# COHERENCE-BASED DOCUMENT CLUSTERING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Latent Dirichlet Allocation or Non-negative Matrix Factorization are just two widely used algorithms for extracting latent topics from large text corpora. While these algorithms differ in their modeling approach, they have in common that hyperparameter optimization is difficult and is mainly achieved by maximizing the extracted topic coherence scores via grid search. Models using word-document embeddings can automatically detect the number of latent topics, but tend to have problems with smaller datasets and often require pre-trained embedding layers for successful topic extraction. We leverage widely used coherence scores by integrating them into a novel document-level clustering approach using keyword extraction methods. The metric by which most topic extraction methods optimize their hyperparameters is thus optimized during clustering, resulting in *ultra-coherent* clusters. Moreover, unlike traditional methods, the number of extracted topics or clusters does not need to be determined in advance, saving an additional optimization step and a time- and computationally-intensive grid search. Additionally, the number of topics is detected much more accurately than by models leveraging word-document embeddings.

## 1 INTRODUCTION

Unsupervised document clustering for extracting latent topics in large collections of documents has gained increasing importance with the ever growing availability of textual data. Prominent algorithms such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003) or Latent Semantic Analysis (LSA) (Landauer et al., 1998) are specifically designed to extract topics from documents. Non-negative Matrix Factorization (NMF)(Lee & Seung, 1999; Xu et al., 2003) or other general clustering algorithms (e.g. K-means) on the other hand are clustering algorithms that are also applicable to document topic extraction. While all of the mentioned document clustering algorithms can produce reliable results, the underlying assumptions differ quite strongly. LDA (Blei et al., 2003) for example assumes that documents are mixtures of latent topics whereas the Gibbs Sampler Dirichlet Multinomial Model (GSDMM) and the Gamma Poisson Mixture Model (GPM) assume that each document is determined from a single topic (Mazarura & De Waal, 2016).

However, all the above methods have in common that the number of extracted topics must be pre-specified manually. The ideal number of topics is often optimised either via human assessment of the created topics or via a grid search by maximizing the extracted topics coherence scores (Syed & Spruit, 2017; Newman et al., 2010; Mimno et al., 2011) with respect to the algorithms hyperparameters, e.g. the number of extracted topics or distributional parameters as in the LDA (Blei et al., 2003; Wallach et al., 2009). Other measures such as likelihood-based perplexity metrics (if suited for the chosen clustering algorithm) are negatively correlated with measures that are based on human evaluation (Chang et al., 2009) and are thus not well suited for optimising hyperparameters.

Other, non-probabilistic methods that do not need a pre-specification of the number of extracted topics make frequent use of word- and document embeddings (Angelov, 2020; Grootendorst, 2020a) arguing that a dense area of documents can be interpreted as these documents covering the same topic. The number of these dense areas, thus, automatically determines the number of extracted topics. While these algorithms tend to perform very well on extremely large data sets, pre-trained embeddings can be used when analysing smaller data sets. However, in our applications we find severe overestimation of the number of topics and a lack of accurate recognition of all topics present, even with pre-trained embeddings for medium sized data sets ($<$5.000 documents).

In this paper, we reverse a commonly used hyperparmeter optimization process and use coherence scores for document clustering. We show that by subsidizing simple keyword extraction methods and clustering documents based on their respective coherence, both, probabilistic as well as semantic embedding based topic extraction methods are outperformed for medium sized data sets. Coherent clusters are formed and appropriate numbers of topics are found without pre-specification. Additionally, we find that *simpler* keyword-extraction methods outperform transformer based keyword extraction methods as introduced by Grootendorst (2020b).

The remainder of the paper is structured as follows: Section 2 outlines our proposed methodology. First, coherence scores are presented in Section 2.1. Second, the extraction of a documents most defining words is described in section 2.2. Section 2.3 outlines our proposed clustering algorithm. In Section 3, applications of our proposed algorithm are presented and bench-marked with conventional methods. Section 4 provides a conclusion.

## 2 METHODOLOGY

Let V $= \{w_1, \ldots, w_n\}$ be the vocabulary of words and D $= \{d_1, \ldots, d_M\}$ be a corpus, i.e. a collection of documents. Each document is then assumed to be a sequence of words $d_i = [w_{i1}, \ldots, w_{in_i}]$ where $w_{ij} \in V$ and $n_i$ denotes the length of document $d_i$. There are different interpretations of topics, but mostly a topic $t_k$ from a set of topics $T = \{t_1, \ldots, t_K\}$ is interpreted to be a distribution over words (Blei et al., 2003; Mazarura & De Waal, 2016). LDA, as well as NMF additionally assume that documents are random mixtures over latent topics.

