Reinforcement Learning for Topic Models

Jeremy Costello and **Marek Z. Reformat**
Department of Electrical and Computer Engineering
University of Alberta
{jeremy1, reformat}@ualberta.ca

Abstract

We apply reinforcement learning techniques to topic modeling by replacing the variational autoencoder in ProdLDA with a continuous action space reinforcement learning policy. We train the system with a policy gradient algorithm REINFORCE. Additionally, we introduced several modifications: modernize the neural network architecture, weight the ELBO loss, use contextual embeddings, and monitor the learning process via computing topic diversity and coherence for each training step. Experiments are performed on 11 data sets. Our unsupervised model outperforms all other unsupervised models and performs on par with or better than most models using supervised labeling. Our model is outperformed on certain data sets by a model using supervised labeling and contrastive learning. We have also conducted an ablation study to provide empirical evidence of performance improvements from changes we made to ProdLDA and found that the reinforcement learning formulation boosts performance.

1 Introduction

The internet contains large collections of unlabeled textual data. Topic modeling is a method to extract information from this text by grouping documents into topics and linking these topics with words describing them. Classical techniques for topic modeling, the most popular being Latent Dirichlet Approximation (LDA) (Blei et al., 2003), have recently begun to be overtaken by Neural Topic Models (NTM) (Zhao et al., 2021).

ProdLDA (Srivastava and Sutton, 2017) is a NTM using a product of experts in place of the mixture model used in classical LDA. ProdLDA uses a variational autoencoder (VAE) (Kingma and Welling, 2013) to learn distributions over topics and words. ProdLDA improved on NVDM (Miao et al., 2016) by explicitly approximating the Dirichlet prior from LDA with a Gaussian distribution and using the Adam optimizer (Kingma and Ba, 2014) with a higher momentum and learning rate.

Perceiving Reinforcement Learning (RL) as probabilistic inference has brought practices of such an inference into the RL field (Dayan and Hinton, 1997) (Levine, 2018). New algorithms using these techniques include MPO (Abdolmaleki et al., 2018) and VIREL (Fellows et al., 2019). MPO optimizes the evidence lower bound (ELBO), which is the same optimization objective used in VAEs.

Inspired by the adoption of probabilistic inference techniques in RL, we look to apply RL techniques to probabilistic inference in the realm of topic models. We use REINFORCE, the simplest policy gradient (PG) algorithm, to train a model which parameterizes a continuous action space, corresponding to the distribution of topics for each document in the topic model. We keep the product of experts from ProdLDA to compute the distribution of words for each document in the topic model.

We additionally improve our topic model by using Sentence-BERT (SBERT) embeddings (Reimers and Gurevych, 2019) rather than bag-of-word (BoW) embeddings, modernizing the neural network (NN) architecture, adding a weighting term to the ELBO, and tracking topic diversity and coherence metrics throughout training. The model architecture is shown in Figure 1. Our method outperforms most other topic models. It is beaten only on some data sets by advanced methods using document labels for supervised learning, while our procedure is fully unsupervised.

2 Related Work

Zhao et al. (2021) provide a survey of NTMs. Variations of VAEs are presented which use different distributions, correlated and structured topics, pre-trained language models, incorporate meta-data, or model on short texts rather than documents. Methods other than VAEs are also used for NTMing,

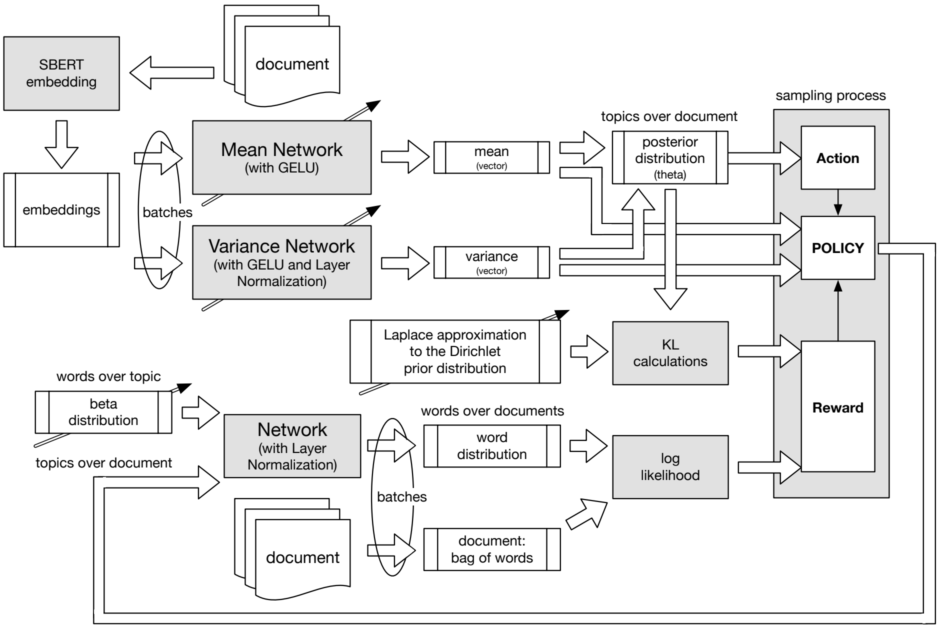

Figure 1: Architecture Diagram: gray boxes - processing; white boxes - models/data/information; arrows across boxes - tune-ability

including autoregressive models, generative adversarial networks, and graph NNs.

Doan and Hoang (2021) compare ProdLDA and NVDM, along with six other NTMs and three classical topic models, in terms of held-out document and word perplexity, downstream classification, and coherence. Scholar (Card et al., 2017), an extension of ProdLDA taking document metadata and labels into account where possible, performed best in terms of coherence. NVDM and NVCTM (Liu et al., 2019), an extension of NVDM which additionally models the correlation between documents, performed best in terms of perplexity and downstream classification. The other NTMs were GSM (Miao et al., 2017), NVLDA (Srivastava and Sutton, 2017), NSMDM (Lin et al., 2019), and NSMTM (Lin et al., 2019). The classical topic models were non-negative matrix factorization (NMF) (Zhao et al., 2017), online LDA (Hoffman et al., 2010), and Gibbs sampling LDA (Griffiths and Steyvers, 2004).

BERTopic (Grootendorst, 2022) and Top2Vec (Angelov, 2020) use dimensionality reduction and clustering to group document embeddings from pre-trained language models into meaningful clusters. Contextualized Topic Models (CTM) (Bianchi et al., 2020a) augments the BoW embeddings used in ProdLDA with SBERT (Reimers and Gurevych, 2019) embeddings, resulting in an improved topic model.

Dieng et al. (2020) develop the embedded topic model (ETM) by using word embeddings to augment a variational inference algorithm for topic modeling. Their method outperforms other topic models, especially on corpora with large vocabularies containing common and very rare words. Nguyen and Luu (2021) augment Scholar (Card et al., 2017) with contrastive learning (Hadsell et al., 2006) and outperform all topic models compared against.

Gui et al. (2019) use RL to filter words from documents, with reward as a combination of the resulting topic model's coherence and diversity, or how few words overlap between topics. Kumar et al. (2021) use REINFORCE (Williams, 1992), a PG RL algorithm, to augment ProdLDA. Their model slightly outperforms ProdLDA in terms of topic coherence.

3 Background

We briefly outline topic models, RL process, KL divergence, and contextual embeddings.

