S2VNTM: SEMI-SUPERVISED VMF NEURAL TOPIC MODELING

Weijie Xu, Jay Desai, Srinivasan Sengamedu, Xiaoyu Jiang & Francis Iannacci
Amazon
weijiexu@amazon.com

ABSTRACT

Language model based methods are powerful techniques for text classification. However, the models have several shortcomings. (1) It is difficult to integrate human knowledge such as keywords. (2) It needs a lot of resources to train the models. (3) It relied on large text data to pretrain. In this paper, we propose Semi-Supervised vMF Neural Topic Modeling (S2vNTM) to overcome these difficulties. S2vNTM takes a few seed keywords as input for topics. S2vNTM leverages the pattern of keywords to identify potential topics, as well as optimize the quality of topics' keywords sets. Across a variety of datasets, S2vNTM outperforms existing semi-supervised topic modeling methods in classification accuracy with limited keywords provided. S2vNTM is at least twice as fast as baselines.

1 INTRODUCTION

Language Model (LM) pre-training Vaswani et al.; Devlin et al. (2018) has proven to be useful in learning universal language representations. Recent language models such as Yang et al. (2019); Sun et al. (2019); Chen et al. (2022); Ding et al. (2021) have achieved amazing results in text classification. Most of these methods need enough high-quality labels to train. To make LM based methods work well when limited labels are available, few shot learning methods such as Bianchi et al. (2021); Meng et al. (2020a;b); Mekala and Shang (2020); Yu et al. (2021); Wang et al. (2021b) have been proposed. However, these methods rely on large pre-trained texts and can be biased to apply to a different environment.

Topic modeling methods generate topics based on the pattern of words. To be specific, unsupervised topic modeling methods Blei et al. (2003); Teh et al. (2006); Miao et al. (2018); Dieng et al. (2020) discover the abstract topics that occur in a collection of documents. Recently developed neural topic modeling achieves faster inference in integrating topic modeling methods with deep neural networks and uncovers semantic relationship Zhao et al. (2020a); Wang and Yang (2020). Compared to unsupervised topic modeling methods, semi-supervised topic modeling methods Mao et al. (2012); Jagarlamudi et al. (2012); Gallagher et al. (2018) allow the model to match the provided patterns from users such as keywords. However, these methods do not have high topic classification accuracy.

After studying topic modeling methods in real world applications Choi et al. (2017); Cao et al. (2019); Kim et al. (2013); Zhao et al. (2020b); Xu et al. (2022), we realize the scenario that cannot be solved by current methods. The scenario involves topic exploration: users have identified a subset of topic keywords. They want to capture topics based on these keywords, while explore additional topics. They value the quality of the resulting topics and want to identify new topics while refining the topics' keywords iteratively Kim et al. (2013); Smith et al. (2018). In addition, users want to use the topic they created on topic classification.

In this work, we propose semi-supervised vMF neural topic modeling (S2vNTM). S2vNTM takes the desired number of topics as well as keywords/key phrases for some subsets of topics as input. It incorporates this information as guideline and leverages negative sampling to create topics that match the pattern of selected keywords. It creates additional topics which align with the semantic structure of the documents. It can help users remove redundant topics. Figure 1 illustrates how users interact with our model. The advantages of this method include:
1. It consistently achieves the best topic classification performance on different datasets compared to similar methods.

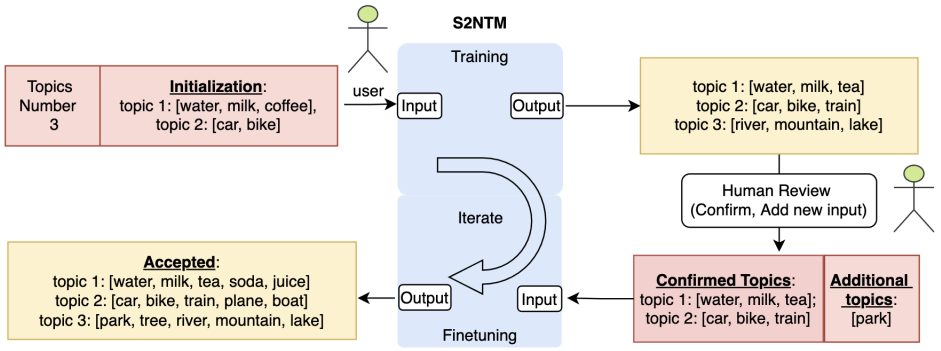

Figure 1: An S2vNTM application scenario. Human experts define topic keywords set and the number of topics first. During the training procedure, S2vNTM outputs keywords for each topic by merging the redundant keywords group and identifying new topics. Human experts then confirm/remove the keywords and/or add new keywords. S2vNTM continues refining the keyword list with a fast fine-tuning procedure. After a few iterations, S2vNTM provides users topics with high-quality keywords and high topic classification accuracy.

2. S2vNTM only requires a few seed keywords per topic, and this makes it suitable for data scarce settings. It does not require any transfer learning.
3. S2vNTM is explainable and easy to fine-tune which makes it suitable for interfacing with subject-matter experts and low resource settings.

