Neural Models for Documents with Metadata: Supplementary Material

Dallas Card[1] **Chenhao Tan**[2] **Noah A. Smith**[3]
[1]Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA, 15213, USA
[2]Department of Computer Science, University of Colorado, Boulder, CO, 80309, USA
[3]Paul G. Allen School of CSE, University of Washington, Seattle, WA, 98195, USA
dcard@cmu.edu chenhao.tan@colorado.edu
nasmith@cs.washington.edu

1 Deriving the ELBO

The derivation of the ELBO for our model is given below, dropping explicit reference to $\mathbf{\Phi}$ and α for simplicity. For document i,

$$\log\, p(\boldsymbol{w}_i, \boldsymbol{y}_i \mid \boldsymbol{c}_i) = \log \int_{\boldsymbol{r}_i} p(\boldsymbol{w}_i, \boldsymbol{y}_i, \boldsymbol{r}_i \mid \boldsymbol{c}_i) d\boldsymbol{r}_i \tag{1}$$

$$= \log \int_{\boldsymbol{r}_i} p(\boldsymbol{w}_i, \boldsymbol{y}_i, \boldsymbol{r}_i \mid, \boldsymbol{c}_i) \frac{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)}{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} d\boldsymbol{r}_i \tag{2}$$

$$= \log \left(\mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\frac{p(\boldsymbol{w}_i, \boldsymbol{y}_i, \boldsymbol{r}_i \mid \boldsymbol{c}_i)}{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \right] \right) \tag{3}$$

$$\begin{aligned} \geq\ & \mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\log p(\boldsymbol{w}_i, \boldsymbol{y}_i, \boldsymbol{r}_i \mid \boldsymbol{c}_i)\right] \\ & - \mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\log q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)\right] \end{aligned} \tag{4}$$

$$\begin{aligned} =\ & \mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\log p(\boldsymbol{w}_i, \boldsymbol{y}_i \mid \boldsymbol{r}_i, \boldsymbol{c}_i)\right] \\ & + \mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\log p(\boldsymbol{r}_i)\right] \\ & - \mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\log q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)\right] \end{aligned} \tag{5}$$

$$\begin{aligned} =\ & \mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\sum_{j=1}^{N_i} \log p(w_{ij} \mid \boldsymbol{r}_i, \boldsymbol{c}_i) \right] \\ & + \mathbb{E}_{q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)} \left[\log p(\boldsymbol{y}_i \mid \boldsymbol{r}_i, \boldsymbol{c}_i)\right] \\ & - \mathrm{D}_{\mathrm{KL}} \left[q(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i) \mid\mid p(\boldsymbol{r}_i)\right] \end{aligned} \tag{6}$$

2 Model details

The KL divergence term in the variational bound can be computed as

$$\begin{aligned} \mathrm{D}_{\mathrm{KL}}[q_{\mathbf{\Phi}}(\boldsymbol{r}_i \mid \boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i) \| p(\boldsymbol{r}_i)] = \frac{1}{2} \Bigg(& \mathrm{tr}(\mathbf{\Sigma}_0^{-1} \mathbf{\Sigma}_i) \\ + (\boldsymbol{\mu}_i - \boldsymbol{\mu}_0)^\top \mathbf{\Sigma}_0^{-1} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_0) - K + \log \frac{|\mathbf{\Sigma}_0|}{|\mathbf{\Sigma}_i|} & \Bigg) \end{aligned} \tag{7}$$

where $\mathbf{\Sigma}_i = \mathrm{diag}(\boldsymbol{\sigma}_i^2(\boldsymbol{w}_i, \boldsymbol{c}_i, \boldsymbol{y}_i))$, and $\mathbf{\Sigma}_0 = \mathrm{diag}(\boldsymbol{\sigma}_0^2(\alpha))$.

