A Study on Topic Modeling for Feature Space Reduction in Text Classification

Daniel Pfeifer[1(✉)] and Jochen L. Leidner[2,3]

[1] Department of Medical Informatics, Heilbronn University of Applied Sciences, Max-Planck-Str. 39, 74081 Heilbronn, Germany
daniel.pfeifer@hs-heilbronn.de
[2] Refinitiv Labs, 30 South Colonnade, London E14 5EP, UK
leidner@acm.org
[3] University of Sheffield, 211 Portobello, Sheffield S1 4DP, UK

Abstract. We examine two topic modeling approaches as feature space reduction techniques for text classification and compare their performance with two standard feature selection techniques, namely Information Gain (IG) and and Document Frequency (DF). Feature selection techniques are commonly applied in order to avoid the well-known "curse of dimensionality" in machine learning. Regarding text classification, traditional techniques achieve this by selecting words from the training vocabulary. In contrast, topic models compute topics as multinomial distributions over words and reduce each document to a distribution over such topics. Corresponding topic-to-document distributions may act as input data to train a document classifier. Our comparison includes two topic modeling approaches – Latent Dirichlet Allocation (LDA) and Topic Grouper. Our results are based on classification accuracy and suggest that topic modeling is far superior to IG and DF at a very low number of reduced features. However, if the number of reduced features is still large, IG becomes competitive and the cost of computing topic models is considerable. We conclude by giving basic recommendations on when to consider which type of method.

Keywords: Topic modeling · Text classification · Feature selection · Feature space reduction

1 Introduction

Feature space reduction is a common step for text classification in order to avoid the well-known "curse of dimensionality" ([3]). Standard approaches achieve this by reducing the *training vocabulary* V: E.g., [19] compare five respective techniques including Information Gain (IG), Document Frequency (DF) and Chi-Square, where all three performed well. [7] is a more extensive study comprising additional word selection techniques and over 200 datasets.

Over the last two decades, probabilistic topic modeling has become an active sub-field of information retrieval and machine learning. Hereby, each topic $t \in T$ is typically represented via a multinomial distribution $p(w|t)$ with $w \in V$ where

A. Cuzzocrea et al. (Eds.): FQAS 2019, LNAI 11529, pp. 403–412, 2019.
https://doi.org/10.1007/978-3-030-27629-4_37

the set of distributions $\Phi = \{p(w|t) \mid w \in V, t \in T\}$ forms the actual *topic model*. Related ideas and solutions were formed in the two seminal publications on *probabilistic Latent Semantic Indexing* (pLSI) ([9]) and *Latent Dirichlet Allocation* (LDA) ([5]). Related models are learned from a training collection D of documents d based on the frequency $f_d(w)$ of each word w per document d. To date, LDA is probably the most commonly used topic modeling approach, and it produces a fixed-size set of non-hierarchical topics (cf. [4] for a general introduction to topic modeling and LDA).

We can apply topic modeling as a feature space reduction method as follows: First, the topic model Φ must be learned from the training collection D. Using Φ, a document d can be characterized by multinomial distributions $p(t|d)$ expressing the prevalence of each topic t in d. Regarding training documents, $p(t|d)$ is an additional output of the topic model's learning procedure. Computing $p(t|d)$ for a test document is called a "fold-in" and will be detailed in Sect. 3. Instead of using word frequencies, a classifier may then be trained and tested via related distributions $p(t|d)_{t \in T}$. Since the number of topics $|T|$ is usually much smaller than the size of the training vocabulary $|V|$, this results in a feature space reduction for the classifier. Figure 1 contrasts the standard word selection and embedding approach with a corresponding approach based on topic modeling.

