Sweeping through the Topic Space: Bad luck? Roll again!

Martin Riedl and **Chris Biemann**
Ubiquitous Knowledge Processing Lab
Computer Science Department, Technische Universität Darmstadt
Hochschulstrasse 10, D-64289 Darmstadt, Germany
riedl@ukp.informatik.tu-darmstadt.de, biem@cs.tu-darmstadt.de

Abstract

Topic Models (TM) such as Latent Dirichlet Allocation (LDA) are increasingly used in Natural Language Processing applications. At this, the model parameters and the influence of randomized sampling and inference are rarely examined — usually, the recommendations from the original papers are adopted. In this paper, we examine the parameter space of LDA topic models with respect to the application of Text Segmentation (TS), specifically targeting error rates and their variance across different runs. We find that the recommended settings result in error rates far from optimal for our application. We show substantial variance in the results for different runs of model estimation and inference, and give recommendations for increasing the robustness and stability of topic models. Running the inference step several times and selecting the last topic ID assigned per token, shows considerable improvements. Similar improvements are achieved with the mode method: We store all assigned topic IDs during each inference iteration step and select the most frequent topic ID assigned to each word. These recommendations do not only apply to TS, but are generic enough to transfer to other applications.

1 Introduction

With the rise of topic models such as pLSI (Hofmann, 2001) or LDA (Blei et al., 2003) in Natural Language Processing (NLP), an increasing number of works in the field use topic models to map terms from a high-dimensional word space to a lower-dimensional semantic space. TMs are 'the new Latent Semantic Analysis' (LSA), (Deerwester et al., 1990), and it has been shown that generative models like pLSI and LDA not only have a better mathematical foundation rooted in probability theory, but also outperform LSA in document retrieval and classification, e.g. (Hofmann, 2001; Blei et al., 2003; Biro et al., 2008). To estimate the model parameters in LDA, the exact computation that was straightforward in LSA (matrix factorization) is replaced by a randomized Monte-Carlo sampling procedure (e.g. variational Bayes or Gibbs sampling).

Aside from the main parameter, the number of topics or dimensions, surprisingly little attention has been spent to understand the interactions of hyperparameters, the number of sampling iterations in model estimation and interference, and the stability of topic assignments across runs using different random seeds. While progress in the field of topic modeling is mainly made by adjusting prior distributions (e.g. (Sato and Nakagawa, 2010; Wallach et al., 2009)), or defining more complex model mixtures (Heinrich, 2011), it seems unclear whether improvements, reached on intrinsic measures like perplexity or on application-based evaluations, are due to an improved model structure or could originate from sub-optimal parameter settings or literally 'bad luck' due to the randomized nature of the sampling process.

In this paper, we address these issues by systematically sweeping the parameter space. For this, we pick LDA since it is the most commonly used TM in the field of NLP. To evaluate the contribution of the TM, we choose the task of TS: this task has received considerable interest from the NLP community, standard datasets and evaluation measures are available for testing, and it

Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pages 19–27,
Avignon, France, April 23 - 27 2012.

has been shown that this task considerably benefits from the use of TMs, see (Misra et al., 2009; Sun et al., 2008; Eisenstein, 2009).

This paper is organized as follows: In the next section, we present related work regarding text segmentation using topic models and topic model parameter evaluations. Section 3 defines the TopicTiling text segmentation algorithm, which is a simplified version of TextTiling (Hearst, 1994), and makes direct use of topic assignments. Its simplicity allows us to observe direct consequences of LDA parameter settings. Further, we describe the experimental setup, our application-based evaluation methodology including the data set and the LDA parameters we vary in Section 4.

Results of our experiments in Section 5 indicate that a) there is an optimal range for the number of topics, b) there is considerable variance in performance for different runs for both model estimation and inference, c) increasing the number of sampling iterations stabilizes average performance but does not make TMs more robust, but d) combining the output of several independent sampling runs does, and additionally leads to large error rate reductions. Similar results are obtained by e) the mode method with less computational costs using the most frequent topic ID that is assigned during different inference iteration steps. In the conclusion, we give recommendations to add stability and robustness for TMs: aside from optimization of the hyperparameters, we recommend combining the topic assignments of different inference iterations, and/or of different independent inference runs.

