manuscript No.
(will be inserted by the editor)

LDAPrototype: A Model Selection Algorithm to Improve Reliability of Latent Dirichlet Allocation

Jonas Rieger · Carsten Jentsch · Jörg Rahnenführer

the date of receipt and acceptance should be inserted later

Abstract For organizing large text corpora topic modeling provides useful tools. A widely used method is Latent Dirichlet Allocation (LDA), a generative probabilistic model which models single texts in a collection of texts as mixtures of latent topics. In this approach, each topic is characterized by its word distribution, which is obtained by assigning words to topics. This assignment usually relies on initial values such that the LDA outcome is to some extent random and not fully reproducible leading to different results in replicated runs on the same text data. This *instability* of the LDA approach is often neglected in everyday practice, where text data analysis is commonly based on only a single LDA run.

In this paper, we propose a new method called LDAPrototype to select the most representative run from a set of replicated LDA runs applied on the same data set. This prototype is the run with highest average similarity to all other runs. By this, we improve the reliability of conclusions drawn from LDA results, since replications of LDAPrototype are more similar to each other than replications of single LDA runs. To measure similarities of LDA runs, we propose the new tailored similarity measure S-CLOP (**S**imilarity of multiple sets by **C**lustering with **LO**cal **P**runing), which is based on a new pruning algorithm for hierarchical clustering results. To quantify topic similarities, we recommend a thresholded version of the Jaccard coefficient and compare it to other potential choices of similarity measures.

The method LDAPrototype method is illustrated by application to six real datasets consisting of newspaper articles or tweets. Our results show that the reproducibility of LDA results using LDAPrototype increases, and so does the reliability of empirical findings based on topic modeling. Overall, the new approach is justified in view of its application, comprehensible, easy to implement, computationally feasible, and can also be applied to other topic modeling procedures with topics characterized by word distributions.

Keywords topic model · stability · similarity · medoid · clustering · pruning · replications

Jonas Rieger ORCiD: 0000-0002-0007-4478
Department of Statistics, TU Dortmund University, 44221 Dortmund, Germany
Tel.: +49 231-755 3127
E-mail: rieger@statistik.tu-dortmund.de

Carsten Jentsch ORCiD: 0000-0001-7824-1697 · Jörg Rahnenführer ORCiD: 0000-0002-8947-440X
Department of Statistics, TU Dortmund University, 44221 Dortmund, Germany

1 Introduction

Understanding unstructured data, e.g. texts, is a big challenge, especially due to the complicated and not always standardized structure and the increasingly large volumes of text data. Text data is one of the most frequent data type in the world today, and text mining tools have become very popular to analyze such data.

Text data are often analyzed using so called probabilistic topic models (Blei, 2012) and the Latent Dirichlet Allocation (Blei et al., 2003) in particular. In the context of probabilistic topic models, the Latent Dirichlet Allocation (LDA) can be described as a successor to the Probabilistic Latent Semantic Indexing (Hofmann, 1999) and a predecessor to the Correlated Topic Models (Blei and Lafferty, 2007). The LDA model has been extended by numerous features in the past years. Many of the implementations aim at the integration of meta variables such as author names in the Author-Topic Model (Rosen-Zvi et al., 2004). Other models have attempted to integrate the temporal component of the modeled texts, e.g. the Continuous Time Dynamic Topic Models (Wang et al., 2008). In addition to the further development of the LDA, the Structural Topic Model (Roberts et al., 2013) has proven to be another reliable model for the analysis of text corpora. Nevertheless, LDA still enjoys the highest popularity due to its simple implementation, flexible assumptions, and usually not significantly worse results compared to the more complex methods.

In this paper, we propose the method LDAPrototype to improve the interpretation of LDA results. In several use cases we show that single models are prone to misinterpretation because of large differences in the results of LDA runs. To do this, we define a similarity measure for LDA models, which we call S-CLOP. This measure can be used to determine pairwise similarities of LDA models. To overcome the mentioned issue of reproducibility, the method LDAPrototype selects the most representative run out of a set of LDA runs according to the novel S-CLOP criterion. This criterion is not based on likelihood-based measures as it is often the case. Instead, our goal is to determine the medoid of the models. In particular, we want to increase the reliability of the results. We understand reliability as a measure to quantify the reproducibility of the results, i.e. a very reliable method should produce results that are as reproducible as possible. We refer to the medoid as the model that agrees most on average with all other models from the same set of models. We call this model the prototype. It is most similar to all other runs taken from the same modeling procedure.

A few approaches exist to make LDA results more reliable, but all of them have weaknesses, which are discussed in detail in Section 2.2. Often the modeling procedure itself is influenced in a way such that LDA loses its flexibility. Other methods do not search in the whole space of possible models, which leads to non-optimal results. To address these weaknesses, our approach does not affect the modeling procedure and is exclusively based on replicated runs.

We do not use likelihood-based measures because there is already good comparative work in this field (e.g. Griffiths and Steyvers, 2004) and Chang et al. (2009) showed that these measures do not correlate well with the human perception of consistent topics. However, the quality of the results is not in the scope of this paper. Subsequent studies with human judgments are needed for this. Instead, the main goal is to address the instability of the LDA through a model selection criterion that significantly increases the reliability of the results. High reliability is the basis and prerequisite to obtain good results in terms of quality. Thus, we do not follow the classical information retrieval approach, but focus first on reliable results in order to finally obtain topics that can be interpreted particularly well. We intentionally do not speak of an improvement in stability because the method LDA is

not changed, and thus is just as stable as before. Our proposed selection algorithm based on replicated runs to obtain the moist representative LDA run, on the contrary, provides an improvement of the reliability.

2 Related Work

Text data are usually organized in large corpora, where each corpus consists of a collection of texts, often also denoted as documents or articles. Each text can be considered as a sequence of tokens of words of the same length as the given text. In common notation token means an individual word at a specific place in the text and the set of words is used synonymously with vocabulary. We refer to these terms in the following to introduce the methodology of LDA as basis for our novel method LDAPrototype.

2.1 Latent Dirichlet Allocation

The method we propose is based on the LDA (Blei et al., 2003) estimated by a Collapsed Gibbs Sampler (Griffiths and Steyvers, 2004). The LDA assumes distributions of latent topics for each text. If K denotes the total number of modeled topics, the set of topics is given by $\boldsymbol{T} = \{T_1, \ldots, T_K\}$. We define $W_n^{(m)}$ as a single token at position n in text m. The set of possible tokens is given by the vocabulary $\boldsymbol{W} = \{W_1, \ldots, W_V\}$ with $V = |\boldsymbol{W}|$, the vocabulary size. Then, let

$$\boldsymbol{D}^{(m)} = \left(W_1^{(m)}, \ldots, W_{N^{(m)}}^{(m)}\right),\ m = 1, \ldots, M, \qquad W_n^{(m)} \in \boldsymbol{W},\ n = 1, \ldots, N^{(m)}$$

be text (or document) m of a corpus consisting of M texts, each text of length $N^{(m)}$. Topics are referred to as $T_n^{(m)}$ for the topic assignment of token $W_n^{(m)}$. Then, analogously the topic assignments of every text m are given by

$$\boldsymbol{T}^{(m)} = \left(T_1^{(m)}, \ldots, T_{N^{(m)}}^{(m)}\right),\ m = 1, \ldots, M, \qquad T_n^{(m)} \in \boldsymbol{T},\ n = 1, \ldots, N^{(m)}.$$

When $n_k^{(mv)}, k = 1, \ldots, K, v = 1, \ldots, V$ describes the number of assignments of word v in text m to topic k, we can define the cumulative count of word v in topic k over all documents by $n_k^{(\bullet v)}$ and, analogously, the cumulative count of topic k over all words in document m by $n_k^{(m\bullet)}$, while $n_k^{(\bullet\bullet)}$ indicates the total count of assignments to topic k. Then, let

$$\boldsymbol{w}_k = \left(n_k^{(\bullet 1)}, \ldots, n_k^{(\bullet V)}\right)^T \in \mathbb{N}_0^V,\ k = 1, \ldots, K$$

denote the vector of word counts for topic k.

Using these definitions, the underlying probability model (Griffiths and Steyvers, 2004) can be written as

$$\begin{aligned}
W_n^{(m)} \mid T_n^{(m)}, \boldsymbol{\phi}_k &\sim \text{Discrete}(\boldsymbol{\phi}_k),\\
\boldsymbol{\phi}_k &\sim \text{Dirichlet}(\eta),\\
T_n^{(m)} \mid \boldsymbol{\theta}_m &\sim \text{Discrete}(\boldsymbol{\theta}_m),\\
\boldsymbol{\theta}_m &\sim \text{Dirichlet}(\alpha).
\end{aligned}$$

For a given parameter set $\{K, \alpha, \eta\}$, LDA assigns one of the K topics to each token. Here K denotes the number of topics and α, η are parameters of a Dirichlet distribution defining the type of mixture of topics in every text and the type of mixture of words in every topic. Higher values for α lead to a more heterogeneous mixture of topics whereas lower values are more likely to produce less but more dominant topics per text. Analogously, η controls the mixture of words in topics. Although the LDA permits α and η to be vector valued Blei et al. (2003), they are usually chosen symmetric because typically the user has no a-priori information about the topic distributions $\boldsymbol{\theta}$ and word distributions $\boldsymbol{\phi}$.

Topic distributions per text $\boldsymbol{\theta}_m = (\theta_{m,1}, \ldots, \theta_{m,K})^T \in (0,1)^K$ and word distributions per topic $\boldsymbol{\phi}_k = (\phi_{k,1}, \ldots, \phi_{k,V})^T \in (0,1)^V$ can be estimated through the Collapsed Gibbs Sampler procedure (Griffiths and Steyvers, 2004) by

$$\hat{\theta}_{m,k} = \frac{n_k^{(m\bullet)} + \alpha}{N^{(m)} + K\alpha} \qquad \text{and} \qquad \hat{\phi}_{k,v} = \frac{n_k^{(\bullet v)} + \eta}{n_k^{(\bullet\bullet)} + V\eta}.$$

2.2 Methods and modifications of LDA to overcome the instability

Inferring LDA using Gibbs sampling is sensitive to the initial assignments, that are often chosen as random, and the reassignment is based on the conditional distributions, which leads to different results in multiple LDA runs for fixed parameters. This instability of LDA leads to a lack of reliability of the modeling results.

This fact is rarely discussed in applications (Agrawal et al., 2018), although several approaches have been proposed to encounter this problem. Agrawal et al. (2018) propose a new algorithm LDADE (LDA **D**ifferential **E**volution) which automatically tunes the parameters of LDA in order to optimize topic similarity in replications using a differential evolution algorithm. This results in a set of input parameters K, α and η which perform best on the given data with respect to reliability. This procedure does not really increase the reliability for a given parameter set, but tries to find the parameter set that produces the most reliable results. The implicit parameter optimization of the mentioned procedure can make one believe that the resulting model is best in terms of reliability and quality. However, it is likely that the tuning algorithm is biased to select parameters that result in systematically better reliability values independent of the underlying dataset, e.g. low α and η parameters.

