

Supplementary Material

Anonymous ACL submission

1 Training Details

We train `tdlm` and `lstm` for 10 epochs, resetting model parameters to the previous epoch's whenever validation loss worsens. Word embeddings are initialised with pre-trained Google News vectors.¹ `tdlm` is optimised using Adam, while `lstm` is optimised using Adagrad. After training, validation perplexity is 59.71 for `lstm` and 65.10 for `tdlm`.

We follow the optimal hyper-parameter settings as suggested by Lau et al. (2017). Configurations for `tdlm` and `lstm` are detailed in Table 1 and 2 respectively.

References

Jey Han Lau, Timothy Baldwin, and Trevor Cohn. 2017. Topically driven neural language model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada, pages 355–365.

¹<https://code.google.com/archive/p/word2vec/>.

Value	Hyper-parameter	Description
3	Output sequence length for topic model	
30	Sequence length for language model	
300	Maximum document length	
64	Minibatch size	
1	Number of LSTM layers	
600	LSTM hidden size	
10	Number of training epochs	
100	Number of topics	
300	Word embedding size	
2	Convolutional filter width	
20	Topic input vector size or number of features for convolutional filter	
50	Topic output vector size	
Adam	Optimiser	
0.001	Learning rate of optimiser	
0.4	Topic model dropout keep probability	
0.6	Language model dropout keep probability	

Table 1: `tdlm` hyper-parameters

Value	Hyper-parameter	Description
30	Sequence length for language model	
300	Maximum document length	
64	Minibatch size	
1	Number of LSTM layers	
600	LSTM hidden size	
10	Number of training epochs	
300	Word embedding size	
Adagrad	Optimiser	
0.2	Learning rate of optimiser	
0.6	Language model dropout keep probability	

Table 2: `lstm` hyper-parameters