BERTopic: Neural topic modeling with a class-based TF-IDF procedure

Maarten Grootendorst
maartengrootendorst@gmail.com

Abstract

Topic models can be useful tools to discover latent topics in collections of documents. Recent studies have shown the feasibility of approach topic modeling as a clustering task. We present BERTopic, a topic model that extends this process by extracting coherent topic representation through the development of a class-based variation of TF-IDF. More specifically, BERTopic generates document embedding with pre-trained transformer-based language models, clusters these embeddings, and finally, generates topic representations with the class-based TF-IDF procedure. BERTopic generates coherent topics and remains competitive across a variety of benchmarks involving classical models and those that follow the more recent clustering approach of topic modeling.

1 Introduction

To uncover common themes and the underlying narrative in text, topic models have proven to be a powerful unsupervised tool. Conventional models, such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and Non-Negative Matrix Factorization (NMF) (Févotte and Idier, 2011), describe a document as a bag-of-words and model each document as a mixture of latent topics.

One limitation of these models is that through bag-of-words representations, they disregard semantic relationships among words. As these representations do not account for the context of words in a sentence, the bag-of-words input may fail to accurately represent documents.

As an answer to this issue, text embedding techniques have rapidly become popular in the natural language processing field. More specifically, Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) and its variations (e.g., Lee et al., 2020; Liu et al., 2019; Lan et al., 2019), have shown great results in generating contextual word- and sentence vector representations. The semantic properties of these vector representations allow the meaning of texts to be encoded in such a way that similar texts are close in vector space.

Although embedding techniques have been used for a variety of tasks, ranging from classification to neural search engines, researchers have started to adopt these powerful contextual representations for topic modeling. Sia et al. (2020) demonstrated the viability of clustering embeddings with centroid-based techniques, compared to conventional methods such as LDA, as a way to represent topics. From these clustered embeddings, topic representations were extracted by embedding words and finding those that are in close proximity to a cluster's centroid. Similarly, Top2Vec leverages Doc2Vec's word- and document representations to learn jointly embedded topic, document, and word vectors (Angelov, 2020; Le and Mikolov, 2014). Comparable to Sia et al. (2020)'s approach, documents are clustered and topic representations are created by finding words close to a cluster's centroid. Interestingly, although the topic representations are generated from a centroid-based perspective, the clusters are generated from a density-based perspective, namely by leveraging HDBSCAN (McInnes and Healy, 2017).

The aforementioned topic modeling techniques assume that words in close proximity to a cluster's centroid are most representative of that cluster, and thereby a topic. In practice, however, a cluster will not always lie within a sphere around a cluster centroid. As such, the assumption cannot hold for every cluster of documents, and the representation of those clusters, and thereby the topic might be misleading. Although (Sia et al., 2020) attempts to overcome this issue by re-ranking topic words based on their frequency in a cluster, the initial candidates are still generated from a centroid-based perspective.

In this paper, we introduce BERTopic, a topic

model that leverages clustering techniques and a class-based variation of TF-IDF to generate coherent topic representations. More specifically, we first create document embeddings using a pre-trained language model to obtain document-level information. Second, we first reduce the dimensionality of document embeddings before creating semantically similar clusters of documents that each represent a distinct topic. Third, to overcome the centroid-based perspective, we develop a class-based version of TF-IDF to extract the topic representation from each topic. These three independent steps allow for a flexible topic model that can be used in a variety of use-cases, such as dynamic topic modeling.

2 Related Work

In recent years, neural topic models have increasingly shown success in leveraging neural networks to improve upon existing topic modeling techniques (Terragni et al., 2021; Cao et al., 2015; Zhao et al., 2021; Larochelle and Lauly, 2012). The incorporation of word embeddings into classical models, such as LDA, demonstrated the viability of using these powerful representations (Liu et al., 2015; Nguyen et al., 2015; Shi et al., 2017; Qiang et al., 2017). Foregoing incorporation into LDA-like models, there has been a recent surge of topic modeling techniques built primarily around embedding models illustrating the potential of embedding-based topic modeling techniques (Bianchi et al., 2020b; Dieng et al., 2020; Thompson and Mimno, 2020). CTM, for example, demonstrates the advantage of relying on pre-trained language models, namely that future improvements in language models may translate into better topic models (Bianchi et al., 2020a).

