***

A reimplementation and evaluation of online Latent Dirichlet Allocation

Christoph Schneider

Abstract

In this work i implement online Latent Dirichlet Allocation (LDA). I evaluate the effectiveness of LDA by comparing LDA generated document vectors to tf-idf document vectors as inputs for a classification task. L

1 Introduction

In the field of Natural Language Processing (NLP) one often finds oneself confronted with the task of representing text documents in a way that can serve as a useful input for subsequent tasks such as classification or sentiment analysis. One method to represent a document as a vector in a relatively low-dimensional space is Latent Dirichlet Allocation (LDA) (Blei et al., 2003). LDA is a model in which each document in a text corpus is modeled as a probability distribution over topics. This probability distribution can serve as a reresenation of the document in low dimensional vector space. Latent Dirichlet Allocation can be trained using Variational Inference (Blei et al., 2017). In batch LDA as described by (Blei et al., 2003) all documents in the corpus have to be processed before a parameter update can be performed. In order to speed up the process Hoffman et al. (2010) have proposed online LDA, which can be used to perform a parameter update after only a minibatch of documents has been processed, thus greatly increasing the training speed of LDA. In this term project i reimplement online LDA. In order to evaluate its effectiveness, i train an LDA model on a corpus consisting of posts from 70 highly active subreddits. I then use the LDA model to obtain document vectors of the reddit posts in my corpus. I use these vectors to train a Random Forest Classifier (Breiman, 2001) to sort documents by subreddit and measure the accuracy of this classifier. As a comparison i use tf-idf vectors of the same documents as input for a Random Forest Classifier and evaluate which vectorization method gives superior accuracy. Conversations in a subreddit typically revolve around only a few topics, so it stands to reason that using a topic model to project reddit posts into vector space would lead to good results.

2 Latent Dirichlet Allocation

Latent Dirichlet Allocation is a model that assumes documents to be generated by a probabilistic process. The process can be described as follows: D is the number of documents to be generated. K is the number of topics.

- Choose $\theta_d \sim Dir(\alpha)$ where $d \in (1, ..., D)$ and $Dir(\alpha)$ is Dirichlet distribution.
- Choose $\beta_k \sim Dir(\eta)$ where $k \in (1, ..., K)$
- For each position (d,j) which is the position of the the j-th word in the d-th document:
- Choose $z_{dj} \sim Multinomial(\theta_d)$
- Choose $w_{dj} \sim Multinomial(\beta_{z_{dj}})$

Here θ_d is the document vector for document d. A document vector is a probability distribution over topics. β_k is the topic vector for topic k. A topic vector is a probability distribution over words in the corpus vocabulary. We sample a topic $z_{dj} \in (1, ..., K)$ for word j in document d from θ_d. Then we sample a word w_{dj} from $\beta_{z_{dj}}$ which is the topic vector for the topic we sampled from the document vector. α and η are hyperaparameters. Figure 1 shows the LDA model in plate notation.

There are several approaches to training such a model. The approach chosen by (Blei et al., 2003) and (Hoffman et al., 2010) is variational inference.

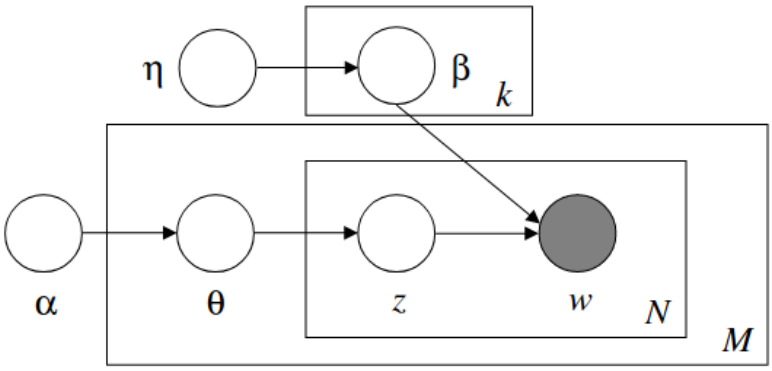

Figure 1: Graphical represantation of an LDA model (Blei et al., 2003)

3 Online Learning for LDA

In order to find good parameters θ β and z for our model, we need to find the posterior probability

$$p(\theta, \beta, z|w, \alpha, \eta) = \frac{p(\theta, \beta, z, w|\alpha, \eta)}{p(w|\alpha, \eta)} \quad (1)$$

so that we can set

$$\theta, \beta, z = argmax_{\theta,\beta,z} p(\theta, \beta, z|w, \alpha, \eta) \quad (2)$$

Since we can not calculate the posterior probability p analytically (Blei et al., 2003), we have to find a probability distribution q to approximate p. We choose q such that it minimizes the Kullback-Leibler Divergence (Kullback and Leibler, 1951) between q and p.