3.1 Topic Models – Approaches

Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is a three-level hierarchical Bayesian model: documents $\rightarrow$ topics $\rightarrow$ words. Each document is a mixture over latent topics, where the topic distribution θ is randomly sampled from a Dirichlet distribution. Each topic is a multinomial distribution over vocabulary words.

Autoencoding Variational Inference for Topic Models (AVITM) (Srivastava and Sutton, 2017) is a neural topic model using a VAE to learn a Gaussian distribution over topics. VAEs use a reparameterization trick (RT) to randomly sample from the posterior distribution to remain fully differentiable. At the time, there was no known RT for Dirichlet distributions, so AVITM used a Gaussian distribution and a Laplace approximation of the Dirichlet prior.

AVITM contains two models: NVLDA and ProdLDA. NVLDA uses the mixture model from LDA to infer a distribution over vocabulary words, while ProdLDA uses a product of experts.

Evidence Lower Bound (ELBO) is the optimization objective for AVITM. ELBO optimization (Jordan et al., 1999) simultaneously tries to maximize the log-likelihood of the topic model and minimize the forward Kullback–Leibler (KL) divergence (Kullback and Leibler, 1951) between the posterior P and prior Q topic distributions.

$$\text{ELBO} = D_{KL}(P||Q) - \textit{log-likelihood} \quad (1)$$

3.2 Topic Models – Evaluation

Topic Coherence is a metric for evaluating topic models. It uses co-occurence in a reference corpus to measure semantic similarity between the top-K words in a topic. Topic model coherence is the average of each topic's coherence.

Normalized pointwise mutual information (NPMI) (Aletras and Stevenson, 2013) was the coherence measure found to correlate best with human judgment (Lau et al., 2014). When computing NPMI, a window size of 20 for co-occurrence counts is used in Srivastava and Sutton (2017), while Dieng et al. (2020) uses full document co-occurrence.

NPMI coherence is calculated for each of the top-K words in a topic and averaged to obtain the coherence for that topic. The overall *topic-coherence* is the average of the coherence for each topic. For a word i, the NPMI coherence is calculated according to Equation 2.

$$\text{NPMI}(w_i) = \sum_{j}^{K-1} \frac{\log \frac{P(w_i,w_j)}{P(w_i)P(w_j)}}{-\log P(w_i, w_j)} \quad (2)$$

where $P(w_i)$ is the probability of word i occurring in a document in the corpus, and $P(w_i, w_j)$ is the probability of words i and j co-occurring in a document in the corpus.

Topic Diversity is another metric for evaluating topic models. It measures the uniqueness of the top-K words across all topics. Dieng et al. (2020) use $K = 25$ for reporting topic diversity.

$$\textit{topic-diversity} = \frac{\textit{number-of-unique-words}}{K * \textit{number-of-topics}} \quad (3)$$

Topic Quality is a topic modeling metric introduced by Dieng et al. (2020).

$$\textit{topic-quality} = \textit{topic-coherence} * \textit{topic-diversity} \quad (4)$$

3.3 Reinforcement Learning

RL is a sequential decision-making framework focused on finding the best sequence of actions executed by an agent. (Sutton and Barto, 2018). An agent takes actions $a \in \mathbf{A}$ to traverse between states $s \in \mathbf{S}$ in an environment, receiving a reward r on each transition. The goal of an RL task is to find the best set of actions —referred to as the policy —which maximizes the reward. RL problems can be episodic, where the agent completes the environment and is reset, or continuing, where the agent continuously traverses the environment without reset. Through traversing the environment, the agent learns a policy π of which actions in each state will maximize return. Return is the cumulative reward received by the agent in an episode or its lifetime. It is usually discounted by a factor γ to favor near-term reward over long-term reward. An alternative to discounting is the average reward formulation.

Policy Gradient (PG) Algorithms Many RL algorithms learn a value function – representing values associated with selecting specific actions – and a corresponding policy that chooses the action or subsequent state with maximum value. PG algorithms (Sutton et al., 1999) provide an alternative approach directly learning a parameterized policy. The parameters of the policy function are optimized through stochastic gradient ascent.

REINFORCE is a Monte Carlo PG algorithm for episodic problems (Williams, 1992). See algorithm 1, where $\boldsymbol{\rho}$ is a vector of optimized parameters.

Algorithm 1: REINFORCE

Input: A differentiable parameterized policy function $\pi(a|s, \boldsymbol{\rho})$

Algorithm Parameters:

step size $\alpha > 0$,

discount factor $\gamma < 1$

```
Initialize ρ (e.g. ρ ~ N(0, 0.02))
for each episode do
    Generate an episode
    s_0, a_0, r_1, ..., s_{T-1}, a_{T-1}, r_T
    following policy π
    for each step in the episode (t from 0 to T − 1) do
        G ← Σ_{k=t+1}^{T} γ^{k−t−1} r_k
        ρ ← ρ + α γ^t G ∇ ln π(a_t|s_t, ρ)
    end
end
```

Continuous Action Spaces are one advantage of PG algorithms (Sutton and Barto, 2018). Parameterized policies allow action spaces that are parameterized by a probability distribution, such as a Gaussian. For Gaussian action spaces, the mean μ and standard deviation σ are given by function approximators parameterized by $\boldsymbol{\rho}$. For a state s, an action a is sampled from the distribution and the policy is updated according to Equation 5.

$$\pi(a|s, \boldsymbol{\rho}) \doteq \frac{1}{\sigma(s, \boldsymbol{\rho})\sqrt{2\pi}} \exp\left(-\frac{(a - \mu(s, \boldsymbol{\rho}))^2}{2\sigma(s, \boldsymbol{\rho}^2)}\right) \tag{5}$$

Kullback-Leibler (KL) Divergence (Kullback and Leibler, 1951) measures the similarity between two probability distributions P and Q. It is used in AVITM (Srivastava and Sutton, 2017) to force the posterior distribution parameterized by the VAE to be the Laplace approximation of the Dirichlet prior. The KL divergence calculation for N topics is shown in Equation 6.

$$D_{KL}(P||Q) = \frac{1}{2}\sum_{1}^{N}\left(\frac{(\mu_P - \mu_Q)^2}{\sigma_Q^2} + \frac{\sigma_P^2}{\sigma_Q^2} - \log\frac{\sigma_P^2}{\sigma_Q^2} - 1\right) \tag{6}$$

3.4 Contextual Embeddings

Contextual embeddings dominate NLP tasks, replacing earlier methods, including Word2Vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), and BoW. Words and sequences of words are encoded into vector embeddings by large Transformer models (Vaswani et al., 2017).

The BoW document representation used in ProdLDA is augmented with contextual embeddings from SBERT Bianchi et al. (2020a). They test three models: one with BoW, one with contextual embeddings, and one with both. They find that using both embeddings produces the best results, and the other two methods perform almost as well. One advantage of using solely contextual embeddings is that multilingual language models can encode documents from different languages into the same embedding space, enabling easy creation of multilingual topic models (Bianchi et al., 2020b).

Sentence-BERT is an extension of BERT using a Siamese network to extract semantically meaningful sentence embeddings (Reimers and Gurevych, 2019). In contrast to BERT, this allows SBERT embeddings to be compared using dot product or cosine similarity, making SBERT more suitable for tasks such as semantic similarity search and clustering.