2 Related Work

2.1 Text Segmentation with Topic Models

Based on the observation of Halliday and Hasan (1976) that the density of coherence relations is higher within segments than between segments, most algorithms compute a coherence score to measure the difference of textual units for informing a segmentation decision. TextTiling (Hearst, 1994) relies on the simplest coherence relation – word repetition – and computes similarities between textual units based on the similarities of word space vectors. The task of text segmentation is to decide, for a given text, how to split this text into segments.

Related to our algorithm (see Section 3.1) are the approaches described in Misra et al. (2009) and Sun et al. (2008): topic modeling is used to alleviate the sparsity of word vectors by mapping words into a topic space. This is done by extending the dynamic programming algorithms from (Utiyama and Isahara, 2000; Fragkou et al., 2004) using topic models. At this, the topic assignments have to be inferred for each possible segment.

2.2 LDA and Topic Model Evaluation

For topic modeling, we use the widely applied LDA (Blei et al., 2003), This model uses a training corpus of documents to create document-topic and topic-word distributions and is parameterized by the number of topics T as well as by two hyperparameters. To generate a document, the topic proportions are drawn using a Dirichlet distribution with hyperparameter α. Adjacent for each word w a topic z_{d_w} is chosen according to a multinomial distribution using hyperparameter $\beta_{z_{d_w}}$. The model is estimated using m iterations of Gibbs sampling. Unseen documents can be annotated with an existing topic model using Bayesian inference methods. At this, Gibbs sampling with i iterations is used to estimate the topic ID for each word, given the topics of the other words in the same sentential unit. After inference, every word in every sentence receives a topic ID, which is the sole information that is used by the TopicTiling algorithm to determine the segmentation. We use the GibbsLDA implementation by Phan and Nguyen (2007) for all our experiments.

The article of Blei et al. (2003) compares LDA with pLSI and Mixture Unigram models using the perplexity of the model. In a collaborative filtering evaluation for different numbers of topics they observe that using too many topics leads to overfitting and to worse results.

In the field of topic model evaluations, Griffiths and Steyvers (2004) use a corpus of abstracts published between 1991 and 2001 and evaluate model perplexity. For this particular corpus, they achieve the lowest perplexity using 300 topics. Furthermore, they compare different sampling methods and show that the perplexity converges faster with Gibbs sampling than with expectation propagation and variational Bayes. On a small artificial testset, small variations in perplexity across different runs were observed in early sampling iterations, but all runs converged to the same limit.

In Wallach et al. (2009) topic models are evaluated with symmetric and asymmetric hyperparameters based on the perplexity. They observe a benefit using asymmetric parameters for α, but cannot show improvement with asymmetric priors for β.

3 Method

3.1 TopicTiling

For the evaluation of the topic models, a text segmentation algorithm called TopicTiling is used here. This algorithm is a newly developed algorithm based on TextTiling (Hearst, 1994) and achieves state of the art results using the Choi dataset, which is a standard dataset for TS evaluation. The algorithm uses sentences as minimal units. Instead of words, we use topic IDs that are assigned to each word using the LDA inference running on sentence units. The LDA model should be estimated on a corpus of documents that is similar to the to-be-segmented documents.

To measure the coherence c_p between two sentences around position p, the cosine similarity (vector dot product) between these two adjacent sentences is computed. Each sentence is represented as a T-dimensional vector, where T is the number of topic IDs defined in the topic model. The t-th element of the vector contains the number of times the t-th topic is observed in the sentence. Similar to the TextTiling algorithm, local minima calculated from these similarity scores are taken as segmentation candidates.

This is illustrated in Figure 1, where the similarity scores between adjacent sentences are plotted. The vertical lines in this plot indicate all local minima found.

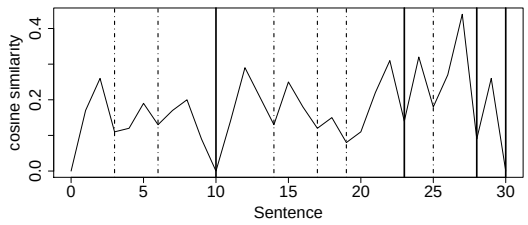

Figure 1: Cosine similarity scores of adjacent sentences based on topic distribution vectors. Vertical lines (solid and dashed) indicate local minima. Solid lines mark segments that have a depth score above a chosen threshold.

