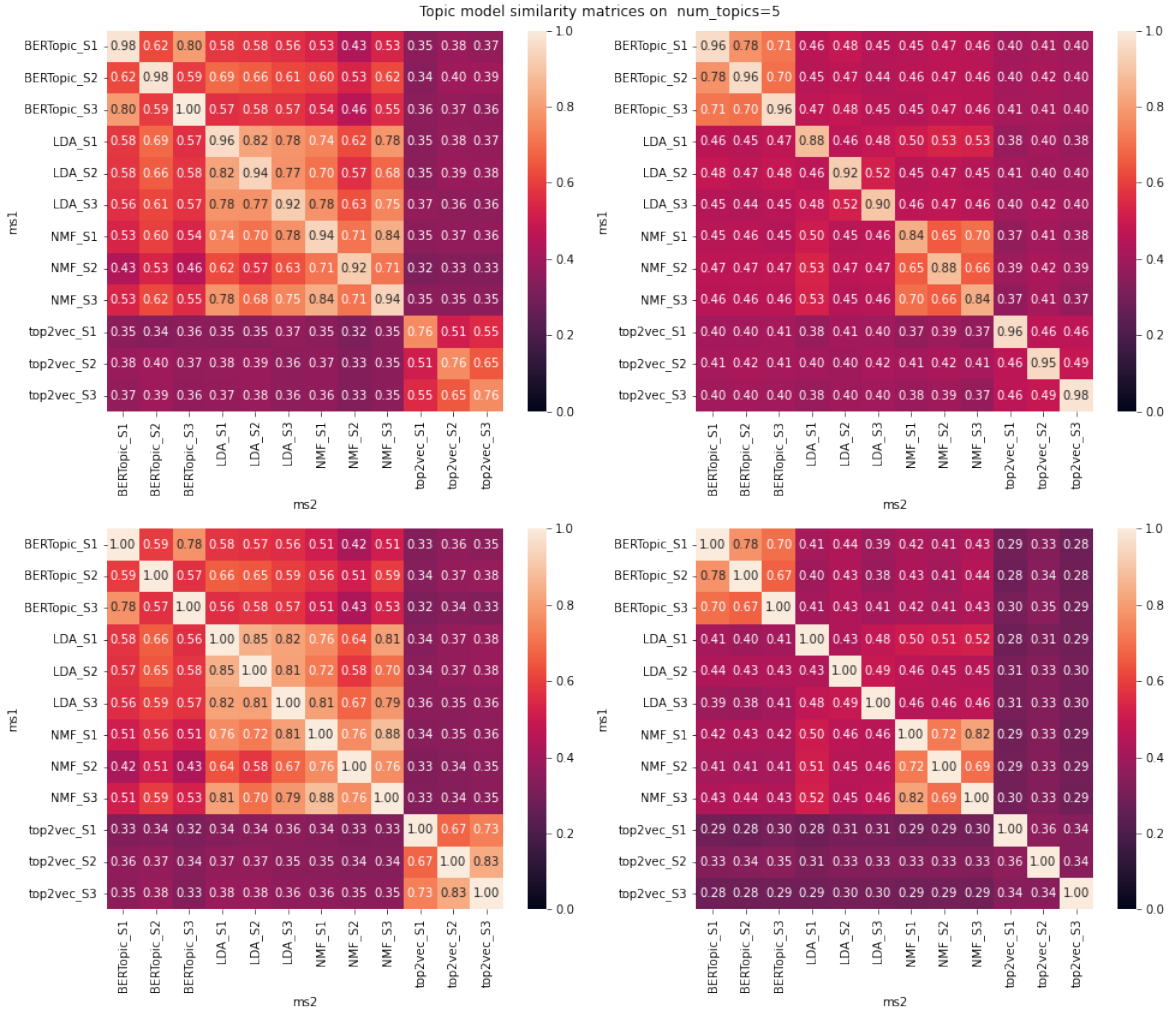

Figure 5.5: The topic model similarity matrix of models with 5 topics with different seeds. In the left column, the results show the news dataset, and in the right the Twitter dataset. In the first row, we use word2vec embeddings and in the second row, we use fasttext embeddings. Each matrix shows MBTS values between topic models in rows and columns. Values are between 0 (no similarity) and 1 (perfect match). Each used initialization seed is denoted with _S.