Another option is to apply a selection criterion to a set of models. The selection can be done by optimizing perplexity (Blei et al., 2003), a performance measure for probabilistic models to estimate how well new data fit into the model (Rosen-Zvi et al., 2004). Alternatively, Nguyen et al. (2014) proposed to average stages of the Gibbs sampling procedure. They present different variations to average iteration steps and show that their approach leads to an increase of perplexity. Averaging LDA models comes with the drawback that one only receives averaged topic proportions, but no specific topic assignment per token. In addition, it was shown that likelihood-based measures like perplexity are negatively correlated with human judgments on topic quality (Chang et al., 2009). Instead, optimizing the semantic coherence of topics should be the aim for a selection criterion. Chang et al. (2009) provide a validation technique called Word/Topic Intrusion (implemented in Koppers et al., 2020) which depends on a human coding process. Automated measures to select the best LDA regarding coherence can be transferred from the Topic Coherence (Mimno et al., 2011; Stevens et al., 2012). But there is no stable and validated aggregation technique of this type of topic quality measure for the results of LDA runs.

Maier et al. (2018) aim for increasing both, reliability and interpretability of the final model simultaneously. Therefore, they maximize topic similarity as well as topic coherence, but discover that standard metrics in general do not perform well in increasing interpretability. Instead, manual approaches as the mentioned intruder validation technique proposed by Chang et al. (2009) are essential. Maier et al. (2018) propose to increase reliability of LDA by initializing topic assignments of the tokens reasonably, e.g. using co-occurences of words (Newman et al., 2011). This initialization technique has the drawback that the model is restricted to a subset of possible results.

There is also a modification of the implementation of LDA that aims to reduce instability. GLDA (**G**ranulated LDA) was proposed by Koltcov et al. (2016) and is based on a modified Gibbs Sampler. The idea of the algorithm is that tokens that are closer to each other are more likely to be assigned to the same topic. The authors show that their algorithm performs comparably well with standard LDA regarding interpretability. Moreover, it leads to more stable results, although, their study is based on only three LDA runs. The implementation is not publicly available. Thus, a validation of this method on other datasets or with larger numbers of replications is pending.

In this work, we propose the novel selection algorithm LDAPrototype based on the tailored similarity measure S-CLOP for LDA models. Thus, our contribution is two-fold. The measure is able to assess the stability of LDA with clustering techniques applied to replicated LDA runs. High stability corresponds to high reliability of findings based on stable models in the sense of improving reproducibility. We introduce a new automated method of clustering topics, more precisely a pruning algorithm for results of hierarchical clustering, based on the optimality criterion that for clustered results of replicated LDA runs, in the ideal case each cluster should contain exactly one topic of every replication of the modeling procedure. This results in our novel tailored similarity measure S-CLOP (**S**imilarity of multiple sets by **C**lustering with **LO**cal **P**runing) for LDA runs. We demonstrate the potential of this measure to quantify stability and to improve reliability by applying it to example corpora. Based on our newly proposed similarity measure S-CLOP, we propose a combination of a repetition strategy and selection criterion to increase the reliability of findings from LDA models leading ti the LDAPrototype algorithm.

3 Methods

We introduce the new method LDAPrototype that selects the medoid of a number of LDA runs. The selection is achieved by choosing the model that maximizes the mean pairwise S-CLOP value to all other LDA runs. For assessing similarities of LDA models using our novel S-CLOP measure, also an adequate similarity measure for topics is required. We define a more robust version of the Jaccard coefficient in the sense that not all words are considered as relevant for each topic. We present other implemented similarity measures, which are compared to our thresholded Jaccard coefficient regarding reliability gain and computation time in Section 5.3. In Section 5, the selection algorithm LDAPrototype is applied to six example corpora to assess the increase in reliability of findings from LDA models.

3.1 LDAPrototype: a new selection algorithm for LDA models

We propose the novel selection algorithm LDAPrototype to improve the reliability of LDA results. This algorithm selects from a set of LDA models the model that is most similar on

Algorithm 1: Selecting the medoid of a number of LDA runs

Data: A set of R LDA models
Result: The LDA with maximal mean pairwise similarity to all other LDA runs
begin
  for $i = 1$ **to** $R-1$ **do**
    for $j = i+1$ **to** R **do**
      Calculate pairwise similarity of LDAs set_i and set_j using a similarity measure sim: $\mathsf{sim}(\mathsf{set}_i, \mathsf{set}_j)$;
    end
  end
  Determine $\mathsf{proto} = \mathsf{set}_{\text{opt}}$ with $\text{opt} = \underset{i \in 1,\dots,R}{\arg\max} \frac{1}{R-1} \sum_{j \neq i} \mathsf{sim}(\mathsf{set}_i, \mathsf{set}_j)$;
  return proto
end

average to all other runs. The approach is similar to the choice of the median in the one-dimensional space. In the multidimensional space, this choice is called medoid and differs from a centroid in the sense that it is not obtained by model averaging, but by model selection. There are methods of model averaging for LDA (cf. Section 2.2), but these have the disadvantage that properties of a single run, such as the assignments of individual tokens to topics, are lost. The proposed selection algorithm preserves this information because it does not influence the modeling itself. The LDAPrototype procedure selects one single model from the set of candidate models.

The corresponding procedure is presented in Algorithm 1. Besides the set of LDAs, a similarity measure for LDA models is needed for the computation. We define a novel tailored measure for this approach in Section 3.4. The measure is called S-CLOP and in turn requires the choice of a topic similarity measure. In the following Section 3.2, we define a default measure for computing these similarities. In Section 3.3 we discuss popular and other promising measures for computing topic similarities, which we then compare to our proposed Jaccard coefficient in Section refcomparesims.

3.2 Thresholded version of the Jaccard coefficient: a similarity measure for topics

A similarity between two topics can be calculated based on the corresponding vectors of counts or set of words. We build on the well established Jaccard coefficient (Jaccard, 1912) and introduce a more robust thresholded version. Its general form is given by

$$\text{Jaccard}(A,B) = \frac{|A \cap B|}{|A \cup B|}, \tag{1}$$

where A, B are sets of words.

Suppose we have a text corpus and we estimate a topic model with LDA, with parameters α, η and K topics. This is done R times independently leading to a set of $N = RK$ topics in total. Then, referring to Section 2.1,

$$\boldsymbol{w}^{(r)} = \left(\boldsymbol{w}_1^{(r)}, \dots, \boldsymbol{w}_K^{(r)}\right) \in \mathbb{N}_0^{V \times K},\ r = 1, \dots, R$$

denotes the matrix of word counts per topic in the r-th replication. Then, for a given lower bound $\boldsymbol{c} = (c_1, \dots, c_N)$ and for two topics (i, j) represented by their word count vectors

$$\boldsymbol{w}_i, \boldsymbol{w}_j \in \left\{\boldsymbol{w}_1^{(1)}, \dots, \boldsymbol{w}_K^{(1)}, \boldsymbol{w}_1^{(2)}, \dots, \boldsymbol{w}_K^{(2)}, \dots, \boldsymbol{w}_1^{(R)}, \dots, \boldsymbol{w}_K^{(R)}\right\}$$

Table 1 Toy example: Assignment counts of two topics and calculation of thresholded version of the Jaccard (tJacc) coefficient

	$\boldsymbol{w}_1$	$\boldsymbol{w}_2$	$\wedge$	$\vee$
trump	1 668	2 860	1	1
trumps	446	854	1	1
president	91	876	1	1
donald	259	693	1	1
news	695	0	0	1
said	500	0	0	1
election	8	474	0	1
will	0	462	0	1
women	397	53	1	1
debate	394	11	0	1
sarcastic	1	4	0	0
Σ	4 459	6 287	5	10

vocabulary size $V = 11$,

relative limit $c_{rel} = 1/500$

$$\Rightarrow \boldsymbol{c} = (n_1^{\bullet\bullet}, n_2^{\bullet\bullet})^T/500 = (4\,459, 6\,287)^T/500 = (8.92, 12.57)^T.$$

$$\text{tJacc}(\boldsymbol{w}_1, \boldsymbol{w}_2) = \frac{5}{10}.$$

our thresholded version of the Jaccard coefficient is calculated by

$$\text{tJacc}(\boldsymbol{w}_i, \boldsymbol{w}_j) = \frac{\sum_{v=1}^{V} \mathbb{1}_{\left\{n_i^{(\bullet v)} > c_i \,\wedge\, n_j^{(\bullet v)} > c_j\right\}}}{\sum_{v=1}^{V} \mathbb{1}_{\left\{n_i^{(\bullet v)} > c_i \,\vee\, n_j^{(\bullet v)} > c_j\right\}}}, \tag{2}$$

Reasonable choices for the threshold vector $\boldsymbol{c} = (c_1, \ldots, c_N)^T \in \mathbb{N}^N$ are an equal absolute lower bound c_{abs} for all words or a topic-specific relative lower bound c_{rel}. A combination of both can be defined by

$$c_l = \max\{c_{abs}, c_{rel} n_l^{\bullet\bullet}\},\ l = 1, \ldots K, \ldots, 2K, \ldots RK = N, \qquad c_{abs} \in \mathbb{N}_0, c_{rel} \in [0,1].$$

To ensure that always enough words per topic are taken, the similarity measure is additionally implemented with a parameter that controls that at least a fixed user-defined number of the most frequent words assigned to the topic are considered.

The interpretation of this tJacc coefficient is the following. It is defined as the ratio of the numbers of the intersection and the union of the words of two topics, but a word is only considered if the number of its occurrences in a text exceeds the topic-specific threshold. In other words, we first restrict ourselves to the most relevant words per topic with respect to the number of assignments, to get rid of heavy tailed word lists. Then the resulting subsets of words are used to measure similarity of topics using the standard Jaccard coefficient.

We demonstrate how the measure is calculated with a small toy example. In Table 1, for eleven selected words the counts of assignments over all articles for the two topics $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are given. We use the relative lower bound with $c_{rel} = 1/500$. In the analysis presented in Section 5, we mainly also use $c_{rel} = 1/500$, which leads to around 100 important words per topic in the setting of the *usatoday* dataset. The last two columns indicate whether the corresponding word belongs to the thresholded intersection or union, respectively. For example, the word *election* does not belong to the intersection because its count is below the topic-specific (relative) threshold of at least nine assignments to the topic belonging to $\boldsymbol{w}_1$. The ratio of the number of entries in the third and the fourth column results in the similarity $\text{tJacc}(\boldsymbol{w}_1, \boldsymbol{w}_2) = \frac{5}{10} = 0.5$ of the two given topics.

3.3 Additionally implemented similarity measures for topics

We choose the tJacc coefficient as main similarity measure for comparing topics based on their word count vectors. In the literature, several alternatives are discussed, from which we compare the three most promising to our thresholded version of the Jaccard coefficient in Section 5.3.

While Agrawal et al. (2018) determine topic similarity with a Jaccard coefficient of the top 9 words per topic across multiple runs and measure stability with the median of the topic similarities, Maier et al. (2018) use the cosine similarity (see below). For repetitions of the same modeling procedure they match topics with the highest cosine similarity, which additionally has to be greater than an arbitrarily selected threshold 0.7. Then, for two models the similarity is calculated as the share of topic matches, and for more than two models by the mean of all pairwise shares.

