Maintaining Topic Models for Growing Corpora

Felix Kuhr, Magnus Bender, Tanya Braun and Ralf Möller
University of Lübeck
Institute of Information Systems
Ratzeburgerallee 160, 23562 Lübeck
{kuhr,m.bender,braun,moeller}@ifis.uni-luebeck.de

***Abstract*—A reference library can be described as a corpus of an individual composition of documents. Over time, the corpus might grow because an agent decides to extend its corpus with additional documents, e.g., new publications, or new articles. Existing approaches use topic modelling techniques to compare documents with each other within the same corpus by the documents' topic distribution. However, for new documents, only the text, and no topic distribution is available. Thus, this paper describes three techniques for estimating topic distributions of new unseen documents considering the initial documents in a corpus. Additionally, we present an extensive evaluation about the performance and runtime of the three topic modelling techniques for various scenarios and different sized corpora.**

I. Introduction

An agent in pursuit of a defined task may work with an individual composition of documents as a reference library, also known as a corpus. A person assembling a range of scientific articles as related work describes such a setting, with the person as the agent, the compiling of related articles as the task, and the articles as the library. From an agent-theoretic perspective, an agent is a rational, autonomous unit acting in a world, perceived through sensors, fulfilling a defined task pursuing goals, e.g., an agent providing document retrieval services given specific requests from users. Working directly with the documents in the corpus leads to high computational costs since all words in each document have an influence. Thus, topic modelling is an approach to reduce the complexity of the documents in a corpus. Topic models are statistical models for discovering abstract *topics*, which can be seen as hidden semantic structures, in the text of documents that occur in a corpus. All topic modelling approaches reduce documents to a fixed number of topics such that an agent can work with the documents at their topic level, e.g., comparing documents not by the words occurring but by their topic similarity. Blei et al. [1] have introduced latent Dirichlet allocation (LDA) – a famous statistical topic modelling approach. LDA generates a topic model from the composition of documents in a corpus and learns a document-topic distribution for each word in the corpus as well as a corpus-specific topic-word distribution.

Over time, the composition of documents in a corpus might change because an agent decides to extend its corpus with additional documents. For these corpus-extending documents, an agent has no information about the topics, since a corpus represents an individual composition of documents, and no topic model has been generated for the composition of documents in an extended corpus. Incorporating corpus-extending documents into a topic model enables an agent to identify for the corpus-extending documents a set of similar documents in the corpus based on the documents' topic similarity. Thus, the new documents need a corpus-specific topic distribution.

There are basically three strategies to estimate topic distributions for new documents: (i) Extend the initial corpus with new documents and, based on the documents in the extended corpus, estimate a new topic model, which also includes topic distributions for the new documents. (ii) Infer topic distributions of new documents based on the topic model generated from the documents in the initial corpus and adapt the topic model based on the new documents. (iii) Infer topic distributions of new documents based on the topic model generated from the documents in the initial corpus without adapting the initial model. The first two strategies incorporate that a topic model should represent all documents in its corpus, including the new corpus-extending documents, which is why in one case, a complete new model is learned and in the other case, the initial topic model is adapted. The latter of the two strategies is faster since the existing topic model is not dropped but adapted given the words in new documents. The third strategy leaves the topic model unchanged, which accepts inaccuracy of the topic model representing all documents in the corpus in exchange for fast processing of new documents.

All three strategies can handle corpus-extending documents leading to topic distribution of new documents. However, the strategies differ in their performance for various scenarios, e.g., extending the corpus with a single document, a series of single documents, or a batch of many documents as well as adding documents sharing similar topics or documents with unknown topics. In this paper, we analyse the performance of the three strategies given varying scenarios. Specifically, the contributions of this paper are: (i) providing techniques to compare topic distributions with each other resulting from different topic models and (ii) a comprehensive evaluation regarding the three strategies handling corpus-extending documents. We use the following three techniques for the evaluation of the aforementioned three strategies (i) LDA, (ii) onlineLDA (OLDA), and (iii) fold-in Gibbs sampling (FIGS).

The remainder of this paper is structured as follows. We start with a look at related work. Then, we recap the three topic modelling techniques LDA, OLDA, and FIGS and provide comparison approaches for topic models. Next, we present an evaluation analysing the performance of the three strategies for various scenarios. The paper ends with a conclusion.

II. Related Work

Finding document representations for efficiently fulfilling tasks such as document retrieval has a long history. An old and famous approach is tf-idf [2] (term frequency-inverse document frequency), which can be used to compare the vector representation of two documents in the corpus. The tf-idf scheme is a straightforward approach to reduce documents to fixed-length lists of numbers. However, tf-idf provides only a small amount of reduction in description length and has its limitations in inter- and intra-document structure. Thus, over the years, researchers have proposed other dimensionality reduction techniques, e.g., latent semantic indexing (LSI) [3], representing a linear combination of the original tf-idf features. Hoffman et al. [4] have introduced a generative probabilistic variant of LSI called probabilistic LSI (pLSI), which is the foundation for the well-known topic model LDA [1]. LDA has been introduced in 2003 as a generative model representing documents as a probability distribution over topics. Many extensions have been proposed to optimize the performance of LDA where extensions (i) relax at least one assumption of LDA to uncover more information about the structure of documents or (ii) optimize topic learning for special categories of documents [5]. Extensions of LDA are, e.g., the author-topic model [6], which extends LDA to couple each author of a document with a multinomial over words, and the dynamic topic model [7], which allows for analysing topic changes over time. Furthermore, there are models available for domains like social networks, analysing relationships between people in networks [8]–[10], or working on short documents, like *tweets* on the microblogging service Twitter [11].