Unlike LDA, we do not assume that documents are mixtures of topics. Rather, we argue that a document itself best describes a single topic. That is, a document deals with only a single topic, while a single topic can be covered by multiple documents. A document that covers different content, by combining these topics into a single document, ensures that it is indeed a single topic with different sub-contents. We follow this reasoning because the very existence of a document that addresses *multiple* topics makes it clear that these *multiple* topics are closely related by their common appearance in a single document. We thus take a clustering approach and assume a cluster to be a collection of documents or even a single document covering similar contents and make use of LDA's most used hyperparameter optimization metric, the coherence score. Subsequently, a topic is merely a set of words best describing the documents of a cluster. A cluster is thus best described as a set of documents and hence as a set of all words contained in the documents, $\gamma_k = \{d_1 = [w_{11}, \ldots, w_{1n_1}], d_2 = [w_{21}, \ldots, w_{2n_2}], \ldots, d_M = [w_{M1}, \ldots, w_{Mn_M}]\}$. Note the difference between a cluster, $\gamma_k$, being a set of documents and a topic, $t_k$, being a distribution over words. The order of the documents or words is not important and the words describing the cluster best, can be found with e.g. term frequency-inverse document frequency (TF-IDF) document representations (Salton & Buckley, 1988) or other keyword extraction methods and thus represent the clusters topic $t_k$.

### 2.1 COHERENCE SCORES

Coherence scores are often used to evaluate the quality of artificially created topics from automated topic modelling. A generated topic is said to be *better* when the topic is more coherent. Coherence, the property of being logical and consistent, is defined in terms of topics as a quality measure that evaluates a topic by the extent to which it consists of words that frequently occur together in the observed documents. A topic, consisting of words or phrases that do not often co-occur within documents[1] would thus be evaluated as a *bad* topic. Interpreting a topic merely as a set of words, a common coherence measure for a topic $t_k$ is given below:

$$C(t_k) = \sum_{j=2}^{J_{max}} \sum_{i=1}^{j-1} \log\Big(\frac{p(w_j, w_i|t_k) + \epsilon}{p(w_i|t_k)}\Big), \tag{1}$$

where $p(w_j, w_i|t_k)$ denotes the probability that the words $w_j$ and $w_i$ co-occur within a document, calculated as the number of documents containing both words divided by the total number of docu-

[1] e.g. *"Deep Learning" and "World War 2", or "Nuclear Power Plant" and "Tennis"*

ments for a given topic $t_k$. $p(w_i|t_k)$ denotes the probability that word $w_i$ occurs in a document and is hence calculated as the number of documents containing word $w_i$ divided by the total number of documents. $\epsilon$ is a regularisation parameter to ensure that $C(t_k) \in \mathbb{R}$ and is often specified as $\frac{1}{M}$. For topic model evaluation these probabilities are dependent on the topic $t_k$, i.e. $p(w_j, w_i|t_k), p(w_i|t_k)$, with $J_{max}$ being set to a fixed number that indicates how many of the topics most probable words are taken into account.

A coherence measure can thus be adapted to the document-level, by simply replacing the sum that is taken over the words contained in a topic by the words contained in that document. A documents coherence is hence evaluated by analyzing the co-occurence of the words contained in that single document, in comparison to the co-occurence of these words in all other documents.

$$C(d_s) = \sum_{j=2}^{n_s} \sum_{i=1}^{j-1} \log\Big(\frac{p(w_j, w_i) + \epsilon}{p(w_i)}\Big) \tag{2}$$

for $w_j \neq w_i$, with $n_s$ being the length of the document. If we adapt the coherence score metric to single documents, the probabilities are hence no longer dependent on a topic $t_k$. Since documents may contain a single word more than once, the score is restricted for the case where $w_i = w_j$. A coherence measure between two documents, $d_1$ and $d_2$ or more precisely a coherence score for document $d_1$, given document $d_2$, can thus be calculated as

$$C(d_1|d_2) = \sum_{j=1}^{n_1} \sum_{i=1}^{n_2} \log\Big(\frac{p(w_j, w_i) + \epsilon}{p(w_i)}\Big) \tag{3}$$

with $p(w_j, w_i) = 1$ if $w_j = w_i$. We found that using $\frac{1}{M^2}$ as $\epsilon$ works best. Hence, $n_1$ denotes the length (number of words) of document $d_1$ and $n_2$ denotes the length of document $d_2$. Since $w_j$ and $w_i$ are words from two different documents, we set $p(w_j, w_i) = 1$ for the case where the words are identical, since documents that have exactly the same wording are obviously very coherent. Note, that the notation $C(d_1|d_2)$ implies a coherence measure for document $d_1$ given document $d_2$. Thus, computing the coherence scores for all document combinations in a corpus and storing them in a matrix, would lead to a $M \times M$ coherence score matrix, $\mathbf{\Phi}$, where the entry $\mathbf{\Phi}_{1,2} = C(d_1|d_2)$ would be the coherence score for document 1 given document 2.