4 Methodology

4.1 Modernizing ProdLDA

Following Liu et al. (2022), we contemporize the architecture of the inference network within ProdLDA. We replace the SoftPlus activation function (Glorot et al., 2011) with a GELU activation function (Hendrycks and Gimpel, 2016), replace batch normalization (Ioffe and Szegedy, 2015) with layer normalization (Ba et al., 2016), and replace all Xavier initialization (Glorot and Bengio, 2010) with $\boldsymbol{\rho} \sim N(0, 0.02)$.

For the inference network, we increase the number of units in each layer from 100 to 128, add weight decay of 0.01 to each layer, and place dropout layers (Srivastava et al., 2014) after each fully connected layer.

We replace the softmax activation after the topic distribution with an RL policy formulation (Equation 5). We use a training batch size of 1024. We clip all gradients to a maximum norm of 1.0 to prevent gradient explosion (Pascanu et al., 2013). Following Bianchi et al. (2020a), we set both distributional priors as trainable parameters. We lower optimizer learning rate to 3e-4 and momentum to 0.9.

4.2 Document Embeddings

Following Bianchi et al. (2020a), we replace the BoW used by ProdLDA with contextualized embeddings from SBERT. We use the "all-MiniLM-L6-v2" model for encoding unpreprocessed documents as embedding vectors. BoW embeddings, used to calculate the log-likelihood of the topic model, are created using preprocessed documents.

4.3 Single-step REINFORCE with a Continuous Action Space

We adopt the view of RL as a statistical inference method (Levine, 2018). The modernized inference network from ProdLDA is used to parameterize a continuous action space from which an action is sampled, and the policy is computed according to Equation 5. The topic model distribution over vocabulary words uses the product of experts from ProdLDA. We use REINFORCE to train the network, with a weighted version of ELBO as the reward. Each document embedding is a state in the environment, and each episode terminates after a single step (i.e., action). Each action is a sample from the topic distribution.

4.4 Weighted Evidence Lower Bound

Following Higgins et al. (2016), we allow modifiable relative entropy between the prior and posterior by weighting the KL divergence term in the ELBO. We define a hyperparameter λ as a multiplier on the KL divergence term.

$$\text{ELBO}_{weighted} = \lambda D_{KL}(P||Q) - \textit{log-likelihood} \quad (7)$$

5 Results

5.1 Initial Experiments

We initially evaluate our topic model on the 20 Newsgroups data set with 20 topics. Results averaged over 30 random seeds are shown: loss in Figure 2, topic coherence in Figure 3, and topic diversity in Figure 4. Mean and 90% confidence intervals are plotted. Topic diversity and coherence are calculated with $K = 10$. Documents are preprocessed following Bianchi et al. (2020a) with the additional step of removing all words with less than three letters. Models are trained for 1000 epochs with the AdamW optimizer ($\alpha = 3e-4$, $\beta_1 = 0.9$, $\beta_2 = 0.999$). We use $\lambda = 5$, inference network dropout of 0.2, and no dropout after the RL policy (policy dropout). All other experiments use these same settings unless otherwise noted.

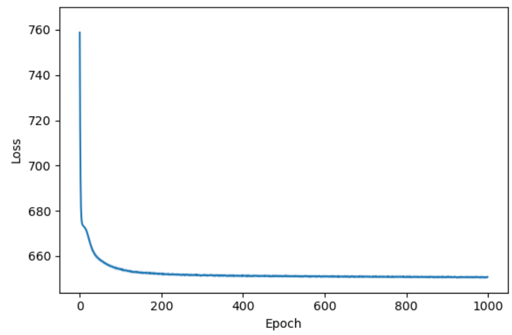

Figure 2: Loss (30 seeds): 20 Newsgroups

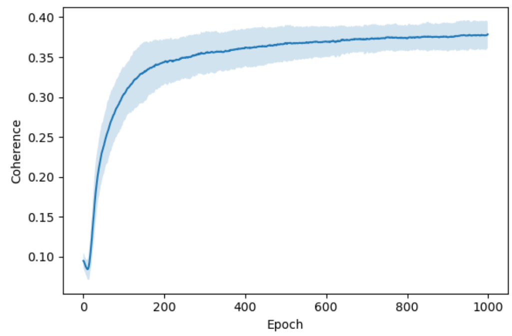

Figure 3: Topic Coherence (30 seeds): 20 Newsgroups

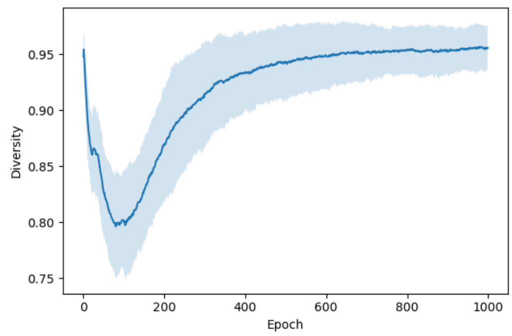

Figure 4: Topic Diversity (30 seeds): 20 Newsgroups

5.2 Comparison to Other Topic Models

We compare our method to recent topic models found in the literature.

5.2.1 Benchmarking Neural Topic Models (BNTM)

In the beginning, our approach is compared with all models evaluated by Doan and Hoang (2021). We use their preprocessed documents and replicate their results using $K = 10$ to calculate topic coherence. Following the authors, we sweep from 0.5*N topics to 3*N topics in intervals of 0.5*N (N being the "correct" number of topics for each data set). Next, we do a hyperparameter sweep over λ of 1,

