On a Topic Model for Sentences

ABSTRACT

Probabilistic topic models describe the content of documents at word level in large document collections. However, the structure of the textual input, and for instance the grouping of words in coherent text spans such as sentences, contains much information which is generally lost with these models. In this paper, we propose sentenceLDA, an extension of LDA whose goal is to overcome this limitation by incorporating the structure of the text in the generative and inference processes. We illustrate the advantages of sentenceLDA by comparing it with LDA using both intrinsic (perplexity) and extrinsic (text classification) evaluation tasks on different text collections.

CCS Concepts

• Information systems → Document topic models;

Keywords

Text Mining; Topic Modeling; Unsupervised Learning

1. INTRODUCTION

Statistical topic models are generative unsupervised models that describe the content of documents in large textual collections. Prior research has investigated the application of topic models such as Latent Dirichlet Allocation (LDA) [2] in a variety of domains ranging from image analysis to political science. Most of the work on topic models treats documents as bag-of-words and as a result, the word ordering and the words’ grouping in coherent text segments such as sentences and phrases are lost.

However, the inner structure of documents is generally useful, when identifying topics. For instance, one would expect that in each sentence, after standard pre-processing steps such as stop-word removal, only a very limited number of latent topics would appear. Thus, we argue that coherent text segments should pose “constraints” on the amount of topics that appear inside those segments.

Previous work on LDA proposed several extensions aiming at incorporating additional information, such as class labels [5], or temporal dependencies between documents in a stream [9], or even multilingual information in parallel or comparable collections [8].

In this paper, we propose sentenceLDA (senLDA), whose purpose is to incorporate part of the text structure in the topic model. Motivated by the argument that coherent text spans should be produced by only a handful of topics, we propose to modify the generative process of LDA. Hence, we argue that the latent topics of short text spans should be consistent across the units of those spans. In our approach, such text spans can vary from paragraphs to sentences and phrases depending on the purpose of the task. It is to be noted that in the extreme case where words are the coherent text segments, the standard LDA model becomes a special case of senLDA.

In the rest of the paper we present the senLDA and we derive its collapsed Gibbs sampler in Section 2, we illustrate its advantages by comparing it with the LDA on intrinsic (in-vitro) and extrinsic (ex-vivo) evaluation experiments using collections of Wikipedia and PubMed articles in Section 3, and we conclude in Section 4.

2. THE PROPOSED MODEL

A statistical topic model represents the words in a collection of \( D \) documents as mixtures of \( K \) “topics”, which are multinomials over a vocabulary of size \( V \). In the case of LDA, for each document \( d_t \), a multinomial over topics is sampled from a Dirichlet prior with parameters \( \alpha \). The probability \( p(w|z=k) \) of a term \( w \), given the topic \( k \), is represented by \( \phi_{k,t} \). We refer to the complete \( K \times V \) matrix of word-topic probabilities as \( \Phi \). The multinomial parameters \( \phi_k \) are again drawn from a Dirichlet prior parametrized by \( \beta \).

Each observed term \( w \) in the collection is drawn from a multinomial for the topic represented by a discrete hidden indicator variable \( z_i \). For simplicity in the mathematical development and notation, we assume symmetric Dirichlet priors but the extension to the asymmetric case is straightforward. Hence, the values of \( \alpha \) and \( \beta \) are model hyper-parameters.

We extend LDA by adding an extra plate denoting the coherent text segments of a document. In the rest, without loss of generality we use sentences as coherent segments. A finer level of granularity can be achieved though, by analysing the structure of sentences and using phrases as such segments. The graphical representation of the senLDA model is shown in Figure 1 and the generative process of a document collection using senLDA is described in Algorithm 1.
the hidden topic variables. We now derive the Gibbs sampler equations by estimating

\[
\text{For inference, we use a collapsed Gibbs sampling method [3]. We now derive the Gibbs sampler equations by estimating the hidden topic variables.}
\]