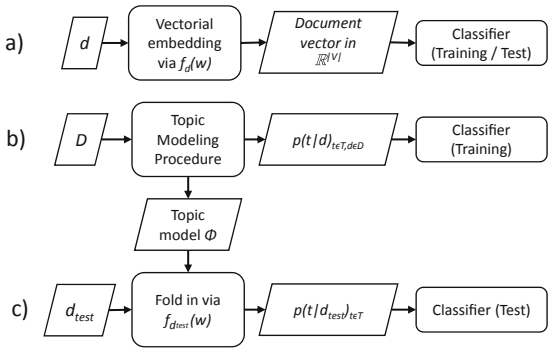

Fig. 1. (a) Standard word embedding for text classification with a potentially reduced vocabulary, (b) classification using topic modeling at training time and (c) classification using topic modeling at test time

To the best of our knowledge, our study is the first to explore *how the performance of topic model-based feature reduction for text classification relates to the number of features*. This covers the following aspects:

1. We describe in reasonable detail, how the steps (b) and (c) from Fig. 1 can be implemented, such that other machine learning practitioners can make best use of it.
2. Based on three datasets we run a direct comparison of feature reduction for text classification using word selection and topic modeling. We include IG, DF, two variants of LDA and Topic Grouper – a more recent topic modeling approach.
3. The results from (2) allow for conclusions and general recommendations on when to prefer which approach.

2 Related Work

We first give a brief overview of *Topic Grouper* (TG) from [15]. as it is incorporated in our study besides LDA: TG has no hyper parameters and partitions V via agglomerative clustering. In this case, a topic t consists of a set of words from V such that $\bigcup_{t \in T} t = V$ and $s \cap t = \emptyset$ for any $s, t \in T$. So each word $w \in V$ belongs to exactly one topic $t \in T$.

To begin with, TG produces a binary clustering tree over the vocabulary V forming a hierarchical topic model. In order to gain a number of non-hierarchical topics T, a dendrogram cut can be made in the produced clustering tree. Depending on the cut position, the range for $|T|$ lies between $|V|$ and 1. Figure 2 illustrates this for an artificial vocabulary $V = \{a, the, it, med, doc\}$. So in case of TG, the number of topics $|T|$ can be chosen *after* training the topic model by resorting to the tree produced *during* training.

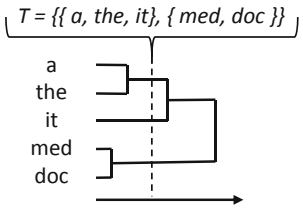

Fig. 2. Obtaining a flat topic model T with two topics via a dendrogram cut of a topic tree produced by Topic Grouper

Blei *et al.* [5] includes an experiment that reduces the feature space via LDA-based topic models in order to classify text using an SVM. However, the authors do not describe their method in detail and a direct comparison to standard word selection techniques is missing. We try to close this gap here.

The study in [16] has a similar focus as ours but it does not clarify how related topic models are generated. As we will show, classification performance depends very much on the details of topic model generation. More importantly, the author does *not* assess classification performance *under a varying number of features*. As will become clear, this aspect is crucial when comparing feature reduction techniques for the given purpose. In addition, it uses only a single dataset which is unsuitable general conclusions.

The authors of [1] study LDA, IG and others as a preprocessing step for text classification based on AdaBoost. Again they work with a fixed number of features for each setting while targeting AdaBoost only. [12] choose LDA as a feature reduction step for news text classification without comparing it to other methods.

The authors of [6] and [11] investigate on topic modeling as a feature reduction method for document clustering, whereas we target supervised document classification.

3 Method

3.1 Datasets

We work with three popular datasets according to Table 1. Regarding "Reuters 21578" we kept only documents with a unique class label, considered only the ten most frequent classes and adopted the so-called "ModApte split" (see [13]). Regarding "Twenty News Groups" we removed all tokens containing non-alphabetical characters or being shorter than three characters. In both cases we performed Porter stemming and stop word filtering. Regarding "OHSUMED" we used the preprocessed dataset extract "ohscal" from [7] with exactly one class label per document.