In sections below, we have shown Method in Section 2 which describes the technical details of S2vNTM, Results in Section 3 and Conclusion and Future work in Section 4. Details on Modularity of S2vNTM is given in Appendix A. Related Work and Challenges are described in Appendix B, Experiments in Appendix C and Ablation Studies in Appendix E.

2 METHOD

Figure 2 shows the overall architecture of S2vNTM. The encoder is based on a Neural Topic Model leveraging von Mises-Fisher distribution. We use von Mises-Fisher distribution because it captures distributions on unit sphere and induces better clustering properties. To improve clustering, we add temperature function to the latent distribution(See details in Appendix A.1). The decoder tries to reconstruct the input from the topics while leveraging user-provided seeds for the topics. The model is trained end-to-end with the objective of minimizing reconstruction error while conforming to user-provided seeds and minimizing topic overlap.

2.1 vNTM

We first introduce notation: the encoder network, ϕ, encodes the bag of words representation of any document X_d and outputs the parameters which can be used to sample the topic distribution t_d. The decoder is represented by a vocabulary embedding matrix e_W and a topic embedding matrix e_t. We use a spherical word embedding Meng et al. (2019) trained on the dataset where we apply the model to create e_W and keep it fixed during the training. Spherical word embedding performs better on word similarity related tasks. If we do not keep embedding fixed, reconstruction loss will make the embeddings of co-occurred words closer which is not aligned with true word similarity. Fewer parameters to train can also make our method more stable. W represents all selected vocabularies and T contains all topics. In this notation, our algorithm can be described as follows: for every document d, (1) input bag of word representation X_d to encoder ϕ. (2) Using ϕ, output direction parameter μ and variation parameter κ for vMF distribution. (3) Based on μ and κ, generate a topic distribution t_d using temperature function. (4) Reconstruct X_d by $t_d \times \text{softmax}(e_t e_W^T)$. The goal of this model is to maximize the marginal likelihood of the documents: $\sum_{d=1}^{D} \log p(X_d|e_t, e_W)$. To make it tractable,

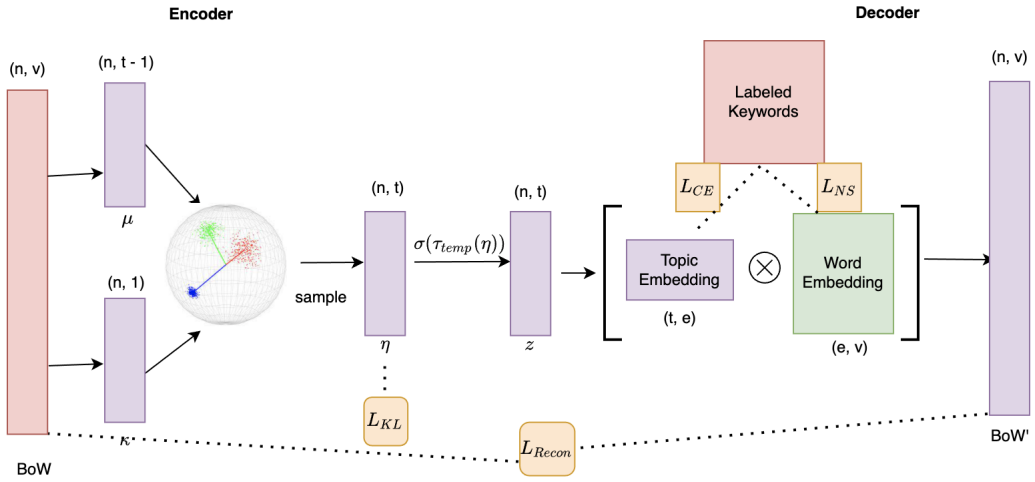

Figure 2: The neural network architecture of S2vNTM. We denote the dimension of the data in the bracket. n is the number of documents. v is the number of vocabularies. t is the number of topics. e is the dimension of embeddings. Word Embedding(green) is fixed during the training. Pink represents user provided data. Orange denotes all loss function including L_{KL}, L_{Recon}, L_{CE} and L_{NS}

the loss function combines reconstruction loss with KL divergence as below:

$$L_{Recon} = (-E_{q_\phi(t_d|X_d)}[logp_\theta(X_d|t_d)] \tag{1}$$

$$L_{KL} = KL[q_\phi(t_d|X_d)||p(t_d)]) \tag{2}$$

Our spherical word embedding is trained on the dataset without any pretraining. This can help embeddings deal with domain specific word. This can also make our model work for the language where there is not much text data available to pre-train. We leverage the vMF distribution as our latent distribution because of its clusterability and stability Xu and Durrett (2018); Ennajari et al. (2021); Reisinger et al. (2010); Davidson et al. (2018). Because of the design of the decoder, for each topic, it can be represented as a distribution of all words in vocabulary ($\mathrm{softmax}(e_t e_W^T)$). *When a document is provided, the user can identify the topics distribution of documents and also related keywords that contribute to these topics. Thus, the model is explainable.*

2.2 Loss Function

Our method allows users to define an arbitrary number of topics and provide keywords for some subsets of those topics. The model takes these two parameters as inputs and generates topics that include user's keywords as well as additional topics that align with topic distribution. With that being said, we want the prior loss similar to

$$L_{CE} = -\sum_{s \in S} max_{t \in T} \log \prod_{x \in s} q(x|t) \tag{3}$$

where S contains all keywords groups, s is a group of keywords and T is the group of topics, $q(x|t)$ stands for the probability of word x given t calculated by decoder.

$$q(x|t) = \frac{\exp\left(e_{t_j} e_{x_i}^T\right)}{\sum_{x \in X} \exp\left(e_{t_j} e_{x^T}\right)} \tag{4}$$

This is the j-th row and i-th column of decoder embedding matrix $\mathrm{softmax}(e_T e_W^T)$. Thus, it uses existed neural network structure to calculate and makes it computationally efficient.