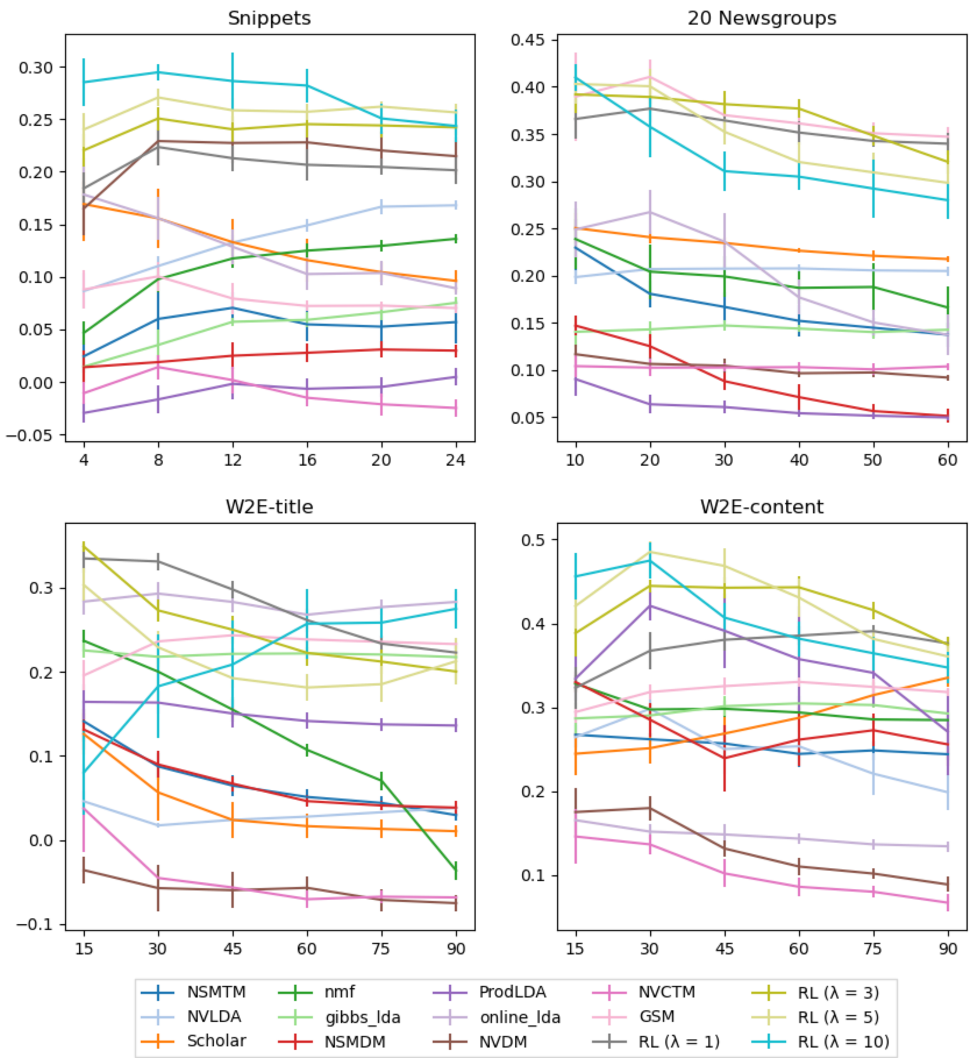

Figure 5: Comparison of RL model (ours) to BNTM models

3, 5, and 10. Results are averaged over ten random seeds and shown in Figure 5.

5.2.2 Topic Modeling in Embedding Spaces

Next, the comparison is done with Dieng et al. (2020) on the New York Times data set with 300 topics and without using stop words. Results are shown in Table 1. We increase batch size to 32768 and only train for 20 epochs on one random seed. Additionally, we increase the number of units in each layer of the inference network to 512, increase dropout in the inference network to 0.5, and decrease λ to 1. Topic diversity is calculated using $K = 25$.

5.2.3 Pre-training is a Hot Topic (PTHT)

We also compare our model, using all metrics, with the best model as evaluated by Bianchi et al. (2020a). Results are shown in Table 2. Metrics are averaged over 25, 50, 75, 100, and 150 topics: 30

Model	Coherence	Diversity	Quality
ETM	0.18	0.22	0.0405
RL model (ours)	**0.24**	**0.32**	**0.0778**

Table 1: Comparison on no stop words data

seeds for each number of topics. We use the same preprocessing as the authors. We use $\lambda = 1$.

5.2.4 Contrastive Learning for NTM (CLNTM)

We compare results with the contrastive Scholar model from Nguyen and Luu (2021). For each data set we perform a hyperparameter search with 50 topics. Search ranges and best results for each data set are shown in Table 3. We use the best hyperparameters from this search for final training runs with 50 and 200 topics. We train for 2000

Data Set	Paper	NPMI	Word2Vec	Inverse RBO
Wiki20K	PTHT best	0.1823	0.2110	**0.9950**
	RL model (ours)	**0.2509**	**0.2368**	0.9799
StackOverflow	PTHT best	0.0280	0.1598	**0.9914**
	RL model (ours)	**0.1249**	**0.1617**	0.9860
Google News	PTHT best	0.1207	0.1325	**0.9965**
	RL model (ours)	**0.3563**	**0.1485**	0.9934
Tweets2011	PTHT best	0.1008	**0.1493**	0.9956
	RL model (ours)	**0.3559**	0.1417	**0.9962**
20 Newsgroups	PTHT best	0.1300	**0.2539**	0.9931
	RL model (ours)	**0.2696**	0.1798	**0.9932**

Table 2: Average metrics from best PTHT model (per metric) and our RL model

Experiment	Layer Size	Inference Dropout	Policy Dropout	λ
Hyperparameter Search	{128, 512}	{0.2, 0.5}	{0.0, 0.25, 0.5}	{1, 5}
20 Newsgroups	128	0.5	0.5	1
IMDb Movie Reviews	512	0.5	0.25	1
Wikitext-103	512	0.5	0.25	5

Table 3: Hyperparameter search and best results per data set for RL model

epochs. Results are averaged over 30 random seeds and shown in Table 4.

To show the tradeoff between topic diversity and coherence, we perform a sweep over policy dropout from 0 to 0.9 at intervals of 0.1 using the 20 Newsgroups data set with 50 topics. Other hyperparameters are kept the same. We train for 2000 epochs. Results are averaged over 30 random seeds and shown in Figure 6.

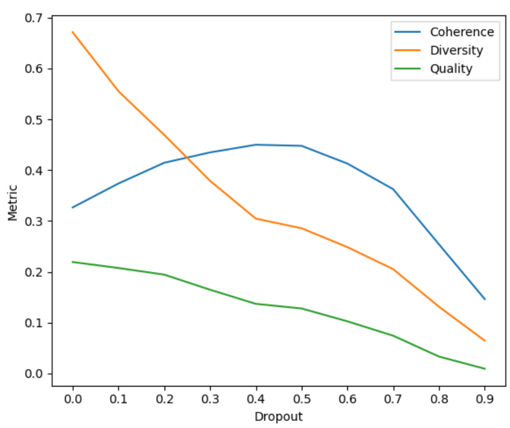

Figure 6: Dropout sweep for 20 Newsgroups

5.3 Ablation Study

To provide empirical evidence that performance improvements come from the RL policy formulation, we do a study ablating relevant changes from the final RL model down to the original ProdLDA model. All comparisons are performed on the 20 Newsgroups data set with 20 topics and use the same settings as subsection 5.1. Results are averaged over 30 random seeds and shown in Table 5.

6 Discussion

For the initial experiments on the 20 Newsgroups data set, the average loss (Figure 2) reaches a near plateau around the 200th epoch. Past this epoch, coherence (Figure 3) continues to increase slowly, and topic diversity (Figure 4) increases substantially until around the 400th epoch, past which it also continues to increase slowly. It shows that training beyond a plateau in loss can still improve NTM performance.

Compared to Doan and Hoang (2021), the RL model performs on par with or better than other models across all four data sets, while the performance of other models varies greatly between data sets. On the Snippets, 20 Newsgroups, and W2E-content data sets, the RL model with lower values of λ usually performs better as the number of topics increases. However, it reverses on the W2E-title data set where $\lambda = 10$ outperforms $\lambda = 1$ on the two highest number of topics.

The RL model outperforms the Labeled ETM model from Dieng et al. (2020) in topic diversity, coherence, and quality. Furthermore, this compari-

Model	20 Newsgroups		IMDb Movie Reviews		Wikitext-103	
	50 Topics	200 Topics	50 Topics	200 Topics	50 Topics	200 Topics
Contrastive Scholar	0.334	0.280	0.197	**0.188**	**0.497**	**0.478**
RL model (ours)	**0.449**	**0.308**	**0.199**	0.139	0.432	0.268

Table 4: Comparison to CLNTM

RL Policy	Embedding	λ	θ Softmax	θ / Policy Dropout	Coherence	Diversity
✓	SBERT	5	×	0.0	**0.3848**	**0.9530**
×	SBERT	5	×	0.0	0.2795	0.453
✓	BoW	5	×	0.0	0.3379	0.9403
✓	SBERT	1	×	0.0	0.3414	0.9070
✓	SBERT	5	✓	0.0	0.1932	0.6927
✓	SBERT	5	×	0.2	0.3769	0.7315
×	BoW	1	✓	0.2	0.2650	0.7390

Table 5: Highlighted results from ablation study

son had no pruning of stop words, showing the RL model can deal with vocabularies containing many common words.