Following the TextTiling definition, not the minimum score c_p at position p itself is used, but a depth score d_p for position p computed by

$$d_i = 1/2 * (c_{p-1} - c_p + c_{p+1} - c_p). \quad (1)$$

In contrast to TextTiling, the directly neighboring similarity scores of the local minima are used, if they are higher than c_p. When using topics instead of words, it can be expected that sentences within one segment have many topics in common, which leads to cosine similarities close to 1. Further, using topic IDs instead of words greatly increases sparsity. A minimum in the curve indicates a change in topic distribution. Segment boundaries are set at the positions of the n highest depth-scores, which is common practice in text segmentation algorithms. An alternative to a given n would be the selection of segments according to a depth score threshold.

4 Experimental Setup

As dataset the Choi dataset (Choi, 2000) is used. This dataset is an artificially generated corpus that consists of 700 documents. Each document consists of 10 segments and each segment has 3–11 sentences extracted from a document of the Brown corpus. For the first setup, we perform a 10-fold Cross Validation (CV) for estimating the TM (estimating on 630 documents at a time), for the other setups we use 600 documents for TM estimation and the remaining 100 documents for testing. While we aim to neglect using the same documents for training and testing, it is not guaranteed that all testing data is unseen, since the same source sentences can find their way in several artificially crafted 'documents'. This problem, however, applies for all evaluations on this dataset that use any kind of training, be it LDA models in Misra et al. (2009) or TF-IDF values in Fragkou et al. (2004).

For the evaluation of the Topic Model in combination of Text Segmentation, we use the P_k measure (Beeferman et al., 1999), which is a standard measure for error rates in the field of TS. This measure compares the gold standard segmentation with the output of the algorithm. A P_k value of 0 indicates a perfect segmentation, the averaged state of the art on the Choi Dataset is $P_k = 0.0275$ (Misra et al., 2009). To assess the robustness of the TM, we sweep over varying

configurations of the LDA model, and plot the results using Box-and-Whiskers plots: the box indicates the quartiles and the whiskers are maximal 1.5 times of the Interquartile Range (IQR) or equal to the data point that is no greater to the 1.5 IQR. The following parameters are subject to our exploration:

- T: Number of topics used in the LDA model. Common values vary between 50 and 500.
- α : Hyperparameter that regulates the sparseness topic-per-document distribution. Lower values result in documents being represented by fewer topics (Heinrich, 2004). Recommended: $\alpha = 50/T$ (Griffiths and Steyvers, 2004)
- β : Reducing β increases the sparsity of topics, by assigning fewer terms to each topic, which is correlated to how related words need to be, to be assigned to a topic (Heinrich, 2004). Recommended: $\beta = \{0.1, 0.01\}$ (Griffiths and Steyvers, 2004; Misra et al., 2009)
- m Model estimation iterations. Recommended / common settings: $m = 500 - 5000$ (Griffiths and Steyvers, 2004; Wallach et al., 2009; Phan and Nguyen, 2007)
- i Inference iterations. Recommended / common settings: 100 (Phan and Nguyen, 2007)
- d Mode of topic assignments. At each inference iteration step, a topic ID is assigned to each word within a document (represented as a sentence in our application). With this option, we count these topic assignments for each single word in each iteration. After all i inference iterations, the most frequent topic ID is chosen for each word in a document.
- r Number of inference runs: We repeat the inference r times and assign the most frequently assigned topic per word at the final inference run for the segmentation algorithm. High r values might reduce fluctuations due to the randomized process and lead to a more stable word-to-topic assignment.

All introduced parameters parameterize the TM. We are not aware of any research that has used several inference runs r and the mode of topic assignments d to increase stability and varying TM parameters in combinations with measures other then perplexity.

5 Results

In this section, we present the results we obtained from varying the parameters under examination.

5.1 Number of Topics T

To provide a first impression of the data, a 10-fold CV is calculated and the segmentation results are visualized in Figure 2.