$$q(z, \beta, \theta) = argmin_q KL(q||p) \quad (3)$$

$q(z, \beta, \theta)$ is indexed by a set of parameters that are optimized to maximize the Evidence Lower Bound (ELBO) which is equivalent to minimizing KL-Divergence (Blei et al., 2017)

$$\begin{aligned} &log\, p(w|\alpha, \eta) \leq L(w, \phi, \gamma, \lambda) \\ &= E_q[log\, p(w, z, \theta, \beta|\alpha, \eta)] - E_q[log\, q(z, \beta, \theta)] \end{aligned} \quad (4)$$

We parameterize z by ϕ, θ by γ and β by λ

Hoffman et al. (2010) show how to optimize L by using coordinate ascent on the variational parameters γ, ϕ and λ They arrive at parameter updates

$$\phi_{dwk} \propto exp(E_q[log\, \theta_{dk}] + E_q[log\, \beta_{kw}]) \quad (5)$$

$$\gamma_{dk} = \alpha + \sum_w n_{dw}\, \phi_{dwk} \quad (6)$$

here n_{dw} is the word-count for word w in document d. The topic updates are

$$\lambda_{kw} = \eta + \frac{D}{S} \sum_s n_{tsk}\, \phi_{tskw} \quad (7)$$

Where S is the number of documents in a mini-batch.

With these update rules Hoffman et al. (2010) derive algorithm 1 to train LDA online.

4 Data

As documents i used reddit posts from 70 subreddits. Tokens in each document were converted to lowercase and lemmatized. Stopwords were ignored. Each document was assigned a label, indicating the subreddit it originally came from. The corpus was split into train, validation and test set. I obtained the reddit posts from the Convokit Reddit corpus (Chang et al., 2020). The vocabulary consists of the 10000 most common lemmata in the corpus.

Table 1 shows dataset statistics for the Corpus.

5 Experiments

LDA can be used to obtain low dimensional vectors that contain the statistically relevant information in the documents of our corpus. From now on i will refer to document vectors obtained by LDA as LDA-vectors. The metric by which i will judge LDA is the effectiveness of LDA-vectors as inputs for a subsequent task. The overall model consists of an LDA model which is trained on the training set. From the trained LDA model i obtain document vectors for the train, test and validation set. I use the vectors from the training set as inputs for a Random Forest Classifier and train it to sort documents by subreddit. I tune hyperparameters for the

Algorithm 1: Online Variational Inference for LDA (Hoffman et al., 2010)

Define $\rho_t = (\tau + t)^{-\kappa}$
initialize λ randomly
while *stopping criterion not met* **do**
  for *d = 1 to D* **do**
    initialize $\gamma_{dk} = 1$ (The constant 1 is arbitrary)
    while $\frac{1}{k}\sum_k |change\ in\ \gamma_{dk}| \leq threshold$ **do**
      Set $\phi_{dwk} \propto exp(E_q[log\,\theta_{dk}] + E_q[log\,\beta_{kw}])$
      Set $\gamma_{dk} = \alpha + \sum_w n_{dw}\,\phi_{dwk}$
    end
  end
  Compute $\hat{\lambda_{kw}} = \eta + \frac{D}{S}\sum_s n_{tsk}\,\phi_{tskw}$
  Set $\lambda = (1 - \rho_t)\lambda + \rho_t\lambda$
end

	train	test	val
documents	22843	2855	2855
avg-docs-per category	408	408	408

Table 1: Corpus statistics

Parameter			
τ	1	**256**	1024
κ	**0.5**	0.75	1
topics	50	75	**100**
threshold	**0.001**		
batch-size	**512**	1024	
vocab-size	**10000**		
epochs	**10**		

Table 2: Hyperparameters used for Grid-Search. Best Parameters are bold.

LDA model using grid-search on the validation set and report the best hyperparameter combination in table 2.

As a baseline comparison i train a second Random Forest model on tf-idf-vectors (Salton and McGill, 1986) of the training set. The results for this experiment can be seen in table 3.

6 Results and Discussion

The Random Forest Classifier achieved accuracy of 41% on the test set using LDA-vectors and 57% using tf-idf vectors. In summary one can say that LDA is able to represent the information in a document in such a way that important statistical information is preserved and can be used for subsequent tasks. In this particular task, LDA-vectors are inferior to tf-idf vectors, which of course does not mean that the succes of LDA in the field of topic modeling is undeserved.

References

David Blei, Andrew Ng, and Michael Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. 2017. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Leo Breiman. 2001. Machine learning, volume 45, number 1 - springerlink. *Machine Learning*, 45:5–32.

Jonathan P. Chang, Caleb Chiam, Liye Fu, Andrew Z. Wang, Justine Zhang, and Cristian Danescu-Niculescu-Mizil. 2020. Convokit: A toolkit for the analysis of conversations.

Matthew Hoffman, David Blei, and Francis Bach. 2010. Online learning for latent dirichlet allocation. volume 23, pages 856–864.

S. Kullback and R. A. Leibler. 1951. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86.

Gerard Salton and Michael J. McGill. 1986. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., USA.