3.2 Topic Model Generation

To generate LDA models we use Gibbs sampling according to [8]. Moreover, we adopt a commonly used heuristic from [8] for LDA's hyper parameters implying $\beta = 0.1$ and $\alpha = 50/|T|$, and call it "LDA with Heuristics".

In addition, we perform a hyper parameter optimization for α and β using the so-called "Minka's Update" according to [14] and [2] and call it "LDA Optimized". For best possible results under "LDA Optimized" and in concordance with Wallach *et al.* [17], we support an asymmetrical optimization for α such that $\alpha \in \mathbb{R}^{|T|}$. In this case, an estimation of α is based on an initially computed topic model Φ_1. The updated α can in turn be used to compute an updated model Φ_2 (while using Φ_1 as a starting point to compute Φ_2) an so forth. After several iterations of such alternating steps, the models Φ_i as well as α converge. [14] provides a theoretical basis for the estimation of Dirichlet parameters via sample distribution data. Regarding α, these are (samples of) estimated distributions $p(t|d)_i$ as computed along with an intermediate model Φ_i. In total, this results in an Expectation Maximization loop for α (but also for β) nesting the actual LDA procedure.

Regarding TG, we simply run the related algorithm according to [15] on the training documents and choose $|T|$ and with it T from the inferred topic tree according to Sect. 2. In order to obtain the distributions $p(w|t)$ as part of a model Φ, we estimate $p(w|t) := \sum_{d \in D} f_d(w) / (\sum_{w \in t} \sum_{d \in D} f_d(w))$ if $w \in t$ and $p(w|t) := 0$ otherwise.

3.3 Choice of Classifier

An important question affecting a our study is which type of classifier to use. Although not the best performing method, we chose Naive Bayes (NB) for the following reasons:

1. It lends itself well to all of the applied feature space reduction approaches as will be shown below.
2. It does *not* mandate additional hyper parameter settings such as SVM, which would complicate the comparison and potentially incur bias.

3. Approaches relying on a TF-IDF embedding (such as Roccio or SVM as in [10]) are problematic with regard to LDA because DF and IDF are undefined for topics.
4. Many classification methods incur a problem-specific preference to a certain number of features: They trade off happens when an increasing number features starts to degrade accuracy due to the curse of dimensionality. In contrast, NB is robust against a large number of features and is known to perform best without any feature space reduction (see [10]). So, by using by NB we avoid a related kind of bias, and we may expect accuracy to rise continuously with an increasing number of features.

To confirm our argument, we tried other classifiers such as SVM variants and indeed experienced the issues according to (2) and (4) from above (not depicted).

3.4 Classification via Topic Models

Let $C = \{c_1, \ldots, c_m\}$ be the set of classes for the training documents D. We assume that the class assignments $l(d) \in C, d \in D$ are unique and known with regard to D. We define D_c as the subset of training documents belonging to class c, so $D_c = \{d \in D | l(d) = c\}$.

As mentioned before we set $f_d(w)$ to be the frequency of $w \in V$ in document d. Further, let $f_d(t)$ be the frequency of topic t in d, $|D|$ be the number of documents in D and $|d| = \sum_{w \in V} f_d(w)$. When using topics, NB determines the class of a test document d_{test} via

$$argmax_{c \in C} \log p(c|d_{test}) \approx argmax_{c \in C} \log(p(c) \cdot \prod_{t \in T} p(t|c)^{f_{d_{test}}(t)})$$

with $p(c) \approx |D_c|/|D|$.

For best possible results under LDA, we estimate $f_{d_{test}}(t) \approx |d_{test}| \cdot p(t|d_{test})$. In order to compute $p(t|d_{test})$ accurately, we resort to the so-called fold-in method: A word-to-topic assignment z_i is sampled for every word occurrence w_i in d_{test} using Gibbs sampling according to [8]. This involves the use of the underlying topic model Φ and leads to a respective topic assignment vector $\mathbf{z}$ of length $|d_{test}|$. The procedure is repeated S times leading to S vectors $\mathbf{z}^{(s)}$. Together, these results form the basis of

$$p(t|d_{test}) \approx 1/S \cdot \sum_{s=1}^{S} 1/|d_{test}| \sum_{i=1}^{|d_{test}|} \delta_{\mathbf{z}_i^{(s)},t.}$$

More details on this sampling method can be found in [18].