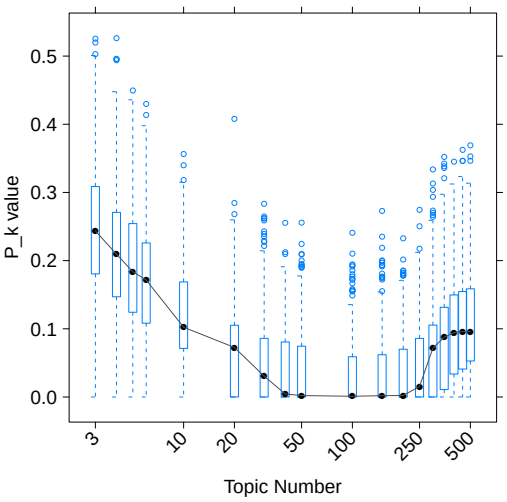

Figure 2: Box plots for different number of topics T. Each box plot is generated from the average P_k value of 700 documents, $\alpha = 50/T$, $\beta = 0.1$, $m = 1000$, $i = 100, r = 1$. These documents are segmented with TopicTiling using a 10-folded CV.

Each box plot is generated from the P_k values of 700 documents. As expected, there is a continuous range of topic numbers, namely between 50 and 150 topics, where we observe the lowest P_k values. Using too many topics leads to overfitting of the data and too few topics result in too general distinctions to grasp text segments. This is in line with other studies, that determine an optimum for T, cf. (Griffiths and Steyvers, 2004), which is specific to the application and the data set.

5.2 Estimation and Inference iterations

The next step examines the robustness of the topic model according to the number of model estimation iterations m needed to achieve stable results. 600 documents are used to train the LDA model

that is applied by TopicTiling to segment the remaining 100 documents. From Figure 2 we know that sampling 100 topics leads to good results. To have an insight into unstable topic regions we also inspect performance at different sampling iterations using 20 and 250 topics. To assess stability across different model estimation runs, we trained 30 LDA models using different random seeds. Each box plot in Figures 3 and 4 is generated from 30 mean values, calculated from the P_k values of the 100 documents. The variation indicates the score variance for the 30 different models.

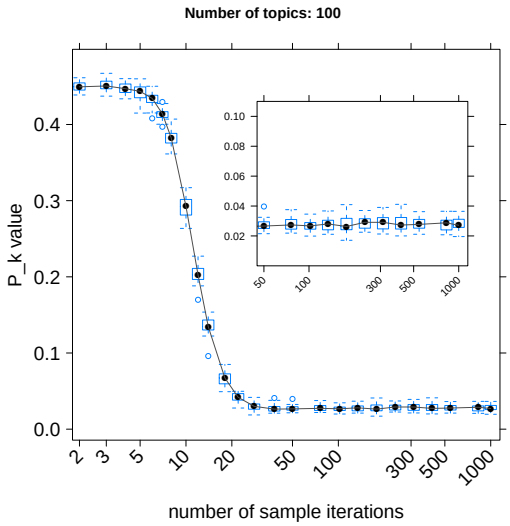

Figure 3: Box plots with different model estimation iterations m, with T=100, $\alpha = 50/T$, $\beta = 0.1$, $i = 100$, $r = 1$. Each box plot is generated from 30 mean values calculated from 100 documents.

Using 100 topics (see Figure 3), the burn-in phase starts with 8–10 iterations and the mean P_k values stabilize after 40 iterations. But looking at the inset for large m values, significant variations between the different models can be observed: note that the P_k error rates are almost double between the lower and the upper whisker. These remain constant and do not disappear for larger m values: The whiskers span error rates between 0.021 - 0.037 for model estimation on document units

With 20 topics, the P_k values are worse as with 100 topics, as expected from Figure 2. Here the convergence starts at 100 sample iterations. More interesting results are achieved with 250 topics. A robust range for the error rates can be found between 20 and 100 sample iterations. With more iterations m, the results get both worse and unstable: as the 'natural' topics of the collection have to be split in too many topics in the model, perplexity optimizations that drive the estimation process lead to random fluctuations, which the TopicTiling algorithm is sensitive to. Manual inspection of models for $T = 250$ revealed that in fact many topics do not stay stable across estimation iterations.

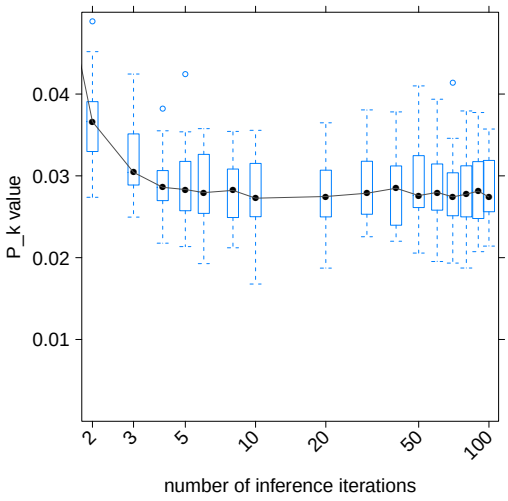

Figure 5: Figure of box plots for different inference iterations i and $m = 1000$, $T = 100$, $\alpha = 50/T$, $\beta = 0.1$, $r = 1$.