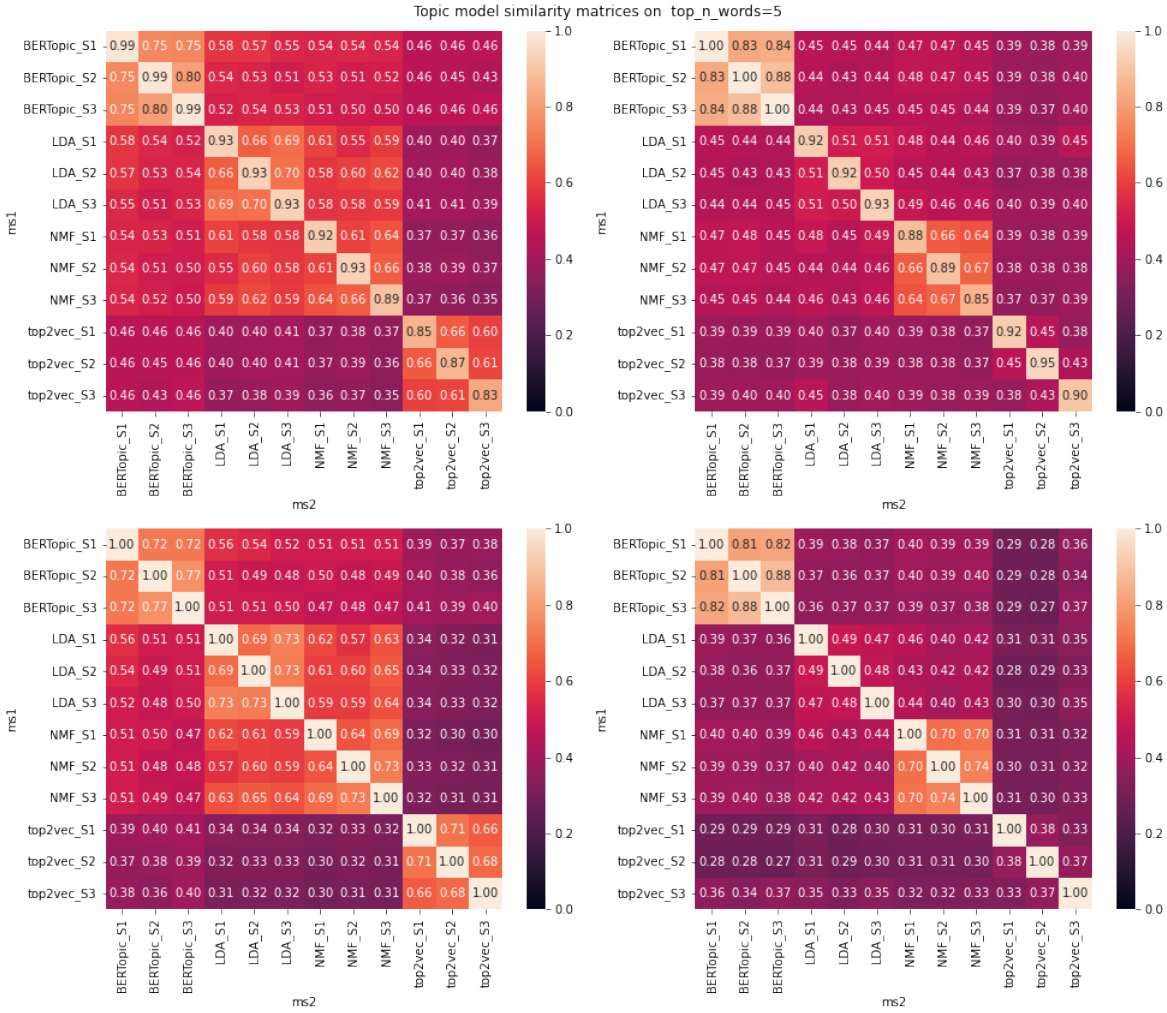

Figure 5.6: The topic model similarity matrix of models with 5 top-n words with different seeds. In the left column, the results show the news dataset, and in the right the Twitter dataset. In the first row, we use word2vec embeddings and in the second row, we use fasttext embeddings. Each matrix shows MBTS values between topic models in rows and columns. Values are between 0 (no similarity) and 1 (perfect match). Each used initialization seed is denoted with _S.