3 Practicalities and Implementation

As observed in past work, inference using a VAE can suffer from component collapse, which translates into excessive redundancy in topics (i.e., many topics containing the same set of words). To mitigate this problem, we borrow the approach used by Srivastava and Sutton (2017), and make use of the Adam optimizer with a high momentum, combined with batchnorm layers to avoid divergence (Ioffe and Szegedy, 2015). Specifically, we add batchnorm layers following the computation of $\boldsymbol{\mu}$, $\log \boldsymbol{\sigma}^2$, and $\boldsymbol{\eta}$.

This effectively solves the problem of mode collapse, but the batchnorm layer on $\boldsymbol{\eta}$ introduces an additional problem, not previously reported. At test time, the batchnorm layer will shift the input based on the learned population mean of the training data; this effectively encodes information about the distribution of words in this model that is not captured by the topic weights and background distribution. As such, although reconstruction error will be low, the document representation $\boldsymbol{\theta}$, will not necessarily be a useful representation of the topical content of each document. In order to alleviate this problem, we reconstruct $\boldsymbol{\eta}$ as a convex combination of two copies of the output of the generator network, one passed through a batchnorm layer, and one not. During training, we then gradually anneal the model from relying entirely on the component passed through the batchnorm layer, to relying entirely on the one that is not. This ensures that the the final weights and document representations will be properly interpretable.

Note that although ProdLDA (Srivastava and Sutton, 2017) does not explicitly include a background term, it is possible that the batchnorm layer applied to $\boldsymbol{\eta}$ has a similar effect, albeit one that is not as easily interpretable. This annealing process avoids that ambiguity.

4 Data

All datasets were preprocessed by tokenizing, converting to lower case, removing punctuation, and dropping all tokens that included numbers, all tokens less than 3 characters, and all words on the stopword list from the snowball sampler.[1] The vocabulary was then formed by keeping the words that occurred in the most documents (including train and test), up to the desired size (2000 for 20 newsgroups, 5000 for the others). Note that these small preprocessing decisions can make a large difference to perplexity. We therefore include our preprocessing scripts as part of our implementation so as to facilitate easy future comparison.

For the UIUC Yahoo answers dataset, we downloaded the documents from the project webpage.[2] However, the file that is available does not completely match the description on the website. We dropped *Cars and Transportation* and *Social Science* which had less than the expected number of documents, and merged *Arts* and *Arts and Humanities*, which appeared to be the same category, producing 15 categories, each with 10,000 documents.

5 Experimental Details

For all experiments we use a model with 300-dimensional embeddings of words, and we take f_e to be only the element-wise softplus nonlinearity (followed by the linear transformations for $\boldsymbol{\mu}$ and $\log \boldsymbol{\sigma}^2$). Similarly, f_y is a linear transformation of $\boldsymbol{\theta}$, followed by a softplus layer, followed by a linear transformation to the size of the output (the number of classes). During training, we set S (the number of samples of $\boldsymbol{\epsilon}$) to be 1; for estimating the ELBO at on test documents, we set $S = 20$.

For the unsupervised results, we use the same set up as (Srivastava and Sutton, 2017): Adam optimizer with $\beta_1 = 0.99$, learning rate $= 0.002$, batch size of 200, and training for 200 epochs. The setup was the same for all datasets, except we only trained for 150 epochs on Yahoo answers because it is much larger. For LDA, we updated the hyperparameters every 10 epochs.

For the external evaluation of NPMI, we use the co-occurrence statistics from all New York Times articles in the English Gigaword published from the start of 2000 to the end of 2009, processed in the same way as our data.

Model	Ppl. ↓	NPMI (int.) ↑	NPMI (ext.) ↑	Sparsity ↑
LDA	**810**	0.20	0.11	0
SAGE	867	0.27	0.15	**0.71**
NVDM	1067	0.18	0.11	0
SCHOLAR - B.G.	928	0.17	0.09	0
SCHOLAR	921	**0.35**	0.16	0
SCHOLAR + W.V.	955	0.29	**0.17**	0
SCHOLAR + REG.	1053	0.25	0.13	0.43

Table 1: Performance of various models on the 20 newsgroups dataset with 20 topics and a 2,000-word vocabulary.