Compared to Bianchi et al. (2020a), the RL model significantly outperforms all other models on all data sets evaluated in terms of NPMI coherence. Furthermore, the RL model performs similarly to the best of the other models in terms of inverse RBO. We state the topic diversity used by Dieng et al. (2020) is a more useful metric than inverse RBO, as it usually has a higher variance in values and is more intuitive to understand. For Word2Vec coherence, the RL model performs on par with the best of the other models, except when compared to ETM (Dieng et al., 2020) on the 20 Newsgroups data set.

If we consider models from Nguyen and Luu (2021), our RL model performs similarly on 50 topics but worse on 200 topics. The RL model's performance on larger topic sizes and vocabularies could be improved by adding supervised labels, applying contrastive learning, scaling up inference layer sizes, or performing a hyperparameter sweep with 200 topics.

Topic diversity and coherence values should be provided when reporting topic model performance. In Figure 6, the highest topic quality is achieved when there is no policy dropout. Topic diversity can be sacrificed for some gain in coherence. Applications of topic models may want to maximize topic diversity, coherence, or both. The description of topic model performance should reflect this.

In the ablation study, removing the RL policy formulation causes the model to perform worse than the original one. It confirms RL policy augments the improvements from other changes to the model. Performance suffers the most when the softmax distribution is re-added to the topic distribution during training. To recapture the softmax distribution of topics, it can be applied to the topic distribution during inference. Adding policy dropout significantly reduces topic diversity and leads to a slight coherence reduction. Performance improves with SBERT embeddings, and the model can still reconstruct the BoW within the ELBO without direct access. Increasing λ to 5 improves performance, but as seen from other experiments, this is only sometimes the case.

7 Conclusion

Inspired by the introduction of probabilistic inference techniques to RL, we take the approach to develop a NTM augmented with RL. Our model builds on the ProdLDA model, which uses a product of experts instead of the mixture model used in classical LDA. We improve ProdLDA by adding SBERT embeddings, an RL policy formulation, a weighted ELBO loss, and the improved NN architecture. In addition, we track topic diversity and coherence during a training process rather than only evaluating these metrics for the final model. Our fully unsupervised RL model outperforms most other topic models. It is only topped by contrastive Scholar —a method using supervised labels during training —in a few select cases.

8 Limitations

The main limitation identified for our RL model is decreased performance as the vocabulary size increases. Our RL model also has a higher variance than some other topic models to which we compared. While our RL model performed well on all the data sets tested, this performance may not generalize to different data sets. The insights from the policy dropout sweep conducted may not apply to other topic models. The performance difference for NPMI coherence compared with Bianchi et al. (2020a) may be overstated since the model in that paper used a deprecated SBERT model that produces sentence embeddings of low quality[1]. For the comparison to Nguyen and Luu (2021), we used slightly different preprocessing than the authors. While the model can work on any languages with associated embedding models, all data sets used in this paper were in English. Our model has additional hyperparameters compared to some other models. So, it may require more tuning and, therefore, more GPU computing. The initial model was developed on a system with 8GB of RAM and a Nvidia GTX 1060 with 3GB of VRAM for a total of approximately 100 GPU hours. A single run of the model for 1000 epochs on this GPU requires less than an hour. Experiments using the New York Times data set were run on a system with 256GB of RAM and a Nvidia RTX 3090 for approximately 100 GPU hours. All other experiments were run on a system with 128GB of RAM and a Nvidia TITAN RTX for approximately 600 GPU hours.

9 Ethics Statement

All data sets used in this paper are cited. The New York Times data set[2] is licensed under "The New York Times Annotated Corpus Agreement"[3]. The Tweets2011 corpus[4] is available under the "TREC 2011 Microblog Dataset Usage Agreement"[5] which additionally requires following the "Twitter terms of service"[6]. All other data sets are obtained from the recent literature. No sensitive information is used or inferred in this paper. The risk of harm from our model is low. Any artifacts in this paper are used following their intended use cases.

Acknolwedgements

We would like to thank Federico Bianchi for assistance in finding data sets. We would like to thank the creators and maintainers of Python and the following Python packages: lda, torch, numpy, biterm, scipy, gensim, tqdm, transformers, nltk, sentence_transformers, sklearn, and pandas. We would like to thank the following GitHub users for code inspiration: maifeng, smutahoang, dicegroup, shion-h, karpathy, estebandito22, akashgit, and MilaNLProc.

References

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. 2018. Maximum a posteriori policy optimisation. *arXiv preprint arXiv:1806.06920*.

Nikolaos Aletras and Mark Stevenson. 2013. Evaluating topic coherence using distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)–Long Papers*, pages 13–22.

Dimo Angelov. 2020. Top2vec: Distributed representations of topics. *arXiv preprint arXiv:2008.09470*.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2020a. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. *arXiv preprint arXiv:2004.03974*.

Federico Bianchi, Silvia Terragni, Dirk Hovy, Debora Nozza, and Elisabetta Fersini. 2020b. Cross-lingual contextualized topic models with zero-shot learning. *arXiv preprint arXiv:2004.07737*.

David M Blei and John D Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Dallas Card, Chenhao Tan, and Noah A Smith. 2017. Neural models for documents with metadata. *arXiv preprint arXiv:1705.09296*.

Peter Dayan and Geoffrey E Hinton. 1997. Using expectation-maximization for reinforcement learning. *Neural Computation*, 9(2):271–278.

[1] https://huggingface.co/sentence-transformers/stsb-roberta-large

[2] https://catalog.ldc.upenn.edu/LDC2008T19

[3] https://catalog.ldc.upenn.edu/license/the-new-york-times-annotated-corpus-ldc2008t19.pdf

[4] https://trec.nist.gov/data/tweets/

[5] https://trec.nist.gov/data/tweets/tweets2011-agreement.pdf

[6] https://twitter.com/en/tos

Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Thanh-Nam Doan and Tuan-Anh Hoang. 2021. Benchmarking neural topic models: An empirical study. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4363–4368.

Matthew Fellows, Anuj Mahajan, Tim GJ Rudner, and Shimon Whiteson. 2019. Virel: A variational inference framework for reinforcement learning. *Advances in neural information processing systems*, 32.

Mikhail Figurnov, Shakir Mohamed, and Andriy Mnih. 2018. Implicit reparameterization gradients. *Advances in neural information processing systems*, 31.

Sarah Filippi, Olivier Cappé, and Aurélien Garivier. 2010. Optimism in reinforcement learning and kullback-leibler divergence. In *2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 115–122. IEEE.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings.

Thomas L Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National academy of Sciences*, 101(suppl_1):5228–5235.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Lin Gui, Jia Leng, Gabriele Pergola, Yu Zhou, Ruifeng Xu, and Yulan He. 2019. Neural topic model with reinforcement learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3478–3483.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. 2016. beta-vae: Learning basic visual concepts with a constrained variational framework.

Tuan-Anh Hoang, Khoi Duy Vo, and Wolfgang Nejdl. 2018. W2e: a worldwide-event benchmark dataset for topic detection and tracking. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1847–1850.

Matthew Hoffman, Francis Bach, and David Blei. 2010. Online learning for latent dirichlet allocation. *advances in neural information processing systems*, 23.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. 1999. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233.