Note, however, that it can be seen in formula (3) that the coherence of $d_1$ given $d_2$ is not necessarily the same as the coherence of $d_2$ given $d_1$, $C(d_1|d_2) \neq C(d_2|d_1)$ since

$$\sum_{j=1}^{J} \sum_{i=1}^{I} \log\Big(\frac{p(w_j, w_i) + \epsilon}{p(w_i)}\Big) \neq \sum_{i=1}^{I} \sum_{j=1}^{J} \log\Big(\frac{p(w_i, w_j) + \epsilon}{p(w_j)}\Big) \tag{4}$$

with $p(w_j, w_i) = 1$ if $w_j = w_i$. Thus, $\mathbf{\Phi}$ is not symmetrical.

### 2.2 MOST DEFINING WORDS

The lengths of the documents have a big impact on the coherence score for the documents, as we take the sums over all words of both documents. Longer documents would thus lead to larger coherence scores. To circumvent this, we adapt the formula by not summing over all word-to-word combinations in both documents, but similar to the topic coherence by summing over the documents $k$ *most defining* words. The documents *most defining* words could easily be extracted by using e.g. the TF-IDF document representation (Salton & Buckley, 1988), formally denoted by:

$$\text{tf-idf}(w) = frequency(w) \cdot [\log \frac{k}{K(w)} + 1], \tag{5}$$

where $k$ is the total number of words in the dictionary, $K$(w) is the total number of documents wherein the word appears and $frequency(w)$ is self explanatory the words frequency in the document. The words with the largest TF-IDF scores can thus be interpreted as the *most defining* words of a document. We compute the representations with Scikit-learns (Pedregosa et al., 2011) built-in

TF-IDF vectorizer. The two sums in formula (3) thus are only built upon a pre-specified number of keywords, e.g. 20. Note, however, that the probabilities $p(w_j, w_i)$, $p(w_i)$ are still taken over all words in all documents.

As a second keyword extraction method we use Transformer based document embeddings (Vaswani et al., 2017). As described by Grootendorst (2020b), we subsidize a pre-trained BERT Model (Devlin et al., 2018; Reimers & Gurevych, 2019) to create these document embeddings. The keywords are extracted by computing the cosine similarity between the single word vectors and the complete document vector. The similarity between word $w$ and document $d$ would thus be

$$similarity(w, d) = \frac{w \times d}{||w|| \times ||d||}, \tag{6}$$

where

$$w \times d = \sum_{i=1}^{nn} w_i \cdot d_i$$

and

$$||w|| \times ||d|| = \sqrt{\sum_{i=1}^{nn} (w_i)^2} \cdot \sqrt{\sum_{i=1}^{nn} (d_i)^2}.$$

$nn$ denotes the vectors dimension in the feature space and is identical for $d$ and $w$. As Grootendorst (2020b) points out, the words best representing the document would have a smaller distance to the document in the embedding space.

### 2.3 CLUSTERING

To create coherent clusters from a corpus, we make use of the documents coherence scores (3) following a simple clustering rule: First, we assume each document to be an independent cluster, where document $d_i$ is denoted as cluster $\gamma_i$. Clusters are being combined into larger clusters, by combining clusters that are the most coherent to one another being clustered together. As such, if $C(\gamma_1|\gamma_2)$ is the maximum coherence score for $\gamma_1$ given all other possible $\gamma$'s, the clusters $\gamma_1$ and $\gamma_2$ would simply be combined in the form of:

$$\gamma_k = \gamma_1 \cup \gamma_2. \tag{7}$$

$\gamma_k$ denotes here a newly formed cluster that is the union of the clusters $\gamma_1$ and $\gamma_2$. A cluster is subsequently a combination of documents and can be interpreted in the same way as a single document. $\gamma_k$ can consequently be expressed as $\gamma_k = \{\gamma_1, \gamma_2\}$.

