Neural Topic Model with Reinforcement Learning

Lin Gui1,§, Jia Leng2,§, Gabriele Pergola1, Yu Zhou1, Ruifeng Xu2,3,4, Yulan He1,†
1Department of Computer Science, University of Warwick, UK
2Harbin Institute of Technology (Shenzhen), China
3Peng Cheng Laboratory, Shenzhen, China
4Joint Lab of Harbin Institute of Technology and RICOH

Lin.gu@warwick.ac.uk, lengjia@stu.hit.edu.cn
gabriele.pergola@warwick.ac.uk, Yu.Zhou.1@warwick.ac.uk
xuruifeng@hit.edu.cn, yulan.he@warwick.ac.uk

Abstract

In recent years, advances in neural variational inference have achieved many successes in text processing. Examples include neural topic models which are typically built upon variational autoencoder (VAE) with an objective of minimizing the error of reconstructing original documents based on the learned latent topics. However, minimizing reconstruction errors does not necessarily lead to high quality topics. In this paper, we borrow the idea of reinforcement learning and incorporate topic coherence measures as reward signals to guide the learning of a VAE-based topic model. Furthermore, our proposed model is able to automatically separating background words dynamically from topic words, thus eliminating the pre-processing step of filtering infrequent and/or top frequent words, typically required for learning traditional topic models. Experimental results on the 20 Newsgroups and the NIPS datasets show superior performance both on perplexity and topic coherence measure compared to state-of-the-art neural topic models.

1 Introduction

Probabilistic topic models have been used widely in natural language processing (Li et al., 2016; Zeng et al., 2018). The fundamental principle is that words are assumed to be generated from latent topics which can be inferred from data based on word co-occurrence patterns (Neal, 1993; Andrieu et al., 2003). In recent years, Variational Autoencoder (VAE) has been proved more effective and efficient to approximating deep, complex and underestimated variance in integrals (Kingma and Welling, 2013; He et al., 2017). However, the VAE-based topic models focus on the construction of deep neural networks to approximate the intractable distribution between observed words and latent topics based on log-likelihood and the learning objective is to minimise the error of reconstructing the original documents based on the learned latent topic vectors rather than improving the quality of learned topics, for example, measured by coherence scores (Kingma and Welling, 2013; Sønderby et al., 2016; Miao et al., 2016; Card et al., 2017; Srivastava and Sutton, 2017; Bouchacourt et al., 2018). The lack of consideration of topic coherence measures during the learning process of VAE-based topic models makes it difficult to control the quality of the generated topics. Intuitively, one solution is to jointly consider coherence scores in the learning objective. However, this is not feasible since coherence score is an unsupervised measure of topics based on a large-scale knowledge source, there is no ground truth “best topics”.

Another limitation of existing approaches is that they typically require a pre-processing step to filter infrequent and/or top frequent words in order to reduce the vocabulary size and achieve better topic extraction results. Word filtering is often done heuristically. Although there have been attempts to automatically distinguishing background words and topic words, existing approaches either require a switch variable defined at each word position to indicate whether the word is a background word, which makes the models cumbersome, or model each latent topic as the deviation in log-frequency from a constant background distribution (Eisenstein et al., 2011; Smith et al., 2018).

In this paper, we propose a new framework to use reinforcement learning (Pan et al., 2018; Qin et al., 2018; Yin et al., 2018) to incorporate the topic coherence measures into the learning of a neural topic model and filter background words dynamically. More concretely, given an input document, its constituent words will first be sampled...
by a weight vector which assigns higher weights to words with higher coherence scores and have more concentrated topic distributions. The sampled words will then be fed into a VAE-based neural topic model to reconstruct the original document. A reward function is deployed to take into account both topic coherence scores and the degree of word overlapping between topics. The reward signal derived is subsequently used to update the sampling weight vector for each word. In this way, we do not need to directly add the coherence scores into the loss function. Our experimental results show that our proposed framework outperforms the traditional topic model and existing neural topic modelling approaches on the 20 Newsgroups (Lang, 1995) and the NIPS data (Tan et al., 2017) in topic coherence and perplexity.

The rest of the paper is organized as follows. Section 2 presents our proposed reinforcement learning framework for topic modelling. Section 3 reports the experimental setup and results. Section 4 concludes the paper and outlines future research directions.