Several approaches have started simplifying the topic building process by clustering word- and document embeddings (Sia et al., 2020; Angelov, 2020). This clustering approach allows for a flexible topic model as the generation of the clusters can be separated from the process of generating the topic representations.

BERTopic builds on top of the clustering embeddings approach and extends it by incorporating a class-based variant of TF-IDF for creating topic representations.

3 BERTopic

BERTopic generates topic representations through three steps. First, each document is converted to its embedding representation using a pre-trained language model. Then, before clustering these embeddings, the dimensionality of the resulting embeddings is reduced to optimize the clustering process. Lastly, from the clusters of documents, topic representations are extracted using a custom class-based variation of TF-IDF.

3.1 Document embeddings

In BERTopic, we embed documents to create representations in vector space that can be compared semantically. We assume that documents containing the same topic are semantically similar. To perform the embedding step, BERTopic uses the Sentence-BERT (SBERT) framework (Reimers and Gurevych, 2019). This framework allows users to convert sentences and paragraphs to dense vector representations using pre-trained language models. It achieves state-of-the-art performance on various sentence embedding tasks (Reimers and Gurevych, 2020; Thakur et al., 2020).

These embeddings, however, are primarily used to cluster semantically similar documents and not directly used in generating the topics. Any other embedding technique can be used for this purpose if the language model generating the document embeddings was fine-tuned on semantic similarity. As a result, the quality of clustering in BERTopic will increase as new and improved language models are developed. This allows BERTopic to continuously grow with the current state-of-the-art in embedding techniques.

3.2 Document clustering

As data increases in dimensionality, distance to the nearest data point has been shown to approach the distance to the farthest data point (Aggarwal et al., 2001; Beyer et al., 1999). As a result, in high dimensional space, the concept of spatial locality becomes ill-defined and distance measures differ little.

Although clustering approaches exist for overcoming this curse of dimensionality (Pandove et al., 2018; Steinbach et al., 2004), a more straightforward approach is to reduce the dimensionality of embeddings. Although PCA and t-SNE are well-known methods for reducing dimensionality, UMAP has shown to preservers more of the local

and global features of high-dimensional data in lower projected dimensions (McInnes et al., 2018). Moreover, since it has no computational restrictions on embedding dimensions, UMAP can be used across language models with differing dimensional space. Thus, we use UMAP to reduce the dimensionality of document embeddings generated in 3.1 (McInnes et al., 2018).

The reduced embeddings are clustering used HDBSCAN (McInnes et al., 2017). It is an extension of DBSCAN that finds clusters of varying densities by converting DBSCAN into a hierarchical clustering algorithm. HDBSCAN models clusters using a soft-clustering approach allowing noise to be modeled as outliers. This prevents unrelated documents to be assigned to any cluster and is expected to improve topic representations.'Moreover, (Allaoui et al., 2020) demonstrated that reducing high dimensional embeddings with UMAP can improve the performance of well-known clustering algorithms, such as k-Means and HDBSCAN, both in terms of clustering accuracy and time.

3.3 Topic Representation

The topic representations are modeled based on the documents in each cluster where each cluster will be assigned one topic. For each topic, we want to know what makes one topic, based on its cluster-word distribution, different from another? For this purpose, we can modify TF-IDF, a measure for representing the importance of a word to a document, such that it allows for a representation of a term's importance to a topic instead.

The classic TF-IDF procedure combines two statistics, term frequency, and inverse document frequency (Joachims, 1996):

$$W_{t,d} = tf_{t,d} \cdot \log(\frac{N}{df_t}) \quad (1)$$

Where the term frequency models the frequency of term t in document d. The inverse document frequency measures how much information a term provides to a document and is calculated by taking the logarithm of the number of documents in a corpus N divided by the total number of documents that contain t.