2.3 TOPIC AND KEYWORDS SET MATCHING

We want to make sure matched topics capture all documents related to the provided keywords. The problem of using L_{CE} is that different keywords set may map to the same topic. It may merge the irrelevant topic set when that topic set is not aligned with most of the topics. To avoid this situation, we first select the topic that is most likely to align with this group of keywords but not align with words in all other groups. To be specific, we first select

$$t_s = \arg\max_{t \in T}(E_{x \in s}(\log q(x|t)) - \max_{x \in S} \log(q(x|t))) \tag{5}$$

This is inspired by Gumbel-Softmax Jang et al. (2016). If one word in keywords set is dissimilar to the topic, the log will penalize it heavily and the topic is less likely to be matched. We also want to separate keyword groups which are different. If a keyword in another group has a higher probability in a topic, then $max_{s \in S} \log(q(s|t))$ will be large, which makes the topic less likely to be the selected topic. If we have two similar keywords' sets, they can have similar and large $E_{x \in s}(\log q(x|t))$. These keywords sets can still map to the same topics. The benefit of this matching method is that it is more stable compare to method such as Gumbel-softmax and it can remove redundant topics by merging it with similar topics.

2.4 NEGATIVE SAMPLING

We also want keywords as guidance to select other related keywords. Similar to Yang et al. (2020), when a keyword set is matched with a topic, we want the topic to be less correlated with words that are unrelated to the matched keyword set. Thus, we leverage negative sampling. We first select the top N words in the selected topic using a decoder embedding matrix and sample each of top N word with sampling probability equal to $max_{x \in s} 1 - cos(x, x_N)$ where x_N stands for a word in top N words in that selected topic and cos stands for cosine similarity. Our goal is to make words that are dissimilar to the provided keywords likely to be sampled, as seen in Table 2. Negative Sampling can also help the model converge faster since it pushes away unrelated words quicker Mimno and Thompson (2017). The penalty we add for each keywords' set is:

$$L_{NS,s} = \gamma \sum_{x \in ns} (\log(q(x|t_s))) \tag{6}$$

where ns contains words sampled from negative sampling. The loss of negative sampling is

$$L_{NS} = \sum_{s \in S} L_{NS,s} \tag{7}$$

β controls input keywords strength on overall loss function and γ controls the strength of negative sampling. The overall loss function is:

$$L = L_{Recon} + L_{KL} + \beta * L_{CE} + \gamma * L_{NS} \tag{8}$$

where L_{NS} is the sum of all keywords set. L_{Recon} is the reconstruction loss and L_{KL} is the KL divergence loss. The benefit of this negative sampling design is that $q(x|t_s)$ can be directly mapped from the decoder. Thus, it does not require additional computation, which saves computation resources.

Model	AG News			R8			DBLP		
Metric	Accuracy	Aucroc	Macro F1	Accuracy	Aucroc	Macro F1	Accuracy	Aucroc	Macro F1
GuidedLDA	0.734 ± 0.037	0.857 ± 0.016	0.735 ± 0.039	0.54 ± 0.012	**0.872±0.012**	0.309± 0.017	0.493±0.009	0.693±0.005	0.47±0.008
CoreEx	0.778±0.003	0.889±0.001	0.765±0.002	0.532± 0.051	0.762±0.025	**0.394±0.024**	0.53± 0.009	**0.8±0.005**	0.492±0.01
S2vNTM	**0.795±0.009**	**0.902±0.007**	**0.792±0.009**	**0.651±0.03**	0.813±0.022	0.362±0.049	**0.598±0.029**	0.793±0.022	**0.545±0.032**

Table 1: Scores and Standard Deviation for Accuracy, Macro F1 and Aucroc of GuidedLDA, CoreEx and S2vNTM models on AG News, R8 and DBLP datasets.

3 RESULTS

We ran our experiments 10 times with different seeds and show the result in Table 1 (and Figure 5 in the Appendix). (1) S2vNTM achieves the best accuracy in all three datasets. In fact, the worst

reported accuracy of S2vNTM is higher than the best from the other two methods. We believe there are 3 reasons contributing to its superior performance. (i) It has high clusterability using vMF as a latent distribution. This makes our method easily clustered. (ii) Negative sampling excludes unrelated keywords from the topics. This makes our method perform better on documents that are related to keywords. (iii) S2vNTM also uses word embedding trained on the dataset. This makes our method perform well on documents that have words that are similar to words in keywords set. (2) S2vNTM keywords make more sense qualitatively in Table 2 in Appendix. This is due to KL divergence loss. Flexible concentration parameter κ makes our method more locally concentrated. This makes topics different from each other. (3) S2vNTM also has a higher aucroc and Macro F1 score than other methods in most cases (from Table 1). This means that our method can deal with imbalanced datasets and can easily distinguish between classes. However, it performs less well on R8, which has 8 imbalanced classes. For class with less than 300 documents, keywords selected by tf-idf are less representative. Thus, it has lower performance and higher variance. Besides, our method using vMF distribution which has higher reconstruction loss when the dimension is high. R8 has 8 classes which make our method perform worse.