In senLDA the joint distribution can be factored:

\[
p(w, z|\alpha, \beta) = p(w|z, \beta)p(z|\alpha)
\]

because the first term is independent of \(\alpha\) and the second from \(\beta\). After standard manipulations one arrives at

\[
p(\vec{z}, \vec{w}|\alpha, \beta) = \frac{\prod \Delta(\vec{n}_s + \beta)}{\prod \Delta(\vec{n}_m + \alpha)} \quad \text{for} \quad i = 1, \ldots, K
\]

where \(\Delta(\vec{x})\) can be seen as a multidimensional extension of the beta function: \(B(x_1, x_2) = \Delta(P\{x_1, x_2\})\). To calculate the full conditional we full account to the structure of the document \(d\) and the fact that \(\vec{w} = \{\vec{w}_{d1}, \ldots, \vec{w}_{dS}\}\), \(\vec{z} = \{\vec{z}_{s1}, \ldots, \vec{z}_{sS}\}\). The subscript \(s\) in \(\vec{w}, \vec{z}\) denotes the words and the topic respectively of sentence \(s\). For the full conditional of topic \(k\) we have:

\[
p(z_s = k|\vec{z}_{s-}, \vec{w}) = \frac{p(w, z_s|\alpha, \beta)}{p(w, \vec{z}_{s-}, \vec{w})} = \frac{p(w, z_s)}{p(w, \vec{z}_{s-}, \vec{w})} = \frac{\Delta(\vec{n}_s + \beta)}{\Delta(\vec{n}_m + \alpha)} \Delta(\vec{n}_s - \beta + \vec{n}_m + \alpha)
\]

For the first term of equation Eq. (3) we have:

\[
\Delta(\vec{n}_s + \beta) = \prod_{w \in \mathcal{W}} \Gamma(\vec{n}_s(w) + \beta) = \prod_{w \in \mathcal{V}} \Gamma(\vec{n}_s(w) + \beta)
\]

\[
\Delta(\vec{n}_m + \alpha) = \prod_{m \in \mathcal{M}} \Gamma(\vec{n}_m + \alpha) = \prod_{m \in \mathcal{M}} \Gamma(\vec{n}_m + \alpha)
\]

\[
\Delta(\vec{n}_s - \beta + \vec{n}_m + \alpha) = \prod_{w \in \mathcal{V}} \Gamma(\vec{n}_s(w) - \beta + \vec{n}_m + \alpha)
\]

\[
= \prod_{w \in \mathcal{V}} (\vec{n}_s(w) + \beta) \cdots (\vec{n}_s(w) + \beta + (n_{s,w} - 1))
\]

\[
= \prod_{w \in \mathcal{V}} \Gamma\left(\sum_{w \in \mathcal{V}}(n_{k,w} + \beta)\right) \cdots \left(\sum_{w \in \mathcal{V}}(n_{k,w} + \beta + (N_{k,w} - 1))\right)
\]

\[
B
\]

Here, for the generation of \(A\) and \(B\) we used the recursive property of the \(\Gamma\) function: \(\Gamma(x + m) = (x + m - 1)(x + m - 2) \cdots (x + 1)\Gamma(x)\); \(w\) is a term that can occur many times in a sentence and \(n_{s,w}\) denotes the term frequency \(w\) in sentence \(s\) given that the sentence \(s\) belongs to topic \(k\); \(N_{k,w}\) denotes how many words of sentence \(s\) belong to topic \(t\).

The development of the second factor in the final step of Eq. (3) is similar to the LDA calculations with the difference that the counts of topics per document are calculated given the allocation of sentences to topics and not the allocation of words to topics. This yields:

\[
p(z_s = k|\vec{z}_{s-}, \vec{w}) = \frac{n_{k,s} + \alpha}{\sum_{k \in \mathcal{K}} n_{k,s} + \beta + \alpha}
\]

\[
\times \prod_{w \in \mathcal{V}} \Gamma\left(\sum_{w \in \mathcal{V}}(n_{k,w} + \beta)\right) \cdots \left(\sum_{w \in \mathcal{V}}(n_{k,w} + \beta + (N_{k,w} - 1))\right)
\]

where \(n_{m,s}\) denotes the number of times that topic \(k\) has been observed with a sentence from document \(d\), excluding the sentence currently sampled. Note that Eq. (5) reduces to the known LDA collapsed Gibbs sampling inference equations [4] if the coherent text spans are reduced to words.