We generalize this procedure to clusters of documents. First, we treat all documents in a cluster as a single document by simply concatenating the documents. Then, TF-IDF is adjusted to account for this representation by translating documents to clusters:

$$W_{t,c} = tf_{t,c} \cdot \log(1 + \frac{A}{tf_t}) \quad (2)$$

Where the term frequency models the frequency of term t in a class c or in this instance. Here, the class c is the collection of documents concatenated into a single document for each cluster. Then, the inverse document frequency is replaced by the inverse class frequency to measure how much information a term provides to a class. 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. To output only positive values, we add one to the division within the logarithm.

Thus, this class-based TF-IDF procedure models the importance of words in clusters instead of individual documents. This allows us to generate topic-word distributions for each cluster of documents.

Finally, by iteratively merging the c-TF-IDF representations of the least common topic with its most similar one, we can reduce the number of topics to a user-specified value.

4 Dynamic Topic Modeling

Traditional topic modeling techniques are static in nature and do not allow for sequentially-organized of documents to be modeled. Dynamic topic modeling techniques, first introduced by (Blei and Lafferty, 2006) as an extension of LDA, overcome this by modeling how topics might have evolved over time and the extent to which topic representations reflect that.

In BERTopic, we can model this behavior by leveraging the c-TF-IDF representations of topics. Here, we assume that the temporal nature of topics should not influence the creation of global topics. The same topic might appear across different times, albeit possibly represented differently. As an example, a global topic about cars might contain words such as "car" and "vehicle" regardless of the temporal nature of specific documents. Car-related documents created in 2020, however, might be better represented with words such as "Tesla" and "self-driving" whereas these words would likely not appear in car-related documents created in 1990. Although the same topic is assigned to car-related documents in 1990 and 2020, its representation might differ. Thus, we first generate a global representation of topics, regardless of their temporal nature, before developing a local representation.

To do this, BERTopic is first fitted on the entire corpus as if there were no temporal aspects to the data in order to create a global view of topics. Then, we can create a local representation of each topic by simply multiplying the term frequency of documents at timestep *i* with the pre-calculated global IDF values:

$$W_{t,c,i} = tf_{t,c,i} \cdot \log(1 + \frac{A}{tf_t}) \quad (3)$$

A major advantage of using this technique is that these local representations can be created without the need to embed and cluster documents which allow for fast computation. Moreover, this method can also be used to model topic representations by other meta-data, such as author or journal.

4.1 Smoothing

Although we can observe how topic representations are different from one time to another, the topic representation at timestep *t* is independent of timestep *t-1*. As a result, this dynamic representation of topics might not result in linearly evolving topics. When we expect linearly evolving topics, we assume that a topic representation at timestep *t* depends on the topic representation at timestep *t-1*.

To overcome this, we can leverage the c-TF-IDF matrices that were created at each timestep to incorporate this linear assumption. For each topic and timestep, the c-TF-IDF vector is normalized by dividing the vector with the L1-norm. When comparing vectors, this normalization procedure prevents topic representations from having disproportionate effects as a result of the size of the documents that make up the topic.

Then, for each topic and representation at timestep *t*, we simply take the average of the normalized c-TF-IDF vectors at *t* and *t-1*. This allows us to influence the topic representation at *t* by incorporating the representation at *t-1*. Thus, the resulting topic representations are smoothed based on their temporal position.

It should be noted that although we might expect linearly evolving topics, this is not always the case. Hence, this smoothing technique is optional when using BERTopic and will be reflected in the experimental setup.

5 Experimental Setup

OCTIS (Optimizing and Comparing Topic models is Simple), an open-source python package, was used to run the experiments, validate results, and preprocess the data (Terragni et al., 2021).