2 Proposed Method

In this section, we introduce our proposed reinforcement learning (RL) framework for topic modelling. A standard RL framework contains three components: action, state, and reward. Here, the action aims to select words with high coherence scores and filter background words. The state is the distribution of latent topics among words, which is obtained from a VAE-based topic model. The reward is a function to measure the quality of topics based on an external corpus and guide the weight updating of the next word selecting action. The overall architecture is illustrated in Figure 1.

We detail our framework in the following.

2.1 Action

For an input document $d = \{w_1, w_2, ..., w_U\}$, where each word $w_i$ is represented by a one-hot representation, the action is determined by a probabilistic vector $P = \{p_1, p_2, ..., p_U\}$ which is used to filter the less topical-coherent and background words at each iteration of model learning. Here, each $p_i$ present the sampling probability for word $w_i$, and $U$ is the full vocabulary size. We aim to select $V$ words from the full vocabulary based on $P$ and mask out other words in $d$, thus $\tilde{d} = \{w'_1, w'_2, ..., w'_V\}$ where $V \leq U$. The goal of our method is to assign higher probabilities to words which contribute more to the topic coherence scores and lower probabilities to those less topical-coherent words and background words which occur equally likely across topics.

2.2 State

After word selection, the new document representation $\tilde{d}$ is fed into a neural topic model to obtain the state, which is the topic distribution in the topic model. Here, we deploy the VAE (Miao et al., 2016; Card et al., 2017) to learn the latent topics, which consists of two main components, the Inference Network and the Generation Network. For the Inference Network, we use VAE to approximate the posterior distribution over topics for all the training instances. In the Generation Network, the words are generated via Gaussian softmax construction from the topic distribution generated by the Inference Network. The architecture of the neural topic model is shown in Figure (1) and we describe the model in more details below.

Inference Network. Following the idea of VAE
which computes a variational approximation to an intractable posterior using MLPs, we define two MLPs, \( f_{\mu_d} \) and \( f_{\Sigma_d} \), which takes as input the word counts in a document and outputs mean and variance of a Gaussian distribution, both being vectors in \( R^K \), \( \mu_d = f_{\mu_d}(w_d), \Sigma_d = \text{diag}(f_{\Sigma_d}(w_d)) \). Here, ‘diag’ converts a column vector to a diagonal matrix. For a document \( d \), its variational distribution is \( q(\theta) \approx \mathcal{N}(\mu_d, \Sigma_d) \). With such a formulation, we can generate samples from \( q(\theta) \) by first sampling \( \epsilon \sim \mathcal{N}(0, I^2) \) and then computing \( \hat{\theta} = \sigma(\mu_d + \Sigma_d^{1/2}\epsilon) \).

**Generation Network.** We feed the sampled \( \hat{\theta} \) to two MLPs to generate \( z_d \). Here, \( z_d \) is a \( K \)-dimensional latent topic representation of document \( d \). The probability of \( d \)-th word in \( n \)-th document \( w_{d,n} \) can be parameterised by another network,

\[
p(w_{d,n}|w_d, z_d) \propto \exp(m_d + W \cdot z_d)
\]

where \( m_d \) is the \( V \)-dimensional background log-frequency word distribution, \( W \in R^{V \times K} \) is a weight matrix. With the sampled \( \hat{\theta} \), for each document \( d \in N_d \), we can estimate the Evidence Lower Bound (ELBO) with a Monte Carlo approximation using \( L \) independent samples:

\[
L_s(w_d) \approx \frac{1}{L} \sum_{t=1}^L \sum_{n=1}^{N_d} \log p(w_{d,n}|\hat{\theta}(t)) - KL(q(z_d|w_d)||p(z_d))
\]

By minimising the ELBO in Eq. (2), the neural topic model reconstructs the input document \( w_d \). At the reconstruction layer, the matrix \( W \in R^{V \times K} \) in a single-layer network, which is used to capture the sampling weights between each word and the latent topics, is the state which produces specific topic coherence scores.

### 2.3 Reward

Intuitively, words with higher topic coherence and lower degree of overlapping among different topics should be assigned higher reward in the next iteration of learning. Hence, the reward function should be composed by two terms for each word: the average coherence score and topic overlapping value. The average coherence score is defined as:

\[
CO_{\text{average}} = (W \cdot C_t(W)) \odot P_{w_i \in w_d}
\]

where matrix \( W \) is the distribution of latent topics among words, \( C_t(W) \) is a \( K \)-dimension vector which contains the coherence score for each topic based on the sampling weight matrix, \( P_{w_i \in w_d} \) is the sampling probability for each word in document \( d \) (i.e., which action to take as described in Section 2.1), and \( \odot \) is the element-wise product. Hence, \( CO_{\text{average}} \) is a \( V \)-dimension weight vector to distribute coherence scores to the selected words based on sampling probabilities in action and topic distribution in topic modelling.

