NPFL097 Assignment 1
Latent Dirichlet Allocation

David Mareček

November 2020

In this assignment, you will get a document collection comprising around 18000 news posts divided to training, development, and test sets. Even though these sets are labeled by topics, you will not use these annotations. Your task is to implement the unsupervised Latent Dirichlet Allocation (LDA) topic model (presented in the previous lectures), and to provide some characteristics of the model you learned. The preferred programming language is Python.

Send me your final source code and commented graphs and results by email: `marecek@ufal.mff.cuni.cz`.

Data

You can access the dataset using the `sklearn.datasets.fetch_20newsgroups` function. The preprocessing steps (tokenization, lemmatization, stemming and dictionary filtration) needed for running the LDA algorithm are given in the enclosed script `lda-data.py`.

Tasks:

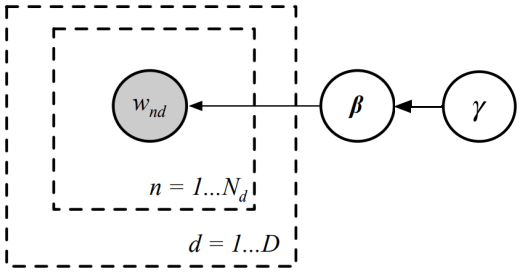
1. Implement the Latent Dirichlet allocation topic model, as described in the previous lectures. Set the hyperparameters $\alpha = 0.1$, $\gamma = 0.1$ and set number of topics $K = 20$. *(4 pts)*

2. Plot the distribution over topics for one chosen document after initialization and after 1st, 2nd, 5th, 10th, 20th, and 50th iteration. Comment on these. *(1 pt)*

3. Compute the word entropy for each of the topics as a function of the number of Gibbs iterations. *(1 pt)*

4. Show histograms of the most frequent 20 words of three chosen topics after 50 Gibbs iterations. *(1 pt)*

5. Preprocess the test data in the same way as the training data. For filtration, use the dictionary from the training data. Compute the per-word perplexity of the test data for the state after 50 Gibbs iterations. Compare it to a simple bayesian model not using any hidden variables and using only one distribution over words β for all documents with symmetric Dirichlet prior with and concentration parameter $\gamma = 0.1$. *(2 pts)*



6. Try to change the number of topics K, the hyperparameters α and γ and the number of Gibbs iterations. How the performance changes? *(1 pt)*