As described in formula (4), $C(\gamma_1|\gamma_2) \neq C(\gamma_2|\gamma_1)$. Hence, it could happen that e.g. $\gamma_1$ has a maximum coherence score given $\gamma_2$, but $\gamma_2$ has a largest coherence score given $\gamma_3$ or even multiple maximum coherence scores given multiple different clusters. If supposedly $C(\gamma_2|\gamma_3) > C(\gamma_2|\gamma_1)$, the formed cluster $\gamma_k$ would be of the form:

$$\gamma_k = \gamma_1 \cup \gamma_2 \cup \gamma_3.$$

For this cluster to be *complete*, it would require that no other $\gamma_i$ has a maximum coherence score given any cluster included in $\gamma_k$ and $\gamma_3$ needs to have a maximum coherence score given either $\gamma_1$ or $\gamma_2$. We call this asymmetrical appearance of maximum values the clusters chain size, $\delta$. It becomes clear that a clusters chain size, $\delta$, is only smaller than the total number of clusters, if at any point in the chain, a cluster $\gamma_i$ has a maximum coherence score given a cluster *further back* in the chain, $\gamma_j$.

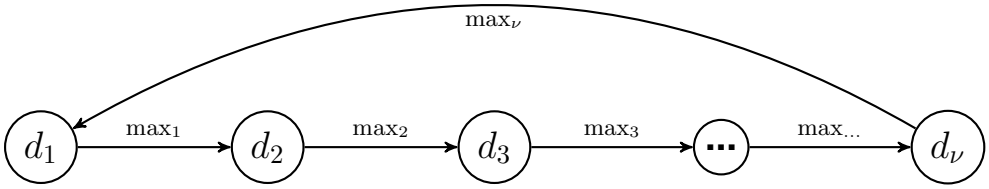


Figure 1: Cluster building for a given maximum chain size $\nu$

Thus, we form clusters based on these maximum coherence value chains, were we limit the cluster size to an arbitrary number, $\nu$. Only clusters are being clustered together, when $\delta \leq \nu$. A limit to this chain size is necessary, as otherwise a cluster that shares no words at all with all other clusters and thus has the same coherence scores given all other clusters could end up forming a *super* cluster. By subsidizing the coherence scores, we make inherently sure that only coherent clusters are formed. Thus, even single documents could end up forming a complete cluster when they do not share any words with all other clusters. Additionally, no pre-specification of topic distributions (Wallach et al., 2009) is necessary and the cluster sizes are detected automatically.

As we form our document clusters in an iterative manner, it is not required to pre-specify an exact number of desired output clusters. However, a pre-specification of a maximum number of formed clusters is advised, as the clustering rule stops clustering, when this number, $\eta$, is reached. Hence, pre-specifying an $\eta$ of e.g. 20 could lead to any number of clusters between 1 and 19, depending on the coherence between the clusters. If this number is not specified, we would either end up with a number of clusters larger than $\nu$, when the clusters all have the same coherence score towards another, or form very few or even a single cluster. Thus, in contrast to LDA or NMF, a researcher does not need to either know the exact number of topics in advance or iterate over every possible number of topics, maximizing after coherence scores.

The presented clustering rule and subsequent topic extraction is used as in the form below:

1. initialize $\nu$ as the maximum chain size
   initialize $\eta$ as the maximum number of extracted clusters
   initialize each document $d_i$ as a cluster $\gamma_i$

2. While number of clusters $> \eta$

   - Compute *most defining words* for clusters
   - Compute coherence score matrix $\mathbf{\Phi}$
   - Combine clusters, $\gamma_a, \gamma_b$ where $\mathbf{\Phi}_{\gamma_a,\gamma_b} \geq \mathbf{\Phi}_{\gamma_a,\gamma_j}$ for all $\gamma_j \neq \gamma_b$ and $\delta \leq \nu$

3. Compute the resulting clusters *most defining words*

## 3 Application and Comparison

For testing we use three different, independent, labelled data sets and compare the presented clustering approach with two of the most frequently used topic modelling approaches, the LDA (Blei et al., 2003) and the NMF (Xu et al., 2003; Lee & Seung, 1999) as well as two state of the art document embedding leveraging topic models, BERTopic (Grootendorst, 2020a) and Top2Vec (Angelov, 2020). For The LDA as well as the NMF we use the gensim implementation (Rehurek & Sojka, 2011) and optimize the number of extracted topics / clusters with the U-mass coherence score, iterating over a sensible range of topics. For text preprocessing we follow the usual techniques and remove stopwords, numbers and symbols and lemmatize the documents. For both algorithms, the LDA and NMF we use the bag-of-words feature representations as the simplest and fastest form of feature extraction. For BERTopic and Top2Vec, no preprocessing is required as is no pre-specification of hyperparameters. For even more accurate testing we use both, pre-trained embedding layers[2] as well as training from scratch. For the presented clustering rule we set the number of extracted keywords to 20.