Qualitatively, as you can see in Table 2 , negative sampling reduces the importance of unrelated keywords such as *call, york, company* while increasing the importance of given keywords such as *military, industry, athlete*. Also, semantically, keywords in each set are closer to each other. For example, in the first set of keywords, *government, war* are semantically more related to *crime, rule* compared to *call, election*. On the other hand, even if CorEx has good topic diversity, the keywords set is not coherent. For example, the last group in Table D has *inc, corp, people, bush, million* in one group. Determining the relationship between these keywords is not obvious.

Speed We run each model 10 times on AG News with different seeds to evaluate how long it takes to fine-tune the model by modifying 20 percent of keywords set. The average fine-tune time for our method is 51.33 seconds. To compare, CatE Meng et al. (2018) takes 888.61 seconds to fine-tune, while CorEx takes 94.98 seconds to fine-tune. This shows that our method is better suitable for iterative topic learning Hu et al. (2014) and resource restrictive environments.

Overall, qualitative results show that *S2vNTM can help users find more coherent and relevant keywords compare to existed methods. Negative sampling makes the topics set more coherent. S2vNTM is at least twice faster than baselines.*

S2vNTM	S2vNTM + Negative Sampling
government, **war**, president, call, election	**government**, **war**, **military**, crime, rule
stock, high, investor, **market**, york	**stock**, investor, **market**, share, **industry**
software, **computer**, system, microsoft, company	**software**, **computer**, microsoft, system, technology
game, sport, champion, season, team	game, sport, champion, season, **athelete**
united, reuters, international, state, union	reuters, united, state, international, plan
reuters, report, target, http, company	reuters, report, target, http, company

Table 2: Comparison of top 5 keywords from each topics on AG News. The keywords that are given are [government,military,war], [stock,market,industry], [computer,telescope,software], [basketball,football,athlete].

4 Conclusion and Future Work

In conclusion, we propose S2vNTM as an approach to integrate keywords as pattern to current neural topic modeling methods. It is based on vMF distribution, negative sampling, modified topic keywords mapping and spherical word embeddings. Our method achieves better classification performance compared to existing semi-supervised topic modeling methods. It is not sensitive to parameters. S2vNTM gives more coherent topics qualitatively. It also performs well when the input keywords set is less common in the dataset. It is also fast to fine-tune. It does not require pretraining or transfer learning. It only needs a few sets of seed words as input.

The ablation study shows the potential of our method to further improve. In the future, we will focus on decreasing the gap between loss function and classification metric, incorporating sequential information and further improving the stability of the model. We will also work on improving its expressability in higher dimensions.

REFERENCES

Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2021. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. In *ACL*.

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

Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. 2016. Hard negative mining for metric learning based zero-shot classification. In *European Conference on Computer Vision*, pages 524–531. Springer.

Sophie Burkhardt and Stefan Kramer. 2019. Decoupling sparsity and smoothness in the dirichlet variational autoencoder topic model. *Journal of Machine Learning Research*, 20(131):1–27.

Buqing Cao, Jianxun Liu, Yiping Wen, Hongtao Li, Qiaoxiang Xiao, and Jinjun Chen. 2019. Qos-aware service recommendation based on relational topic model and factorization machines for iot mashup applications. *Journal of parallel and distributed computing*, 132:177–189.

Qianben Chen, Richong Zhang, Yaowei Zheng, and Yongyi Mao. 2022. Dual contrastive learning: Text classification via label-aware data augmentation.

Hye-Jeong Choi, Minho Kwak, Seohyun Kim, Jiawei Xiong, Allan S Cohen, and Brian A Bottge. 2017. An application of a topic model to two educational assessments. In *The Annual Meeting of the Psychometric Society*, pages 449–459. Springer.

Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. 2018. Hyperspherical variational auto-encoders. *arXiv preprint arXiv:1804.00891*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

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.

SiYu Ding, Junyuan Shang, Shuohuan Wang, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2021. ERNIE-Doc: A retrospective long-document modeling transformer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2914–2927, Online. Association for Computational Linguistics.

Hafsa Ennajari, Nizar Bouguila, and Jamal Bentahar. 2021. Combining knowledge graph and word embeddings for spherical topic modeling. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15.

Ryan J. Gallagher, Kyle Reing, David Kale, and Greg Ver Steeg. 2018. Anchored correlation explanation: Topic modeling with minimal domain knowledge.

Ian Gemp, Ramesh Nallapati, Ran Ding, Feng Nan, and Bing Xiang. 2019. Weakly semi-supervised neural topic models.

Kelvin Guu, Tatsunori B. Hashimoto, Yonatan Oren, and Percy Liang. 2018. Generating sentences by editing prototypes.

Bahareh Harandizadeh, J. Hunter Priniski, and Fred Morstatter. 2022. Keyword assisted embedded topic model. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. ACM.