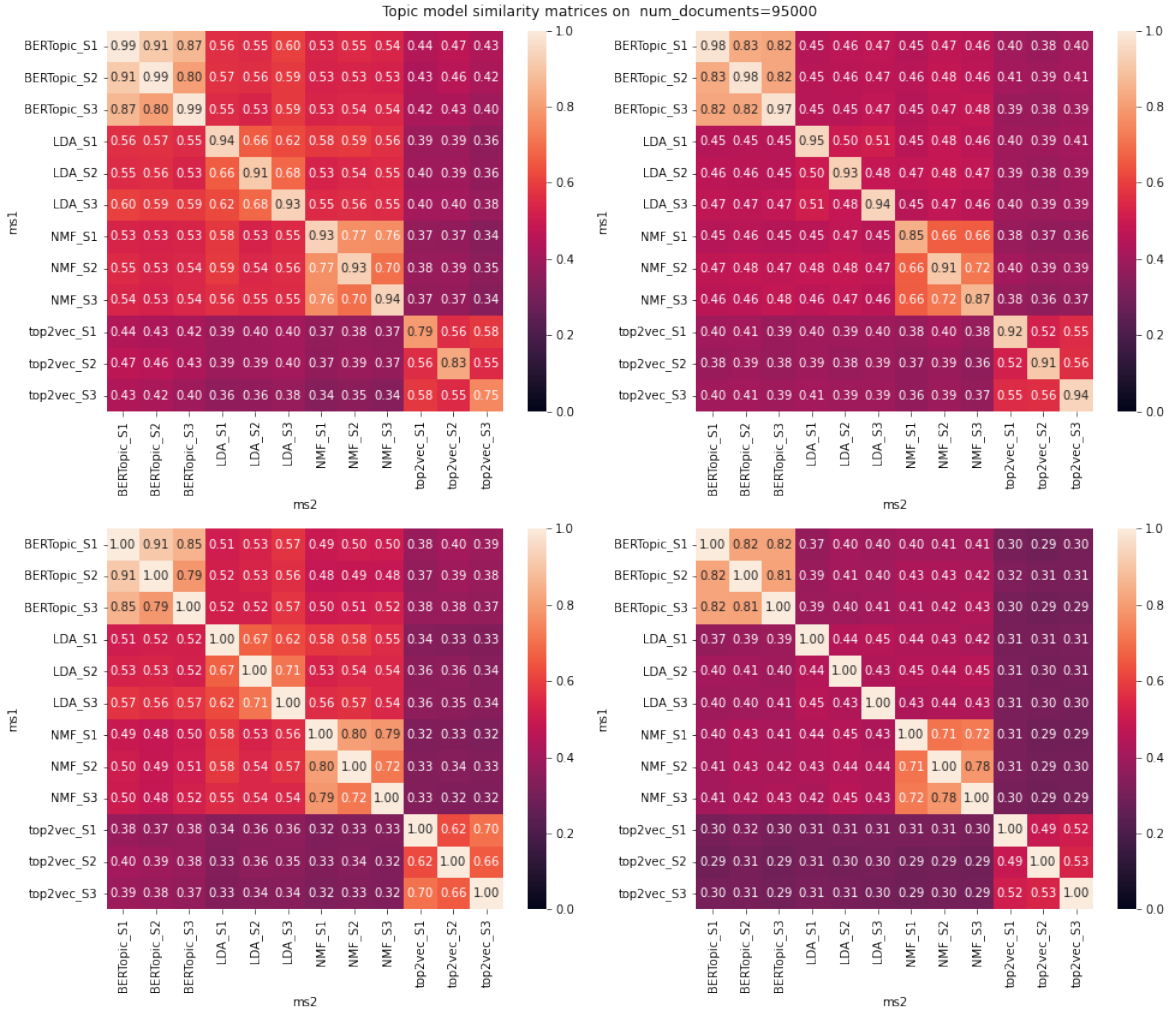

Figure 5.7: The topic model similarity matrix of models on 95k documents with different seeds. In the left column, the results show the news dataset, and in the right the Twitter dataset. In the first row, we use word2vec embeddings and in the second row, we use fasttext embeddings. Each matrix shows MBTS values between topic models in rows and columns. Values are between 0 (no similarity) and 1 (perfect match). Each used initialization seed is denoted with _S.

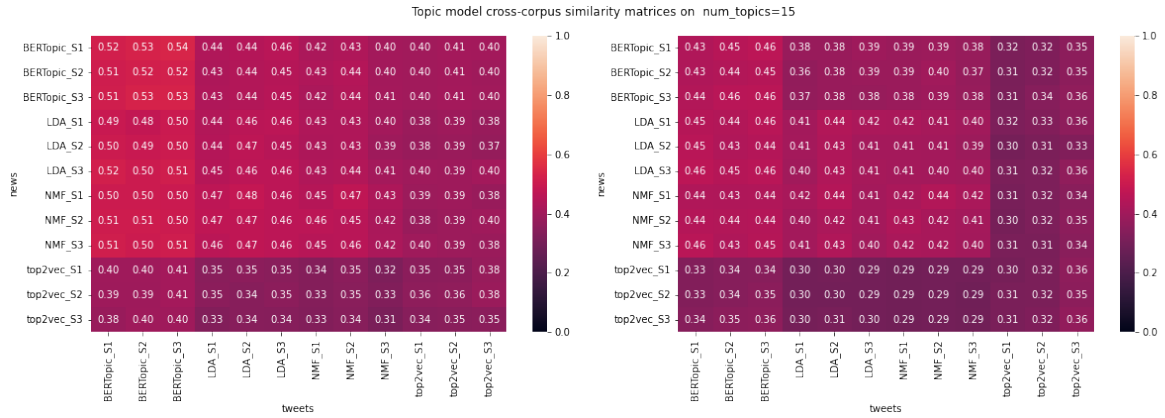

Figure 5.8: The topic model similarity matrix of models with default parameters on different datasets with different seeds. We compare similarities of discovered topics on news versus tweets dataset with different models and random seeds. On the left plot we use word2vec embeddings and on the right we use fasttext embeddings. Each matrix shows MBTS values between topic models on the news dataset in rows and topic models on the Twitter dataset columns. Values are between 0 (no similarity) and 1 (perfect match). Each used initialization seed is denoted with _S.

5.3.5 Cross corpus similarity matrix

We can also use MBTS to measure topic similarities between two corpora. On Figure 5.8, we can see similarities of different models on the news and Twitter dataset. Again, BERTopic finds the most similar topics. As news and tweets usually cover many different topics we can not really say if the topics in our corpora are very similar. It would be more helpful to compare these two datasets to some other related corpus.

Chapter 6

Conclusion

In this thesis we explored the application of four topic modeling algorithms NMF, LDA, top2vec, and BERTopic for the Slovene language, utilizing two distinct datasets composed of news articles and tweets. Quantitative assessment was conducted using topic coherence and topic diversity metrics, with variations in the number of topics, top-n words, and the corpus size. We found that all models exhibited lower topic coherence in the Twitter dataset compared to the news dataset, although coherence increased with more documents. Topic coherence and diversity both decreased when increasing the number of topic words across both datasets. The news dataset appeared to have more diverse and specific topics, as topic coherence increased for all models when increasing the number of topics. While BERTopic often outperformed other models in terms of topic coherence, confidence intervals frequently overlapped, making it difficult to identify a clear winner and a manual inspection of the outputs is recommended.