Given a topic distribution for each document, one can compare documents using their topic distributions and retrieve documents similar to a given document regarding their topic distributions or cluster the documents in a corpus based on their topic distributions. To compare topic distributions of a document, generated from different topic models, there exist techniques, e.g., visually comparing the documents' topics [12], comparing top-k words of topics from different models [13], or using distances like Hellinger distance [14], Kullback-Leibler divergence [15], or Bhattacharyya distance [16].

All these approaches assume that an underlying corpus does not change. In this paper, we consider documents to extend a given corpus, meaning the topic model no longer accurately represents the initial composition of documents. At least topic distributions for the corpus-extending documents have to be inferred. OLDA [17] infers a topic distribution for a batch of new documents while adapting the existing topic model to the new documents. Fold-in Gibbs sampling [18] only infers a topic distribution for new documents using the given topic-word distribution and document-topic distribution, leaving the existing distributions unchanged. The idea relates to incremental LSI [19] considering the previous computations of the model. We analyse the performance of different topic modelling techniques estimating the topic distribution for corpus-extending documents.

III. Topic Models and Their Maintenance

This section describes LDA [1] as the main topic modelling technique as well as OLDA [17] and FIGS [18] for maintaining a corpus when new documents arrive.

Topic modelling techniques basically estimate topics from a collection of documents by calculating for each of the documents a topic probability distribution and topics represent co-occurring words of the documents. LDA assumes that documents in a corpus $\mathcal{D}$ represent a mixture of topics where each topic is characterized by a distribution of words from a fixed vocabulary $\mathcal{V}$ containing all words from the documents in $\mathcal{D}$. LDA generates a topic model from the documents in $\mathcal{D}$, learning latent structures of two forms,

(i) a *document-topic distribution* θ_d for each document $d \in \mathcal{D}$, representing the degree with which the content of d is about each of the K topics, and

(ii) a *topic-word distribution* ϕ describing the probability of each word from $\mathcal{V}$ occurring in each of the K topics.

Both, the document-topic distribution and topic-word distribution depend on the documents in $\mathcal{D}$. The inputs for LDA are a corpus $\mathcal{D}$ of documents as defined above, the number of topics K, and two hyperparameters α and β, where α conditions the per-document topic distributions and β conditions the per-corpus topic distributions. Hyperparameters α and β trade off the following two goals to identify groups of co-occurring words: (i) Allocate the words in a document to as few topics as possible (α). (ii) Assign high probability to as few terms as possible in each topic (β). The two goals are conflicting, since assigning all words to a single topic within a document achieves the first goal but makes it difficult to achieve the second goal. Achieving the second goal and assigning only few words to each topic makes it difficult to reach the first goal for documents containing many words. To cover all words in a document, many topics have to be assigned; however, the first goal is to assign as few topics as possible within the document.

Formally, for each document $d \in \mathcal{D}$, LDA learns a discrete probability distribution θ_d over the K topics, which contains for each topic $k \in \{1, \dots, K\}$ a value between 0 and 1 s.t. the sum of all values is 1, and a discrete probability distribution ϕ_k for each topic $k \in \{1, \dots, K\}$ over the words in $\mathcal{V}$, which contains for each $w \in \mathcal{V}$ a value between 0 and 1 s.t. the sum of all values is 1. Both distributions represent a corpus-specific topic model $\mathcal{M} = (\theta_d \forall d \in \mathcal{D}, \phi_k \forall k \in \{1, \dots, K\})$. Given a corpus, only $w_{d,j}$ are visible in each d. The key inference problem is computing the posterior distribution of hidden variables given a document (with α, β chosen):

$$p(\theta, z \mid w, \alpha, \beta) = \frac{p(\theta, z, w \mid \alpha, \beta)}{p(w \mid \alpha, \beta)}, \tag{1}$$

where z represents a single topic chosen from θ. Exactly calculating the posterior distribution of the hidden variables is intractable. Instead, approximative inference algorithms are used such as mean-field variational expectation maximisation [1], expectation propagation [20], Gibbs sampling [18], or online variational Bayes [17].

If a new document d' extends a given corpus with an existing topic model $\mathcal{M}$, there are three main strategies to provide a document-topic distribution $\theta_{d'}$ for the new document d':

(i) Extend corpus $\mathcal{D}$ with document d' and calculate a new topic model $\mathcal{M}'$ for $\mathcal{D}' = \mathcal{D} \cup \{d'\}$ using LDA.
(ii) Approximate $\theta_{d'}$ for d' by inferring $\theta_{d'}$ based on $\mathcal{M}$
  (a) either updating the parameters of $\mathcal{M}$ considering the content of document d' along the way (OLDA),
  (b) or without updating the parameters of $\mathcal{M}$ (FIGS).

Strategy (i) is the most *accurate* version of handling d' since a topic model is based on all words in a corpus. If the corpus changes, a new topic model has to be learned. Since learning a new topic model is computationally intensive, the adaptive methods OLDA and FIGS have been introduced.

OLDA [17] can analyse large collections of documents. It is optimized for handling streams of documents extending a corpus. OLDA efficiently adapts topic models and calculates document-topic distributions for new documents by approximating the posterior probability in Eq. (1) using online stochastic optimization converging to a local optimum of a variational Bayes objective function. To extend an initial corpus $\mathcal{D}$ with a new document d', OLDA updates the initial topic-word distributions $\phi_k, k \in \{1, \ldots, K\}$ and the document-topic distributions $\theta_d, d \in \mathcal{D} \cup \{d'\}$ by using an EM-algorithm iterating over the extended corpus until the model performance converges or a fixed number of iterations is reached. For details, please refer to Alg. 2 in [17]. In contrast to using LDA for learning a new model, OLDA reuses the distributions of the old topic-model by adapting it given new documents, saving computation costs. Adapting a topic model given a new document might drag down the performance of the topic model on the original documents, though.