Yuening Hu, Jordan Boyd-Graber, Brianna Satinoff, and Alison Smith. 2014. Interactive topic modeling. *Machine learning*, 95(3):423–469.

Jagadeesh Jagarlamudi, Hal Daumé III, and Raghavendra Udupa. 2012. Incorporating lexical priors into topic models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–213, Avignon, France. Association for Computational Linguistics.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Hyun Duk Kim, Malu Castellanos, Meichun Hsu, ChengXiang Zhai, Thomas Rietz, and Daniel Diermeier. 2013. Mining causal topics in text data: iterative topic modeling with time series feedback. In *Proceedings of the 22nd ACM international conference on information & knowledge management*, pages 885–890.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. 2014. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pages 3581–3589.

David D. Lewis. 1997. Reuters-21578 text categorization test collection, distribution 1.0.

Xirong Li, CeesG M Snoek, Marcel Worring, Dennis Koelma, and Arnold WM Smeulders. 2013. Bootstrapping visual categorization with relevant negatives. *IEEE Transactions on Multimedia*, 15(4):933–945.

Xian-Ling Mao, Zhaoyan Ming, Tat-Seng Chua, Si Li, Hongfei Yan, and Xiaoming Li. 2012. Sshlda: a semi-supervised hierarchical topic model. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 800–809.

Dheeraj Mekala and Jingbo Shang. 2020. Contextualized weak supervision for text classification. In *ACL*.

Yu Meng, Jiaxin Huang, Guangyuan Wang, Zihan Wang, Chao Zhang, Yu Zhang, and Jiawei Han. 2020a. Discriminative topic mining via category-name guided text embedding. In *WWW*.

Yu Meng, Jiaxin Huang, Guangyuan Wang, Chao Zhang, Honglei Zhuang, Lance Kaplan, and Jiawei Han. 2019. Spherical text embedding.

Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018. Weakly-supervised hierarchical text classification.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. 2020b. Text classification using label names only: A language model self-training approach. In *EMNLP*.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2018. Discovering discrete latent topics with neural variational inference.

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, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

David Mimno and Laure Thompson. 2017. The strange geometry of skip-gram with negative sampling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2873–2878, Copenhagen, Denmark. Association for Computational Linguistics.

Andriy Mnih and Koray Kavukcuoglu. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In *Advances in neural information processing systems*, pages 2265–2273.

Shirui Pan, Jia Wu, Xingquan Zhu, Chengqi Zhang, and Yang Wang. 2016. Tri-party deep network representation. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 1895–1901. IJCAI/AAAI Press.

Jeffrey Pennington, Richard Socher, and Christopher 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, Doha, Qatar. Association for Computational Linguistics.

Joseph Reisinger, Austin Waters, Bryan Silverthorn, and Raymond J Mooney. 2010. Spherical topic models. In *ICML*.

Tian Shi, Kyeongpil Kang, Jaegul Choo, and Chandan K Reddy. 2018. Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *Proceedings of the 2018 World Wide Web Conference*, pages 1105–1114.

Alison Smith, Varun Kumar, Jordan Boyd-Graber, Kevin Seppi, and Leah Findlater. 2018. Closing the loop: User-centered design and evaluation of a human-in-the-loop topic modeling system. In *23rd International Conference on Intelligent User Interfaces*, pages 293–304.

Leslie N. Smith and Nicholay Topin. 2018. Super-convergence: Very fast training of neural networks using large learning rates.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification?

Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476):1566–1581.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. [link].

Wei Wang, Bing Guo, Yan Shen, Han Yang, Yaosen Chen, and Xinhua Suo. 2021a. Neural labeled lda: a topic model for semi-supervised document classification.

Xinyi Wang and Yi Yang. 2020. Neural topic model with attention for supervised learning. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 1147–1156. PMLR.

Zihan Wang, Dheeraj Mekala, and Jingbo Shang. 2021b. X-class: Text classification with extremely weak supervision. In *NAACL*.

Xiaobao Wu, Chunping Li, Yan Zhu, and Yishu Miao. 2020. Short text topic modeling with topic distribution quantization and negative sampling decoder. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1772–1782, Online. Association for Computational Linguistics.

Jiacheng Xu and Greg Durrett. 2018. Spherical latent spaces for stable variational autoencoders.

Weijie Xu, Xiaoyu Jiang, Jay Desai, Bin Han, Fuqin Yan, and Francis Iannacci. 2022. Kdstm: Neural semi-supervised topic model-ing with knowledge distillation.

Ji Yang, Xinyang Yi, Derek Zhiyuan Cheng, Lichan Hong, Yang Li, Simon Xiaoming Wang, Taibai Xu, and Ed H Chi. 2020. Mixed negative sampling for learning two-tower neural networks in recommendations. In *Companion Proceedings of the Web Conference 2020*, pages 441–447.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang. 2021. Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. In *NAACL*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2016. Character-level convolutional networks for text classification.

He Zhao, Dinh Phung, Viet Huynh, Trung Le, and Wray Buntine. 2020a. Neural topic model via optimal transport.

Jinjin Zhao, Kim Larson, Weijie Xu, Neelesh Gattani, and Candace Thille. 2020b. Targeted feedback generation for constructed-response questions.