We analyzed the topics generated by each topic model and discovered that all models yielded satisfactory results for the news dataset. However, only BERTopic managed to extract coherent topics from the Twitter dataset. Topics derived from the other models were ambiguous, often containing words that seemed out of place.

We presented a novel method, MBTS, for assessing topic model stability and similarity, which relies on semantic similarity and maximum bipartite matching. This approach enables semantic comparisons of topic models and

evaluations of their stability by measuring topic similarity across different initializations. Our findings revealed that BERTopic is the most stable model on our datasets, followed by NMF. LDA and NMF exhibit the highest similarity, while top2vec differs significantly from the other three models.

To enhance BERTopic and top2vec performance, we could experiment with alternative pretrained sentence encoders, such as LaBSE, or even develop a Slovene-specific encoder from scratch. Additionally, ensemble techniques may improve model performance and stability. MBTS could be applied to evaluate topic models across various datasets and languages, and its potential could be further explored by employing different word embeddings and alternative methods for matching similar topics.

Bibliography

[1] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent Dirichlet allocation, Journal of machine learning research 3 (Jan) (2003) 993–1022.

[2] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (6755) (1999) 788–791.

[3] D. Angelov, Top2vec: Distributed representations of topics, arXiv preprint arXiv:2008.09470 (2020).

[4] M. Grootendorst, Bertopic: Neural topic modeling with a class-based TF-IDF procedure, arXiv e-prints (2022) arXiv–2203.

[5] Z. S. Harris, Distributional structure, Word 10 (2-3) (1954) 146–162.

[6] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Inf. Process. Manag. 24 (1988) 513–523.

[7] P. Jaccard, Étude comparative de la distribution florale dans une portion des Alpes et des Jura, Bull Soc Vaudoise Sci Nat 37 (1901) 547–579.

[8] L. R. Dice, Measures of the amount of ecologic association between species, Ecology 26 (3) (1945) 297–302.

[9] J. H. W. Jr., Hierarchical grouping to optimize an objective function, Journal of the American Statistical Association 58 (301) (1963) 236–244. doi:10.1080/01621459.1963.10500845.

[10] J. MacQueen, Classification and analysis of multivariate observations, in: 5th Berkeley Symp. Math. Statist. Probability, 1967, pp. 281–297.

[11] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, R. A. Harshman, Indexing by latent semantic analysis, J. Am. Soc. Inf. Sci. 41 (1990) 391–407.

[12] T. Hofmann, Probabilistic latent semantic analysis, in: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, UAI'99, 1999, p. 289–296.

[13] Y. Bengio, R. Ducharme, P. Vincent, A neural probabilistic language model, Advances in neural information processing systems 13 (2000).

[14] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: 1st International Conference on Learning Representations, ICLR 2013, Workshop Track Proceedings, 2013.

[15] J. Pennington, R. Socher, C. D. Manning, GloVe: Global vectors for word representation, in: Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543.

[16] A. Joulin, É. Grave, P. Bojanowski, T. Mikolov, Bag of tricks for efficient text classification, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, 2017, pp. 427–431.

[17] J. Sarzynska-Wawer, A. Wawer, A. Pawlak, J. Szymanowska, I. Stefaniak, M. Jarkiewicz, L. Okruszek, Detecting formal thought disorder by deep contextualized word representations, Psychiatry Research 304 (2021) 114135.

[18] Q. Le, T. Mikolov, Distributed representations of sentences and documents, in: International conference on machine learning, PMLR, 2014, pp. 1188–1196.

[19] N. Reimers, I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: Proceedings of the 2019 Conference on

Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3982–3992.

[20] L. McInnes, J. Healy, N. Saul, L. Großberger, UMAP: Uniform Manifold Approximation and Projection, Journal of Open Source Software 3 (29) (2018) 861.

[21] L. McInnes, J. Healy, S. Astels, HDBSCAN: Hierarchical density based clustering., J. Open Source Softw. 2 (11) (2017) 205.

[22] K. Lang, Newsweeder: Learning to filter netnews, in: Machine Learning Proceedings 1995, 1995, pp. 331–339.