The Topic Overlapping (TO) is defined as:

\[
TO = \sum_{\text{row}}(\text{abs}(I - W \cdot W^T))
\]

where \( I \) is a \( V \times V \) identity matrix. \( TO \in R^{|v|} \) is to measure the separation based on mean distribution. In TO, the high value indicates that the associated word appears frequently across topics and hence could be considered as background words.

Based on the average coherence score and the topic overlapping value, the reward function is:

\[
R_t = CO_{\text{average}} - \alpha \cdot TO
\]

\[
Q_t = \beta \cdot Q_{t-1} + (1 - \beta) \cdot R_{t-1}
\]

where \( \alpha \) and \( \beta \in (0, 1) \) are trade-off coefficient. Then, the reward at the current time step, \( R_t \), and the history rewards encoded by \( Q_t \) will be used to update the sampling weight in the action:

\[
P_t = \max(P_{t-1} + \lambda_p \cdot (R_t - Q_t), 0 + \epsilon)
\]

where \( P_t \) is the sampling vector in action, \( \epsilon > 0 \) is a minimal value, and \( \lambda_p \) is the learning rate for \( P \). We choose a ramp function with \( \epsilon \) to ensure the sampling probability is positive.

### 2.4 Training

For pre-processing, we performed stop word removal* and use Adam to optimise the parameters in the neural networks. The learning rate for \( VAE \) and \( \theta_p \) are both 0.0001, the mini-batch size is 32, \( \alpha \) is 0.1, \( \beta \) is 0.5, \( \epsilon \) is 0.01, and the coherence scores are obtained from Wikipedia. The parameters in VAE are updated in each mini-batch, and the probabilistic vector \( P \) for action selection is updated every 2,000 mini-batches.

### 3 Experiments

We evaluate our model on the 20 Newsgroups† consisting of 18K documents, and NIPS (Tan

---

*http://mallet.cs.umass.edu/import-stoplist.php

†http://qwone.com/~jason/20Newsgroups/
<table>
<thead>
<tr>
<th>Methods</th>
<th>20News PPL</th>
<th>20News ( C_v )</th>
<th>NIPS PPL</th>
<th>NIPS ( C_v )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>#Topics = 30</strong>, frequency-based vocab.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>1,213.1</td>
<td>0.503</td>
<td>1,042.7</td>
<td>0.507</td>
</tr>
<tr>
<td>NVDM</td>
<td>980.8</td>
<td>0.497</td>
<td>931.6</td>
<td>0.492</td>
</tr>
<tr>
<td>NGTM</td>
<td>929.3</td>
<td>0.479</td>
<td>938.9</td>
<td>0.503</td>
</tr>
<tr>
<td>Scholar</td>
<td>1,345.9</td>
<td>0.537</td>
<td>1,350.9</td>
<td>0.512</td>
</tr>
<tr>
<td><strong>#Topics = 30</strong>, RL-based vocab.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>1,451.7</td>
<td>0.522</td>
<td>1,093.1</td>
<td>0.534</td>
</tr>
<tr>
<td>NVDM</td>
<td>845.8</td>
<td>0.510</td>
<td>768.7</td>
<td>0.509</td>
</tr>
<tr>
<td>NGTM</td>
<td>791.5</td>
<td>0.517</td>
<td>757.2</td>
<td>0.527</td>
</tr>
<tr>
<td>Scholar</td>
<td>1,158.4</td>
<td>0.560</td>
<td>1,273.6</td>
<td>0.548</td>
</tr>
<tr>
<td>VTMRL</td>
<td>803.7</td>
<td><strong>0.577</strong></td>
<td><strong>730.6</strong></td>
<td>0.568</td>
</tr>
<tr>
<td><strong>#Topics = 50</strong>, frequency-based vocab.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>1,015.9</td>
<td>0.501</td>
<td>995.5</td>
<td>0.503</td>
</tr>
<tr>
<td>NVDM</td>
<td>1,014.0</td>
<td>0.471</td>
<td>927.6</td>
<td>0.506</td>
</tr>
<tr>
<td>NGTM</td>
<td>903.5</td>
<td>0.491</td>
<td>908.8</td>
<td>0.498</td>
</tr>
<tr>
<td>Scholar</td>
<td>1,514.5</td>
<td>0.521</td>
<td>1,373.2</td>
<td>0.508</td>
</tr>
<tr>
<td>VTMRL</td>
<td>725.2</td>
<td><strong>0.559</strong></td>
<td><strong>712.2</strong></td>
<td>0.566</td>
</tr>
<tr>
<td><strong>#Topics = 50</strong>, RL-based vocab.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>1,251.6</td>
<td>0.518</td>
<td>921.4</td>
<td>0.527</td>
</tr>
<tr>
<td>NVDM</td>
<td>837.9</td>
<td>0.502</td>
<td>767.0</td>
<td>0.514</td>
</tr>
<tr>
<td>NGTM</td>
<td>772.2</td>
<td>0.514</td>
<td>749.7</td>
<td>0.511</td>
</tr>
<tr>
<td>Scholar</td>
<td>1,335.9</td>
<td>0.526</td>
<td>1,299.8</td>
<td>0.530</td>
</tr>
<tr>
<td>VTMRL</td>
<td><strong>725.2</strong></td>
<td><strong>0.559</strong></td>
<td><strong>712.2</strong></td>
<td>0.566</td>
</tr>
</tbody>
</table>