Greene et al. (2014) and Su et al. (2016) determine topic similarities with an average Jaccard coefficient

$$\text{AverageJaccard}(A,B) = \frac{1}{N}\sum_{n=1}^{N}\frac{|A_n \cap B_n|}{|A_n \cup B_n|}, \tag{3}$$

where A_n and B_n define the sets of the first n words of the ordered lists from the word sets A and B. They choose $N = 5$ and find the best matching topics of different LDA runs based on this measure with the hungarian method (Kuhn, 1955). The authors try to encounter the problem that more than two runs of topics have to be matched by learning a reference model. They calculate the similarity of one LDA to the reference LDA as the mean average Jaccard coefficient over all matched topics, characterized by their word sets Z_{k^*} to the topics $Z_k^{(\text{ref})}$ of the reference model. Here Z_{k^*} denotes the reordered topics' word lists Z_k of the LDA run, so that matched topics have the same index. Analogously, they calculate stability over a number of R replications as the mean over the pairwise similarities against the topics' word lists of the (predetermined) reference model $Z_k^{(\text{ref})}$ by

$$\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{K}\sum_{k=1}^{K}\text{AverageJaccard}\left(Z_k^{(\text{ref})}, Z_{k^*}^{(r)}\right)\right). \tag{4}$$

One drawback of this approach is the specification of the reference model, which should be a good representative of all other LDAs. It is non-trivial to determine this representative model. Therefore, our approach follows an opposite strategy. We first calculate similarities between models and then determine the prototype model, i.e. the most representative LDA run, based on these values.

Aletras and Stevenson (2014) argue that Jensen-Shannon Divergence (Lin, 1991) is one of the best similarity measures based on word distributions considering correlation with human judgments. It is a symmetric version of the Kullback-Leibler Divergence (Kullback and Leibler, 1951)

$$\text{KLD}(\boldsymbol{q}_1, \boldsymbol{q}_2) = \sum_{v=1}^{V} q_{1,v}\log\frac{q_{1,v}}{q_{2,v}},\ \boldsymbol{q}_1, \boldsymbol{q}_2 \in (0,1]^V \tag{5}$$

and treated as similarity measure defined as

$$\begin{aligned} \mathrm{JS}(\boldsymbol{w}_i, \boldsymbol{w}_j) &= 1 - \left(\mathrm{KLD}\left(\boldsymbol{p}_i, \frac{\boldsymbol{p}_i + \boldsymbol{p}_j}{2}\right) + \mathrm{KLD}\left(\boldsymbol{p}_j, \frac{\boldsymbol{p}_i + \boldsymbol{p}_j}{2}\right)\right)/2 \\ &= 1 - \mathrm{KLD}(\boldsymbol{p}_i, \boldsymbol{p}_i + \boldsymbol{p}_j)/2 - \mathrm{KLD}(\boldsymbol{p}_j, \boldsymbol{p}_i + \boldsymbol{p}_j)/2 - \log(2), \\ \boldsymbol{p}_l &= \left(p_{l,1}, ..., p_{l,V}\right) = \left(n_l^{(\bullet 1)}, ..., n_l^{(\bullet V)}\right)/n_l^{(\bullet\bullet)}. \end{aligned} \tag{6}$$

Moreover, Aletras and Stevenson (2014) found out that a standard Jaccard coefficient is able to realize higher correlations to human judgments than other common similarity measures on specific datasets.

Kim and Oh (2011) showed that Jaccard coefficients perform on par with Jensen-Shannon Divergence and outperform a number of other popular similarity measures like cosine similarity, which is defined as

$$\cos(\boldsymbol{w}_i, \boldsymbol{w}_j) = \frac{\sum_{v=1}^{V} n_i^{(\bullet v)} n_j^{(\bullet v)}}{\sqrt{\sum_{v=1}^{V} \left(n_i^{(\bullet v)}\right)^2} \sqrt{\sum_{v=1}^{V} \left(n_j^{(\bullet v)}\right)^2}}. \tag{7}$$

and the mentioned Kullback-Leibler Divergence. To quantify the quality of the similarity measures they compare the negative log-likelihood of the model as an indicator how well the model explains the data. They swap the best matching topics from models of two time slices and interpret an increase of the negative log-likelihood as deficiency of the specific similarity measure.

Another option for measuring topic similarity introduced by Mantyla et al. (2018) is the Rank Biased Overlap (RBO) (Webber et al., 2010) for comparing ranked lists. The similarity is defined as

$$\mathrm{RBO}(A, B) = 2p^k \frac{|A_k \cap B_k|}{|A_k| + |B_k|} + \frac{1-p}{p} \sum_{d=1}^{k} 2p^d \frac{|A_d \cap B_d|}{|A_d| + |B_d|} \tag{8}$$

with parameters $k \in \mathbb{N}$ setting the maximum depth of evaluation and $p \in (0,1)$ controlling the influence of higher ranked words. While the measure seems to be useful because it implements a more flexible form of a Jaccard coefficient, the authors do not investigate stability of LDA models based on RBO. Moreover, the calculation of the measure is very time consuming.

For our analysis we prefer the thresholded version of the Jaccard coefficient tJacc as defined in formula 2, because in Section 5.3 we show that it performs on par with the others. In addition, it is calculated faster than the Jensen-Shannon Divergence and especially than the RBO. In comparison to the cosine similarity, calculation of the tJacc is computationally more demanding, but may lead to an increased interpretability. In view of the large computation demand for the LDA modeling in big data scenarios, this runtime increase is acceptable and rather minor.

3.4 S-CLOP: a new similarity measure for LDA models

We introduce the new similarity measure S-CLOP for comparing different LDA runs consisting of topics that are represented by word count vectors. The pairwise distances are calculated with the tJacc coefficient in formula 2. However, the measure can be applied using any appropriate distance respectively similarity measure, see Section 3.3.

The general idea of the measure S-CLOP is the following. First, join all LDA runs to one overall set, then cluster the topics with subsequent local pruning, and then check how many members of the original different LDA runs are contained in the resulting clusters. Then the deviation from the perfect situation of one representative from each LDA run is quantified. Two models are very similar, if always one topic of the first model is clustered together with one topic from the other model. High S-CLOP similarity values suggest that many topics can be identified that have a representative in each LDA run.

For the initial clustering step, we use hierarchical clustering with complete linkage (Hastie et al., 2009, pp. 520–525). We prefer complete linkage over single or average linkage because it uses the maximum distance between topics to identify clusters. This is consistent with our aim of identifying highly homogeneous groups. Of course, topic similarities must first be transformed to distances to apply hierarchical clustering.

Measuring disparity of LDA runs Consider a cluster (respectively a group) g of topics, after clustering R LDA runs in one joint cluster analysis, using all $R \cdot K$ topics from all runs. The goal is to quantify the deviation from the desired situation that each run is represented exactly once in g. The vector $\boldsymbol{t}^{(g)} = (t_1^{(g)}, \ldots, t_R^{(g)})^T \in \mathbb{N}_0^R$ contains the number of topics that belong to the different LDA runs. Then, we define the disparity measure

$$U(g) := \frac{1}{R} \sum_{r=1}^{R} |t_r^{(g)} - 1| \cdot \sum_{r=1}^{R} t_r^{(g)}. \tag{9}$$

The first factor $|t_r^{(g)} - 1|$ measures the deviation from the best case of exactly one topic per run in g. The second factor determines the number of members in the cluster and is required to penalize large clusters. Without this adjustment, the algorithm presented below for minimizing the sum of disparities would prefer one large cluster over a number of small clusters. In particular, without the second term, joining two perfect clusters as well as splitting one perfect cluster in two clusters would result in the same value for the mean disparity ($R/R = 1$), and we prefer the second situation, where two different topics from one run are not clustered together. The disparity of one overall cluster g containing all topics, e.g. defined by the root of a dendrogram, is given by $U(g) = (K-1) \cdot N$.

Finding the best cluster result by minimizing average disparity The goal is to minimize the sum of disparities $U(g)$ over all groups $g \in G$ of a cluster result. Hierarchical clustering of all N objects (topics) provides cluster results with $1, \ldots, N$ clusters. A common approach is to globally cut the dendrogram according to a target value. Here, we propose to prune the resulting dendrogram locally to obtain the final clusters. The pruning algorithm requires as input a hierarchical clustering result and minimizes the sum of disparities, with respect to the dendrogram structure, i.e.

$$U_\Sigma(G) := \sum_{g \in G} U(g) \to \min, \tag{10}$$

where G is a set of clusters (of topics), and the set of all topics is a disjoint union of the members of the single clusters $g \in G$.

Denote by G^* the optimal set of clusters resulting from splits identified from the dendrogram, and by $U^* := U_\Sigma(G^*)$ the corresponding minimal sum of disparities. The root of the dendrogram contains as a disjoint union the members of the two nodes obtained by the first split. Likewise, iteratively, each node contains as a disjoint union the members of the

Algorithm 2: Determining the minimal sum of disparities $U^*(g)$ of a cluster g

```
Data: A node of a dendrogram
Result: The minimal possible sum of disparities for this node
begin
    if is.leaf(node) then
        return (R − 1)/R
    else
        return min{U(node), Recall(node.left) + Recall(node.right)}
    end
end
```

Algorithm 3: Finding the optimal set of clusters G^*

```
Data: A dendrogram with a root
Result: A list correponding to the optimal set of clusters G*, obtained by local pruning of the
        dendrogram
begin
    node = root;
    if U(node) == U*(node) then
        Add all objects belonging to the cluster corresponding to node as one cluster to list;
    else
        Recall(node.left);
        Recall(node.right);
    end
end
return list
```

two nodes on a clustering level one step below this specific node, as denoted in Algorithm 2 by node.left and node.right. The optimal sum U^* can be calculated recursively with Algorithm 2.

For a node in the dendrogram, we denote by $U(\mathsf{node})$ the disparity of the corresponding cluster and by $U^*(\mathsf{node})$ the minimal sum of disparities of the dendrogram induced by (or below) this node. Algorithm 3 can now be used to find the best set of clusters. A cluster is added to the list of final clusters, if its disparity is lower than every sum of disparities obtained when further splitting this node.

Measuring similarity with aggregated disparities Finally, we can calculate the similarity of a set of LDA runs using the optimized set of clusters. We normalize the sum of disparities of the optimal clustering, such that its values lie in the interval $[0,1]$, where 0 corresponds to the worst case and 1 to the best case. The worst case is a pruning state with R clusters, each consisting of all topics from one LDA run. Then the pruning of Algorithm 3 would lead to a set $\tilde{G}$ of N single topic clusters, resulting in the highest possible value for the sum of disparities

$$U_{\Sigma,\mathsf{max}} := \sum_{g\in\tilde{G}} U(g) = N \cdot \frac{R-1}{R}. \tag{11}$$

The similarity measure S-CLOP (**S**imilarity of multiple sets by **C**lustering with **LO**cal **P**runing) then is defined by

$$\text{S-CLOP}(G) := 1 - \frac{1}{U_{\Sigma,\mathsf{max}}} \sum_{g\in G} U(g) \ \in [0,1] \tag{12}$$

Table 2 Specifications of the six considered datasets

Dataset	Type	Time	M	V (Limit)	K	Source
reuters	Newspaper	1987	91	2141	5–15	Rieger (2020)
economy	Wikinews	2004–2018	1855	7099 (5)	20	Koppers et al. (2020)
politics	Wikinews	2004–2009	4178	12 138 (5)	30	Koppers et al. (2020)
usatoday	Newspaper	06–11/2016	7453	25 486 (5)	50	LexisNexis (2019)
tweets	Twitter	03–06/2020	3 706 740	17 208 (250)	25	Rieger et al. (2021)
nyt	Newspaper	1999–2019	1 993 182	74 218 (250)	100	LexisNexis (2019)

and $\text{S-CLOP}(G^*) = \max_{g \in G} \text{S-CLOP}(G)$ defines the similarity of replicated LDA runs based on the identified optimal set of clusters G^*.