FIGS refers to adding a new document d' to the initial corpus $\mathcal{D}$ and performs Gibbs sampling [18] only on the words in d', without adapting the initial document-topic distributions $\theta_d, d \in \mathcal{D}$ and the topic-word distributions $\phi_k, k \in \{1, \ldots, K\}$. Thus, FIGS is even faster than OLDA compared to Strategy (i). FIGS assigns the most probable topic for each word $w \in d'$ using the topic-word distributions $\phi_k, k \in \{1, \ldots, K\}$. Then, FIGS computes for each word in d' the probability being assigned to each of the K topics, samples a topic from the document's topic distribution, and assigns the word to the new topic. If d' contains a new word w not part in any document $d \in \mathcal{D}$, the Gibbs sampling process randomly assigns a topic for this word. The topic assignment of the words in d' yields the distribution of topics in d'. FIGS requires only few iterations taking the topic structure into account, i.e., allocate words of documents to as few topics as possible.

Technically, there is no limitation in the number of documents for extending a corpus using OLDA or FIGS, but the document-topic distributions $\theta_d, d \in \mathcal{D}$, topic-word distributions $\phi_k, k \in \{1, \ldots, K\}$, and the number of topics (K) changes with each corpus-extending document. Thus, it might be a good idea to generate a new topic model after a while.

IV. Comparing Topic Models

This section discusses how to compare topic distributions and complete topic models, which contain various topic distributions with topics not necessarily matching one-to-one between different topic models.

A. *Comparing Topic Distributions within a Topic Model*

The Hellinger distance allows for measuring the distance between two probability distributions [14]. As such, we can use the distance to compare document-topic and topic-word distributions for a single topic model $\mathcal{M}$ generated from the documents in a corpus $\mathcal{D}$. Given document-topic distributions θ_{d_i} and θ_{d_j} for two documents $d_i, d_j \in \mathcal{D}$, the Hellinger distance $H(\theta_{d_i}, \theta_{d_j})$ between θ_{d_i} and θ_{d_j} is defined as

$$H(\theta_{d_i}, \theta_{d_j}) = \frac{1}{\sqrt{2}} \sqrt{\sum_{k=1}^{K} \left(\sqrt{\theta_{d_i,k}} - \sqrt{\theta_{d_j,k}} \right)^2}, \qquad (2)$$

where k refers to the topics. Since K is usually small, it is computationally feasible to calculate $H(\theta_{d_i}, \theta_{d_j})$ for two given distributions. To compute the Hellinger distance between two topic-word distributions ϕ_{k_i}, ϕ_{k_j}, the inner sum of Eq. (2) goes over the words in the vocabulary of $\mathcal{D}$.

The inner sum of Eq. (2) assumes that the two distributions are indexed over the same variables, i.e., topics K in case of document-topic distributions $\theta_{d_i}, \theta_{d_j}$ and vocabulary $\mathcal{V}$ in case of topic-word distributions ϕ_{k_i}, ϕ_{k_j}. If comparing two distributions from two different topic models $\mathcal{M}$ and $\mathcal{M}'$ based on the same corpus, the assumption may be violated. In case of topic-word distributions ϕ_k and $\phi_{k'}$, it is reasonable to assume $\mathcal{V}$ is the same and can therefore be matched between ϕ_k and $\phi_{k'}$. But, assuming that K is identical for both $\mathcal{M}$ and $\mathcal{M}'$, the K topics, over which the inner sum iterates, may not be as easily matched. Topics are only abstract structures representing a distribution over the words in a vocabulary, i.e., ϕ_k for all $k \in \{1, \ldots, K\}$. Learning the topic distributions does not guarantee that topic $k = 1$, represented by ϕ_1 in $\mathcal{M}$, matches the topic $k = 1$, represented by ϕ_1 in $\mathcal{M}'$. Therefore, we need a way to match the topics from one topic model to the topics from another topic model.

B. *Matching Topics from Different Topic Models*

As mentioned above, comparing topic distributions of documents from different topic models is difficult since topics have no names and the first topic from a model $\mathcal{M}$ does not necessarily represent the first topic from another model $\mathcal{M}'$. Thus, we need a technique mapping K topics from one topic model $\mathcal{M}$ to K' topics from another model $\mathcal{M}'$ such that we can compare the topics from different models. We say the best match between the K topics of two topic models $\mathcal{M}, \mathcal{M}'$ is given by the mapping σ of the K topics from $\mathcal{M}$ to the topics of $\mathcal{M}'$ that has the minimal sum of the Hellinger distances between the K topic-word distributions:

$$\min_{\sigma} \sum_{k=1}^{K} H(\phi_k, \phi'_{\sigma(k)}), \qquad (3)$$

where σ denotes a function that maps each topic $k \in \{1, \dots, K\}$ in $\mathcal{M}$ to a topic $k' \in \{1, \dots, K'\}$ in $\mathcal{M}'$. If $K = K'$, we may impose bijectivity on σ to require that each topic in $\mathcal{M}$ is mapped to exactly one topic in $\mathcal{M}'$, and vice versa. We consider the following techniques estimating the best mapping between the topics of the two models $\mathcal{M}$ and $\mathcal{M}'$:

(i) *Full Permutation:* Calculate the Hellinger distance for each possible mapping between the topics of $\mathcal{M}$ to the topics of $\mathcal{M}'$ to identify the best mapping, i.e., exactly determine Eq. (3), yielding a bijective mapping. The complexity lies in $O(K! \cdot T_H)$, where T_H refers to the complexity of calculating the Hellinger distance, which depends on the number of topics K. Thus, this technique is only applicable for small K.