Appendix

A MODULARITY OF S2VNTM

Our methods can be plugged into variational autoencoder based topic modeling methods such as NVDM Miao et al. (2016) and NSTM Zhao et al. (2020a). For NVDM, since their decoder is a multinomial logistic regression, we can consider that as the distribution of word over the topic. For L_{CE}, we can change $P(e_w|e_t)$ to $P_\theta(x_i|h)$ (formula (6) in Miao et al. (2016)) as it also represents the probability of certain word given all other words. For L_{NS}, we just sample it the same way as Mikolov et al. (2013). For NSTM, since they also maintain topics and word embeddings (They name it G and E in the paper), we can use cosine similarity of these embeddings to create the loss functions L_{CE} and L_{NS} respectively. For that being said, this work can be easily extended by existed unsupervised neural topic modeling methods.

A.1 TEMPERATURE FUNCTION AND FLEXIBLE κ

Step (3) in Section 2.1 introduced the concept of a temperature function. Temperature is a function that applies to the sample generated by vMF distribution to form a topic distribution. To be specific,

$$t_d = \text{softmax}(\tau_{temp}(\eta_d)) \tag{9}$$

where η_d is the vector of sampled vMF distribution. Since the sample from vMF is on the surface of a sphere, we have

$$\sum(\eta_d^2) = 1 \tag{10}$$

In cases where the number of topics equals to 10, the most polarized η_d is $(1, 0, 0, 0, 0, ...)$. If we apply softmax to this η_d, the highest topic proportion is 0.23, making latent space entangled and limit the clusterability.

To overcome the expressibility concern mentioned in Related Work in Appendix B.5, temperature function τ_{temp} is used to increase expressibility. For example, if we let $\tau_{temp}(\eta_d) = 10 * \eta_d$, the highest topic proportion of the above example becomes 0.99. This makes the produced topics more clustered. Also, we make κ flexible. The KL divergence of vMF distribution makes the distribution more concentrated while not influence the direction of latent distribution.

B RELATED WORK AND CHALLENGES

In this section, we touch on key concepts utilized in S2vNTM and their limitations.

B.1 WEAKLY-SUPERVISED TEXT CLASSIFICATION

Weakly supervised text classification methods aim to predict labels of texts using limited or noisy labels. Given class names, Wang et al. (2021b) first estimates class representations by adding the most similar word to each class. It then obtains document representation by averaging contextualized word representations. Finally, it picks the most confident cdocuments from each cluster to train a text classifier. Yu et al. (2021) improves weakly text classification on existed LM using contrastive regularization and confidence based reweighting. Meng et al. (2020b) associates semantically related words with the label names. It then finds category-indicative words and trains the model to predict their implied categories. Finally, it generalizes the model via self-training. However, all these methods are time consuming to train and fine-tune which make it hard to be interactive. It is also hard to explain the reason behind certain classification.

B.2 TOPIC MODELING

Latent Dirichlet Allocation (LDA) Blei et al. (2003) is the most fundamental topic modeling approach based on Bayesian inference on Markov chain Monte Carlo (MCMC) and variational inference; however, it is hard to be expressive or capture large vocabularies. It is time consuming to train the model. It also has the tendency to identify obvious and superficial aspects of a corpus Jagarlamudi

et al. (2012) Neural topic model Miao et al. (2018)(NTM) leverages an autoencoder Kingma et al. (2014) framework to approximate intractable distributions over latent variables which makes the training faster. To increase semantic relationship with topics, Embedded topic model (ETM) Dieng et al. (2020) uses it during the decoder/reconstruction process to make topic more coherent and reduces the influence of stop words. However, the generated topics are not well clustered. Besides, using pre-trained embeddings cannot help the model identify domain specific topics. For example, topics related to Covid cannot be identified easily using pre-trained Glove embeddings Pennington et al. (2014) since Covid is not in the embeddings. To improve clusterabilityGuu et al. (2018), NSTM Zhao et al. (2020a) uses optimal transport to replace KL divergence to improve clusterability. It learns the topic distribution of a document by directly minimizing its optimal transport distance to the document's word distributions. Importantly, the cost matrix of the optimal transport distance models the weights between topics and words, which is constructed by the distances between topics and words in an embedding space. Due to the instability of latent distribution, it makes it difficult to integrate external knowledge into these models. Existed semi-supervised NTM methods either are not stable Wang et al. (2021a); Harandizadeh et al. (2022) or need specific twists Gemp et al. (2019).

B.3 SEMI-SUPERVISED TOPIC MODELING

Semi-supervised Topic Modeling methods take few keyword sets as input and create topics based on these keyword sets. Correlation Explanation (CorEx) Gallagher et al. (2018) is an information theoretic approach to learn latent topics over documents. It searches for topics that are "maximally informative" about a set of documents. To be specific, the topic is defined as group of words and trained to minimize total correlation or multivariate mutual information of documents conditioned on topics. CorEx also accepts keywords by add a regularization term for maximizing total correlation between that group of keywords to a given topics. There is a trade-off between total correlation between documents conditioned on topics and total correlation between keywords to topics. GuidedLDA Jagarlamudi et al. (2012) incorporates keywords by combining two techniques. The first one defines topics as a mixture of a seed topic and a regular topic where topic distribution only generates words from a group of keywords. The second one associates each group of keywords with a Multinomial distribution over the regular topics. It transfers the keywords information from words into the documents that contain them by first sampling a seed set and then using its group-topic distribution as prior to draw the document-topic distribution. However, both methods fail to capture the semantic relationship between words. This means that when the provided keywords are less frequent in the corpus, the model's performance drop sharply.