[23] D. Greene, P. Cunningham, Practical solutions to the problem of diagonal dominance in kernel document clustering, in: Proceedings of the 23rd international conference on machine learning, 2006, pp. 377–384.

[24] D. Newman, C. Chemudugunta, P. Smyth, M. Steyvers, Analyzing entities and topics in news articles using statistical topic models, in: International conference on intelligence and security informatics, 2006, pp. 93–104.

[25] J. Tang, Z. Meng, X. Nguyen, Q. Mei, M. Zhang, Understanding the limiting factors of topic modeling via posterior contraction analysis, in: International conference on machine learning, PMLR, 2014, pp. 190–198.

[26] T.-I. Yang, A. Torget, R. Mihalcea, Topic modeling on historical newspapers, in: Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, 2011, pp. 96–104.

[27] A. Abuzayed, H. Al-Khalifa, BERT for Arabic topic modeling: an experimental study on BERTopic technique, Procedia Computer Science 189 (2021) 191–194.

[28] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, X. Li, Comparing Twitter and traditional media using topic models, in: Advances in Information Retrieval, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 338–349.

[29] A. F. Hidayatullah, E. C. Pembrani, W. Kurniawan, G. Akbar, R. Pranata, Twitter topic modeling on football news, in: 2018 3rd International Conference on Computer and Communication Systems (ICCCS), 2018, pp. 467–471. doi:10.1109/CCOMS.2018.8463231.

[30] P. Ma, Q. Zeng-Treitler, S. J. Nelson, Use of two topic modeling methods to investigate COVID vaccine hesitancy, in: Int. Conf. ICT Soc. Hum. Beings, Vol. 384, 2021, pp. 221–226.

[31] R. Egger, J. Yu, A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts, Frontiers in Sociology 7 (2022).

[32] J. Bajt, M. Robnik-Šikonja, Strojna analiza tematik in sentimenta slovenskih novičarskih medijev, Uporabna informatika 30 (1) (maj 2022). doi:10.31449/upinf.159.

[33] N. Berginc, N. Ljubešić, Gigafida in slWaC: tematska primerjava, Slovenščina 2.0: empirical, applied and interdisciplinary research 1 (2013) 78–110. doi:10.4312/slo2.0.2013.1.78-110.

[34] A. B. Dieng, F. J. Ruiz, D. M. Blei, Topic modeling in embedding spaces, Transactions of the Association for Computational Linguistics 8 (2020) 439–453.

[35] J. H. Lau, D. Newman, T. Baldwin, Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality, in: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, 2014, pp. 530–539. doi:10.3115/v1/E14-1056.

[36] A. Srivastava, C. Sutton, Autoencoding variational inference for topic models, Stat 1050 (2017) 4.

[37] Y. Miao, E. Grefenstette, P. Blunsom, Discovering discrete latent topics with neural variational inference, in: International Conference on Machine Learning, PMLR, 2017, pp. 2410–2419.

[38] L. Hong, B. Davison, Empirical study of topic modeling in Twitter, SOMA 2010 - Proceedings of the 1st Workshop on Social Media Analytics (01 2010). doi:10.1145/1964858.1964870.

[39] I. Škrjanec, S. Pollak, Topic ontologies of the Slovene blogosphere: a gender perspective, in: Proceedings of the 4th Conference on CMC and Social Media Corpora for the Humanities, 2016, p. 62–65.

[40] M. S. Maučec, Z. Kačič, B. Horvat, Modelling highly inflected languages, Information Sciences 166 (1-4) (2004) 249–269.

[41] K. Miok, E. Hidalgo-Tenorio, P. Osenova, M.-A. Benitez-Castro, M. Robnik-Sikonja, Multi-aspect, multilingual and cross-lingual parliamentary speech analysis, arXiv preprint arXiv:2207.01054 (2022).

[42] M. Röder, A. Both, A. Hinneburg, Exploring the space of topic coherence measures, WSDM '15, New York, NY, USA, 2015, p. 399–408. doi:10.1145/2684822.2685324.