The three used data sets as well as their topic prevalences can be seen in Table 1.

[2] *universal-sentence-encoder* (Cer et al., 2018) for Top2Vec and *paraphrase-MiniLM-L6-v2* (Reimers & Gurevych, 2019) for BERTopic

Table 1: Used Data Sets and their Topic Distributions

| | Prevalence | | |
|---|---|---|---|
| Topics | Data Set **A** | Data Set **B** | Data Set **C** |
| Business | 0.23 | | 0.1 |
| Politics | 0.19 | | 0.1 |
| Entertainment | 0.17 | | 0.1 |
| Sports | 0.23 | | 0.1 |
| Technology | 0.18 | | 0.1 |
| Graphics | | | 0.1 |
| Food | | | 0.1 |
| Space | | | 0.1 |
| Medicine | | | 0.1 |
| History | | 0.16 | 0.1 |
| Physics | | 0.25 | |
| Computer Science | | 0.20 | |
| Biology | | 0.20 | |
| Maths | | 0.07 | |
| Geography | | 0.03 | |
| Accounts | | 0.09 | |

All algorithms are tested on all three data sets and either the optimal number of topics after computing coherence scores or the respective algorithms optimal solution is analysed. We find that both NMF and LDA are close to the true number of topics for all three data sets but create completely meaningless topics. While BERTopic and Top2Vec create very meaningful topics, they severely overestimate the number of true topics in nearly all cases. Only for a perfectly balanced data set BERTopic and Top2Vec accurately extract the true number as well as the true meaning of the topics. The presented clustering rule, abbreviated as CBC in the following, very accurately detects the true number of topics, especially for TF-IDF keywords, generates meaningful and human-interpretable clusters, and is the only algorithm that can handle severe imbalancedness in a data set. The true number of topics as well as the algorithms extracted number of topics can be seen in Table 2.

Table 2: Data Set Comparison For All Used Methods. The presented method is abbreviated to CBC. For LDA and NMF Coherence scores are used for finding the optimal number of topics.

| | Data Set **A** | Data Set **B** | Data Set **C** |
|---|---|---|---|
| Original number of topics | **5** | **7** | **10** |
| LDA | 2 | 9 | 3 |
| NMF | 2 | 5 | 7 |
| Top2Vec | 29 | 22 | 13 |
| BERTopic | 44 | 45 | 15 |
| Top2Vec pre-trained | 10 | 19 | 13 |
| BERTopic pre-trained | 40 | 45 | 15 |
| CBC - TFIDF | 5 | 8 | 9 |
| CBC - BERT Keywords | 4 | 7 | 14 |

### 3.1 Data Set **A**: BBC Data

First, a BBC data set covering five topics, namely *business, entertainment, politics, sports* and *technology* is analyzed. The categories are fairly balanced across the data set which is comprised of 2.225 documents. We initialize $\nu$ as 15 and $\eta$ as 80. The TF-IDF based rule[3] extracts 5 clusters as can be seen in Figure 2 . The four topics *business, politics, technology* and *sports* are clearly identified, whereas *entertainment*, the least prevalent category, is merged within the *sports* cluster.

[3]See appendix for the results using the BERT keywords (6)

Interesting is also the distinction between Russian and American *business* as seen in the wordclouds #1 and #2.


Figure 2: Cluster extraction for data set **A**, covering 5 topics: *business, entertainment, politics, sports, tech*. $\nu$ = 15, $\eta$= 80. 5 clusters are extracted.

In Comparison, both the NMF and LDA have the largest coherence score for a number of two topics. Whereas the LDA uncovers the topics *sport* and *politics*, the NMF generates two nearly identical topics (see Figure 3).


(a) Extracted Topics from LDA

(b) Extracted Topics from NMF

Figure 3: 2 Extracted Topics from LDA and NMF for data set **A**

BERTopic as well Top2Vec overestimate the number of true topics severely, whereas Top2Vec with pre-trained embeddings is closest to the truth with 10 extracted topics (see wordclouds in the appendix, (7)). However, Top2Vec fails completely to recognize a humanly interpretable topic of business which is the most prevalent topic in the data set. Instead, several very similar topics describing *sports* and *technology* are extracted.