B.4 NEGATIVE SAMPLING

Negative Sampling Mikolov et al. (2013) is proposed as a simplified version of noise contrastive estimation Mnih and Kavukcuoglu (2013). It is an efficient way to compute the partition function of an non-normalized distribution to accelerate the training of word2vec. Mikolov et al. (2013) sets the negative sampling distribution proportional to the $\frac{3}{4}$ power of degree by tuning the parameters. Uncertainty based negative sampling Li et al. (2013) selects the most informative negative pairs and iteratively updates how informative those pairs are. Some methods Bucher et al. (2016) also account for the intra-class correlation. Negative sampling is used in topic modeling methods since it can leverage the word-context semantic relationships Shi et al. (2018) or generate more diverse topics Wu et al. (2020). Both methods are applied in fully unsupervised scenario. In general, it needs to compute the similarity between the topic and all vocabularies. This step adds additional time and space complexity to the model which makes related methods less practical.

B.5 VON MISES-FISHER BASED METHODS

In low dimensions, the gaussian density presents a concentrated probability mass around the origin. This is problematic when the data is partitioned into multiple clusters. An ideal prior should be non informative and uniform over the parameter space. Thus, the von Mises-Fisher(vMF) is used in VAE. vMF is a distribution on the (M-1)-dimensional sphere in R^M, parameterized by $\mu \in R^M$ where $||\mu||= 1$ and a concentration parameter $\kappa \in R_{\geq 0}$. The probability density function of the vMF distribution for $t \in R^D$ is defined as:

$$q(t|\mu, \kappa) = C_M(\kappa) \exp(\kappa \mu^T t)$$

$$C_M(\kappa) = \frac{\kappa^{\frac{M}{2}-1}}{(2\pi)^{\frac{M}{2}} I_{\frac{M}{2}-1}(\kappa)} + \log 2$$

where I_v denotes the modified Bessel function of the first kind at order v. The KL divergence with vMF(., 0) Davidson et al. (2018) is

$$KL(vMF(\mu,\kappa)|vMF(.,0)) = \kappa \frac{I_{\frac{M}{2}}(\kappa)}{I_{\frac{M}{2}-1}(\kappa)}$$
$$+(\frac{M}{2}-1)\log\kappa - \frac{M}{2}\log(2\pi) - \log I_{\frac{M}{2}-1}(\kappa)$$
$$+\frac{M}{2}\log\pi + \log 2 + \log\Gamma(\frac{M}{2})$$

vMF based VAE has better clusterability of data points especially in low dimensions Guu et al. (2018).

Xu and Durrett (2018) proposes using vMF(.,0) in place of Gaussian as $p(Z)$, avoiding entanglement in the center. They also approximate the posterior $q_\phi(Z|X) = vMF(Z;\mu,\kappa)$ where κ is fixed to avoid posterior collapse. The above approach does not work well for two reasons. First of all, fixing κ causes KL divergence to be constant which reduces the regularization effect and increases the variance of latent distribution. Another concern with vMF distribution is its limited expressability when its sample is translated into a probability vector. Due to the unit constraint, $softmax$ of any sample of vMF will not result in high probability on any topic even under strong direction μ. For example, when topic dimension M equals to 10, the highest topic proportion of a certain topic is 0.23.

C EXPERIMENTS

In this section, we report experimental results for S2vNTM and show that it performs significant better compared to two baselines.

Datasets: We use three datasets: DBLP Pan et al. (2016), AG News Zhang et al. (2016), R8 Lewis (1997). These datasets are all labeled. AG News has 4 classes and 30000 documents per class with an average of 45 words per document. We select AG News since it is a standard dataset for semi-supervised topic modeling evaluation. DBLP has 4 classes. Documents per class varies from 4763 to 20890. Average document length is 5.4. We select DBLP to see how our model performs when document is short and categories are unbalanced. R8 is a subset of the Reuters 21578 dataset, which consists of 7674 documents from 8 different reviews groups. We select R8 dataset to see how our model performs when the number of keywords set and topics are large. We use the same keywords as Meng et al. (2018) for our experiments for AG News. For others, we use 20 percent of corpus as the training set to get our keywords by tf-idf score for each classes. To form the vocabulary, we keep all words that appear more than 15 times depending on the size of the dataset. We remove documents that are less than 2 words. We also remove stop words, digits, time and symbols from vocabulary. We also include bigram and trigram that appear more than 15 times.

Settings: The hyperparameter setting used for all baseline models and vNTM are similar to Burkhardt and Kramer (2019). We use a fully-connected neural network with two hidden layers of [256, 64] unit and ReLU as the activation function followed by a dropout layer (rate = 0.5). We use Adam Kingma and Ba (2017) as optimizer with learning rate 0.002 and use batch size 256. We use Smith and Topin (2018) as scheduler and use learning rate 0.01 for maximally iterations equal to 50. We use 50 dimension embeddings Meng et al. (2019) trained on the dataset where we apply the model. We set the number of topics equal to the number of classes plus one. Our code is written in pytorch and all the models are trained on AWS using ml.p2.8xlarge (NVIDIA K80). We use 80 percent data as test set.