Both the implementation of BERTopic as well as the experimental setup are freely available online. [1] [2]

5.1 Datasets

Three datasets were used to validate BERTopic, namely 20 NewsGroups, BBC News, and Trump's tweets. We choose to thoroughly preprocess the 20 NewsGroups and BBC News datasets, and only slightly preprocess Trump's tweets to generate more diversity between datasets.

The 20 NewsGroups dataset[3] contains 16309 news articles across 20 categories (Lang, 1995). The BBC News dataset[4] contains 2225 documents from the BBC News website between 2004 and 2005 (Greene and Cunningham, 2006). Both datasets were retrieved using OCTIS, and preprocessed by removing punctuation, lemmatization, removing stopwords, and removing documents with less than 5 words.

To represent more recent data in a short-text form, we collected all tweets of Trump[5] before and during his presidency. The data contains 44253 tweets, excluding re-tweets, between 2009 and 2021. In both datasets, we lowercased all tokens.

To evaluate BERTopic in a dynamic topic modeling setting, Trump's tweets were selected as they inherently had a temporal nature to them. Additionally, the transcriptions of the United Nations (UN) general debates between 2006 and 2015[6] were analyzed (Baturo et al., 2017). The Trump dataset was binned to 10 timesteps and the UN datasets to 9 timesteps.

5.2 Models

BERTopic will be compared to LDA, NMF, CTM, and Top2Vec. LDA and NMF were run through OCTIS with default parameters. The *"all-mpnet-base-v2"* SBERT model was used as the embedding

[1] https://github.com/MaartenGr/BERTopic

[2] https://github.com/MaartenGr/BERTopic_evaluation

[3] https://github.com/MIND-Lab/OCTIS/tree/master/preprocessed_datasets/20NewsGroup

[4] https://github.com/MIND-Lab/OCTIS/tree/master/preprocessed_datasets/BBC_news

[5] https://www.thetrumparchive.com/faq

[6] https://runestone.academy/runestone/books/published/httlads/_static/un-general-debates.csv

	20 NewsGroups		BBC News		Trump	
	TC	TD	TC	TD	TC	TD
LDA	.058	.749	.014	.577	-.011	.502
NMF	.089	.663	.012	.549	.009	.379
T2V-*MPNET*	.068	.718	-.027	.540	-.213	.698
T2V-*Doc2Vec*	.192	.823	.171	.792	-.169	.658
CTM	.096	.886	.094	.819	.009	.855
BERTopic-*MPNET*	.166	.851	.167	.794	.066	.663

Table 1: Ranging from 10 to 50 topics with steps of 10, topic coherence (TC) and topic diversity (TD) were calculated at each step for each topic model. All results were averaged across 3 runs for each step. Thus, each score is the average of 15 separate runs.

model for BERTopic and CTM (Song et al., 2020). Two variations of Top2Vec were modeled, one with Doc2Vec and one with the *"all-mpnet-base-v2"* SBERT model[7].

For fair comparisons between BERTopic and Top2Vec, the parameters of HDBSCAN and UMAP were fixed between topic models.

To measure the generalizability of BERTopic across language models, four different language models were used in the experiments with BERTopic, namely the Universal Sentence Encoder (Cer et al., 2018), Doc2Vec, and the *"all-MiniLM-L6-v2"* (MiniLM) and *"all-mpnet-base-v2"* (MPNET) SBERT models.

Finally, BERTopic, with and without the assumption of linearly-evolving topics, was compared with the original dynamic topic model, referred hereto as LDA Sequence.

5.3 Evaluation

The performance of the topic models in this paper is reflected by two widely-used metrics, namely topic coherence and topic diversity. For each topic model, its topic coherence was evaluated using normalized pointwise mutual information (NPMI, (Bouma, 2009)). This coherence measure has been shown to emulate human judgment with reasonable performance (Lau et al., 2014). The measure ranges from [-1, 1] where 1 indicates a perfect association. Topic diversity, as defined by (Dieng et al., 2020), is the percentage of unique words for all topics. The measure ranges from [0, 1] where 0 indicates redundant topics and 1 indicates more varied topics.