(ii) *Topic coherence:* Estimate for each topic k of $\mathcal{M}$ and k' of $\mathcal{M}'$ the documents having a high probability for the respective topic (*top-doc*) and compare the topics between $\mathcal{M}$ and $\mathcal{M}'$ in both directions using the Jaccard coefficient J on the assigned documents. This results in two sets, each containing for each topic a set of documents. Additionally, we compare the *top-c* words of each topic k of $\mathcal{M}$ with the *top-c* words of all topics k' of $\mathcal{M}'$ using the Jaccard coefficient J, leading again to two sets of document-topic assignment. Thus, for each topic k of $\mathcal{M}$, we have four possible topics in $\mathcal{M}'$ for our mapping and use a majority vote to map k to k'. The basic assumption is that the documents characterise a topic. The mapping is not necessarily bijective as two or more topics from $\mathcal{M}$ may be mapped to one topic in $\mathcal{M}'$ given the topic-assigned documents. The upside of this technique is its superior runtime in $O(K^2 \cdot T_J)$, where T_J refers to the complexity of calculating J, which depends on the number of documents in the corpus.

(iii) *Minimal Hellinger distance:* Calculate the Hellinger distance between each topic k in model $\mathcal{M}$ and all topics k' in model $\mathcal{M}'$.

$$k' = \underset{k' \in \{1, \dots, K'\}}{\arg\min} \; H(\phi_k, \phi'_{k'})$$

Again, the mapping is not necessarily bijective. Compared to topic coherence, the best match is based not on the *top-doc* documents but on the distribution over all topics. The complexity lies in $O(K^2 \cdot T_H)$ where T_H again refers to the complexity of calculating the Hellinger distance

We generate 50 topic models from the documents in a corpus $\mathcal{D}$ and compare the three techniques plus the average distance between randomly selected mappings for topic-distribution of two topic models $\mathcal{M}, \mathcal{M}'$ learned for one corpus $\mathcal{D}$ (without extending $\mathcal{D}$). Figure 1 presents the Hellinger distance of the matched topics given the mapping σ generated by (a) random permutation – randomly selecting one mapping for each of the 50 topic models, (b) topic coherence, (c) best permutation – selecting for each of the 50 topic models only its best mapping, and (d) minimal Hellinger distance.

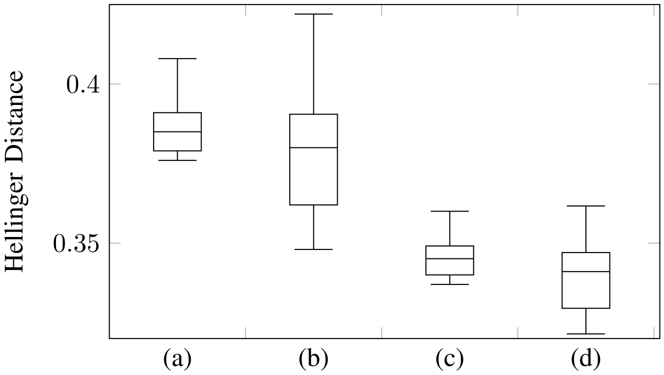

Fig. 1: Topic mapping approaches using (a) a random mapping, (b) topic coherence, (c) best permutation, and (d) minimal Hellinger distance for 50 topic models from the corpus.

The average performance of all techniques is similar, with the Hellinger distance lying between 0.3 and 0.4. The performance of the topic coherence varies the most, the performance of the best mapping from the full permutation the least. The best performance can be reached using the minimal Hellinger distance, which is what we will use for the evaluation. Since the minimal Hellinger distance allows us to map two topics in $\mathcal{M}$ to the same topic in $\mathcal{M}'$, it is possible that the performance of the minimal Hellinger distance is better than the best results from a full permutation. Given a way to match topics from different topic models, we specify two ways to compare topic models with each other.

C. Comparing Topic Models

We are interested in evaluating the performance between LDA, OLDA, and FIGS in estimating document-topic distributions of new documents extending an initial corpus $\mathcal{D}$.

A famous benchmark in the NLP community is the *perplexity*. The perplexity of a topic model describes how well the generated model predicts a sample. The smaller the perplexity of a topic model, the better is the prediction performance of the model for samples. However, different experiments have shown that the perplexity does not strongly correlate to human judgment [21]. Besides perplexity, we focus on the document-topic distribution of documents as a benchmark.

As described in Section III, calculating the hidden variables of a document d is based on approximative techniques. Generating two topic models from the same composition of documents in $\mathcal{D}$ using the same initial topic modelling parameters lead to two topic models distinguishing in their topic-word distribution ϕ and the document-topic distribution θ for each document in $\mathcal{D}$, i.e., the document-topic distribution θ_d of document d generated by one topic model $\mathcal{M}$ is different from θ_d generated by another model $\mathcal{M}'$ from the same composition of documents in $\mathcal{D}$. One reason for the difference in the document-topic distribution is given by the approximative inference algorithms generating the topic-word distributions and document-topic distributions. Thus, we say that there is an "excused error" that we attribute to the approximate nature of the calculations. We call this error a baseline error $b_{err}(\mathcal{M}, \mathcal{M}')$ between the two topic models $\mathcal{M}$ and $\mathcal{M}'$ and define it as follows:

$$b_{err}(\mathcal{M}, \mathcal{M}') = \frac{\sum_{k=1}^{K} H(\phi_k, \phi_{\sigma(k)})}{K}$$

Calculating H requires a mapping σ between $\mathcal{M}, \mathcal{M}'$ for the inner sum of Eq. (2).