In the next step we sweep over several inference iterations i. Starting from 5 iterations, error rates do not change much, see Figure 5. But there is still substantial variance, between about 0.019 - 0.038 for inference on sentence units.

5.3 Number of inference runs r

To decrease this variance, we assign the topic not only from a singe inference run, but repeat the inference calculations several times, denoted by the parameter r. Then the frequency of assigned topic IDs per token is counted across the r runs, and we assign the most frequent topic ID (frequency ties are broken randomly). The box plot for several evaluated values of r is shown in Figure 6.

This log-scaled plot shows that both variance and P_k error rate can be substantially decreased. Already for $r = 3$, we observe a significant improvement in comparison to the default setting of $r = 1$ and with increasing r values, the error rates are reduced even more: for $r = 20$, variance and error rates are is cut in less than half of their original values using this simple operation.

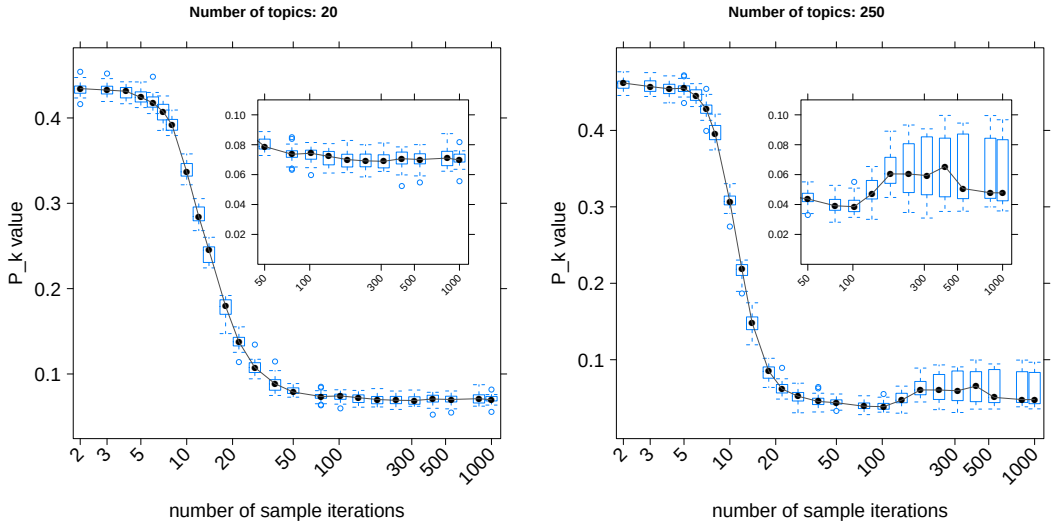

Figure 4: Box plots with varying model estimation iterations m applied with $T = 20$ (left) and $T = 250$ (right) topics, $\alpha = 50/T$, $\beta = 0.1$, $i = 100$, $r = 1$

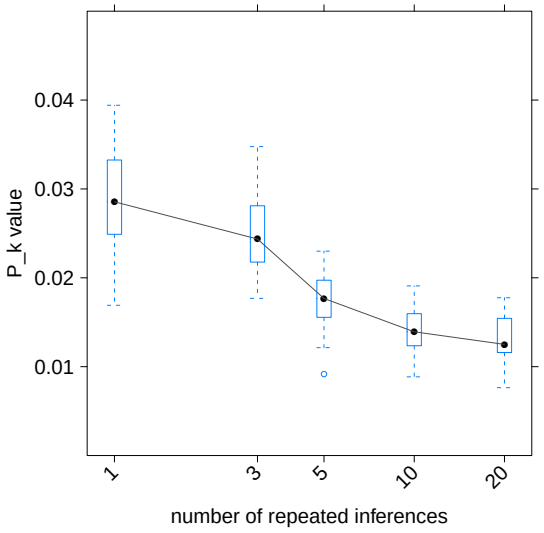

Figure 6: Box plot for several inference runs r, to assign the topics to a word with $m = 1000$, $i = 100$, $T = 100$, $\alpha = 50/T$, $\beta = 0.1$.