Hilbert J Kappen, Vicenç Gómez, and Manfred Opper. 2012. Optimal control as a graphical model inference problem. *Machine learning*, 87(2):159–182.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Taisuke Kobayashi. 2022. Optimistic reinforcement learning by forward kullback–leibler divergence optimization. *Neural Networks*, 152:169–180.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

Amit Kumar, Nazanin Esmaili, and Massimo Piccardi. 2021. A reinforced variational autoencoder topic model. In *International Conference on Neural Information Processing*, pages 360–369. Springer.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*, pages 331–339. Elsevier.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539.

Sergey Levine. 2018. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*.

Tianyi Lin, Zhiyue Hu, and Xin Guo. 2019. Sparsemax and relaxed wasserstein for topic sparsity. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 141–149.

Luyang Liu, Heyan Huang, Yang Gao, Yongfeng Zhang, and Xiaochi Wei. 2019. Neural variational correlated topic modeling. In *The World Wide Web Conference*, pages 1142–1152.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. 2022. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Richard McCreadie, Ian Soboroff, Jimmy Lin, Craig Macdonald, Iadh Ounis, and Dean McCullough. 2012. On building a reusable twitter corpus. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 1113–1114.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. In *International Conference on Machine Learning*, pages 2410–2419. PMLR.

Yishu Miao, Lei Yu, and Phil Blunsom. 2016. Neural variational inference for text processing. In *International conference on machine learning*, pages 1727–1736. PMLR.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Thong Nguyen and Anh Tuan Luu. 2021. Contrastive learning for neural topic model. *Advances in Neural Information Processing Systems*, 34:11974–11986.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. PMLR.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Jipeng Qiang, Zhenyu Qian, Yun Li, Yunhao Yuan, and Xindong Wu. 2020. Short text topic modeling techniques, applications, and performance: a survey. *IEEE Transactions on Knowledge and Data Engineering*, 34(3):1427–1445.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. 2015. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. 1999. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12.

Yuan Tian, Minghao Han, Chetan Kulkarni, and Olga Fink. 2022. A prescriptive dirichlet power allocation policy with deep reinforcement learning. *Reliability Engineering & System Safety*, 224:108529.

Naonori Ueda and Kazumi Saito. 2002. Parametric mixture models for multi-labeled text. *Advances in neural information processing systems*, 15.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Nino Vieillard, Tadashi Kozuno, Bruno Scherrer, Olivier Pietquin, Rémi Munos, and Matthieu Geist. 2020. Leverage the average: an analysis of kl regularization in reinforcement learning. *Advances in Neural Information Processing Systems*, 33:12163–12174.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

He Zhao, Dinh Phung, Viet Huynh, Yuan Jin, Lan Du, and Wray Buntine. 2021. Topic modelling meets deep neural networks: A survey. *arXiv preprint arXiv:2103.00498*.

Renbo Zhao, Vincent Tan, and Huan Xu. 2017. Online nonnegative matrix factorization with general divergences. In *Artificial Intelligence and Statistics*, pages 37–45. PMLR.

A KL Divergence in RL

KL divergence has recently become popular in continuous action space RL algorithms. One application is to prevent policy updates from making large changes to the policy that could result in poorer performance. Two algorithms using KL divergence for this are TRPO (Schulman et al., 2015) and MPO (Abdolmaleki et al., 2018). Another application is for optimistic RL (Filippi et al., 2010) (Kobayashi, 2022). Vieillard et al. (2020) investigate the usage of KL divergence as regularization in RL. KL divergence has also been used in optimal control (Kappen et al., 2012), which is closely related to RL.

B Data Sets

We evaluate models on the test set where available, and on the training set if there is no test set. Coherence and diversity for the training and test set are the same, as they are evaluated on the word distribution over topics which doesn't change per document. In the code, training coherence and diversity are computed after each batch, while test coherence and diversity are computed after each epoch. Number of training/test documents and vocabulary sizes are shown in Table 6. Average original and preprocessed training document lengths are shown in Table 7.

B.1 20 Newsgroups

The 20 Newsgroups data set (Lang, 1995) consists of around 19,000 newsgroup posts from 20 topics. We perform experiments on this data set with three different preprocessing methods. For our initial experiments, we follow the preprocessing in Bianchi et al. (2020a) and additionally remove all words with less than 3 letters. For the comparisons with Bianchi et al. (2020a) and Nguyen and Luu (2021), we follow the preprocessing in Bianchi et al. (2020a). For the comparison with Doan and Hoang (2021), we use their already preprocessed data set.

B.2 New York Times

The New York Times data set (Sandhaus, 2008) consists of over 1.8 million articles written by the New York Times between 1987 and 2007. We follow the preprocessing from Bianchi et al. (2020a), but do not remove stopwords.

B.3 Snippets

The Web Snippets data set (Ueda and Saito, 2002) consists of around 12,000 snippets of text from websites linked on "yahoo.com". The snippets are grouped into 8 domains. We use the already preprocessed data set from Doan and Hoang (2021).

B.4 W2E

The W2E data set (Hoang et al., 2018) consists of news articles from media channels around the world. The W2E-title subset is the titles from the news articles, while the W2E-content subset is the text content of the articles. The articles are grouped into 30 topics. We use the already preprocessed data set from Doan and Hoang (2021).

B.5 Wiki20K

The Wiki20K data set (Bianchi et al., 2020b) consists of 20,000 English Wikipedia abstracts randomly sampled from DBpedia. We follow the preprocessing from Bianchi et al. (2020a).

B.6 StackOverflow

The StackOverflow data set (Qiang et al., 2020) consists of around 16,000 question titles randomly sampled from 20 different tags in a larger data set crawled from the website "stackoverflow.com" between July and August 2012. We use the already preprocessed data set from Qiang et al. (2020).

B.7 Google News

The Google News data set (Qiang et al., 2020) consists of around 11,000 titles and short samples from Google News articles clustered into 152 groups. We use the already preprocessed data set from Qiang et al. (2020).

B.8 Tweets2011

The Tweets2011 data set (Qiang et al., 2020) consists of around 2,500 tweets in 89 clusters sampled from the larger Tweets2011 corpus (McCreadie et al., 2012) crawled from Twitter between January and February 2011. We use the already preprocessed data set from Qiang et al. (2020).

Data Set	Comparison Paper	Training Docs	Test Docs	Vocab Size
20 Newsgroups	This one (Bianchi et al., 2020a) (Nguyen and Luu, 2021)	11,314	7,532	2,000
	(Doan and Hoang, 2021)	15,465	N/A	4,134
New York Times	(Dieng et al., 2020)	1,864,470	N/A	10,283
Snippets	(Doan and Hoang, 2021)	12,295	N/A	4,666
W2E-title	(Doan and Hoang, 2021)	105,457	N/A	3,703
W2E-content	(Doan and Hoang, 2021)	83,548	N/A	10,508
Wiki20K	(Bianchi et al., 2020a)	20,000	N/A	2,000
StackOverflow	(Bianchi et al., 2020a)	16,407	N/A	2,236
Google News	(Bianchi et al., 2020a)	11,108	N/A	8,099
Tweets2011	(Bianchi et al., 2020a)	2,472	N/A	5,097
IMDb Movie Reviews	(Nguyen and Luu, 2021)	25,000	25,000	5,000
Wikitext-103	(Nguyen and Luu, 2021)	28,472	60	20,000

Table 6: Data Sets - Documents and Vocabularies

B.9 IMDb Movie Reviews

The IMDb Movie Reviews data set (Maas et al., 2011) consists of 50,000 movie reviews, each with an associated sentiment label, from the website "imdb.com". We follow the preprocessing from Bianchi et al. (2020a).