[7]For an overview of SBERT models and their performance, see https://www.sbert.net/docs/pretrained_models.html

Ranging from 10 to 50 topics with steps of 10, the NPMI score was calculated at each step for each topic model. All results were averaged across 3 runs for each step. To evaluate the dynamic topic models, the NPMI score was calculated at 50 topics for each timestep and then averaged. All results were averaged across 3 runs.

Validation measures such are topic coherence and topic diversity are proxies of what is essentially a subjective evaluation. One user might judge the coherence and diversity of a topic differently from another user. As a result, although these measures can be used to get an indication of a model's performance, they are just that, an indication.

It should be noted that although NPMI has been shown to correlate with human judgment, recent research states that this may only be the case for classical models and that this relationship might not exist with neural topic models (Hoyle et al., 2021). In part, the authors suggest a needs-driven approach to evaluation as topic modeling's primary use is in computer-assisted content analysis.

To this purpose, the differences in running times of each model were explored as they can greatly impact their usability. Here, we choose to focus on the wall times as it more accurately reflects how the topic modeling techniques would be used in practice. All the models are run on a machine with 2 cores of Intel(R) Xeon(R) CPU @ 2.00GHz and a Tesla P100-PCIE-16GB GPU.

Moreover, in Section 7, the strengths and weaknesses of the proposed model across use cases will be discussed extensively to further shed a light on what the model can and cannot do.

6 Results

Our main results can be found in Table 1.

	20 NewsGroups		BBC News		Trump	
	TC	TD	TC	TD	TC	TD
BERTopic-*USE*	.149	.858	.158	.764	.051	.684
BERTopic-*Doc2Vec*	.173	.871	.168	.819	-.088	.536
BERTopic-*MiniLM*	.159	.833	.170	.802	.060	.660
BERTopic-*MPNET*	.166	.851	.167	.792	.066	.663

Table 2: Using four different language models in BERTopic, coherence score (TC) and topic diversity (TD) were calculated ranging from 10 to 50 topics with steps of 10. All results were averaged across 3 runs for each step. Thus, each score is the average of 15 separate runs.

6.1 Performance

From Table 1, we can observe that BERTopic generally has high topic coherence scores across all datasets. It has the highest scores on the slightly preprocessed dataset, Trump's tweets, whilst remaining competitive on the thoroughly preprocessed datassets, 20 NewsGroups and BBC News. Although BERTopic demonstrates competitive topic diversity scores, it is consistently outperformed by CTM. This is consistent with their results indicating high topic diversity, albeit using a different topic diversity measure (Bianchi et al., 2020a).

6.2 Language Models

The results in Table 2 demonstrate the stability of BERTopic, in terms of both topic coherence and topic diversity, across SBERT language models. As a result, the smaller and faster model, *"all-MiniLM-L6-v2"*, might be preferable when limited GPU capacity is available.

Although the USE and Doc2Vec language in BERTopic generally have similar performance, Doc2Vec scores low on the Trump dataset. This is reflected in the results we find in 1 where Top2Vec with Doc2Vec has poor performance. These results suggest that Doc2Vec struggles with creating accurate representations of the Trump dataset.

On topic coherence, Top2Vec with Doc2Vec embeddings shows competitive performance. However, when MPNET embeddings are used both its topic coherence and diversity drop across all datasets suggesting that Top2Vec might not be best suited with embeddings outside of those generated through Doc2Vec. This is not unexpected as both word and documents vectors in Doc2Vec are jointly embedded in the same space, which does not hold for all language models.

In turn, this also suggests why BERTopic remains competitive regardless of the embedding model. By separating the process of embedding documents and constructing the word-topic distribution, BERTopic is flexible in its embedding procedure.

6.3 Dynamic Topic Modeling

From Table 3, we can observe that BERTopic with and without the assumption of linearly evolving topics performs consistently well across both datasets. For Trump, it outperforms LDA on all measures whereas it only achieves the top score on topic coherence for the UN dataset.