Having a baseline error, we can define the classification performance $\mathcal{K}$ for evaluating the performance between LDA, OLDA, and FIGS in estimating the topic distribution for an extend corpus $\mathcal{D}' = \mathcal{D} \cup \{d_i'\}_{i=0}^{T}$ containing T new documents as follows:

$$\mathcal{K}(\mathcal{M}, \mathcal{M}') = \max\left(0, \frac{\sum_{d \in \mathcal{D}'} H(\theta_d, \theta_d')}{|\mathcal{D}'|} - b_{err}(\mathcal{M}, \mathcal{M}')\right),$$

where $\mathcal{M}'$ represents the topic model generated by LDA and represents the ground-truth for the document-topic distribution and topic-word distribution. $\mathcal{M}$ represents the topic model generated by FIGS (OLDA) so that we can compare the performance of FIGS (OLDA) with LDA generating a completely new topic model from an extend corpus, while considering the baseline error. $\mathcal{K}$ indicates the average Hellinger distance of the document-topic distributions. The smaller the classification value of $\mathcal{K}$, the better is the performance of a model.

V. Empirical Evaluation

This section shows an empirical evaluation of LDA, OLDA, and FIGS. Each of the approaches estimates a topic distribution given the text of corpus-extending documents. However, the performance of the three approaches differ in (i) the perplexity of the overall corpus, (ii) the held-out set perplexity, (iii) the classification performance, and (iv) the runtime for corpora of different size.

The documents in corpus $\mathcal{D}$, the length of the documents, and the words in the documents influence the difference between the inferred topic distributions of new documents $\{d_i'\}_{i=0}^{T}$ and the topic distributions of the same documents generated by a new topic model for $\mathcal{D} \cup \{d_i'\}_{i=0}^{T}$. This evaluation focusses on which approach performs best for different scenarios, extending the corpus by adding batches of 1000 corpus-extending documents to the initial corpus such that

(type 1) the categories of unseen documents are not part of the documents in the initial corpus or

(type 2) the documents in the initial corpus contain the same categories as the corpus-extending documents.

We expect FIGS and OLDA to perform better with documents of a known category compared to documents of an unknown category. We also compare for OLDA two settings:

(incr.) Short for incremental, where in each iteration, we add a batch of documents, perform OLDA, and continue with the extended corpus when adding another batch in the next iteration. That is in each iteration, a batch of 1000 documents is added to the growing corpus.

(init.) Short for initial, where we retain the original corpus and add an increasingly larger batch of documents in each iteration, meaning, in the first iteration, we add 1000 documents, in the second, 2000, and so on to the original corpus.

Thus, we are interested in the performance for the documents in a corpus and the held-out set for the *incremental* and *initial* technique extending the corpus since new documents are generated from time to time and not only once.

We compare the performance of OLDA and FIGS against the performance of LDA, which acts as a baseline, on the well-known 20Newsgroup data set [22]. The data set contains 20 different newsgroups, each corresponding to a different topic. However, some of the newsgroups are closely related, e.g., *autos* and *motorcycles*, *baseball* and *hockey*, or *pc hardware* and *mac hardware*, resulting in 11 to 13 distinct topics. We remove 1000 duplicated documents and preprocess the remaining 18.846 documents using the following four techniques: (i) lowercasing all characters, (ii) stemming the words, (iii) tokenizing the result, and (iv) eliminating tokens part of a stop-word list containing 337 words.

A. Perplexity

We present the topic model perplexity of LDA, OLDA, and FIGS for the following two cases in the right plot of Fig. 2: In the first case (type 1), the set of corpus-extending documents represents content form newsgroups that is not represented in the documents of the initial corpus $\mathcal{D}$. In the second case (type 2), the set of corpus-extending documents represents content form newsgroups that are also represented in documents of the initial corpus $\mathcal{D}$.

For LDA, we calculate a new topic model from the initial composition of documents in corpus $\mathcal{D}$ and the set of corpus-extending documents $\{d_i'\}_{i=0}^{T}$. For FIGS, we use the available topic-word distributions ϕ_k, $k \in \{1, ..., K\}$ and document-topic distribution θ_d, $d \in \{1, ..., |D|\}$ from the initial corpus $\mathcal{D}$ inferring the document-topic distribution $\theta_{d_i'}$ for each corpus-extending document in $\{d_i'\}_{i=0}^{T}$. In the setting of OLDA, we update the initial topic model after extending the corpus with T corpus-extending documents $\{d_i'\}_{i=0}^{T}$. The performance of OLDA depends on the occurring categories in the documents of the initial corpus and the categories of the corpus-extending documents. Generally, the performance of OLDA is worse when the categories of new documents are distinct (type 1) to those categories of the documents in the initial corpus (type 2).

In the left plot of Figure 2, we evaluate the topic model perplexity of FIGS, LDA, and both variants of OLDA considering only unseen documents of type 1. The corpus perplexity increases with corpus-extending documents for the incremental variant of OLDA, which strongly adapts the initial model. Adding all batches to the initial corpus and adapting the initial topic model using OLDA init. leads to a similar corpus perplexity as using LDA which calculates a new topic model from all documents. In the right plot of Figure 2, we evaluate the topic model perplexity for FIGS, LDA, and both variantes of OLDA considering only unseen documents containing the