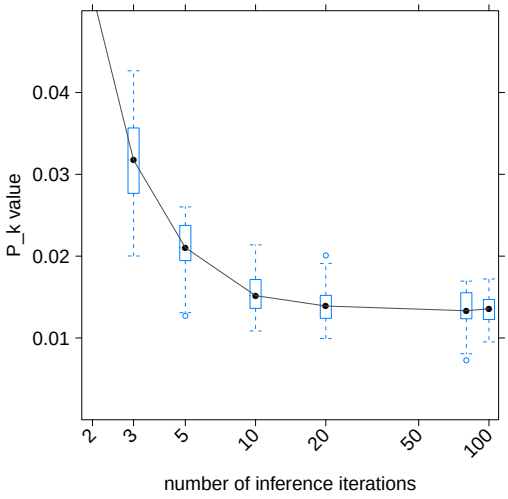

Figure 7: Box plot using the mode method $d = true$ with several inference iterations i with $m = 500$, $T = 100$, $\alpha = 50/T$, $\beta = 0.1$.

5.4 Mode of topic assignment d

In the previous experiment, we use the topic IDs that have been assigned most frequently at the last inference iteration step. Now, we examine something similar, but for all i inference steps of a single inference run: we select the mode of topic ID assignments for each word across all inference steps. The impact of this method on error and variance is illustrated in Figure 7. Using a single inference iteration, the topic IDs are almost assigned randomly. After 20 inference iterations P_k values below 0.02 are achieved. Using further iterations, the decrease of the error rate is only marginal. In comparison to the repeated inference method, the additional computational costs of this method are much lower as the inference iterations have to be carried out anyway in the default application setting.

5.5 Hyperparameters α and β

In many previous works, hyperparameter settings $\alpha = 50/T$ and $\beta = \{0.1, 0.01\}$ are commonly used. In the next series of experiments we investigate how different parameters of these both parameters can change the TS task.

For α values, shown in Figure 8, we can see that the recommended value for $T = 100$, $\alpha =$

0.5 leads to sub-optimal results, and an error rate reduction of about 40% can be realized by setting $\alpha = 0.1$.

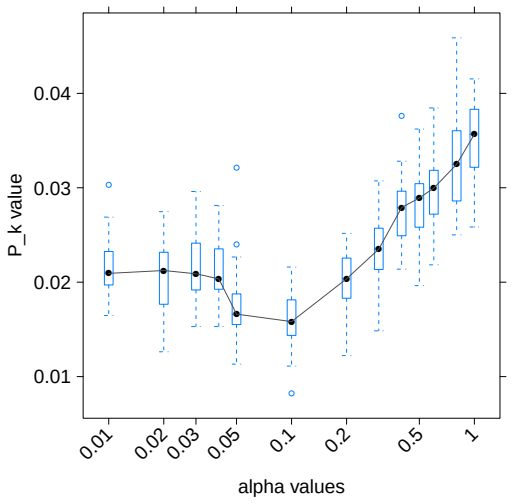

Figure 8: Box plot for several alpha values α with $m = 500$, $i = 100$, $T = 100$, $\beta = 0.1$, $r = 1$.

Regarding values of β, we find that P_k rates and their variance are relatively stable between the recommended settings of 0.1 and 0.01. Values larger than 0.1 lead to much worse performance. Regarding variance, no patterns within the stable range emerge, see Figure 9.

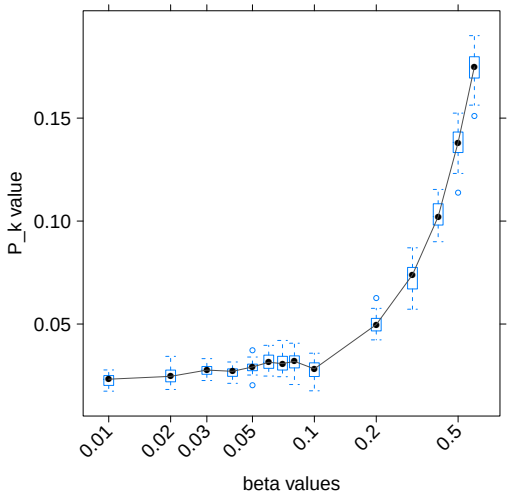

Figure 9: Box plot for several beta values β with $m = 500$, $i = 100$, $T = 100$, $\alpha = 50/T$, $r = 1$.