[43] D. Newman, J. H. Lau, K. Grieser, T. Baldwin, Automatic evaluation of topic coherence, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, California, 2010, pp. 100–108. URL https://aclanthology.org/N10-1012

[44] A. Hoyle, P. Goel, A. Hian-Cheong, D. Peskov, J. Boyd-Graber, P. Resnik, Is automated topic model evaluation broken? The incoherence of coherence, Advances in Neural Information Processing Systems 34 (2021) 2018–2033.

[45] J. Lee, J.-H. Kang, S. Jun, H. Lim, D. Jang, S. Park, Ensemble modeling for sustainable technology transfer, Sustainability 10 (2018) 2278. doi:10.3390/su10072278.

[46] T. Hofmann, Probabilistic latent semantic indexing, The 22nd International Conference on Research and Development in Information Retrieval (SIGIR'99).

[47] J. D. M.-W. C. Kenton, L. K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of NAACL-HLT, 2019, pp. 4171–4186.

[48] S. R. Bowman, G. Angeli, C. Potts, C. D. Manning, A large annotated corpus for learning natural language inference, in: EMNLP, 2015.

[49] A. Williams, N. Nangia, S. R. Bowman, A broad-coverage challenge corpus for sentence understanding through inference, in: 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2018, Association for Computational Linguistics (ACL), 2018, pp. 1112–1122.

[50] N. Reimers, I. Gurevych, Making monolingual sentence embeddings multilingual using knowledge distillation, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 4512–4525.

[51] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE., Journal of machine learning research 9 (11) (2008).

[52] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Knowledge Discovery and Data Mining, 1996.

[53] L. McInnes, How HDBSCAN works, https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html, accessed: 2022-11-17.

[54] X. Rong, Word2vec parameter learning explained, arXiv preprint arXiv:1411.2738 (2014).

[55] J. Lau, K. Grieser, D. Newman, T. Baldwin, Automatic labelling of topic models., 2011, pp. 1536–1545.

[56] J. Chang, S. Gerrish, C. Wang, J. Boyd-Graber, D. Blei, Reading tea leaves: How humans interpret topic models, Advances in neural information processing systems 22 (2009).

[57] M. Röder, A. Both, A. Hinneburg, Exploring the space of topic coherence measures, in: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, WSDM '15, New York, NY, USA, 2015, p. 399–408. doi:10.1145/2684822.2685324.

[58] P. J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20 (1987) 53–65. doi:https://doi.org/10.1016/0377-0427(87)90125-7.

[59] W. M. Rand, Objective criteria for the evaluation of clustering methods, Journal of the American Statistical Association 66 (1971) 846–850.

[60] L. Hubert, P. Arabie, Comparing partitions, Journal of classification 2 (1985) 193–218.

[61] S. Bhagwani, S. Satapathy, H. Karnick, Sranjans : Semantic textual similarity using maximal weighted bipartite graph matching, in: *SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012), 2012, pp. 579–585.

[62] M. Fares, A. Kutuzov, S. Oepen, E. Velldal, Word vectors, reuse, and replicability: Towards a community repository of large-text resources,

in: Proceedings of the 21st Nordic Conference on Computational Linguistics, NODALIDA 2017, 2017, pp. 271–276.

[63] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, T. Mikolov, Learning word vectors for 157 languages, in: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), 2018.

[64] J. Mu, S. Bhat, P. Viswanath, All-but-the-top: Simple and effective postprocessing for word representations, arXiv preprint arXiv:1702.01417 (2017).

[65] N. Ljubešić, D. Fišer, T. Erjavec, TweetCaT: A tool for building Twitter corpora of smaller languages, in: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), 2014, pp. 2279–2283.

[66] N. Ljubešić, K. Dobrovoljc, What does neural bring? Analysing improvements in morphosyntactic annotation and lemmatisation of Slovenian, Croatian and Serbian, in: Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing, 2019, pp. 29–34. doi:10.18653/v1/W19-3704.

[67] R. Řehůřek, P. Sojka, Software framework for topic modelling with large corpora, in: Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks, 2010, pp. 46–50.