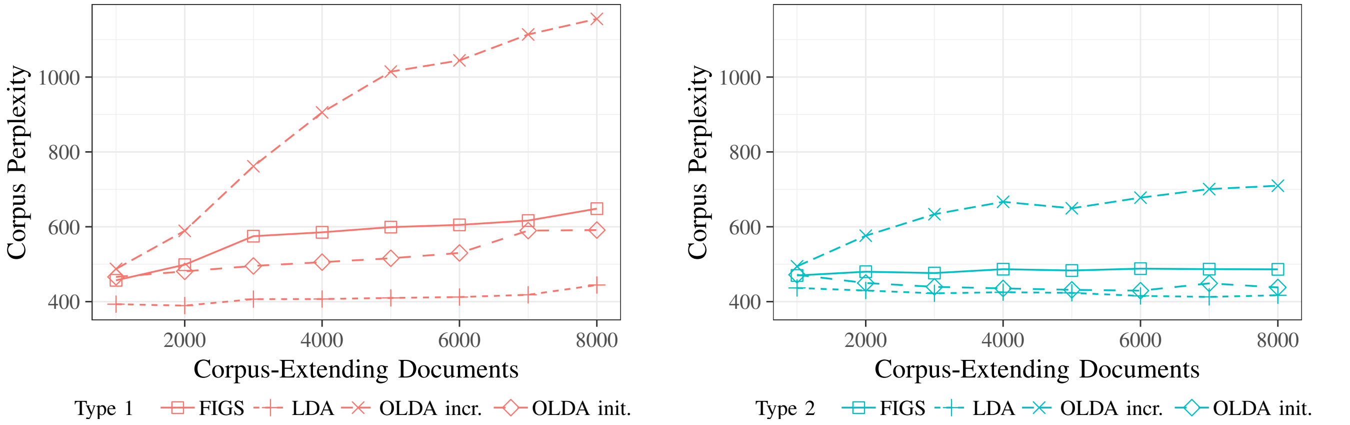

Fig. 2: Both plots show the perplexity using LDA, FIGS, OLDA incr., and OLDA init.. We present corpus-extending strategy using *type 1* (left plot) and *type 2* (right plot). Setting: initial corpus size: 8k, α: 0.1, β: 0.1, topics: 11, iterations: 10k.

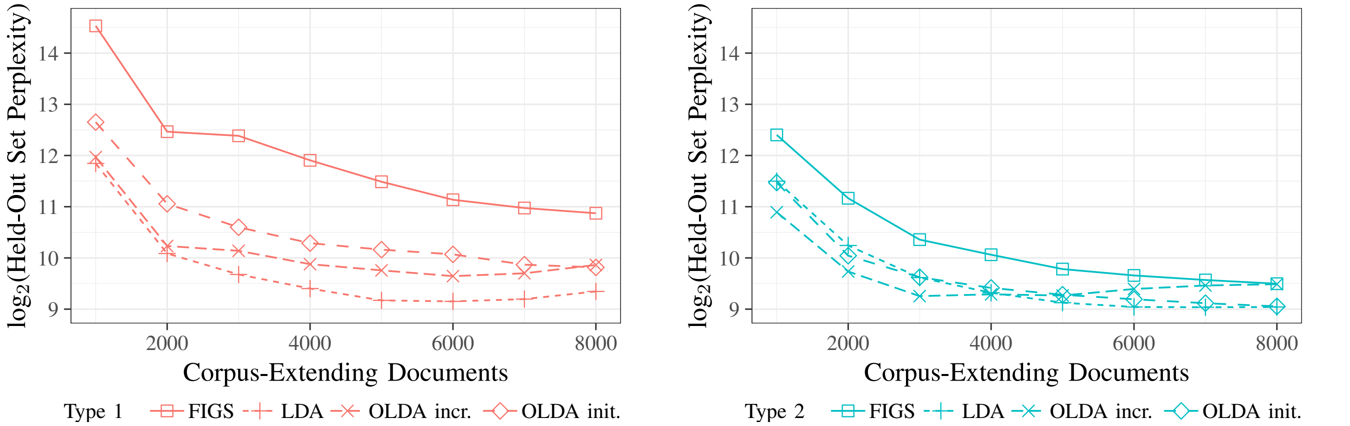

Fig. 3: Both plots show the held-out set perplexity using LDA, FIGS and both variants of OLDA.

same categories as the documents in the initial corpus. Again, the performance of OLDA incr. is worst. For all three other techniques, the corpus perplexity is similar.

In Figure 3, we present the held-out perplexity for a fixed set of documents using FIGS, LDA, and OLDA for type 1 (left) and type 2 (right). Initially, we generate one topic model from all documents within a corpus $\mathcal{D}$ containing 10k documents. In each step, we extend $\mathcal{D}$ with 1k documents and calculate the perplexity for the new documents in the following way:

(i) For LDA, we add documents to the initial corpus and calculate a new topic model from the extended corpus, (ii) for OLDA incr. we add in each step the new documents to the actual corpus and update the actual topic model leading to new topic distributions for all documents in the corpus, and (iii) for OLDA init. we add in each step all so far new documents to the initial corpus and update the initial topic model leading to new topic distributions for all documents in the corpus, and (iv) for FIGS, we use the initial topic model for estimating a the topic distribution for new documents without changing the topic distribution of all other documents in the corpus. LDA and both OLDA variants have similar held-out perplexity and the performance of both approaches is better than for FIGS.

B. Runtime

In Figure 6, we compare the runtime performance of the different approaches considering the following three corpora differing in their initial corpus size: (i) small corpus containing 4k documents, (ii) medium corpus containing 8k documents, and (iii) large corpus containing 18k documents.

We analyse the runtime performance for each approach adding three batches to each of the three different corpora. In case of the LDA topic modelling technique, we calculate four topic models for each size of the corpus; one initial topic model and three additional models, each after adding a batch of corpus-extending documents to the corpus. Using FIGS requires only one initial topic model and no additional models, since FIGS use the initial topic model to infer the document-topic distribution $\theta_{d'_i}$ of each corpus-extending document in the batch of size T, represented by $\{d'_i\}_{i=0}^{T}$. Additionally, we compare the performance of OLDA incr. and OLDA init. For OLDA incr. we calculate one topic model from the initial corpus $\mathcal{D}$ and update the topic model after adding the corpus-extending documents to $\mathcal{D}$ and go on working with the updated topic model. For OLDA init. we always perform the update operation on a topic model representing the initial corpus $\mathcal{D}$.