5.6 Putting it all together

Until this point, we have examined different parameters with respect to stability and error rates one at the time. Now, we combine what we have learned from this and strive at optimal system performance. For this, we contrast TS results obtained with the default LDA configuration with the best systems obtained by optimization of single parameters, as well as to a system that uses these optimal settings for all parameters. Table 1 shows P_k error rates for the different systems. At this, we fixed the following parameters: $T = 100$, $m = 500$, $i = 100$, $\beta = 0.1$. For the computations we use 600 documents for the LDA model estimation, apply TopicTiling and compute the error rate for the 100 remaining documents and repeat this 30 times with different random seeds.

System	P_k	error red.	σ^2	var. red.
default	0.0302	0.00%	2.02e-5	0.00%
$\alpha = 0.1$	0.0183	39.53%	1.22e-5	39.77%
$r = 20$	0.0127	57.86%	4.65e-6	76.97%
$d = true$	0.0137	54.62%	3.99e-6	80.21%
combined	0.0141	53.45%	9.17e-6	54.55%

Table 1: Comparison of single parameter optimizations, and combined system. P_k averages and variance are computed over 30 runs, together with reductions relative to the default setting. Default: $\alpha = 0.5$, $r = 1$. combined: $\alpha = 0.1$, $r = 20$, $d = true$

We can observe a massive improvement for optimized single parameters. The α-tuning tuning results in an error rate reduction of 39.77% in comparison to the default configurations. Using $r = 20$, the error rate is cut in less than half its original value. Also for the mode mechanism ($d = true$) the error rate is halved but slightly worse than than when using the repeated inference. Using combined optimized parameters does not result to additional error decreases. We attribute the slight decline of the combined method in both in the error rate P_k and in the variance to complex parameter interactions that shall be examined in further work. In Figure 10, we visualize these results in a density plot. It becomes clear that repeated inference leads to slightly better and more robust performance (higher peak) than the mode method. We attribute the difference to situations, where there are several highly probable topics in our sampling units, and by chance the same one is picked for adjacent sentences that belong to different segments, resulting in failure to recognize the segmentation point. However, since the differences are miniscule, only using the mode method might be more suitable for practical purposes since its computational cost is lower.

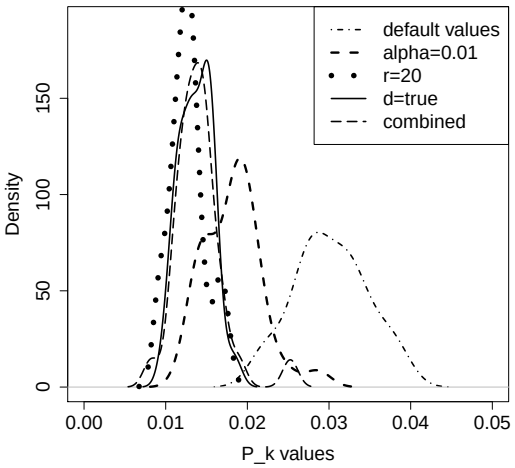

Figure 10: Density plot of the error distributions for the systems listed in Table 1

6 Conclusion

In this paper, we examined the robustness of LDA topic models with respect to the application of Text Segmentation by sweeping through the topic model parameter space. To our knowledge, this is the first attempt to systematically assess the stability of topic models in a NLP task.

The results of our experiments are summarized as follows:

- Perform the inference r times using the same model and choosing the assigned topic ID per word token taken from the last inference iteration, improves both error rates and stability across runs with different random seeds.
- Almost equal performance in terms of error and stability is achieved with the mode mechanism: choose the most frequent topic ID assignment per word across inference steps. While error rates were slightly higher for our data set, this method is probably preferable in practice because of its lower computation costs.
- As found in other studies, there is a range for the number of topics T, where optimal results are obtained. In our task, performance showed to be robust in the range of 50 - 150 topics.
- The default setting for LDA hyperparameters α and β can lead to sub-optimal results. Especially α should be optimized for the task at hand, as the utility of the topic model is very sensitive to this parameter.
- While the number of iterations for model estimation and inference needed for convergence is depending on the number of topics, the size of the sampling unit (document) and the collection, it should be noted that after convergence the variance between different sampling runs does not decrease for a larger number of iterations.