Moreover, we estimate $p(t|c) \approx (\sum_{d \in D_c} p(t|d) \cdot |d|)/\sum_{d \in D_c} |d|$. In this case, an approximation of $p(t|d)$ is already known from running LDA on the training documents.

Since TG partitions V, each word $w \in V$ belongs to exactly one topic t. So we can determine $f_d(t) := \sum_{w \in t} f_d(w)$, which results in the following estimate under TG: $p(t|c) \approx ((1 + \sum_{d \in D_c} f_d(t))/(|T| + \sum_{d \in D_c} |d|))$.[1]

[1] The "1+" and "$|T|$+" in the expression form a standard Lidstone smoothing accounting for potential zero probabilities. Other than that, its practical effect is negligible.

4 Results and Discussion

Figures 3, 4 and 5 present classification accuracy as a function of the number topics or selected words using micro averaging. Given a small number of topics, our findings confirm the impressive abilities of LDA for feature space reduction as reported in [5] when applying hyper parameter optimization. Beyond 700 topics, the heuristic setting degrades LDA's performance in two cases. In accordance with [19], the results confirm that IG performs better than DF. The performance of TG depends on the dataset and ranges below "LDA Optimized", is considerably above IG in Fig. 5 but remains below IG in Fig. 4. In Fig. 3 "LDA Optimized", IG and TG are close above 200 topics or words, respectively.

When applying topic modeling this way, an important point to consider is the computational overhead for topic model generation but also for feature space reduction of new documents at classification time: LDA's runtime is in $O(|T|(\sum_{d \in D} |d|)^2)$ (see [5]) without hyper parameter optimization. The additional EM loop for hyper parameter optimization further drives up the computational cost but as seen in Figs. 5 and 3 this kind of optimization is relevant. E.g., regarding our experiments, computing related topic models with more than a thousand topics took several hours. At test-time, LDA requires the relatively complex fold-in computation of $p(t|d_{test})$ which is on the order of $S \cdot |d_{test}| \cdot |T|$ for a test document.

The runtime of TG is on the order of $|V|^2|D|$ (see [15]), and therefore is it important to limit the size of the vocabulary of the training collection. Once a TG model is built, its use for feature space reduction incurs minimal overhead: i.e., a word from a test document d_{test} can be reduced in constant time via the unique word-to-topic assignment. Thus the total feature space reduction cost for a test document remains on the order of $|d_{test}|$. As noted before, TG assesses all values for $|T|$ between $|V|$ and one within an single training run. This allows to adjust the degree of feature space reduction in hindsight without the need for topic model recomputations.

Altogether, these observations lead us to the following recommendation on when to consider topic modeling as a feature reduction technique for text classification: The primary criterion for a related decision is *the desired number of features* for the classifier:

- If it is well over a few thousand, the computational overhead for LDA does not outweigh the potential improvement in classification accuracy. In this case TG might offer some improvement but our results show that this is also depends on the dataset. It is therefore advisable to try out IG first and then to consider TG for potential improvements in accuracy.
- If the desired number of features is well under a few hundred, then LDA with hyper parameter optimization becomes attractive. However, this case seems unlikely as many popular classifiers can well handle a few hundred features without suffering from the curse of dimensionality.

– If a classifier can deal with a few thousand features or more, standard word selection techniques are an overall good choice since in this case, at least IG approximates or matches approaches based on topic modeling. Also, it incurs lower complexity and less computational overhead.