On both datasets, there seems to be no effect of the assumption of linearly evolving topics on both topic coherence and topic diversity indicating that from an evaluation perspective, the proposed assumption does not impact performance.

	Trump		UN	
	TC	TD	TC	TD
LDA Sequence	.009	.715	.173	**.820**
BERTopic	**.079**	.862	**.231**	.779
BERTopic-*Evolve*	**.079**	**.863**	.226	.769

Table 3: The topic coherence (TC) and topic diversity (TD) scores were calculated on dynamic topic modeling tasks. The TC and TD scores were calculated for each of the 9 timesteps in each dataset. Then, all results were averaged across 3 runs for each step. Thus, each score represents the average of 27 values.

6.4 Wall time

From the left graph in Figure 1, CTM using a MPNET SBERT model, is quite slow compared to all other models. If we remove that model from the results, we can more easily compare the wall time of the topic models that are more close in speed. Then, We can observe that the classical models, NMF

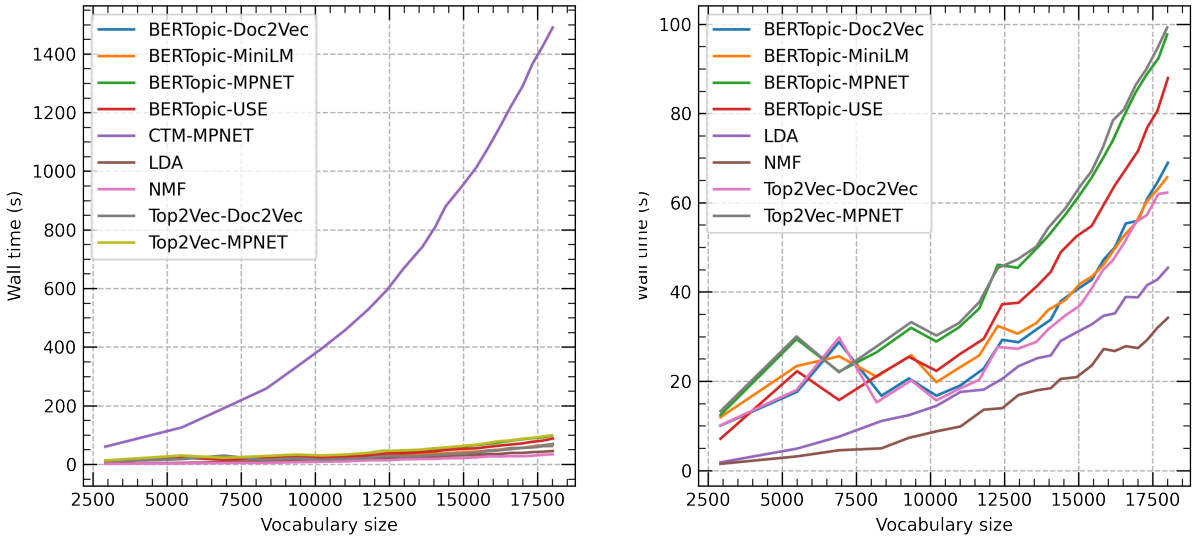

Figure 1: Computation time (wall time) in seconds of each topic model on the Trump dataset. Increasing sizes of vocabularies were regulated through selection of documents ranging from 1000 documents until 43000 documents with steps of 2000. **Left**: computational results with CTM. **Right**: computational results without CTM as it inflates the y-axis making differentiation between other topic models difficult to visualize.

and LDA, are faster than the neural network-based topic modeling techniques. Moreover, BERTopic and Top2Vec are quite similar in wall times if they are using the same language models. Interestingly, the MiniLM SBERT model seems to be similar in speed compared with Doc2Vec indicating that in BERTopic, MiniLM is a good trade-off between speed and performance.