Determining the medoid As described in Section 3.1, the LDAPrototype algorithm (Algorithm 1) relies on finding the medoid using a suitable similarity measure for LDA models. Note that in the special case of comparing just two LDA runs with the same number of topics K, as it is the case in the setting of determining the medoid, the normalization factor is $U_{\Sigma,\mathsf{max}} = 2 \cdot K \cdot \frac{1}{2} = K$. With respect to the definition of U we can simplify formula 12 to

$$\text{S-CLOP}(G) = 1 - \frac{1}{2K} \sum_{g \in G} |g| \left(||g_{|1}| - 1| + ||g_{|2}| - 1| \right) \in [0, 1], \tag{13}$$

where $g_{|1}$ and $g_{|2}$ denote groups of g restricted to topics of the corresponding LDA run. Then, the medoid is determined by the run that maximizes the average pairwise S-CLOP value to all other LDA runs from the same set, which is obtained with Algorithm 1.

The introduced methods have been implemented as R package (Rieger, 2020) and are available at `https://github.com/JonasRieger/ldaPrototype` as continuously developing GitHub repository.

4 Data

In Section 5 we consider six different datasets, three of which are freely available through R packages (R Core Team, 2020). Table 2 gives an overview of the datasets. Among them are three corpora consisting of traditional newspaper articles. The dataset *reuters* contains 91 articles from 1987 and is included in the package ldaPrototype (Rieger, 2020). The datasets *usatoday* and *nyt* are available to us via the paid service of LexisNexis (2019), with more than 7000 articles and almost two million texts, respectively. They offer a good possibility to test the method on datasets of common and large size. For the latter, we consider all articles from the New York Times from 01/01/1999 to 12/31/2019. From the R package tosca (Koppers et al., 2020), we also use the *economy* and *politics* datasets, which consist of nearly 2000 and just over 4000 Wikinews articles from 2004 – 2018 and 2004 – 2009, respectively. Also, in contrast to the other datasets, we consider a collection of nearly four million German-language tweets from March 19 – June 27, 2020, the first 101 days after then-German Chancellor Merkel's TV address on the coronavirus outbreak. For this purpose, 50 000 tweets with keywords related to the coronavirus were scraped every hour over the mentioned period using the Twitter API and duplicates were removed (Rieger and von Nordheim, 2021).

All six corpora are preprocessed in R with the packages tosca and tm (Feinerer et al., 2008), using common procedures in natural language processing (NLP). That is, duplicates

from articles that occur more than once are removed, so that every unique article remains once. As an example, 204 articles were removed from the *usatoday* dataset, which previously contained 7 657 articles. As common in practice, characters are formatted to lowercase; numbers and punctuation are removed. In addition, a trusted stopword list (Feinerer et al., 2008) is applied to remove words that do not help in classifying texts in topics. Moreover, the texts are tokenized and words with a total count less or equal to a given limit are neglected. For example, for the *usatoday* dataset we choose this limit to be 5. This reduces the vocabulary size from 79 734 to $V = 25\,486$. For the larger datasets *tweets* and *nyt* we have set the limit higher to 250.

For all corpora, we heuristically choose a reasonable number of topics to model. We choose a higher number for larger and more general datasets. As can be seen in Section 5.4, for the *reuters* dataset we try different numbers of topics, namely K = 5, ..., 15, and investigate them with respect to the effects on modeling and runtime behavior. Other well known and widely used packages for preprocessing and/or modeling of text data are quanteda (Benoit et al., 2018), topicmodels (Grün and Hornik, 2011) and stm (Roberts et al., 2019).

5 Results

In the following, we apply the previously defined methods on the six presented datasets in different comparisons. For this purpose, the statistical programming software R 4.0.2 (R Core Team, 2020) and in particular the package ldaPrototype are used. This is based on an effective implementation in C/C++ of the LDA from the package lda (Chang, 2015). For computation on a batch system or local parallelization, we use the packages batchtools (Lang et al., 2017) or parallelMap (Bischl et al., 2020). For modeling we always use the default parameters unless otherwise specified. For the number of topics to be modeled K, the methods deliberately do not provide a default. Our chosen parameters depending on the dataset are given in Table 2. The parameters α and η are chosen by default as `alpha` $=$ `eta` $= 1/K$, the Gibbs Sampler runs for `num.iterations` $= 200$ iterations each time. For the computation using the thresholded version of the Jaccard coefficient tJacc defined in formula 2, we choose `limit.abs` $= 10$, `atLeast` $= 0$, and, unless otherwise specified, `limit.rel` $= 0.002$. In addition, $R = 100$ runs of LDAs are modeled by default.

The general procedure is as follows: We select a prototypical LDA from a set of R LDAs using the presented method LDAPrototype. We repeat this procedure 100 times. Thus, we obtain the pairwise S-CLOP values of all combinations of R LDAs, 100 times each. The 100 most representative LDAs, each selected from R LDAs, can then again be compared pairwise using the S-CLOP measure. The distribution of mean similarities of the simple replications can then be compared to the distribution of mean similarities of the prototypical LDAs. A location shift of the distribution represents an increase in reliability due to the use of the selection mechanism.

Table 3 shows the runtimes for determining the LDAPrototype depending on the dataset. Due to the size of the *nyt* dataset, only one prototype was determined, and only on 50 modeled LDAs. In addition, the calculation was not parallelized due to the required memory, so that it lasts around 130 days. For all other datasets 100×100 LDAs, and thus 100 LDAPrototypes were calculated. The runtimes are observed under parallelization on 4 cores and range from less than a minute for the *reuters* dataset to just over a day for the *tweets* dataset.

In the following, we first show a minimal example of the application of the S-CLOP measure to $R = 4$ runs. In doing so, we analyze the clustering behavior of the topics and exemplify the instability and thus limited reproducibility and reliability of the LDA results.

Table 3 Runtimes for determining the LDAPrototype on the six different datasets

Dataset	M	R	K	Cores	Min.	Mean	Max.	Unit
reuters	91	100	5	4	43.38	43.81	45.30	secs
economy	1855	100	20	4	8.25	8.33	8.69	mins
politics	4178	100	30	4	27.21	27.32	27.76	mins
usatoday	7453	100	50	4	3.33	3.42	3.56	hours
tweets	3 706 740	100	25	4	28.15	28.64	30.67	hours
nyt	1 993 182	50	100	1	-	130	-	days

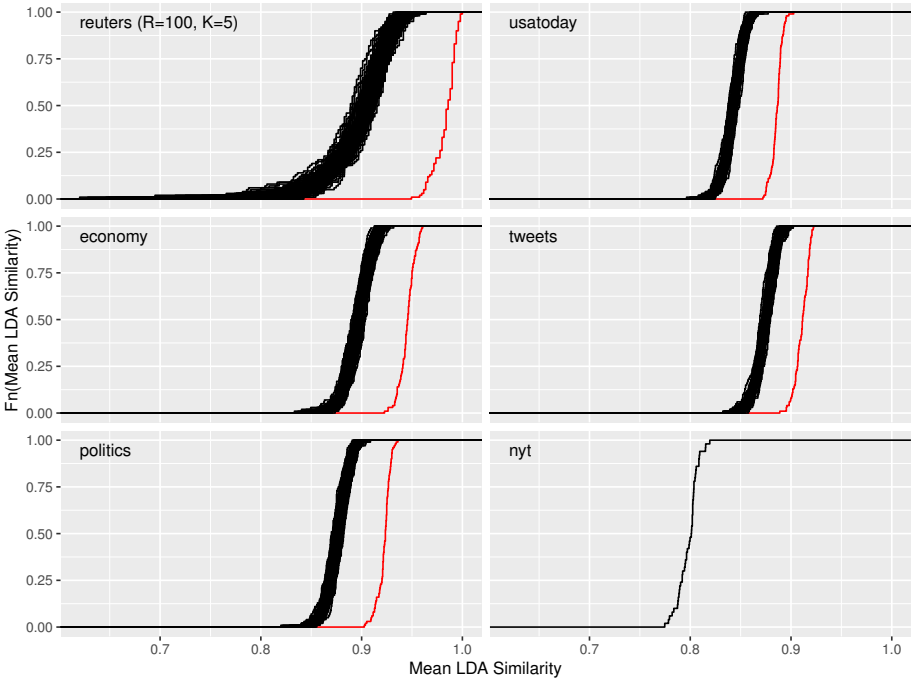

Fig. 1 Increase of reliability for five of the six different datasets and reliability of a single LDAPrototype for the *nyt* dataset; black: empirical cumulative distribution functions (ecdf) of the mean pairwise LDA similarites, red: ecdf of the mean pairwise LDAPrototype similarities

We then show in Section 5.2 the improvement in reliability on the same dataset (*usatoday*) in dependence of the parameter R, the number of potential LDAs. This is followed by a comparison of the use of the presented Jaccard coefficient tJacc in formula 2 with the other similarity measures cosine (7), Jensen-Shannon (6) and Rank Biased Overlap (8), from Section 3.3. For this, we also use the *usatoday* dataset and also compare the similarity measures in terms of their runtime and parallelizability. Then, on a smaller dataset (*reuters*), we compare the runtime and reliability gains for different numbers of topics K and different numbers of modeled LDA runs R. Finally, we analyze all six datasets and show that the method yields an increase in reliability regardless of the dataset and at the same time remains computable for large datasets. Figure 1 shows this increase. The (red) curves, indicating the empirical cumulative distribution functions (ecdf) of the mean pairwise similarities of the LDAProtoypes, differ significantly from the (black) ecdfs corresponding to the simple LDA replications.

5.1 Cluster analysis and similarity calculation

In this Section we present an example analysis of a clustering result of four independent LDA runs based on newspaper articles from USA Today. We run the basic LDA four times, such that the number of runs to be compared is $R = 4$ and the total number of topics to be clustered is $N = R \cdot K = 4 \cdot 50 = 200$. To demonstrate how dissimilar replicated LDA runs can be, we cluster the $N = 200$ topics from the $R = 4$ independent runs with $K = 50$ topics each using the tJacc coefficient from formula 2, complete linkage and the new introduced algorithm for pruning. The four runs were selected from 10000 total runs. In fact, the runs *Run1* and *Run2* were chosen as the top two models in mean similarity of the 100 prototypes, which means their points lie at the top of the very right (red) curve in the plot from *usatoday* in Figure 1. Their similarity values are 0.902 and 0.898 in the set of prototypes or 0.877 and 0.871 in the original sets, respectively. The model *Run3* was chosen as the worst of the 100 prototype models with a similarity value of 0.872 in the set of prototypes and 0.863 in its original set. *Run4* was chosen randomly as one of the worst models realizing a mean similarity to all other models in its original set of 0.807.