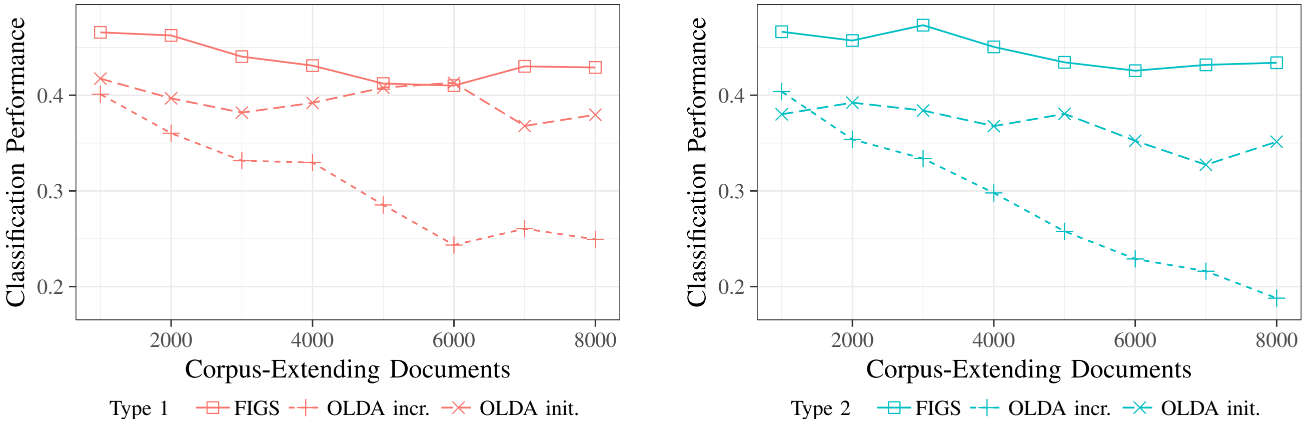

Fig. 4: Classification performance of FIGS and both OLDA variants on corpora of type 1 left and type 2 right.

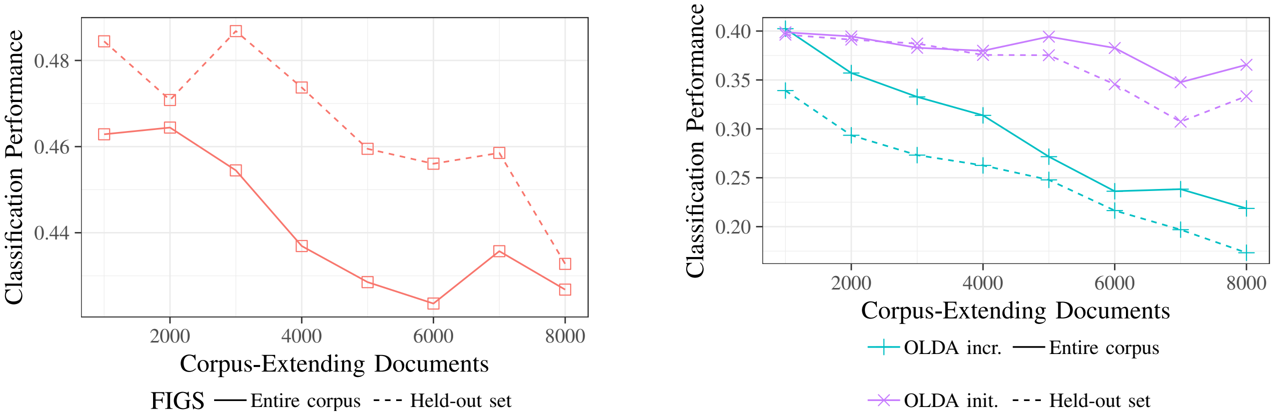

Fig. 5: Classification performance of FIGS and OLDA on the entire corpus compared to the held-out set.

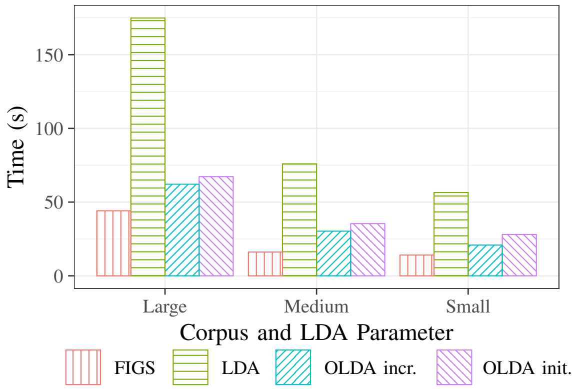

Fig. 6: Runtime on different sized corpora.

Thus, in each step updating $\mathcal{D}$ with new documents add the actual batch of documents and the previous batches to the initial corpus. Afterwards, using OLDA to update the topic model to the documents in the extended corpus.

The runtime proportion between LDA, FIGS, OLDA incr. and OLDA init. is the same for all corpora. LDA requires most time. FIGS is the fastest technique for each corpora. Both variants of OLDA have similar runtime performance.

C. Classification Performance

Finally, we compare the classification performance between LDA, both variants of OLDA, and FIGS. In Figure 4, we present two plots of the classification performance. In both plots, we add eight batches of 1k documents to an initial corpus containing 8k documents using one of the three approaches. Again, the left plot shows the classification performance for type 1 and the right plot of type 2. For type 1 the initial corpus contains documents with content from different categories compared to the corpus-extending documents, and for type 2 the initial corpus contains documents from the same categories as the corpus-extending documents. Viewing both plots in Figure 4 we can see that FIGS has the worst classification performance. Both, OLDA incr. and OLDA init. start at nearly the same value after adding the first batch of corpus-extending documents. Adding more batches to the initial corpus, the OLDA incr. becomes better while OLDA init. stays the same value. The worse values of FIGS result from ignoring an update of the initial model parameters, while adapting a model by FIGS. The adaptation process of OLDA indeed changes the topic distributions and results in a better classification performance. The performance of OLDA incr. is better than for OLDA init.