For the text classification experiments, we use the `scikit-learn` implementation of logistic regression. We give it access to the same input data as our model (using the same vocabulary), and tune the strength of l_2 regularization using cross-validation. For our model, we only tune the number of epochs, evaluating on development data. Our models for this task did not use regularization or word vectors.

6 Additional Experimental Results

In this section we include additional experimental results in the unsupervised setting.

Table 1 shows results on the 20 newsgroups dataset, using 20 topics with a 2,000-word vocabulary. Note that these results are not necessarily directly comparable to previously published results, as preprocessing decisions can have a large impact on perplexity. These results show a similar pattern to those on the IMDB data provided in the main paper, except that word vectors do not improve internal coherence on this dataset, perhaps because of the presence of relatively more names and specialized terminology. Also, although the NVDM still has worse perplexity than LDA, the effects are not as dramatic as reported in (Srivastava and Sutton, 2017). Regularization is also more beneficial for this data, with both SAGE and our regularized model obtaining better coherence than LDA. The topics from SCHOLAR for this dataset are also shown in Table 3.

Table 2 shows the equivalent results for the Yahoo answers dataset, using 250 topics, and a 5,000-word vocabulary. These results closely match those for the IMDB dataset, with our model having higher perplexity but also higher internal coherence than LDA. As with IMDB, the use of word vectors improves coherence, both internally and externally, but again at the cost of worse perplexity. Surpris-

[1] `snowball.tartarus.org/algorithms/english/stop.txt`

[2] `cogcomp.org/page/resource_view/89`

Model	Ppl. ↓	NPMI (int.) ↑	NPMI (ext.) ↑	Sparsity ↑
LDA	**1035**	0.29	0.15	0
NVDM	4588	0.20	0.09	0
SCHOLAR - B.G.	1589	0.27	**0.16**	0
SCHOLAR	1596	0.33	0.13	0
SCHOLAR + W.V.	1780	**0.37**	0.15	0
SCHOLAR + REG.	1840	0.34	0.13	0.44

Table 2: Performance of various models on the Yahoo answers dataset with 250 topics and a 5,000-word vocabulary. SAGE did not finish in 72 hours so we omit it from this table.

ingly, our model without a background term actually has the best *external* coherence on this dataset, but as described in the main paper, these tend to give high weight primarily to common words, and are more repetitive as a result.

References

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of ICML*.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. In *Proceedings of ICLR*.

NPMI	Topic
0.77	turks armenian armenia turkish roads escape soviet muslim mountain soul
0.52	escrow clipper encryption wiretap crypto keys secure chip nsa key
0.49	jesus christ sin bible heaven christians church faith god doctrine
0.43	fbi waco batf clinton children koresh compound atf went fire
0.41	players teams player team season baseball game fans roger league
0.39	guns gun weapons criminals criminal shooting police armed crime defend
0.37	playoff rangers detroit cup wings playoffs montreal toronto minnesota games
0.36	ftp images directory library available format archive graphics package macintosh
0.33	user server faq archive users ftp unix applications mailing directory
0.32	bike car cars riding ride engine rear bmw driving miles
0.32	study percent sexual medicine gay studies april percentage treatment published
0.32	israeli israel arab peace rights policy islamic civil adam citizens
0.30	morality atheist moral belief existence christianity truth exist god objective
0.28	space henry spencer international earth nasa orbit shuttle development vehicle
0.27	bus motherboard mhz ram controller port drive card apple mac
0.25	windows screen files button size program error mouse colors microsoft
0.24	sale shipping offer brand condition sell printer monitor items asking
0.21	driver drivers card video max advance vga thanks windows appreciated
0.19	cleveland advance thanks reserve ohio looking nntp western host usa
0.04	banks gordon univ keith soon pittsburgh michael computer article ryan

Table 3: Topics from the unsupervised SCHOLAR on the 20 newsgroups dataset, and the corresponding internal coherence values.