We apply hierarchical clustering with complete linkage to the 200 topics. The topics are labeled with meaningful titles (words or phrases). These labels were obtained by hand, based on the ranked list of the 20 most important words per topic. For this, the importance of a word $v = 1, \ldots, V$ in topic $k = 1, \ldots, K$ (Chang, 2015) is calculated by

$$I(v,k) = \frac{n_k^{(\bullet v)}}{n_k^{(\bullet\bullet)}} \left[\log \left(\frac{n_k^{(\bullet v)}}{n_k^{(\bullet\bullet)}} + \varepsilon \right) - \frac{1}{K} \sum_{l=1}^{K} \log \left(\frac{n_l^{(\bullet v)}}{n_l^{(\bullet\bullet)}} + \varepsilon \right) \right], \tag{14}$$

where ε is a small constant value which ensures numerical computability, which we choose as $\varepsilon = 10^{-5}$. The importance measure is intuitive, because it scores words high which occur often in the present topic, but less often in average in all other topics.

Figure 2 shows two dendrograms visualizing the result of clustering the 200 topics and of Algorithm 3 for clustering with local pruning. The horizontal axis describes the complete linkage distance based on our tJacc coefficient with `limit.rel` $= 0.002$. In the left dendrogram, the topic labels are colored with respect to the LDA run (*Run1*: grey, *Run2*: green, *Run3*: orange, *Run4*: purple). In the right dendrogram, topic labels are colored according to the clusters obtained with our proposed pruning algorithm. In addition, every topic label is prefixed by its run number.

Looking at the right dendrogram, we see that there are a number of clusters with identical labels for topics. This demonstrates that the four LDA runs produce a number of similar topics that are represented by similar word distributions. Examples for such stable topics are *Trump vs Clinton Campaign* colored yellow and *Olympics Medals* colored green.

However, there are also considerable differences visible. In the left dendrogram in Figure 2, strikingly, there are several topics from *Run4*, highlighted in red, where no other topic is within a small distance. It is remarkable that *Run4* creates such a great number of individual topics, e.g. *Video Games*, *Gender Debate*, *TV Sports* (which includes words for describing television schedules of sport events) and *Terrorism*. Also, *Run4* leads to six explicit stopword topics, which is the maximum number compared to the other runs with four to six stopword topics.

In the right dendrogram, the color depends on cluster membership. We measure combined stability of these four LDAs by applying the proposed pruning algorithm (Algorithm 3) to the dendrogram. This leads to 61 clusters and a S-CLOP score of 0.83. The normalization factor is given by $U_{\Sigma,\max} = K \cdot (R-1) = 50 \cdot 3 = 150$, and the minimization of

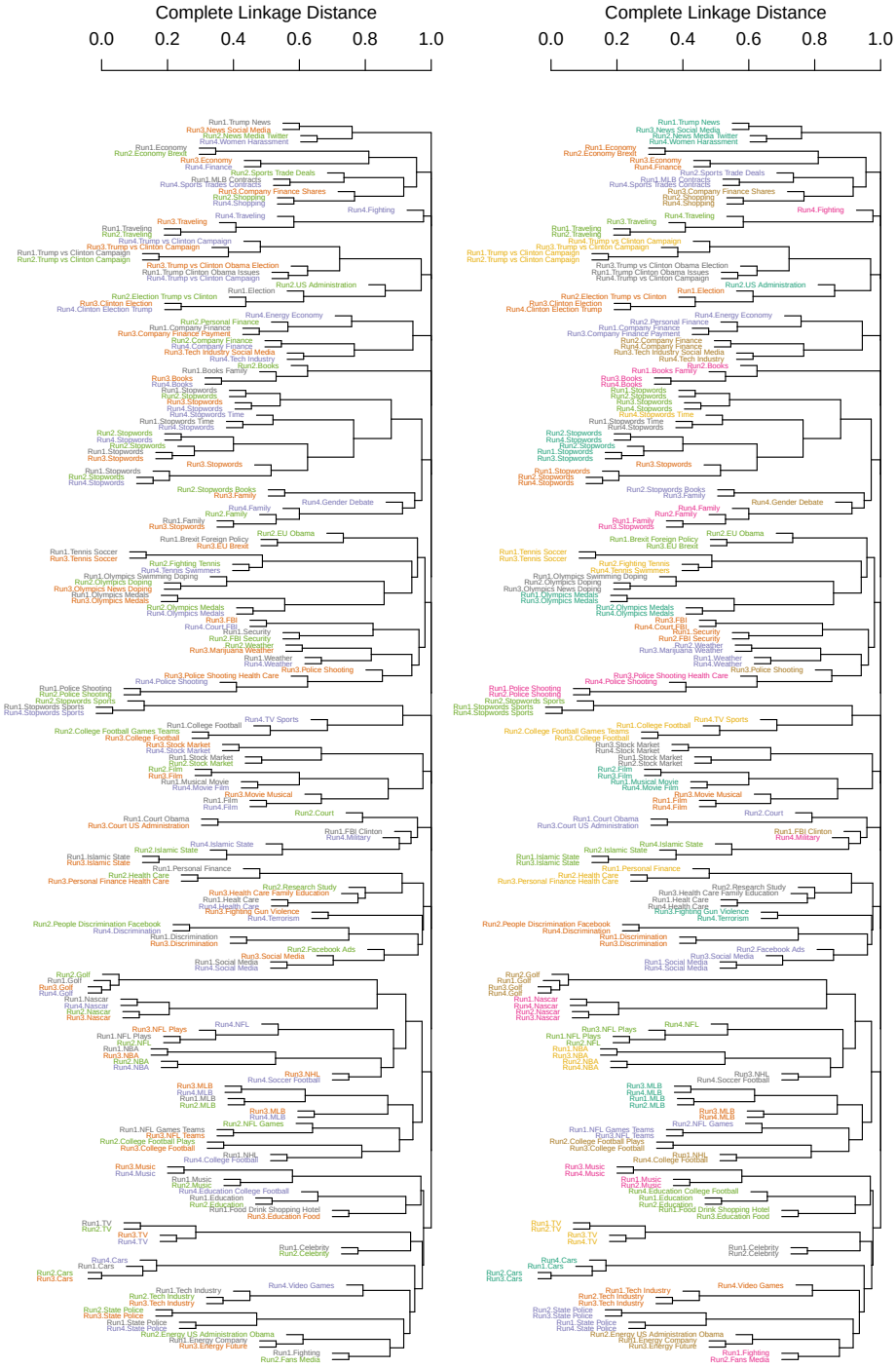

Fig. 2 Dendrograms of $N = 200$ topics from $R = 4$ selected LDA runs on the *usatoday* dataset with $K = 50$ topics each; left: colored by runs; right: colored by cluster membership

the sum of disparities yields $U^* = U_\Sigma(G^*) = 25$, resulting in S-CLOP $= 1 - 25/150 = 0.83$. There are seven single topics, one from each of the first three runs and four from *Run4*. The eleven clusters which consist of exactly three topics contain ten times a topic from *Run1*. Topics from *Run2* and *Run3* are represented nine times each, whereas only five of the mentioned clusters contain a topic from *Run4*. This shows that LDA run *Run4* strongly differs from the others. There are a lot of cases, where only a topic from this run is missing to obtain perfect topic clusters with exactly one topic from each run.

For comparison to the proposed local pruning algorithm, we once applied an established global criterion. Since 50 topics were originally modeled for each run, it is not reasonable to determine less than 50 clusters. Therefore, we try the global criterion with $50, \ldots, 70$ as the target cluster count. The largest similarity value according to S-CLOP based on the resulting clusters is obtained as 0.3 for 59 clusters. However, considering the dendrogram, there are not such large differences in the topic structures between the runs as to justify such a low similarity. This shows the necessity of the presented local pruning method.

In addition, the dendrograms illustrate that random selection can lead to a poor model regarding interpretability and especially to some kind of an outlier model as *Run4*. This means that random selection can lead to low reliability.

5.2 Increase of reliability

As an improvement, we recommend to use the introduced approach based on prototyping of replications. It increases mean similarity, which comes along with an increase in reliability. We demonstrate how to determine a prototype LDA run as the most representative run out of a set of runs, based on our novel pruning algorithm. We show that this technique leads to systematically higher LDA similarities, which suggests a higher reliability of LDA findings from such a prototype run.

The introduced similarity measure S-CLOP (12) quantifies pairwise similarity of two LDA runs by

$$1 - \frac{1}{50} \sum_{g \in G^*} U(g) = 1 - \frac{1}{100} \sum_{g \in G^*} |g| \left(||g_{|1}| - 1| + ||g_{|2}| - 1| \right),$$

where $K = 50$ is the number of topics per model and G^* an optimized set of topic clusters identified by our proposed pruning algorithm. We investigate the mean S-CLOP scores per LDA on the corpus from USA Today newspaper articles.

We propose to select the LDA run with highest mean pairwise similarity to all other runs. The following study shows that this is a suitable way to identify a stable prototype LDA, thus leading to improved reliability of LDA findings based on this particular run. We fit $R = 100$ LDA models and select the model with highest mean similarity as prototype. This procedure is repeated 100 times, which results in 100 prototype models. Then, for the 100 prototypes, also mean pairwise similarities to the other prototypes are calculated. The results are visualized in Figure 3. The very right curve describes the empirical cumulative distribution function of the mean similarities obtained for the 100 prototypes. At the very left there are the 100 curves of the 100×100 original runs. In addition, we also determine 100 prototypes from subsamples. For this, only 10, 20, 30, 40 or 50 runs from each original set of 100 runs are randomly selected and are used for the following calculation steps. The resulting curves are also plotted and labeled in Figure 3.

The minimum of the mean similarities from the original 100 sets of 100 models is 0.796, while the maximum is 0.877. Based on the findings from Figure 3, we recommend to fit at

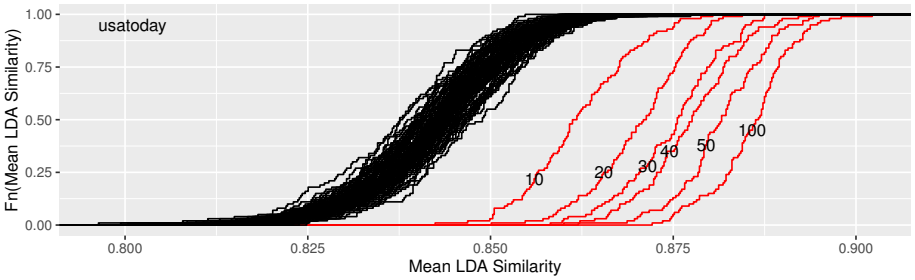

Fig. 3 Increase of reliability in dependence of the number of replications R on the *usatoday* dataset: ecdfs of the mean similarities calculated on 100 replications of randomly selected LDA runs (black) and on the 100 most representative prototype LDA runs (red) based on subsamples of $10, 20, 30, 40, 50$ or all 100 LDA runs

least 50 replications because this leads to an increase of similarity to 0.862 at the minimum and 0.895 at the maximum. Higher values for the number of repetitions are desirable. In general, the choice depends on the complexity of the corpus. Encapsulated topics or certain complicated dependency structures make the modeling procedure more prone to a larger span of possible fits and therefore to smaller mean similarity values. However, if computational power is limited, already taking the prototype model from 10 candidates considerably increases the reliability. Here, the minimum and maximum of mean similarity are 0.842 and 0.880, considerably higher values than these associated with random selection.

5.3 Comparison of the implemented similarity measures

Up to this point, we have performed all calculations with our tJacc coefficient, from formula 2 in Section 3.2. Now we study differences in reliability with different choices of measures of topic similarity. For this we compare cosine similarity (7), Jensen-Shannon similarity (6) and Rank Biased Overlap (RBO, formula 8) with the tJacc coefficient. We consider, where applicable, effects of different parameter constellations on the correlation of the similarities. We show that the tJacc coefficient is a good choice for the topic similarity measure because it improves reliability while not having a disproportionate runtime.

