Project Proposal: Transfer Learning Between Inference Methods for Topic Models

Tommy Jones

03/03/2019

Background

The two primary methods for training Latent Dirichlet Allocation (LDA) are Gibbs sampling and variantional Bayesian inference (VB). Gibbs sampling is significantly more computationally expensive than VB. Yet Gibbs has theoretical guarantees to converge to the correct posterior distribution. VB estimates parameters based on an approximation of the posterior. It does not have the same theoretical guarantees as Gibbs.

Recently, researchers have developed methoeds using variational autoencoders (VAEs) as a computational method for VB. Traditionally, using VAEs for LDA has remained elusive. The distributions involved in LDA do not retain the form needed for typical VAEs. Yet in 2017, two researchers developed a viable method for using VAEs to train LDA and LDA-like models. They refer to this method as "autoencoded variational inference for topic models" or AVITM.

Their approach appears to have two key advantages. First, it is at least as fast as traditional VB for training topic models and even faster for inference on new documents. Second, it is a "black" box. Gibbs and traditional VB require one to derive new equations for even small model tweaks, a considerable barrier. AVITM, like most neural network-based methods, does not require such re-derivations.

In spite of these advantages, Gibbs is unassailable in one key regard. It is guaranteed to converge to the true posterior. (To clarify: Gibbs does not guard against an analyst pathologically misspecifying a model. Rather, it is guaranteed to converge to the model as specified. VB and AVITM don't even meet that guarantee.) Perhaps there is a way to leverage the advantages of both methods for LDA?

Objectives

I propose exploring the effects of transfer learning between Gibbs sampling and AVITM. Both methods estimate the same parameters: Θ whose rows represent $P(\text{topic}_k|\text{document}_d)$ and Φ whose rows represent $P(\text{word}_v|\text{topic}_k)$. One could theoretically use estimates of Φ and Θ trained using one method in the other. Such scenarios may include:

- Pre-training using AVITM for speed and then switching to Gibbs to finish off with theoretical guarantees.
- Taking a model trained with Gibbs (purely Gibbs or using the hybrid approach as above) but using AVITM for inference on new documents (again for a speed advantage).

If this method works, the following should hold. For the first case, abov: pre-training with AVITM and switching to Gibbs should result in a converged model (measured with some convergence statistic such as the Gelman-Rubin statistic) in less time than Gibbs alone. For the second case, inference using AVITM should result in similar estimates as Gibbs but get there with lower time complexity. In both cases, one would have to use the same hardware.

Methods

Methods for this project fall into four categories: computational methods for Gibbs sampling LDA, computational methods for AVITM LDA, data used for experiments, and evaluation metrics. For Gibbs sampling, I intend to use the R package `textmineR`.

I intend to implement AVITM using three different frameworks. My ultimate objective is to have a sequential implementation in C++, called from R, to directly compare it to `textmineR`'s Gibbs implementation. However, first I will implement it in Keras (using R's `keras` package) following the example at https://github.com/nzw0301/keras-examples/blob/master/prodLDA.ipynb. Next, I will implement it natively in R since I understand it well. Finally, I intend to implement AVITM in C++ using the `Rcpp` framework that provides a C++ interfact through R. (This is how Gibbs is implemented in textmineR.)

Data used for experiments may come from any number of sources and the choice is somewhat arbitrary. A common data set is the 20 Newsgroups data set, which was used in the AVITM paper. I have worked with grant abstracts from NIH's database quite a bit in the past. I've also worked with NHTSA's automobile complaints database in the past. I am also considering a corpus of papers from the last few years of the Association for Computational Linguistics's annual conference. (A very meta choice.) Since we are comparing computational methods for topic models, it is important to choose a corpus that does not have properties that result in odd modeling behavior. From the three examples above, I'm inclined to say that the 20 news groups documents are often too short, while whole research papers from ACL may be too long. I have not decided yet, but do not have a shortage of potential sources.

Evaluation metrics for topic models can be problematic. As a result, researchers use heuristic "coherence" metrics to measure topic quality. My aim is not to say that Gibbs or AVITM results in a better model. Rather, I would like to show that it is possible to perform tranfer learning without a significant difference in results. Therefore, I will use three metrics: coherence to compare overall topic quality, R-squared for topic models to compare goodness-of-fit [1], and a convergence statistic as mentioned above.

Relevant Literature

The most relevant work of course is the AVITM paper published in 2017 by Akash Srivastava and Charles Sutton. Most papers related to transfer learning for topic models focus on transfering parameters trained on one corpus to another corpus. They are largely silent about tranferring parameters between computational methods as I propose. There is one slight exception.

In 2009, Asuncion et al. compared Gibbs samping, VB, and maximum a-posteriori (MAP) estimation for LDA. They find that these methods largely differ in the amount of smoothing applied to the counts. However, when they optimize hyperparameters, the difference is performance between methods diminish. This implies that for my research, I should not expect that AVITM and Gibbs would result in the same model when trained independently.

Timeline

The tasks I have to accomplish are: (1) Literature review (in progress), (2) Implement AVITM, and (3) Perform experiments.

As of this writing, I have 6 - 8 weeks to accomplish this, depending on when I present. The biggest challenge will be implementing AVITM. I expect I would need a week to do the literature review and a week to code and run my experiments. I have identified a path to implementing AVITM. However, since it is the biggest technical risk of this project, I am going to prioritize this part of the project. If I plan on 6 weeks, then I have 4 weeks to implement a satisfactory version of AVITM.

[1] I have a working paper which derives the coefficient of variation for topic models. The paper may be viewed here.