### 3.2 Data Set **B**

Second, we analyze a data set covering 7 topics, namely *computer science, physics, math, biology, geography, history* and *accounting*. The categories prevalence in this data set is fairly unbalanced with the least prevalent category, *geography* only making up 3% of the total amount of documents (3.035).

We initialize the maximum chain size $\nu$ as 15 and the maximum number of clusters $\eta$ as 80. The clustering rule clusters the documents to 8 clusters for the TF-IDF keyword extraction. The words best describing the clusters and thus representing the topics can be seen in Figure 4. CBC subsidizing BERT keywords extracts very accurately 7 topics, but fails to identify all present topics (see Appendix, Figure (10)).

Figure 4: Cluster extraction for data set **B** covering 7 topics: *computer science, physics, math, biology, geography, history, accounting*. $\nu = 15$, $\eta = 80$. 8 clusters are extracted.

Except for the least prevalent category, *geography*, all categories are identified, with the category *accounts* being clustered into three separate topics (wordclouds 3 - 5 in Figure 4). In Comparison, neither the LDA nor the NMF identify a *geography* topic, whereas the LDA identifies 9 topics and the NMF 5 topics (see Appendix). Neither BERTopic or Top2Vec either with or without pre-trained embeddings are close to the true number of topics and again severely overestimate the true number of topics also failing to extract a cluster covering the topic of *geography*.

### 3.3 Data Set **C**

Last, a data set covering 10 topics, namely *graphics, technology, space, sport, business, politics, medicine, history, entertainment* and *food* is analyzed. This data set is perfectly balanced with all categories making up 10% of all 1.000 documents. We initialize the maximum chain size $\nu$ as 15 and the maximum number of clusters $\eta$ as 80. Our Clustering approach extracts 9 clusters using the TF-IDF keywords which are represented in Figure 5.

Figure 5: Cluster extraction for data set **C** covering 10 topics: *graphics, technology, space, sport, business, politics, medicine, history, entertainment, food*. $\nu = 15$, $\eta = 80$. 9 clusters are extracted

Most of the initial 10 topics can be clearly identified and are easily distinguishable from one another. Only the *medical* topic is not clearly identifiable. In comparison, after optimizing with coherence scores, the LDA identified 3 topics and the NMF 7 topics. Interestingly, neither identifies any form of cluster, related to *food* (see figures in Appendix). This perfectly balanced data set is the only data

set for which both BERTopic as well as Top2Vec accurately recognize the true number of topics and precisely extract humanly interpretable topics representing the true categories covered in the data set.

## 4 CONCLUSION

The proposed clustering rule accurately detects the inherent topics in the dataset and outperforms LDA and NMF in terms of human-interpretable topics. Even BERTopic and Top2Vec are outperformed in terms of finding the perfect number of topics as well as recognizing all topics prevalent in the data set. Interestingly, we find that TF-IDF keyword extraction outperforms Transformer based keyword extraction (Grootendorst, 2020b). Through the simple optimization process of subsidizing the documents coherence scores, we successfully create *ultra-coherent* clusters and thus *ultra-coherent* topics. As Rosner et al. (2014) found that the U-mass coherence score can even be outperformed by other, similar scores, an extension of the method using different types of coherence scores promises additional insights into creating even more coherent and humanly interpretable clusters.

## REFERENCES


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

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

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

Jonathan Chang, Sean Gerrish, Chong Wang, Jordan L Boyd-Graber, and David M Blei. Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems*, pp. 288–296, 2009.

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

Maarten Grootendorst. Bertopic: Leveraging bert and c-tf-idf to create easily interpretable topics., 2020a. URL https://doi.org/10.5281/zenodo.4381785.

Maarten Grootendorst. Keybert: Minimal keyword extraction with bert., 2020b. URL https://doi.org/10.5281/zenodo.4461265.

Thomas K Landauer, Peter W Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse processes*, 25(2-3):259–284, 1998.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

J. Mazarura and A. De Waal. A comparison of the performance of latent dirichlet allocation and the dirichlet multinomial mixture model on short text. *In 2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech)*, pp. 1–6, 2016.

David Mimno, Hanna Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pp. 262–272, 2011.

David Newman, Jey Han Lau, Karl Grieser, and Timothy 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*, pp. 100–108, 2010.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Radim Rehurek and Petr Sojka. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2), 2011.

Nils Reimers and Iryna Gurevych. 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, 11 2019. URL http://arxiv.org/abs/1908.10084.