We again consider the *usatoday* dataset for the comparison. For the tJacc coefficient we set `limit.abs` $= 10$ and `atLeast` $= 0$ and vary `limit.rel` $\in \{0, 0.001, 0.002, 0.005\}$, for the RBO we choose eight combinations of $k \in \{10, 20, 50\}$ and $p \in \{0.1, 0.5, 0.8, 0.9\}$. We compute all pairwise topic similarities of the total $R \cdot K = 5000$ topics using the given measures and first consider the correlation of the pairwise LDA similarities based on these. In Figure 4 all pairwise correlations of the similarity values are given.

Thereby, clear patterns can be identified. The similarities obtained with cosine and Jensen-Shannon are correlated with a value of 0.50. The similarity values based on the tJacc coefficient correlate with the cosine similarity depending on the parameter in the range from 0.40 to 0.47. It is noticeable that the correlation initially increases from 0.41 to 0.47 when considering a longer tail of words belonging to a topic, but drops to 0.40 when considering the complete tail. In contrast, the correlation with Jensen-Shannon LDA similarities increases steadily from 0.38 for `limit.rel` $= 0.005$ to 0.66 when considering the complete tail, or these words that were assigned to that topic at least 11 times (`limit.abs` $= 10$). It is interesting to note that even the different versions of the tJacc coefficient are mostly less correlated with each other than the tJacc LDA similarities are with the Jensen-Shannon

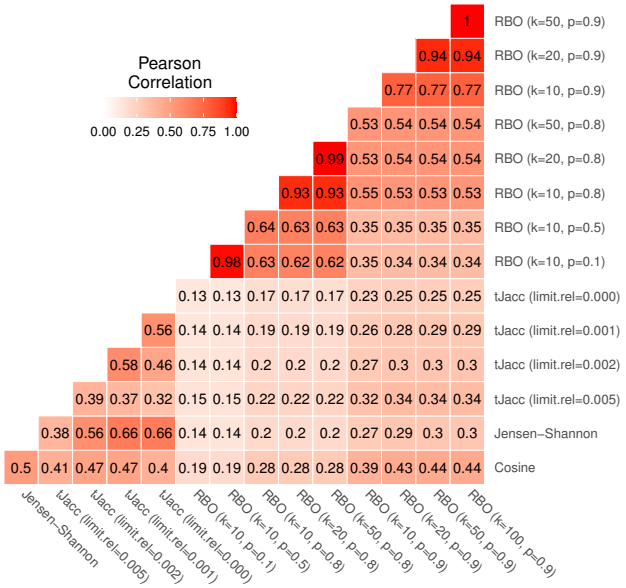

Fig. 4 Correlation matrix of pairwise LDA similarity values calculated using the four similarity measures with selected parameter combinations for 5000 topics on the *usatoday* dataset

LDA similarities. This illustrates that there can be significant differences between different choices for the threshold based on `limit.rel`.

As mentioned in Section 3.2, choosing the parameter `limit.rel` as 0.002 determines around 100 words per topic as relevant. Thus, one could assume that the corresponding similarities should correlate strongly with those of the RBO with $k = 100$. In fact, we see that the RBO is rather weakly correlated with the other three measures overall; in particular, for low values of $k \leq 50$ and $p \leq 0.8$. The corresponding correlations are all below 0.30. However, it is also noticeable that for increasing k and p, the RBO similarities appear to be increasingly correlated with the other three similarity measures. This is due to the fact that for larger values of k and p the RBO converges to a measure very similar to the Jaccard coefficient or for $p \neq 1$ to the AverageJaccard defined in formula 3 in Section 3.3. This means that RBO similarities calculated with parameters $k = 100$ and $p = 1$ should be very close to tJacc similarities with `limit.rel` $= 0.002$. In the present case, however, even with 0.9, p is still much smaller than 1. Note that $0.9^{100} \approx 0$ and thus the 100th word in the ranked list gets practically no weight, which is exactly the idea of the RBO. Therefore it is not recommended to choose the parameter p close to 1. Alternatively, a better approach would be to choose an implementation based on Jaccard, e.g. AverageJaccard, because it is faster to implement. These findings show that k and p should always be chosen dependent on each other. One can also see in Figure 4 that words from rank 50 onwards no longer have a significant influence with a choice of $p = 0.9$. The correlation between the LDA similarities based on RBO with $k = 50, p = 0.9$ and $k = 100, p = 0.9$ is 1.

The four measures considered have different complexities in terms of their implementations. While the cosine similarity can be computed very quickly, the tJacc coefficient and the Jensen-Shannon similarity require computationally intensive precomputations, which increases the runtime. Cosine similarity and Jensen-Shannon similarity have no parameters to

Table 4 Runtime and parallelizability comparison of the four similarity measures computing pairwise similarities of $R \cdot K = 5000$ topics on the *usatoday* dataset; all runtimes refer to hours on 4 cores, a measure is better implemented in parallel for a larger parallelizability score $\in [0.25, 1]$

	Measure k	Cos -	tJacc -	JS -	RBO 10	RBO 20	RBO 50	RBO 100
Parallel (4 Cores)	Min.	0.26	0.65	1.33	4.54	9.09	22.47	47.15
	Mean	0.29	0.68	1.39	5.05	11.40	24.75	51.62
	Max.	0.31	0.76	1.48	6.07	12.13	29.37	59.05
Serial	Time	0.86	2.49	4.58	-	27.26	-	-
Parallelizability	Score	0.75	0.93	0.83	-	0.60	-	-

be set. For the tJacc coefficient, the calculations do not depend on the chosen parameters. The calculation of the RBO is generally time-consuming, since values must be calculated individually for all considered depths up to the maximum rank, which also means that the runtime for the calculation of the RBO strongly depends on the parameter k.

The runtimes are documented in Table 4. The times here are in hours, the calculations were run on four cores in parallel and are based on at least 100 different values. For each measure, a serial calculation was performed once in each case in order to be able to give a value for estimating the parallelizability of the calculations or implementations. It can be seen that the cosine similarity is the fastest to calculate with well under one hour. The thresholded Jaccard coefficient requires slightly more than twice as much time, while the Jensen-Shannon similarity computes with 83 and 275 minutes, respectively, again almost twice as long as the tJacc coefficient. The Rank Biased Overlap with $k = 100$ takes even over 2 days in parallel computation. This is also due to the fact that the implementation is only parallelized with a score of 0.6, i.e. one achieves only 60% of the maximum possible time saving through parallelization. With a maximum score of 1, the calculation would only take 30.97 hours instead of 51.62. The cosine similarity is also not implemented maximally parallelized. This is due to the overall low runtime. The tJacc coefficient, instead, is implemented in parallel with an approximate maximum time saving of 0.93.

After comparing the measures in terms of runtime and correlation of the resulting pairwise S-CLOP values, we restrict the analysis to one parameter combination per measure and compare the increase in reliability in dependence of the measure. In Figure 5, these are plotted against each other. We compare the cosine similarity, Jensen-Shannon similarity, the tJacc coefficient with `limit.rel = 0.002` and the RBO with $k = 20, p = 0.9$. On the diagonal of the plot matrix, the known ecdfs of the LDA similarities are shown in black and the ecdf of the LDAPrototypes in red, respectively. In addition, we calculated - using the same measure - the similarities of the LDAPrototypes, that were determined based on the other three topic similarities. The colors of the ecdfs correspond to the color of the box of similarity measures. This allows the different measures to be analyzed in terms of their selection behavior taking into account the level differences of the similarity values. The lower triangular plot matrix provides the LDA similarities as pairwise correlation plots, the upper matrix the corresponding correlations themselves. Determining the pairwise LDA similarities based on 100 experiments with $R = 100$ replications each yields $100 \cdot (R-1)R/2 = 495\,000$ values per similarity measure, on which the heatmaps and correlations are both based.

It can be seen that the LDA similarities according to the tJacc coefficient and according to the Jensen-Shannon similarity are most highly correlated with each other. The point cloud is least circular, but rather narrow and elliptical in shape. It can also be seen that the RBO

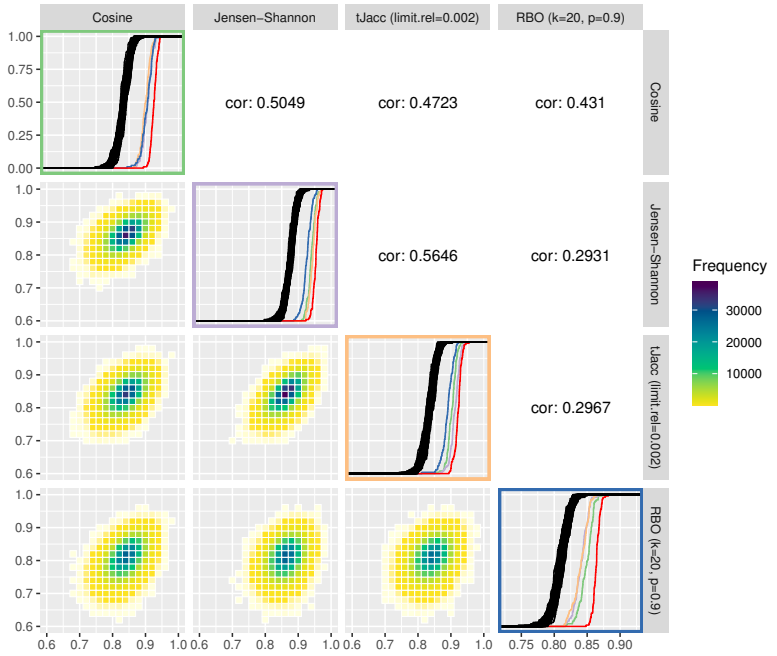

Fig. 5 Comparison of four selected similarity measures regarding their increase of the reliability of LDA results: on the diagonal the ecdfs of the LDA similarities based on the different similarity measures are shown in black, in addition the similarities of the LDAPrototypes are shown in red or in further colors for the LDAPrototypes determined on the basis of the other measures (green for cosine, purple for Jensen-Shannon, orange for tJacc and blue for RBO); below the diagonals correlation plots of the LDA similarities according to the similarity measure are shown as heatmap, above the diagonals the corresponding correlations are given

LDA similarities are very different from the others. In the lower right plot (outlined in blue), the three ecdfs of different colors show significant differences from the red curve, with the cosine curve showing even slightly more similarity than the curves of the other two measures. Similarly, the blue curve is in the Jensen-Shannon (purple) and tJacc (orange) windows far behind the other colored ecdfs. The cosine LDA similarities (top left, green) differ barely for all the foreign-determined LDA prototypes, but the red curve sets itself apart from the others. Thus, the selection by cosine similarity obviously also differs significantly from the others.

The plots from Figure 5 suggest that all four measures differ with regard to their selection criteria. However, a qualitative ranking which of the measures increases the reliability of the results the most is difficult because for this question the true evaluation measure has to be identified first, which is a vicious circle. For this reason, all these measures have their justification to be used within the procedure. We prefer to use the thresholded Jaccard (tJacc) coefficient because it has plausible heuristics and reasonable runtime. Its selection of the LDAPrototype strongly correlates with that of the Jensen-Shannon similarity, for which, according to Section 3.3, besides the tJacc coefficient itself, the best results in terms of correlations with human perceptions could be obtained.