The presented results can all be reproduced via a prototypical Java library named TopicGrouperJ published on GitHub.[2] Besides script code for the experiments, it features implementations of TG and offers an LDA Gibbs Sampler with options for hyper parameter optimization according to Sect. 3.2.

Table 1. Datasets used for the classification task ($|C|$ is the number of classes, MF is the minimum frequency to keep a stem)

Dataset	$\|D\|$	$\|D_{test}\|$	$\|V\|$	$\|C\|$	MF
Reuters 21578	7,142	2,513	9,567	10	3
OHSUMED	8,374	2,788	11,423	10	3
Twenty news groups	14,134	4711	25,826	20	5

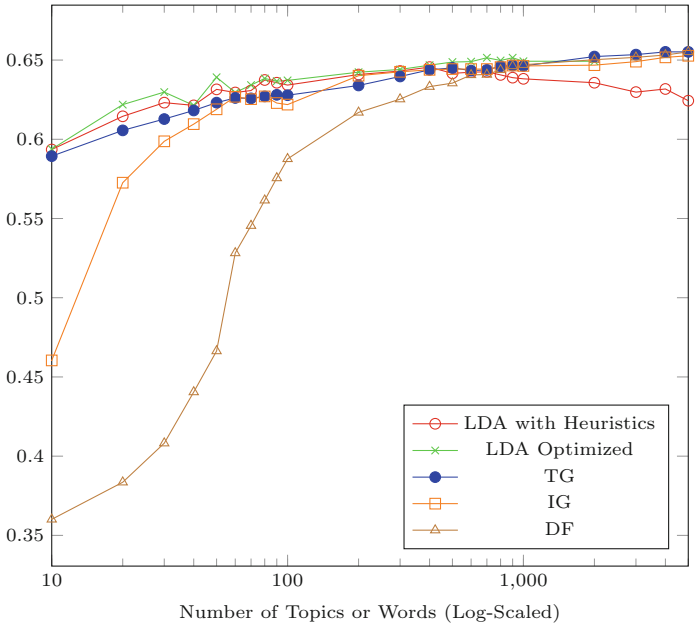

Fig. 3. Accuracy for Reuters 21578

[2] See https://github.com/pfeiferd/TopicGrouperJ.