Frank Rosner, Alexander Hinneburg, Michael Röder, Martin Nettling, and Andreas Both. Evaluating topic coherence measures. *arXiv preprint arXiv:1403.6397*, 2014.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

Shaheen Syed and Marco Spruit. Full-text or abstract? examining topic coherence scores using latent dirichlet allocation. In *2017 IEEE International conference on data science and advanced analytics (DSAA)*, pp. 165–174. IEEE, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Hanna M Wallach, David M Mimno, and Andrew McCallum. Rethinking lda: Why priors matter. In *Advances in neural information processing systems*, pp. 1973–1981, 2009.

Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 267–273, 2003.

# A Appendix

## A.1 Results for data set **A**

The cluster extraction of CBC for data set **A** is here presented subsidizing the BERT-keywords. Similar to the TF-IDF extraction, $\nu$ and $\eta$ are set to 15 and 80 respectively. The topics *politics* and *technology* are visibly detected whereas the topic *sports* seems to be included in two topics (#1, #3).


Figure 6: Cluster extraction for data set **A** covering 5 topics: *business, entertainment, politics, sports, tech*. $\nu = 15$, $\eta = 15$. 4 clusters are extracted.

Due to these unclear distinctions between the topics we favour the TF-IDF based clustering rule. Top2Vec topic extraction using pre-trained embeddings comes with 10 identified topics fairly close to the true number of 5 topics. However, the most prevalent topic, *business* is not identified at all. There are 4 very clearly identifiable *sports* topics, even distinguishing between tennis, Olympia and football.

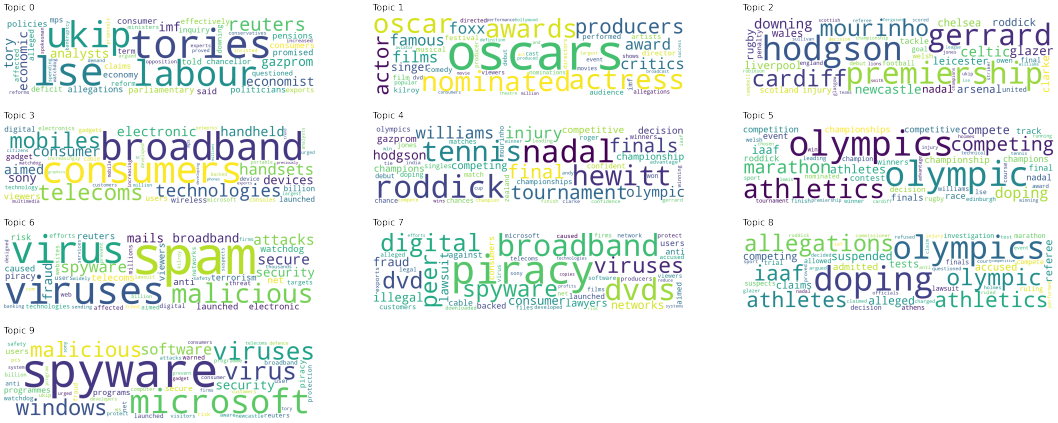


Figure 7: Top2Vec topic extraction for data set **A** covering 5 topics: *business, entertainment, politics, sports, tech*.

### A.2 RESULTS FOR DATA SET **B**

For the second data set covering 7 topics, LDA extracts 9 topics (Figure 8). Topics as *accounts* (#2, #8) or *biology* (#5) are clearly distinguishable.

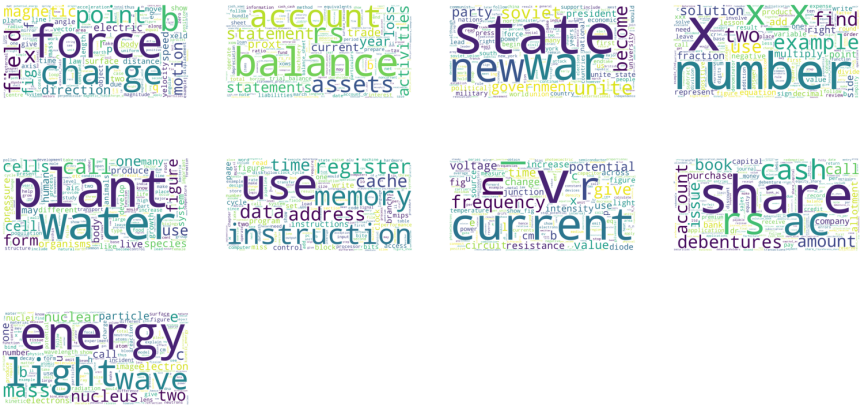


Figure 8: LDA topic modeling for data set **B**

The NMF topic model extracts 5 topics for the 7 topic data set. The topics *geography* as well as *accounts* and *biology* are not distinguishable at all. *History* seems to be split up into two topics (#2 and #5).