5.4 Comparison of different values for the parameters R and K

In addition to the choice of the similarity measure, the choice of the parameters K, number of topics to be modeled, and R, number of replications, also has a large influence on the runtime of the method. In Section 5.2 it has already been shown that larger values for R result in a larger increase in reliability. We will confirm these findings based on the *reuters* dataset on the one hand and on the other hand bring them into a combined comparison with the number of modeled topics.

Table 5 Runtime comparison of LDAPrototypes on the *reuters* dataset for choices of R and K; all runtimes refer to minutes on 4 cores

R	50	50	50	100	100	100	200	200	200	500	500	500
K	5	10	15	5	10	15	5	10	15	5	10	15
Min.	0.25	0.47	0.79	0.72	1.43	2.57	2.46	5.47	9.81	16.01	36.45	69.28
Mean	0.26	0.48	0.80	0.73	1.47	2.65	2.49	5.57	10.03	16.31	37.03	70.03
Max.	0.31	0.53	0.82	0.76	1.64	2.81	2.77	5.84	10.37	17.16	38.25	71.58

In Table 5 the runtimes for the determination of the LDAPrototypes based on the calculations of the topic similarities by the tJacc coefficient are given. Obviously, the runtime increases more than linear in both in the number of topics K to be modeled and in the number of replications R. This is due to the fact that the computation of the matrices of topic similarities have a quadratic complexity, since pairwise similarities are computed. The runtime for modeling the LDAs is linear in the parameters R and K.

Figure 6 shows the corresponding plots for the increase in reliability for all combinations of $R = 50, 100, 200, 500$ and $K = 5, \ldots, 15$. Consistent with expectations, for fixed K, increasing the replication number from 50 to 500 does not change the location parameter for the ecdfs of the mean pairwise LDA similarities. However, the variance of the similarities decreases due to the higher replicate number. It can be seen that for fixed R and increasing topic number K the level of similarities decreases. From an average similarity of 0.90 for $K = 5$ for the LDA replications, the value decreases to 0.75 for $K = 10$ and 0.65 for $K = 15$. The increase in reliability is marked by the red ecdfs and is clearly pronounced for all parameter combinations. The gain is larger for higher topic numbers, so the level of LDAPrototype similarity does not decrease quite as much with increasing parameter K. The gain is also larger for increasing R. This is an expected behavior because then each prototype is determined from a larger set of individual LDAs and thus each LDAPrototype becomes even more reliable. For $K = 5$, the similarities of the prototypes thus increase from 0.97 for $R = 50$ to close to 1 for $R = 500$ (with LDA similarities around 0.90). For $K = 9$ they increase from 0.91 to 0.95 (0.78) and for $K = 14$ from 0.84 to 0.90 (0.67).

The findings from Figure 3 are confirmed here. With increasing R the reliability gain increases. However, already for small values of R a clear increase is recognizable. Accordingly, R should be chosen as large as possible depending on the available computing power.

5.5 Comparison of the introduced datasets

For the corpus of 7453 newspaper articles from the USA Today from 01/06 to 11/30/2016 and the *reuters* dataset with $M = 91$, an increase in reliability could clearly be shown. Lastly,

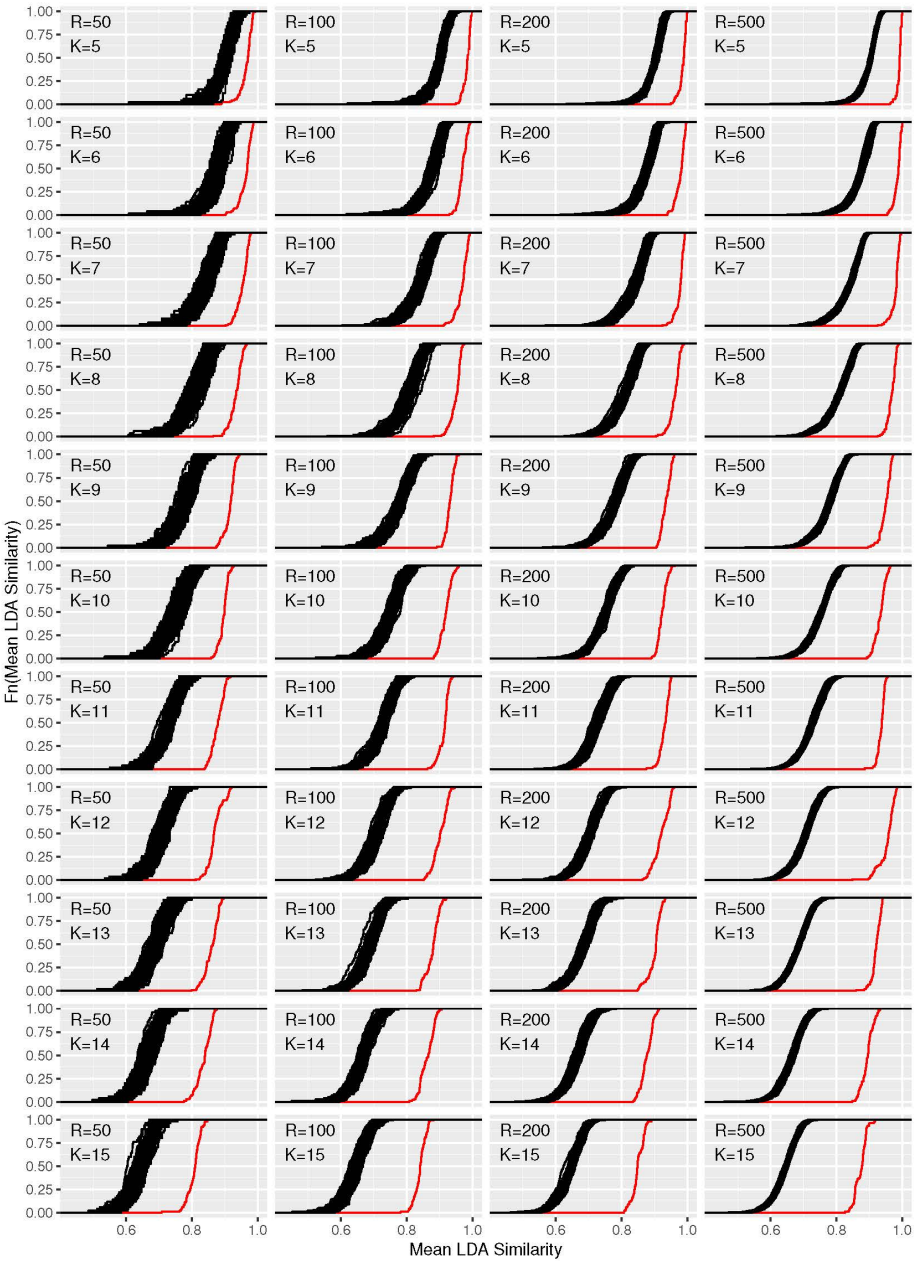

Fig. 6 Increase of reliability for the *reuters* dataset for $R = 50, 100, 200, 500$ and $K = 5, \ldots, 15$

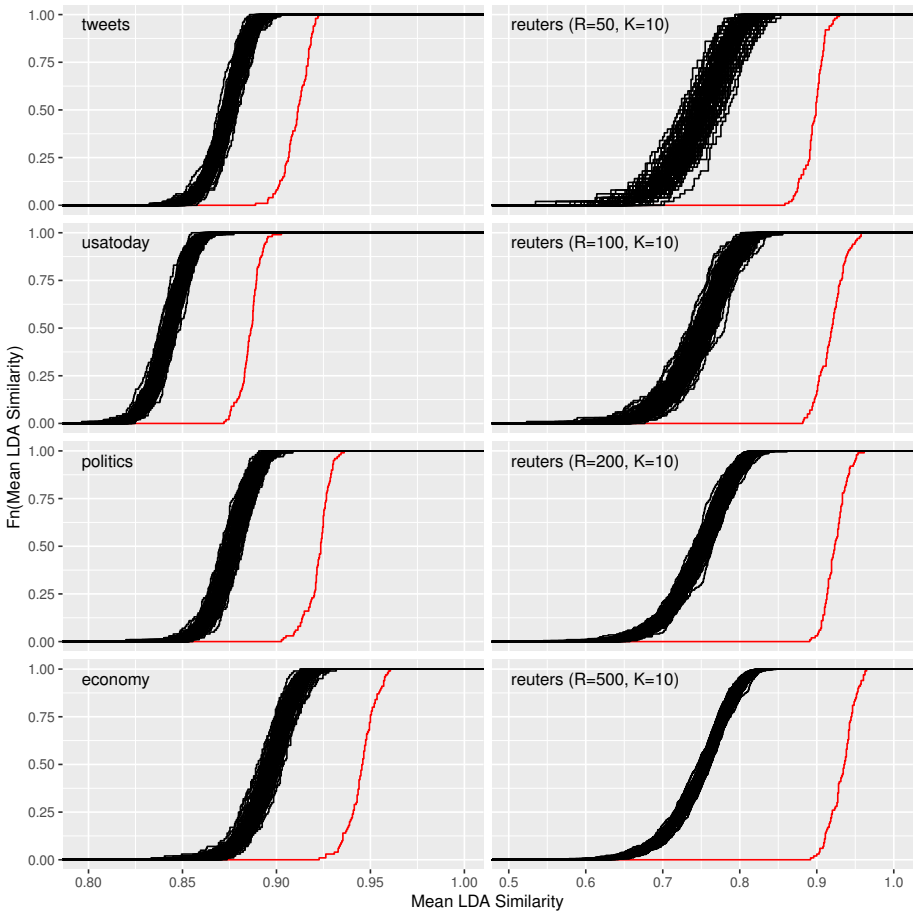

Fig. 7 Increase of reliability for the datasets *tweets*, *usatoday*, *politics*, and *economy* with $R = 100$ and for *reuters* with $R = 50, 100, 200, 500$

we show that this increase results independently of the dataset, so that the LDAPrototype method can be reasonably applied to many different types of text data.

Figure 7 in particular shows the increase in reliability for the datasets *tweets*, *politics*, and *economy* that have not yet been considered so far. For comparison, the corresponding plots for the *usatoday* dataset are shown, for *reuters* with $K = 10$ and the four different values for $R = 50, 100, 200, 500$ (cf. Section 5.4). Figure 1 already showed the computability for large datasets on the example of the *nyt* dataset. We did not repeat the computation of the LDAPrototype 100 times for this dataset due to the comparatively higher runtime. The *tweets* dataset, with 3 706 740, does consist of a larger number of individual documents than the *nyt* dataset (1 993 182, cf. Table 2). However, due to the more specific topic structure, we chose a smaller topic number for this dataset. Together with the fact that the modeled tweets are much shorter than journalistic texts from the New York Times, this leads to a significantly lower runtime of just over one day per LDAPrototype than for the *nyt* dataset with about 130 days (cf. Table 3).

The plots show a lower gain in reliability for *tweets* than for the dataset *economy*, for example. This might surprise because the similarities of the basic LDA repelications are already somewhat higher for the latter. To be concrete, the similarities increase from about 0.88 to 0.94 for *economy* and from 0.87 to 0.91 for the dataset consisting of tweets. This reduced gain could be due to the shorter texts and the resulting higher uncertainty in the modeling. As a result, the instability of the LDA on this dataset is more pronounced, so that higher reliability increases are only possible with larger values for R. On the right side of Figure 7 the results from the Sections 5.2 and 5.4 are recapitulated. The increase of reliability increases steadily with an increasing number of LDA replications R.