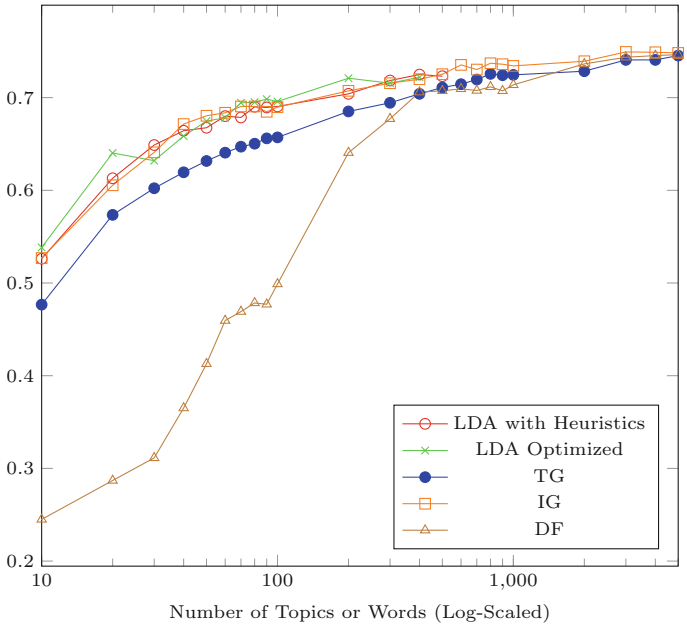

Fig. 4. Accuracy for OHSUMED

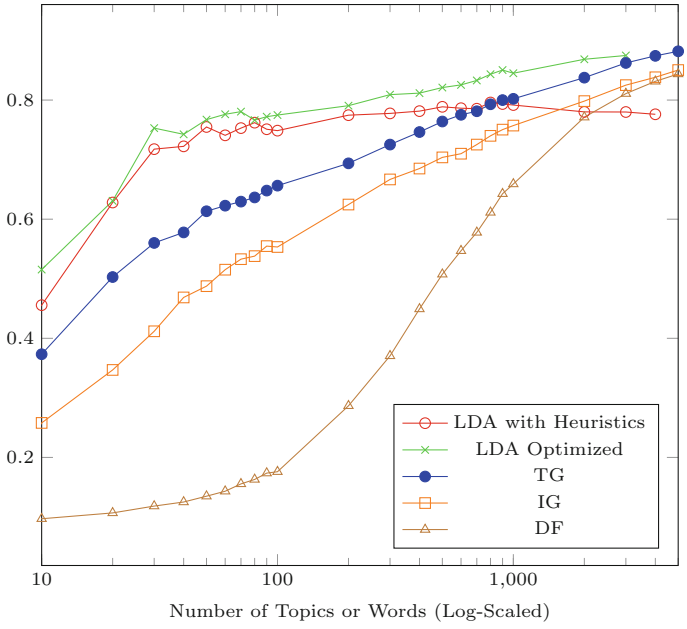

Fig. 5. Accuracy for twenty new groups

References

1. Al-Salemi, B., Ayob, M., Noah, S.A.M., Aziz, M.J.A.: Feature selection based on supervised topic modeling for boosting-based multi-label text categorization. In: 2017 6th International Conference on Electrical Engineering and Informatics (ICEEI), pp. 1–6 November 2017
2. Asuncion, A., Welling, M., Smyth, P., Teh, Y.W.: On smoothing and inference for topic models. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, pp. 27–34. UAI 2009. AUAI Press, Arlington, VA, USA (2009)
3. Bellman, R.: Adaptive Control Processes: A Guided Tour. Princeton University Press, Princeton (1961)
4. Blei, D.M.: Probabilistic topic models. Commun. ACM **55**(4), 77–84 (2012)
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
6. Drummond, A., Vagena, Z., Jermaine, C.: Topic models for feature selection in document clustering. In: Proceedings of the SIAM International Conference on Data Mining, pp. 521–529 (2013)
7. Forman, G.: An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. **3**, 1289–1305 (2003)
8. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proc. Nat. Acad. Sci. **101**(Suppl. 1), 5228–5235 (2004)
9. Hofmann, T.: Probabilistic latent semantic analysis. In: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, UAI 1999, pp. 289–296. Morgan Kaufmann, San Francisco (1999)
10. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0026683
11. Kumar, B.S., Ravi, V.: LDA based feature selection for document clustering. In: Proceedings of the 10th Annual ACM India Compute Conference, Compute 2017, pp. 125–130. ACM, New York (2017)
12. Li, Z., Shang, W., Yan, M.: News text classification model based on topic model. In: 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), pp. 1–5, June 2016
13. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
14. Minka, T.P.: Estimating a Dirichlet distribution, Technical report, Carnegie Mellon University, Pittsburgh, PA, USA (2000). https://tminka.github.io/papers/dirichlet/minka-dirichlet.pdf
15. Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.): ECIR 2019. LNCS, vol. 11437. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-15712-8
16. Sriurai, W.: Improving text categorization by using a topic model. Adv. Comput. **2**(6), 21 (2011)
17. Wallach, H.M., Mimno, D.M., McCallum, A.: Rethinking LDA: why priors matter. In: Bengio, Y., Schuurmans, D., Lafferty, J.D., Williams, C.K.I., Culotta, A. (eds.) NIPS, pp. 1973–1981. Curran Associates, Inc. (2009)

18. Wallach, H.M., Murray, I., Salakhutdinov, R., Mimno, D.: Evaluation methods for topic models. In: Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, pp. 1105–1112. ACM, New York (2009)
19. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning, ICML 1997, pp. 412–420. Morgan Kaufmann, San Francisco (1997)