3. EMPIRICAL RESULTS

We conduct experiments to verify the applicability and evaluate the performance of senLDA compared to LDA. The process is divided into two steps: (i) the training phase, where the model is trained to learn the model parameters, and (ii) the inference phase that is for new, unseen documents their topic distributions are estimated. We use the Gibbs sampling inference approach given by Eq. (5). The hyper-parameters \(\alpha\) and \(\beta\) are set to \(\frac{1}{K}\), with \(K\) being the number of topics. Table 1 shows the datasets we used. They come from the publicly available collections of Wikipedia [6] and PubMed [7]. The first four datasets (WikiTrain and PubMedTrain) were used for learning the topic model parameters; they differ in their respective size. Also, the vocabulary of the PubMed datasets is significantly larger due to the medical terms used in those documents. During preprocessing we applied lower-casing, stop-word removal and lemmatization using the WordNet Lemmatizer.\(^1\)

\(^1\)Our code will be made available to foster reproducibility.
Intrinsic evaluation

Topic model evaluation has been the subject of intense research. For intrinsic evaluation we report here perplexity [1], which is probably the dominant measure for topic models evaluation in the bibliography. The perplexity of \( d \) held out documents given the model parameters \( \vec{\vartheta} \) is defined as the reciprocal geometric mean of the token likelihoods of those data, given the parameters of the model:

\[
p(w_{\text{heldOut}}) = \exp \left( -\frac{\sum_{i=1}^{d} \sum_{j=1}^{w_{i,j}} \log p(w_{i,j}|\vec{\vartheta})}{\sum_{i=1}^{d} \sum_{j=1}^{w_{i,j}}} \right)
\]

Note that \( senLDA \) samples per sentence and thus results in less flexibility at the word level that perplexity is calculated. Even though, the comparison between \( senLDA \) and LDA, at word level using perplexity, gives insights in the relative merits of the the proposed model.

Figure 2 depicts the ratio between the perplexity values achieved using \( senLDA \) and LDA. Values higher (resp. lower) than one signify that \( senLDA \) achieves lower (resp. higher) perplexity than LDA. The figure demonstrates that in the first iterations before convergence of both models, \( senLDA \) performs better. What is more, \( senLDA \) converges after only around 30 iterations, whereas LDA converges after 160 iterations on Wikipedia and 200 iterations on the PubMed datasets respectively. The shaded area highlights the period while \( senLDA \) performs better. It is to be noted, that although competitive, \( senLDA \) does not outperform LDA given unlimited time resources. However, that was expected since for \( senLDA \) the training instances are sentences, thus the model’s flexibility is restricted when a word-based evaluation measure is considered.

An important difference between both models however, lies in the way they converge. From Figure 2 it is clear that \( senLDA \) converges faster. We highlight this by providing exact timings for the first 25 iterations of the models (column “Timing” of Table 1) on a machine using an Intel Xeon CPU E5-2643 v3 at 3.40GHz. For both models we use our own Python implementations with the same speed optimisations. Using “WikiTrain2” and 125 topics, for 25 iterations the \( senLDA \) needs 332 secs, whereas LDA needs 434 sec., an improvement of 30%. Furthermore, comparing the convergence \( senLDA \) needs 332 secs (25 iterations ) whereas LDA needs more than 2770 secs (more than 160 iterations) making \( senLDA \) more than 8 times faster. Similarly for the “PubMedTrain2” dataset which is more complex due to its larger vocabulary size, \( senLDA \) converges around 12 (an order of magnitude) times faster. Note that \( senLDA \)’s fast convergence is a strong advantage and can be highly appreciated in different application scenarios where unlimited time resources are not available.