We are interested in comparing new corpus-extending documents with the documents in an initially given corpus, besides the classification performance of FIGS and OLDA on the entire corpus we are also interested in the classification performance only on the added batches of documents. In Figure 5, we compare the classification performance on the entire corpus consisting of the initial documents and the batch of corpus-extending documents with the classification performance on the added batches. The left plot of Figure 5 shows the results for FIGS. As FIGS does not change the underlying model, the entire corpus reaches a better performance, while the dashed line representing only the batches stays above. By adding more batches to the initial set of documents, both lines converge since the impacts of the additional batches rise whereas they decrease for the initial documents. In the right plot of Figure 5, we present the results for OLDA on the entire corpus (solid line) and a held-out set (dashed lines). Additionally, we compare OLDA for the incremental variant (plus) and the initial variant (cross). In contrast to FIGS, the performance for OLDA incr. and OLDA init. is better on the held-out set than on the entire corpus and the classification performance increases with an increasing number of documents extending to the initial corpus.

VI. Conclusion

Topic modelling techniques estimate topic distributions for documents in a corpus. For corpus-extending documents, this paper evaluates three main strategies to incorporate new documents: (i) learn a new topic model using LDA, (ii) adapt an existing topic model and infer topic distributions for new documents using onlineLDA, and (iii) infer topic distributions for new documents using fold-in Gibbs sampling, leaving the topic model as is. We also compare different techniques to match the topics generated from different topic models. To the best of our knowledge, this is the first evaluation comparing different topic modelling techniques with a focus on corpus-extending documents. In the context of our evaluation, we evaluate the performance of LDA, FIGS, and two variants of OLDA regarding the perplexity of the documents in the corpus, the held-out set perplexity, the classification performance, and the runtime for corpora of different size. In conclusion, each of the three approach has its advantages and disadvantages and the best approach estimating a topic distribution for corpus-extending documents depends on the individual use case. Calculating a new topic model from all documents in an extended corpus is time-consuming but allows easy comparison between topic distributions of documents. OLDA allows for efficiently updating the initial topic model, accepting an increased corpus perplexity. FIGS is fast in estimating a topic distribution for new documents without changing the topic distribution of the documents in the corpus. The evaluation shows that for a small set of corpus-extending documents it is worth it to use the FIGS technique estimating topic distribution for corpus-extending documents.

References

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[2] K. Sparck Jones, “A statistical interpretation of term specificity and its application in retrieval,” *Journal of Documentation*, vol. 28, no. 1, pp. 11–21, 1972.

[3] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, “Indexing by latent semantic analysis,” *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.

[4] T. Hofmann, “Probabilistic latent semantic indexing,” in *22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 50–57.

[5] D. M. Blei, “Surveying a suite of algorithms that offer a solution to managing large document archives,” *Communication of the ACM*, vol. 55, no. 4, pp. 77–84, 2012.

[6] Rosen-Zvi, Michal and Griffiths, Thomas and Steyvers, Mark and Smyth, Padhraic, “The author-topic model for authors and documents,” in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI Press, 2004, pp. 487–494.

[7] D. M. Blei and J. D. Lafferty, “Dynamic topic models,” in *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, 2006, pp. 113–120.

[8] L. Hong and B. D. Davison, “Empirical study of topic modeling in twitter,” in *1st workshop on social media analytics*. ACM, 2010, pp. 80–88.

[9] J. She and L. Chen, “Tomoha: Topic model-based hashtag recommendation on twitter,” in *23rd International Conference on World Wide Web*. ACM, 2014, pp. 371–372.

[10] A. Ahuja, W. Wei, and K. M. Carley, “Topic modeling in large scale social network data,” 2015.

[11] Yan, Xiaohui and Guo, Jiafeng and Lan, Yanyan and Cheng, Xueqi, “A biterm topic model for short texts,” in *International conference on World Wide Web*. ACM, 2013, pp. 1445–1456.

[12] P. J. Crossno, A. T. Wilson, T. M. Shead, and D. M. Dunlavy, “Topicview: Visually comparing topic models of text collections,” in *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*. IEEE, 2011, pp. 936–943.

[13] K. Stevens, P. Kegelmeyer, D. Andrzejewski, and D. Buttler, “Exploring topic coherence over many models and many topics,” in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 2012, pp. 952–961.

[14] E. Hellinger, “Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen.” *Journal für die reine und angewandte Mathematik*, vol. 136, pp. 210–271, 1909.

[15] M. Paul, “Cross-collection topic models: Automatically comparing and contrasting text,” *Urbana*, vol. 51, p. 61801, 2009.

[16] A. Bhattacharyya, “On a measure of divergence between two multinomial populations,” *Sankhyā: the indian journal of statistics*, pp. 401–406, 1946.

[17] M. D. Hoffman, D. M. Blei, and F. R. Bach, “Online learning for latent dirichlet allocation,” in *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada.*, 2010, pp. 856–864.

[18] T. L. Griffiths and M. Steyvers, “Finding scientific topics,” *National academy of Sciences*, vol. 101, no. suppl 1, pp. 5228–5235, 2004.

[19] H.-Y. Jiang, T. N. Nguyen, X. Chen, H. Jaygarl, and C. K. Chang, “Incremental latent semantic indexing for automatic traceability link evolution management,” in *Proceedings of the 2008 23rd IEEE/ACM International Conference on Automated Software Engineering*. IEEE Computer Society, 2008, pp. 59–68.

[20] T. Minka and J. Lafferty, “Expectation-propagation for the generative aspect model,” in *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2002, pp. 352–359.

[21] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei, “Reading tea leaves: How humans interpret topic models,” in *Advances in neural information processing systems*, 2009, pp. 288–296.

[22] K. Lang, “Newsweeder: Learning to filter netnews,” in *Machine Learning Proceedings 1995*. Elsevier, 1995, pp. 331–339.