Table 1: Perplexity and topic coherence results of different models. ‘frequency-based vocab.’ denotes that the vocabulary is constructed by filtering out rare words while ‘RL-based vocab.’ denotes that the vocabulary is dynamically generated by our model using RL.

et al., 2017) consisting of 6.6k documents. We use 10% training data as the validation set to fine-tune the parameters. We compare our results with those obtained from the following baselines:

**LDA:** Latent Dirichlet Allocation Model (Blei et al., 2003).

**NVDM:** Neural Variational Document model (Miao et al., 2016).

**NGTM:** Neural Generative Topic model (Card et al., 2017).

**Scholar:** Topic model with metadata (Card et al., 2018).

**VTMRL:** Our proposed Variational Topic Model with Reinforcement Learning.

For all the baseline models, we follow the common pre-processing step in existing approaches by performing stop word removal, and selecting the most frequent 2,000 words as the vocabulary. Since our proposed framework dynamically select words at each iteration of learning, we do not need to pre-set the vocabulary prior to model learning. Instead, we only activate 2,000 words at each mini-batch of training based on the word sampling probabilities. As our model dynamically select words during the training process, in order to ensure fair comparison with other models, we also report the results of training baselines using the vocabulary dynamically generated by our model.

In our experiments, the models are evaluated based on the perplexity (PPL, lower is better) and topic coherence measure \( C_v \) based on external corpus (Röder et al., 2015) (higher is better). The results with 30 and 50 topics are shown in Table 1.

LDA is a conventional topic model, while all the other models are neural topic models. It can be observed from Table 1 that NVDM and NGTM achieve better perplexities compared to LDA. However, in terms of topic coherence measure, NVDM and NGTM perform slightly worse than LDA. A similar observation has been reported in (Card et al., 2017). Scholar achieves better coherence compared to other neural models. Nevertheless, after using reinforcement learning based on the topic coherence scores in our proposed model, VTMRL outperforms all the other models on the topic coherence measure by a large margin. RL could activate words which are semantically related to topics regardless of their occurrence frequency. The inclusion of some rare words would impact the models’ predictive probabilities. As such, we observe worse perplexity results for models trained with RL-based vocabulary compared to frequency-based vocabulary in 20 Newsgroups, though the converse is true for NIPS. Nevertheless, the coherence scores improve for all the models with RL-based vocabulary.

As incorporating RL could increase the computational complexity of VTMRL, we report in Table 2 the total number of parameters and average training time per epoch when the vocabulary size is 2,000 and the number of topics is 50.