Extrinsic evaluation

Previous studies have shown that perplexity does not always agree with human evaluations of topic models [1] and it is recommended to evaluate topic models on real tasks. To better support our development for \( senLDA \) applicability we also evaluate it using text classification as the evaluation task. Details about the document collections we use for learning the topic model parameter (WikiTrain*, PubMedTrain*) and classification purposes (datasets Wiki*, PubMed*) are shown in Table 1. For text classification, each document is represented by its topic distribution, which is the vectorial input to Support Vector Machines (SVMs). The classification collections are split on train/test (75%/25%) parts. The SVM regularization hyper-parameter \( \lambda \) is selected from \( \lambda \in [10^{-4}, \ldots, 10^4] \) using 5-fold cross-validation on the training part of the classification data. The PubMed testsets are multilabel, that is each instance is associated with several classes, 1.4 in average in the sets of Table 1. We attacked the multilabel problem for the SVMs simply by obtaining the classes with positive distance from the separating decision hyperplane. To assess the classification performance, we report the \( F_1 \) evaluation measure, which is the harmonic mean of precision and recall.

The classification performance on \( F_1 \) measure for the different classification datasets is shown in Figure 3. First note that in the majority of the classification scenarios, \( senLDA \) outperforms LDA. In most cases, the performance difference increases when the larger train sets (“WikiTrain2” and “PubMedTrain2”) are available. For instance, in the second line of figures with the PubMed classification experiments, increasing the topic models’ training data benefits both LDA and \( senLDA \) , but \( senLDA \) still performs better. More importantly though and in consistence with the perplexity experiments, the advantage of \( senLDA \) remains: the faster \( senLDA \) convergence benefits the classification performance. The \( senLDA \) curves are steeper in the first training iterations and stabilize after roughly 25 iterations when the model converges. We believe that assigning the latent topics to coherent groups of words such as sentences results in document representations of finer level. In this sense, working with coherent text spans benefits classification performance since text spans larger than single words can capture and

---

| Documents | \(|V|\) | Classes | Timing (sec) |
|-----------|-------|---------|--------------|
| WikiTrain1 | 10,000 | 46,051 | - | 182/271 |
| WikiTrain2 | 30,000 | 65,820 | - | 332/434 |
| PubMedTrain1 | 10,000 | 55,115 | - | 304/433 |
| PubMedTrain2 | 60,000 | 150,440 | - | 1830/2799 |
| Wiki37 | 7,248 | 40,173 | 25 | - |
| Wiki46 | 3,657 | 27,914 | 46 | - |
| PubMed50 | 9,035 | 47,199 | 50 | - |

Table 1: Description of the data used after preprocessing. “Timing” refers to the 25 first training iterations with the left (resp. right) values corresponding to \( senLDA \) (resp. LDA).

---

Figure 2: The ratio of perplexities of \( senLDA \) and LDA calculated on Wiki37 and PubMed25. The topic models are trained with 125 topics.
express the document’s content more efficiently.

To investigate the correlation of topic model representations learned on different levels of text, we report the classification performance using as document representations the concatenation of a document’s topic distributions output by LDA and senLDA. For instance, the vectorial representation of a document when there are \( K = 25 \) topics found by both models is a vector of 50 dimensions. The resulting concatenated representation is denoted by “senLDA+” in Figure 3. As it can be seen, “senLDA+” performs better compared to both LDA and senLDA. Its performance combines the advantages of both models: at the first iterations it is as steep as the senLDA representations and in the later iterations benefits by the LDA convergence to outperform the simple senLDA representation. Hence, the concatenation of the two distributions creates a richer representation where the two models contribute complementary information that achieves the best classification performance. Achieving the optimal performance using those representations suggests that the relaxation of the independence assumptions of the text structural units can be beneficial; this is also one of the contributions of this work.

4. CONCLUSION

We proposed the senLDA model, an extension of LDA where topics are sampled per coherent text spans. This resulted in very fast convergence and good classification and perplexity performance. LDA and senLDA differ in that the second assumes a very strong dependence between the words of sentences whereas the first assumes independency between words of sentences and documents in general. In our future research, we plan to investigate this dependency using language processing techniques such as parsing. Our goal is to further adapt the sampling process of topic models to cope with the rich, text inner structure.

5. REFERENCES