However, it should be noted that in the environment used in this experiment a GPU was available for creating the embeddings. As a result, the wall time is expected to increase significantly when embedding documents without a GPU. Although Doc2Vec can be used as a language model instead, previous experiments in this study have put its stability with respect to topic coherence and topic diversity into question.

7 Discussion

Although we attempted to validate BERTopic across several experiments, topic modeling techniques can be validated through many other evaluation metrics, such as metrics for unsupervised and supervised modeling performance. Moreover, topic modeling techniques can be used across a variety of use cases, many of which are not covered in this study. For those reasons, we additionally discuss the strengths and weaknesses of BERTopic to further describe when, and perhaps most importantly, when not to use BERTopic.

7.1 Strengths

There are several notable strengths of BERTopic compared to the topic models used in this study.

First, the experiments demonstrate that BERTopic remains competitive regardless of the language model used to embed the documents and that performance may increase when leveraging state-of-the-art language models. This indicates its ability to scale performance with new developments in the field of language models whilst still remaining competitive if classical language models are used. Moreover, its stability across language models allows it to be used in a wide range of situations. For example, when a user does not have access to a GPU, Doc2Vec can be used to generate competitive results.

Second, separating the process of embedding documents from representing topics allows for significant flexibility in the usage and fine-tuning of BERTopic. Different preprocessing procedures can be used when embedding the documents and when generating the topic representations. For example, one might want to remove stopwords in the topic representations but not before creating document embeddings. Similarly, once the documents have been clustered, the topic generation process can be fine-tuned, by, for example, increasing the n-gram of words in the topic representation, without the need to re-cluster the data.

Third, by leveraging a class-based version of TF-

IDF, we can represent topics as a distribution of words. These distributions have allowed BERTopic to model the dynamic and evolutionary aspects of topics with little changes to the core algorithm. Similarly, with these distributions, we can also model the representations of topics across classes.

7.2 Weaknesses

No model is perfect and BERTopic is definitely no exception. There are several weaknesses to the model that should be addressed. First, BERTopic assumes that each document only contains a single topic which does not reflect the reality that documents may contain multiple topics. Although documents can be split up into smaller segments, such as sentences and paragraphs, it is not an ideal representation. However, as HDBSCAN is a soft-clustering technique, we can use its probability matrix as a proxy of the distribution of topics in a document. This resolves the issue to some extent but it does not take into account that documents may contain multiple topics during the training of BERTopic.

Second, although BERTopic allows for a contextual representation of documents through its transformer-based language models, the topic representation itself does not directly account for that as they are generated from bags-of-words. The words in a topic representation merely sketch the importance of words in a topic whilst those words are likely to be related. As a result, words in a topic might be similar to one another and can be redundant for the interpretation of the topic. In theory, this could be resolved by applying maximal marginal relevance to the top n words in a topic but it was not explored in this study (Carbonell and Goldstein, 1998).

8 Conclusion

We developed BERTopic, a topic model that extends the cluster embedding approach by leveraging state-of-the-art language models and applying a class-based TF-IDF procedure for generating topic representations. By separating the process of clustering documents and generating topic representations, significant flexibility is introduced in the model allowing for ease of usability.

We present in this paper an in-depth analysis of BERTopic, ranging from evaluation studies with classical topic coherence measures to analyses involving running times. Our experiments suggest that BERTopic learns coherent patterns of language and demonstrates competitive and stable performance across a variety of tasks.

References

Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. 2001. On the surprising behavior of distance metrics in high dimensional space. In *International conference on database theory*, pages 420–434. Springer.

Mebarka Allaoui, Mohammed Lamine Kherfi, and Abdelhakim Cheriet. 2020. Considerably improving clustering algorithms using umap dimensionality reduction technique: A comparative study. In *International Conference on Image and Signal Processing*, pages 317–325. Springer.

Dimo Angelov. 2020. Top2vec: Distributed representations of topics. *arXiv preprint arXiv:2008.09470*.