B.10 Wikitext-103

The Wikitext-103 data set (Merity et al., 2016) consists of around 28,500 Wikipedia articles classified as either Featured articles or Good articles by Wikipedia editors. We follow the preprocessing from Bianchi et al. (2020a).

C Evaluation Metrics

We track topic diversity, coherence, perplexity, and loss for the training and test sets if applicable. Topic diversity and coherence are calculated based on the top-K words in each topic, with K noted for each experiment. We use NPMI coherence with co-occurence based on full document windows.

Most previous NTMs have only reported the coherence of the final model, presumably because coherence is not tracked during training for computational reasons. To enable tracking of coherence during training, we modify a vectorized implementation of UMass coherence[7] to calculate NPMI coherence and add caching for further speed-up. We also implement a GPU-optimized algorithm to calculate topic diversity during training.

Tracking these metrics during training provides two main benefits. The first benefit is that if training is going poorly, it can be terminated. Poor training could be caused by component collapse (low topic diversity), or if the model is unable to fit to coherent topics (low coherence). The second benefit is enabling deeper performance comparisons between models and between training runs for a single model. Most existing NTMs only track loss and perplexity during training, so additionally tracking topic diversity and coherence could provide additional insights on model performance.

D Expanded Results

D.1 Topic Words from Initial Experiments

We choose one example of the top 10 words for all 20 topics from the initial experiments on the 20 Newsgroups data set. We choose the seed with the 15th highest coherence (out of 30 seeds). Topic words are shown in Table 8. Each document in the Twenty Newsgroups data set is labeled as belonging to one of 20 categories. These 20 categories are shown in Table 9.

D.2 Pre-training is a Hot Topic

We show a further comparison between the contextual embedding model from Bianchi et al. (2020a) and our RL model in Table 10. Average NPMI coherence over 30 seeds is compared for each number of topics: 25, 50, 75, 100, and 150.

[7]https://github.com/maifeng/Examples_UMass-Coherence

Data Set	Comparison Paper	Average Training Document Length	
		Original	Preprocessed
20 Newsgroups	This one	287.5	95.9
	(Bianchi et al., 2020a) (Nguyen and Luu, 2021)	287.5	107.6
	(Doan and Hoang, 2021)	N/A	73.5
New York Times	(Dieng et al., 2020)	558.1	484.5
Snippets	(Doan and Hoang, 2021)	N/A	14.4
W2E-title	(Doan and Hoang, 2021)	N/A	6.8
W2E-content	(Doan and Hoang, 2021)	N/A	209.1
Wiki20K	(Bianchi et al., 2020a)	49.8	17.5
StackOverflow	(Bianchi et al., 2020a)	N/A	4.9
Google News	(Bianchi et al., 2020a)	N/A	6.2
Tweets2011	(Bianchi et al., 2020a)	N/A	8.6
IMDb Movie Reviews	(Nguyen and Luu, 2021)	233.8	101.7
Wikitext-103	(Nguyen and Luu, 2021)	295.8	133.2

Table 7: Data Sets - Training Document Lengths

D.3 Hyperparameters

We show the hyperparameters for each experiment we performed. Experiment seeds are generated with a meta-seed for reproducibility. The meta-seed is randomly chosen from integers between 0 and 2^{32}. Values in {curly brackets} indicate a search over multiple parameters. Values in [square brackets] indicate NN layer sizes (e.g. [128, 128] represents two layers of size 128).

D.3.1 Initial Experiments and Ablation Study

We use the same meta-seed for the ablation study as we did for the initial experiments. Hyperparameters for the initial experiments can be found in Table 11. Further tables for all experiments will only show hyperparameters that differ from this table. Hyperparameters for the ablation study can be found in Table 12.

D.3.2 Benchmarking Neural Topic Models

We show hyperparameters for the comparison with Doan and Hoang (2021). Hyperparameters for Snippets can be found in Table 15. 20 Newsgroups in Table 16. W2E-title in Table 17. W2E-content in Table 19.

D.3.3 Topic Modeling in Embedding Spaces

Hyperparameters for the comparison with Dieng et al. (2020) can be found in Table 20.

D.3.4 Pre-training is a Hot Topic

We show hyperparameters for the comparison with Bianchi et al. (2020a). Data set and seed information can be found in Table 13. All other hyperparameters are the same for each data set; these can be found in Table 18.

D.3.5 Contrastive Learning for NTM

We show hyperparameters for the comparison with Nguyen and Luu (2021). Some hyperparameters are already shown in Table 3 and won't be shown again here. Data set and seed information can be found in Table 14. Other hyperparameters are the same for each data set; these can be found in Table 21. Hyperparameters for the policy dropout sweep can be found in Table 22.

D.4 Ablation Study

We show full results from the ablation study in Table 23.

E Model Parameter Count

The number of parameters (P) in the model differs based on the total number of parameters across all inference layers (L), the number of topics (N), and the vocabulary size (V). Trainable parameters are the inference layers, the prior distribution of topics (N x 1), and the distribution of words over topics (V * N). Total parameters can be calculated with Equation 8.

$$P = L + N + V * N \tag{8}$$

Topic Words
max giz bhj chz pts buf air det pit bos
morality objective cramer moral livesey optilink keith homosexual clayton gay
window xterm widget lib windows font usr mouse motif application
gun guns militia firearms weapons cops weapon amendment semi arms
team players hockey game season nhl games play teams leafs
max giz bhj sale chz shipping offer monitor copies condition
jesus god bible christ christians faith church christian heaven lord
geb banks msg patients gordon pitt disease pain doctor medical
fbi batf koresh compound atf waco sandvik udel fire kent
car insurance cars dealer oil saturn honda engine bmw miles
jpeg image bits display gif file program files format color
clipper encryption key chip escrow keys privacy crypto secure nsa
wire ground circuit connected cable atheism electrical universe keyboard output
israel israeli arab jews arabs peace palestinian attacks bony villages
turkish armenian armenians armenia turks serdar argic turkey genocide soviet
pub ftp anonymous tar graphics privacy mailing archive motif faq
moon space lunar orbit nasa spacecraft henry launch shuttle solar
dog bike dod riding ride motorcycle rider bmw went cops
scsi ide drive controller drives bus disk floppy bios isa
stephanopoulos president jobs myers russia russian administration package launch clinton

Table 8: Initial Experiment Topic Words

Category
alt.atheism
comp.graphics
comp.os.ms-windows.misc
comp.sys.ibm.pc.hardware
comp.sys.mac.hardware
comp.windows.x
misc.forsale
rec.autos
rec.motorcycles
rec.sport.baseball
rec.sport.hockey
sci.crypt
sci.electronics
sci.med
sci.space
soc.religion.christian
talk.politics.guns
talk.politics.mideast
talk.politics.misc
talk.religion.misc

Table 9: 20 Newsgroups Categories

The largest model we use is for the Wikitext-103 data set with 200 topics. This model has 4,001,224 parameters.

F Future Work

We have identified some possible paths for future work. The SBERT embeddings could be fine-tuned during training rather than calculating them during pre-processing and freezing them during training. The RL formulation of our model could be extended to dynamic topic models (Blei and Lafferty, 2006). More complex PG RL algorithms could be used rather than REINFORCE, or a baseline could be added to REINFORCE. Exploration techniques from RL could be applied. The influence of hyperparameters (e.g. inference network layer sizes) on varied corpora (e.g. those with large vocabularies) could be explored. The Laplace approximation of the Dirichlet prior could be replaced by a true Dirichlet prior, making use of the Dirichlet RT (Figurnov et al., 2018) and a Dirichlet RL policy (Tian et al., 2022). Finally, λ and the policy dropout could be scheduled during training to provide an automated tradeoff between topic diversity and coherence.