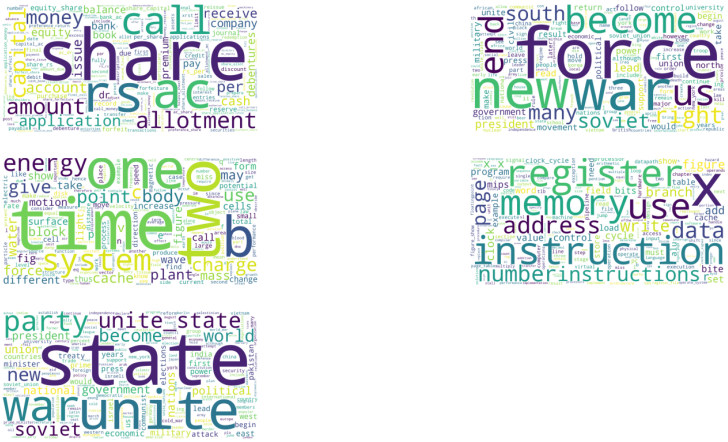


Figure 9: NMF topic modeling for data set **B**

CBC in combination with BERT-keywords extracts accurately 7 topics for the 7 topic data set. The topics *physics* (#7) and *history* (#2) are clearly distinguishable. The topic *accounts* is visibly mixed into multiple clusters.


Figure 10: Cluster extraction for data set **B** covering 7 topics: *computer science, physics, math, biology, geography, history, accounting*. $\nu = 15$, $\eta = 80$. 8 clusters are extracted.

### A.3 Results for data set **C**

LDA topic modelling extracts three topics for the last, 10 topic data set. The three extracted topics seem to cover *history*, *space* and *technology*.

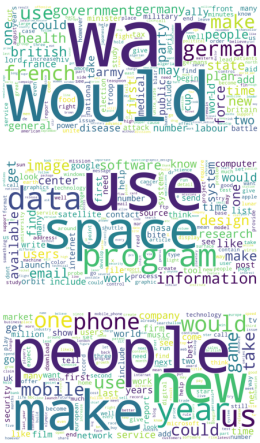


Figure 11: LDA topic modeling for data set **C**

NMF topic modelling extracts 7 topics. *Medicine* is visibly split up into two topics (#4 and #6). *Space, history* and *image* are visibly distinguishable.


Figure 12: NMF topic modeling for data set **C**

CBC subsidizing BERT-keywords extracts 14 clusters. However. setting $\eta$ to 80 as done previously would result in 76 clusters. Pre-specifying $\eta$ anywhere between 15 and 75 results in 14 clusters and specifying $\eta < 15$ would result in 2 clusters.

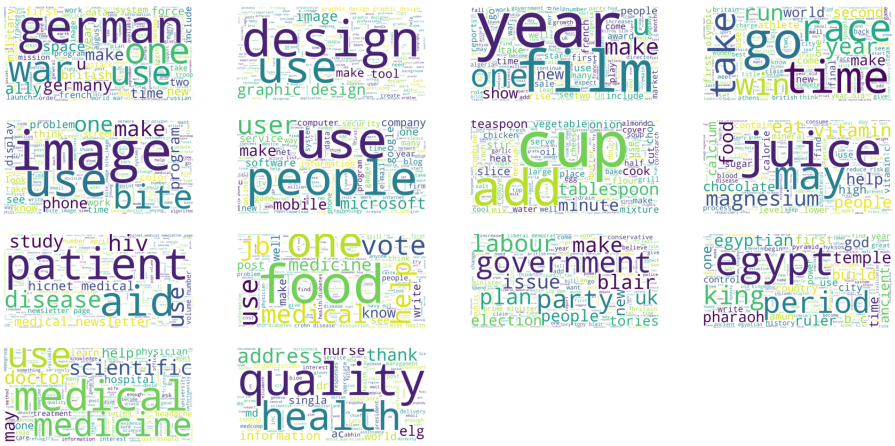


Figure 13: Cluster extraction for data set **C** covering 10 topics: *graphics, technology, space, sport, business, politics, medicine, history, entertainment, food.* $\nu$ = 15, $\eta$ = 15. 14 clusters are extracted