Alexander Baturo, Niheer Dasandi, and Slava J Mikhaylov. 2017. Understanding state preferences with text as data: Introducing the un general debate corpus. *Research & Politics*, 4(2):2053168017712821.

Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. 1999. When is "nearest neighbor" meaningful? In *International conference on database theory*, pages 217–235. Springer.

Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2020a. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. *arXiv preprint arXiv:2004.03974*.

Federico Bianchi, Silvia Terragni, Dirk Hovy, Debora Nozza, and Elisabetta Fersini. 2020b. Cross-lingual contextualized topic models with zero-shot learning. *arXiv preprint arXiv:2004.07737*.

David M Blei and John D Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40.

Ziqiang Cao, Sujian Li, Yang Liu, Wenjie Li, and Heng Ji. 2015. A novel neural topic model and its supervised extension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2210–2216.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings*

of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, pages 335–336.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Cédric Févotte and Jérôme Idier. 2011. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural computation*, 23(9):2421–2456.

Derek Greene and Pádraig Cunningham. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proc. 23rd International Conference on Machine learning (ICML'06)*, pages 377–384. ACM Press.

Alexander Hoyle, Pranav Goel, Andrew Hian-Cheong, Denis Peskov, Jordan Boyd-Graber, and Philip Resnik. 2021. Is automated topic model evaluation broken? the incoherence of coherence. *Advances in Neural Information Processing Systems*, 34.

Thorsten Joachims. 1996. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Technical report, Carnegie-mellon univ pittsburgh pa dept of computer science.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*, pages 331–339. Elsevier.

Hugo Larochelle and Stanislas Lauly. 2012. A neural autoregressive topic model. *Advances in Neural Information Processing Systems*, 25.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. 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*, pages 530–539.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2015. Topical word embeddings. In *Twenty-ninth AAAI conference on artificial intelligence*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

L. McInnes, J. Healy, and J. Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv e-prints*.

Leland McInnes and John Healy. 2017. Accelerated hierarchical density based clustering. In *Data Mining Workshops (ICDMW), 2017 IEEE International Conference on*, pages 33–42. IEEE.

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, 2(11):205.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. 2018. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861.

Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. 2015. Improving topic models with latent feature word representations. *Transactions of the Association for Computational Linguistics*, 3:299–313.

Divya Pandove, Shivan Goel, and Rinkl Rani. 2018. Systematic review of clustering high-dimensional and large datasets. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(2):1–68.

Jipeng Qiang, Ping Chen, Tong Wang, and Xindong Wu. 2017. Topic modeling over short texts by incorporating word embeddings. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 363–374. Springer.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. *arXiv preprint arXiv:2004.09813*.

Min Shi, Jianxun Liu, Dong Zhou, Mingdong Tang, and Buqing Cao. 2017. We-lda: a word embeddings augmented lda model for web services clustering. In *2017 ieee international conference on web services (icws)*, pages 9–16. IEEE.

Suzanna Sia, Ayush Dalmia, and Sabrina J Mielke. 2020. Tired of topic models? clusters of pretrained word embeddings make for fast and good topics too! *arXiv preprint arXiv:2004.14914*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *arXiv preprint arXiv:2004.09297*.

Michael Steinbach, Levent Ertöz, and Vipin Kumar. 2004. The challenges of clustering high dimensional data. In *New directions in statistical physics*, pages 273–309. Springer.

Silvia Terragni, Elisabetta Fersini, Bruno Giovanni Galuzzi, Pietro Tropeano, and Antonio Candelieri. 2021. Octis: Comparing and optimizing topic models is simple! In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 263–270.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2020. Augmented sbert: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. *arXiv preprint arXiv:2010.08240*.

Laure Thompson and David Mimno. 2020. Topic modeling with contextualized word representation clusters. *arXiv preprint arXiv:2010.12626*.

He Zhao, Dinh Phung, Viet Huynh, Yuan Jin, Lan Du, and Wray Buntine. 2021. Topic modelling meets deep neural networks: A survey. *arXiv preprint arXiv:2103.00498*.