6 Discussion

Topic modeling is popular for understanding text data, though the analysis of the reliability of topic models is rarely part of applications. This is caused by plenty of possibilities for measuring reliability, but missing strategies for increasing reliability without touching the original fitting procedure.

We have presented a novel method to address the instability of the LDA procedure. For this purpose, we want to improve the reproducibility of the results and we talk about highly reliable results if they can be reproduced very well. The presented method is based on the idea of modeling a set of LDAs and then selecting the best, in this case the most central, model through a selection mechanism. We call this medoid of several LDA runs LDAPrototype. We deliberately choose not the best-fit model according to one of the well-known - mostly likelihood-based - measures (Griffiths and Steyvers, 2004; Grün and Hornik, 2011), but the LDA that agrees most with all other LDAs from the same set.

In various analyses, we have shown that the presented method increases the reliability of the results. By applying it to different datasets, we have first shown its feasibility due to the implemented R package and at the same time that it produces the desired increase in reliability independent of the dataset. We also investigated the influence of the parameters of the number of modeled topics K and number of LDA replications R on this increase. Furthermore, we presented differences in the determination of the LDAPrototype based on the presented similarity measures. All four measures under consideration resulted in an increase in reliability. No clear ranking could be obtained. The decision for the thresholded Jaccard (tJacc) coefficient as the default measure is based on the combination of visible increase in reliability, interpretability of the measure, fast implementation, and supporting arguments from studies (Aletras and Stevenson, 2014; Kim and Oh, 2011) regarding correlation with human perception.

The quality of the results of LDAPrototype in terms of correlation with human perception was not in the scope of the paper, but we plan to investigate it in further studies. For this, a study with human coders is necessary and different models like the basic LDA, LDAPrototype, Structural Topic Model and potentially other models are evaluated regarding their quality by human coders. For this purpose, we focus on meaningful and distinct topics.

In addition, it is of interest to generalize our novel S-CLOP measure to a similarity measure for models based on different text corpora. There are several difficulties to consider. For example, it is an open question how to deal with differences in the number of topics K of the compared models. Furthermore, it needs to be analyzed whether a comparison of a number of runs per corpus or a comparison of LDAPrototypes is more practical. Such similarity measures based on S-CLOP can open several other application fields. For example,

similarities in newspaper coverage or differences in coverage on different media channels such as Twitter, online and print could be quantified.

The presented idea of selecting a prototypical LDA from a set of LDA runs can be transferred to other topic models as well. For example, the Structural Topic Model offers not only pre-initialized topics but also the possibility of random initialization. This causes the issue of limited reliability of interpretations due to the lack of reproducibility of the results. At this point, the reliability may also increase using an analogous procedure to the method LDAPrototype.

Acknowledgements The present study is part of a project of the Dortmund Center for data-based Media Analysis (DoCMA). In addition, the authors gratefully acknowledge the computing time provided on the Linux HPC cluster at TU Dortmund University (LiDO3), partially funded in the course of the Large-Scale Equipment Initiative by the German Research Foundation (DFG) as project 271512359.

Conflict of interest

The authors declare that they have no conflict of interest.

References

Agrawal A, Fu W, Menzies T (2018) What is wrong with topic modeling? And how to fix it using search-based software engineering. Information and Software Technology 98:74–88, DOI 10.1016/j.infsof.2018.02.005

Aletras N, Stevenson M (2014) Measuring the Similarity between Automatically Generated Topics. In: Proceedings of the 14th EACL-Conference, Volume 2: Short Papers, ACL, pp 22–27, URL `http://www.aclweb.org/anthology/E14-4005`

Benoit K, Watanabe K, Wang H, Nulty P, Obeng A, Müller S, Matsuo A (2018) quanteda: An R package for the quantitative analysis of textual data. Journal of Open Source Software 3(30):774, DOI 10.21105/joss.00774

Bischl B, Lang M, Schratz P (2020) parallelMap: Unified Interface to Parallelization Back-Ends. URL `https://CRAN.R-project.org/package=parallelMap`, R package version 1.5.0

Blei DM (2012) Probabilistic Topic Models. Communications of the ACM 55(4):77–84, DOI 10.1145/2133806.2133826

Blei DM, Lafferty JD (2007) A Correlated Topic Model of Science. The Annals of Applied Statistics 1(1):17–35, DOI 10.1214/07-AOAS114

Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet Allocation. Journal of Machine Learning Research 3:993–1022, DOI 10.1162/jmlr.2003.3.4-5.993

Chang J (2015) lda: Collapsed Gibbs Sampling Methods for Topic Models. URL `https://CRAN.R-project.org/package=lda`, R package version 1.4.2

Chang J, Boyd-Graber J, Gerrish S, Wang C, Blei DM (2009) Reading Tea Leaves: How Humans Interpret Topic Models. In: NIPS: Advances in Neural Information Processing Systems, Curran Associates Inc., pp 288–296, URL `https://papers.nips.cc/paper/2009/hash/f92586a25bb3145facd64ab20fd554ff-Abstract.html`

Feinerer I, Hornik K, Meyer D (2008) Text Mining Infrastructure in R. Journal of Statistical Software 25(5):1–54, DOI 10.18637/jss.v025.i05

Greene D, O'Callaghan D, Cunningham P (2014) How Many Topics? Stability Analysis for Topic Models. In: ECML PKDD: Machine Learning and Knowledge Discovery in Databases, Springer, LNCS, vol 8724, pp 498–513, DOI 10.1007/978-3-662-44848-9_32

Griffiths TL, Steyvers M (2004) Finding scientific topics. Proceedings of the National Academy of Sciences 101(suppl 1):5228–5235, DOI 10.1073/pnas.0307752101

Grün B, Hornik K (2011) topicmodels: An R Package for Fitting Topic Models. Journal of Statistical Software 40(13):1–30, DOI 10.18637/jss.v040.i13

Hastie T, Tibshirani R, Friedman J (2009) The Elements of Statistical Learning: Data Mining, Inference and Prediction, 2nd edn. Springer Series in Statistics, Springer

Hofmann T (1999) Probabilistic Latent Semantic Indexing. In: Proceedings of the 22nd International SIGIR-Conference, ACM, pp 50–57, DOI 10.1145/312624.312649

Jaccard P (1912) The distribution of the flora in the alpine zone. New Phytologist 11(2):37–50, DOI 10.1111/j.1469-8137.1912.tb05611.x

Kim D, Oh A (2011) Topic Chains for Understanding a News Corpus. In: Gelbukh A (ed) Computational Linguistics and Intelligent Text Processing, Springer, pp 163–176

Koltcov S, Nikolenko SI, Koltsova O, Filippov V, Bodrunova S (2016) Stable Topic Modeling with Local Density Regularization. In: Internet Science, Springer, LNCS, vol 9934, pp 176–188, DOI 10.1007/978-3-319-45982-0_16

Koppers L, Rieger J, Boczek K, von Nordheim G (2020) tosca: Tools for Statistical Content Analysis. DOI 10.5281/zenodo.3591068, R package version 0.2-0

Kuhn HW (1955) The Hungarian method for the assignment problem. Naval Research Logistics Quarterly 2(1-2):83–97, DOI 10.1002/nav.3800020109

Kullback S, Leibler RA (1951) On Information and Sufficiency. The Annals of Mathematical Statistics 22(1):79–86, DOI 10.1214/aoms/1177729694

Lang M, Bischl B, Surmann D (2017) batchtools: Tools for R to work on batch systems. The Journal of Open Source Software 2(10), DOI 10.21105/joss.00135

LexisNexis (2019) Nexis: LexisNexis Academic & Library Solutions. URL: `https://www.nexis.com` and `https://www.lexisnexis.de/`

Lin J (1991) Divergence measures based on the Shannon entropy. IEEE Transactions on Information Theory 37(1):145–151, DOI 10.1109/18.61115

Maier D, Waldherr A, Miltner P, Wiedemann G, Niekler A, Keinert A, Pfetsch B, Heyer G, Reber U, Häussler T, Schmid-Petri H, Adam S (2018) Applying LDA Topic Modeling in Communication Research: Toward a Valid and Reliable Methodology. Communication Methods and Measures 12(2-3):93–118, DOI 10.1080/19312458.2018.1430754

Mantyla MV, Claes M, Farooq U (2018) Measuring LDA Topic Stability from Clusters of Replicated Runs. In: Proceedings of the 12th ACM/IEEE International ESEM-Symposium, ACM, DOI 10.1145/3239235.3267435

Mimno D, Wallach HM, Talley E, Leenders M, McCallum A (2011) Optimizing Semantic Coherence in Topic Models. In: Proceedings of the 2011 EMNLP-Conference, ACL, pp 262–272

Newman D, Bonilla EV, Buntine W (2011) Improving Topic Coherence with Regularized Topic Models. In: NIPS: Advances in Neural Information Processing Systems, Curran Associates Inc., pp 496–504, URL `https://proceedings.neurips.cc/paper/2011/hash/5ef698cd9fe650923ea331c15af3b160-Abstract.html`

Nguyen VA, Boyd-Graber J, Resnik P (2014) Sometimes Average is Best: The Importance of Averaging for Prediction using MCMC Inference in Topic Modeling. In: Proceedings of the 2014 EMNLP-Conference, ACL, pp 1752–1757, DOI 10.3115/v1/D14-1182

R Core Team (2020) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, URL `https://www.R-project.org/`

Rieger J (2020) ldaPrototype: A method in R to get a Prototype of multiple Latent Dirichlet Allocations. Journal of Open Source Software 5(51):2181, DOI 10.21105/joss.02181

Rieger J, von Nordheim G (2021) corona100d – German-language Twitter dataset of the first 100 days after Chancellor Merkel addressed the Coronavirus outbreak in TV. DoCMA Working Paper #4, DOI TBA

Roberts ME, Stewart BM, Tingley D, Airoldi EM (2013) The Structural Topic Model and Applied Social Science. In: NIPS-Workshop on Topic Models: Computation, Application, and Evaluation

Roberts ME, Stewart BM, Tingley D (2019) stm: An R Package for Structural Topic Models. Journal of Statistical Software 91(2):1–40, DOI 10.18637/jss.v091.i02

Rosen-Zvi M, Griffiths T, Steyvers M, Smyth P (2004) The Author-Topic Model for Authors and Documents. In: Proceedings of the 20th UAI-Conference, AUAI Press, pp 487–494

Stevens K, Kegelmeyer P, Andrzejewski D, Buttler D (2012) Exploring Topic Coherence over Many Models and Many Topics. In: Proceedings of the 2012 Joint EMNLP/CoNLL-Conference, ACL, pp 952–961

Su J, Greene D, Boydell O (2016) Topic Stability over Noisy Sources. In: Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT), COLING, pp 85–93, URL `http://aclweb.org/anthology/W16-3913`

Wang C, Blei DM, Heckerman D (2008) Continuous Time Dynamic Topic Models. In: Proceedings of the 24th UAI-Conference, pp 579–586

Webber W, Moffat A, Zobel J (2010) A Similarity Measure for Indefinite Rankings. ACM Transactions on Information Systems 28(4):20:1–20:38, DOI 10.1145/1852102.1852106