Equipped with the insights gained from experiments on single parameter variation, we were able to implement a very simple algorithm for text segmentation that improves over the state of the art on a standard dataset by a large margin. At this, the combination of the optimal α, and a high number of inference repetitions r and the mode method ($d = true$) produced slightly more errors than a high r alone. While the purpose of this paper was mainly to address robustness and stability issues of topic models, we are planning to apply the segmentation algorithm to further datasets.

The most important takeaway, however, is that especially for small sampling units like sentences, tremendous improvements in applications can be obtained when looking at multiple inference assignments and using the most frequently assigned topic ID in subsequent processing – either across diffeent inference steps or across diffeent inference runs. These two new strategies seem to be able to offset sub-optimal hyperparameters to a certain extent. This scheme is not only applicable to Text Segmentation, but in all applications where performance crucially depends on stable topic ID assignments per token. Extensions to this scheme, like ignoring tokens with a high topic variability (stop words or general terms) or dynamically deciding to conflate several topics because of their per-token co-occurrence, are left for future work.

7 Acknowledgments

This work has been supported by the Hessian research excellence program “Landes-Offensive zur Entwicklung Wissenschaftlich-konomischer Exzellenz” (LOEWE) as part of the research center “Digital Humanities”. We would also thank the anonymous reviewers for their comments, which greatly helped to improve the paper.

References

D. Beeferman, A. Berger, and J. Lafferty. 1999. Statistical models for text segmentation. *Machine learning*, 34(1):177–210.

Istvan Biro, Andras Benczur, Jacint Szabo, and Ana Maguitman. 2008. A comparative analysis of latent variable models for web page classification. In *Proceedings of the 2008 Latin American Web Conference*, pages 23–28, Washington, DC, USA. IEEE Computer Society.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March.

Freddy Y. Y. Choi. 2000. Advances in domain independent linear text segmentation. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, NAACL 2000, pages 26–33, Stroudsburg, PA, USA. Association for Computational Linguistics.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Jacob Eisenstein. 2009. Hierarchical text segmentation from multi-scale lexical cohesion. *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics on - NAACL '09*, page 353.

P. Fragkou, V. Petridis, and Ath. Kehagias. 2004. A Dynamic Programming Algorithm for Linear Text Segmentation. *Journal of Intelligent Information Systems*, 23(2):179–197, September.

Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *PNAS*, 101(suppl. 1):5228–5235.

M A K Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*, volume 1 of *English Language Series*. Longman.

Marti a. Hearst. 1994. Multi-paragraph segmentation of expository text. *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, (Hearst):9–16.

Gregor Heinrich. 2004. Parameter estimation for text analysis. Technical report.

Gregor Heinrich. 2011. Typology of mixed-membership models: Towards a design method. In *Machine Learning and Knowledge Discovery in Databases*, volume 6912 of *Lecture Notes in Computer Science*, pages 32–47. Springer Berlin / Heidelberg. 10.1007/978-3-642-23783-6 3.

Thomas Hofmann. 2001. Unsupervised Learning by Probabilistic Latent Semantic Analysis. *Computer*, pages 177–196.

Hemant Misra, Joemon M Jose, and Olivier Cappé. 2009. Text Segmentation via Topic Modeling : An Analytical Study. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management - CIKM '09*, pages 1553—-1556.

Xuan-Hieu Phan and Cam-Tu Nguyen. 2007. GibbsLDA++: A C/C++ implementation of latent Dirichlet allocation (LDA). http://jgibblda.sourceforge.net/.

Issei Sato and Hiroshi Nakagawa. 2010. Topic models with power-law using pitman-yor process categories and subject descriptors. *Science And Technology*, (1):673–681.

Qi Sun, Runxin Li, Dingsheng Luo, and Xihong Wu. 2008. Text segmentation with LDA-based Fisher kernel. *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies Short Papers - HLT '08*, (June):269.

Masao Utiyama and Hitoshi Isahara. 2000. A Statistical Model for Domain-Independent Text Segmentation. *Communications*.

Hanna Wallach, David Mimno, and Andrew McCallum. 2009. Rethinking lda: Why priors matter. In *NIPS*.