Strictly speaking, the number of parameters in LDA is not directly comparable with neural models. Neural models have similar parameter size. With the incorporation of RL, VTMRL only increased the parameter size by 1.4%. Due to the efficiency of GPU, the running costs of neural models are better than that of LDA. Although our proposed VTMRL used full vocabulary, the active words in each epoch are limited. Hence, there is no significant increase in terms of the running cost.
<table>
<thead>
<tr>
<th>Model</th>
<th>#parameters</th>
<th>Training time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>$1.00 \times 10^5$</td>
<td>660.89</td>
</tr>
<tr>
<td>NVDM</td>
<td>$1.38 \times 10^6$</td>
<td>62.16</td>
</tr>
<tr>
<td>NGTM</td>
<td>$1.39 \times 10^6$</td>
<td>72.12</td>
</tr>
<tr>
<td>Scholar</td>
<td>$1.15 \times 10^6$</td>
<td>3.41</td>
</tr>
<tr>
<td>VTMRL</td>
<td>$1.41 \times 10^6$</td>
<td>5.78</td>
</tr>
</tbody>
</table>

Table 2: Number of parameters for each model and the average training time per epoch with vocabulary size 2,000 and topic number 50.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Topic Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>university &lt;NUM&gt; subject host idea</td>
</tr>
<tr>
<td>2</td>
<td>organization article writes surrender lines</td>
</tr>
<tr>
<td>3</td>
<td>people organization posting article lines</td>
</tr>
<tr>
<td>1</td>
<td>mouse x11r5 keyboard serial remote</td>
</tr>
<tr>
<td>2</td>
<td>chip design products build system</td>
</tr>
<tr>
<td>3</td>
<td>drives friend sports espn michigan</td>
</tr>
</tbody>
</table>

Table 3: Example topic words with/without RL by VTMRL (#Topics = 50) in 20 Newsgroup.

We show in Table 3 example topics with/without RL by VTMRL in 20 Newsgroups. The RL method seems producing more interpretable topics. Also, due to the reward-based words sampling in RL, words with low occurrence frequency would still have a chance to be promoted in specific topics, such as ‘x11r5’, which is a serial number of the Windows system.

We next compare the topic coherence changes during model training. We observe that for VTMRL the coherence value increases at the beginning of the training and remains relatively stable in subsequent training iterations. As a contrast, the coherence value of our model without RL is not stable, and decreases rapidly after 10 training epochs. This is not surprising since the model without RL did not consider topic coherence in its learning process.

We also evaluate the effectiveness of using the learned topics as features to train text classifiers on the 20 Newsgroups data. The results are obtained by using logistic regression as the classifier trained from the topics generated by various aforementioned models. We also report the results by training logistic regression from the combination of word features (tf-idf) and topic features (#topic = 30). In addition, we include the results using neural models such as CNN and RNN in Table 4.

<table>
<thead>
<tr>
<th>Model</th>
<th>Acc</th>
<th>Model</th>
<th>Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>0.412</td>
<td>tf-idf + LDA</td>
<td>0.822</td>
</tr>
<tr>
<td>NGTM</td>
<td>0.375</td>
<td>tf-idf + NGTM</td>
<td>0.825</td>
</tr>
<tr>
<td>NVDM</td>
<td>0.303</td>
<td>tf-idf + NVDM</td>
<td>0.824</td>
</tr>
<tr>
<td>VTMRL</td>
<td>0.478</td>
<td>tf-idf + VTMRL</td>
<td>0.830</td>
</tr>
<tr>
<td>CNN</td>
<td>0.515</td>
<td>RNN</td>
<td>0.509</td>
</tr>
</tbody>
</table>

Table 4: Text classification accuracy of different models on the 20 Newsgroups data.

Figure 2: Topic coherence changes with/without RL (#Topics = 50).

We show in Figure 2 topic coherence changes with/without RL by VTMRL in 20 Newsgroups.

In this paper, we have proposed a new reinforcement learning (RL) framework for neural topic modelling, where words are activated dynamically by RL according to topic coherence scores and topic overlapping values. The experiments on the 20 Newsgroups and NIPS datasets show encouraging results both on perplexity and topic coherence measures in comparison with existing neural topic models. In future work, we will explore extending our model for temporal topic modelling.

4 Conclusion

In this paper, we have proposed a new reinforcement learning (RL) framework for neural topic modelling, where words are activated dynamically by RL according to topic coherence scores and topic overlapping values. The experiments on the 20 Newsgroups and NIPS datasets show encouraging results both on perplexity and topic coherence measures in comparison with existing neural topic models. In future work, we will explore extending our model for temporal topic modelling.

Acknowledgment

This work was supported by the Innovate UK (grant no. 103652), the EU-H2020 (grant no. 794196), the National Natural Science Foundation of China U1636103, 61876053.
References