Data Set	Paper	NPMI Coherence				
		25 Topics	50 Topics	75 Topics	100 Topics	150 Topics
Wiki20K	PTHT	0.17	0.19	0.18	0.19	0.17
	RL model (ours)	0.33	0.30	0.25	0.22	0.19
StackOverflow	PTHT	0.05	0.03	0.02	0.02	0.02
	RL model (ours)	0.17	0.14	0.12	0.11	0.10
Google News	PTHT	0.03	0.10	0.15	0.18	0.19
	RL model (ours)	0.38	0.41	0.38	0.34	0.30
Tweets2011	PTHT	0.05	0.10	0.11	0.12	0.12
	RL model (ours)	0.36	0.39	0.38	0.35	0.31
20 Newsgroups	PTHT	0.13	0.13	0.13	0.13	0.12
	RL model (ours)	0.35	0.30	0.27	0.25	0.22

Table 10: NPMI coherence comparison between PTHT model and RL model for each number of topics

Hyperparameter	Value(s)
Meta-seed	4174224060
Num. Seeds	30
Num. Epochs	1000
Data Set	20 Newsgroups
Vocab Size	2000
Embedding	SBERT
Num. Topics (N)	20
Inference Dropout	0.2
Policy Dropout	0.0
Inference Layers	[128, 128]
Activation	GELU
Initialization	$\boldsymbol{\rho} \sim N(0, 0.02)$
Normalization	Layer
λ	5
Topic Words (K)	10
RL policy	✓
θ Softmax	$\times$
Learning Rate (α)	3e-4
Adam β_1, β_2	0.9, 0.999
Weight Decay	0.01
Batch Size	1024
Gradient Clipping	1.0

Table 11: Initial Experiments

Hyperparameter	Value(s)
Meta-seed	4174224060
Num. Seeds	30
Data Set	20 Newsgroups
Embedding	{BoW, SBERT}
θ / Policy Dropout	{0.0, 0.2}
λ	{1, 5}
RL policy	{✓, $\times$}
θ Softmax	{✓, $\times$}

Table 12: Ablation Study

Data Set	Vocab Size	Meta-seed	Num. Seeds
Wiki20K	2000	359491602	30
StackOverflow	2236	1459046441	30
Google News	8099	925040003	30
Tweets2011	5097	1321150024	30
20 Newsgroups	2000	3277797161	30

Table 13: PTHT Data Set Seeds

Data Set	Vocab Size	Meta-seed	Num. Seeds
20 Newsgroups	2000	1553571489	30
IMDb Movie Reviews	5000	3747305026	30
Wikitext-103	20000	2672751736	30

Table 14: CLNTM Data Set Seeds

Hyperparameter	Value(s)
Meta-seed	193270011
Num. Seeds	10
Data Set	Snippets
Vocab Size	4666
Num. Topics (N)	{4, 8, 12, 16, 20, 24}
λ	{1, 3, 5, 10}

Table 15: BNTM Snippets

Hyperparameter	Value(s)
Meta-seed	1216545997
Num. Seeds	10
Data Set	20 Newsgroups
Vocab Size	4157
Num. Topics (N)	{10, 20, 30, 40, 50, 60}
λ	{1, 3, 5, 10}

Table 16: BNTM 20 Newsgroups

Hyperparameter	Value(s)
Meta-seed	4014169843
Num. Seeds	10
Data Set	W2E-title
Vocab Size	3703
Num. Topics (N)	{15, 30, 45, 60, 75, 90}
λ	{1, 3, 5, 10}

Table 17: BNTM W2E-title

Hyperparameter	Value(s)
Num. Topics (N)	{25, 50, 75, 100, 150}
λ	1

Table 18: Pre-training is a Hot Topic

Hyperparameter	Value(s)
Meta-seed	1359128464
Num. Seeds	10
Data Set	W2E-content
Vocab Size	10508
Num. Topics (N)	{15, 30, 45, 60, 75, 90}
λ	{1, 3, 5, 10}

Table 19: BNTM W2E-content

Hyperparameter	Value(s)
Meta-seed	2337766308
Num. Seeds	1
Num. Epochs	20
Data Set	New York Times
Vocab Size	10283
Num. Topics (N)	300
Inference Dropout	0.5
Inference Layers	[512, 512]
λ	1
Topic Words (K)	10*
Batch Size	32768

Table 20: Topic Modeling in Embedding Spaces (*We use $K = 25$ to calculate topic diversity for the final model.)

Hyperparameter	Value(s)
Num. Epochs	2000
Num. Topics (N)	{50, 200}

Table 21: Contrastive Learning for NTM

Hyperparameter	Value(s)
Meta-seed	3432645033
Num. Seeds	30
Data Set	20 Newsgroups
Num. Epochs	2000
Num. Topics (N)	50
Inference Dropout	0.5
Policy Dropout	{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9}
Inference Layers	[128, 128]
λ	1

Table 22: CLNTM Dropout Sweep

RL Policy	Embedding	λ	θ Softmax	θ / Policy Dropout	Coherence	Diversity
×	BoW	1	✓	0.0	0.2906	0.8457
×	BoW	1	×	0.0	0.2373	0.6943
✓	BoW	1	✓	0.0	0.2748	0.8905
✓	BoW	1	×	0.0	0.2738	0.8707
×	BoW	5	✓	0.0	0.2526	0.6598
×	BoW	5	×	0.0	0.2619	0.6928
✓	BoW	5	✓	0.0	0.2032	0.5965
✓	BoW	5	×	0.0	0.3379	0.9403
×	BoW	1	✓	0.2	0.2650	0.7390
×	BoW	1	×	0.2	0.2193	0.5195
✓	BoW	1	✓	0.2	0.2082	0.5692
✓	BoW	1	×	0.2	0.2798	0.7740
×	BoW	5	✓	0.2	0.2526	0.6222
×	BoW	5	×	0.2	0.2257	0.5768
✓	BoW	5	✓	0.2	0.1222	0.314
✓	BoW	5	×	0.2	0.3284	0.8092
×	SBERT	1	✓	0.0	0.2845	0.6207
×	SBERT	1	×	0.0	0.2948	0.5995
✓	SBERT	1	✓	0.0	0.2158	0.8080
✓	SBERT	1	×	0.0	0.3414	0.9070
×	SBERT	5	✓	0.0	0.2726	0.4458
×	SBERT	5	×	0.0	0.2795	0.4530
✓	SBERT	5	✓	0.0	0.1932	0.6927
✓	SBERT	5	×	0.0	**0.3848**	**0.9530**
×	SBERT	1	✓	0.2	0.2532	0.6063
×	SBERT	1	×	0.2	0.2554	0.5430
✓	SBERT	1	✓	0.2	0.1133	0.5520
✓	SBERT	1	×	0.2	0.3649	0.7663
×	SBERT	5	✓	0.2	0.2435	0.4478
×	SBERT	5	×	0.2	0.2080	0.3698
✓	SBERT	5	✓	0.2	0.0967	0.9227
✓	SBERT	5	×	0.2	0.3769	0.7315

Table 23: Full Results from Ablation Study