Baselines: We compare our methods with GuidedLDA Jagarlamudi et al. (2012) and CorEx Gallagher et al. (2018). CorEx are finetuned by anchor strength from 1 to 7 with step equal to 1 on the training set. GuidedLDA is finetuned using best seed confidence from 0 to 1 with step equal to 0.05 on the training set.

Metrics: To evaluate the classification performance of these models, we report **Accuracy**, **Macro F1** and **AUC**. We omit micro f1 since most of classes in these datasets are balanced and micro f1 is very similar to accuracy.

In addition, we want keywords in each topic to be diverse. This can help users to explore and identify new topics. We define **Topic Diversity** to be the percentage of unique words in the top 25 words of all topics Dieng et al. (2020). Diversity close to 0 indicates redundant topics while diversity close to 1 indicates more varied topics.

D QUALITATIVE STUDY

GuidedLDA	CorEx
iraq, kill, reuters, president, minister	**government**, **war**, **military**, iraq, kill
reuters, **stock**, oil, price, profit	**stock**, **market**, **industry**, price, oil
microsoft, company, **software**, service, internet	**software**, **computer**, microsoft, internet, service
win, game, team, season, lead	**footable**, **basketball**, game, win, season
space, reuters, win, quot, world	court, executive, chief, commission, union
quot, year, company, million, plan	inc, corp, people, bush, million

E ABLATION STUDIES

In this section we analyze and investigate the effect of various techniques and hyperparameters on S2vNTM. We use AG News as the dataset since it is standard and has balanced classes. We run each experiment 10 times and report the barplot. Specifically, for parameters we analyze: 1. Number of topics 2. Different keyword sets 3. Temperature function 4. γ (L_{NS} multiplier) for topic modeling. For techniques, we analyze 1. Batch normalization and dropout and 2. Learnable distribution temperature. The first two are reported here and the rest are discussed in the Appendix.

E.1 EFFECT OF NUMBER OF TOPICS

In this section, we analyze the effect of increasing in the number of topics from 5 to 13 shown in Figure 9. We see that the accuracy drops as the number of topics increases. This is because with increased number of topics, there is an increase in probability of adding additional topics that are similar to anchored topics and so the model gets confused while assigning words to topics. This could be either because of lack of topics in dataset or the latent space becoming very crowded i.e. space between vectors is less so it becomes difficult for models to discriminate between topics. Also, it seems that vMF based variational autoencoder performs less well in high dimension data. This could be addressed with an increase in distribution temperature discussed in Appendix E.3.

E.2 EFFECT OF DIFFERENT KEYWORDS SETS:

For traditional method such as CorEx and GuidedLDA, their performance drops when less frequent words are selected as keywords. To check the performance of our method on the less frequent keywords, we select top 30 keywords based on tf-idf score. Then we sort them based on frequency. The keywords set is shown in Figure 7. We then check its performance. See Figure . As you can see, the classification metric does not change in most of cases. This means our method is robust to keywords change. This is because we leverage semantic information using word embedding trained on the dataset. And negative sampling helps our model identify words semantically related keywords. This helps our method leverage more information beyond bag of word representations. This experiment shows that our method can perform well when the input keywords is less common in corpus. This section continues the ablation studies reportede in the main paper.

E.3 EFFECT OF INCREASE IN TEMPERATURE

Temperature is the constant multiplied to the sampled distribution from vMF before softmax. Because of this trick, the topic distribution become more representative and therefore it becomes easier for the model to identify those topics or clusters. Now if the temperature is too high, the distance between topic clusters will increase and the model will have difficulty in adjusting clusters based on keywords set since keywords may be far from each other in latent space. This can be observed in Figure 3 when the temperature is increased from 20 and above we see a decrease in accuracy. On the contrary, if the temperature is too small, the latent distribution is less representative which makes the boundary between clusters vague. This again decreases the performance of the model. This can be observed in

Figure 3 from values 5 to 15. The benefits of lower temperature is to make topic more diverse as you can see in second Figure 3. The optimal value for temperature given number of topics (=5 for this experiment) and the dataset is anywhere between 15-20 where model can easily identify topics. The temperature within this range has high clusterability and expressibility.

E.4 Effect of Gamma

We explore the effect of various values of gamma shown in Figure 4. With the increase in gamma, we observe a minimal increase in standard deviation and mean in accuracy and macro. Higher gamma makes L_{NS} stronger which makes the model less stable. We observe a strong increase in diversity score. This is because higher gamma score can push unrelated keywords further away. This makes each topic more coherent and different from other topics. So, at higher gamma, there is significant increase in diversity with negligible sacrifice in accuracy. This indicates stability of the method.

E.5 Effect of batch normalization and dropout

We explore various combinations of batch normalization (bn) and dropout which are shown in Figure 6. Independently, S2vNTM_Drop0.5 i.e. S2vNTM with dropout 0.5 has high standard deviation. Reducing dropout to 0.2 S2vNTM_Drop0.2 or adding bn S2vNTM_bn_Drop0.5 have very similar effect of reduced variance for accuracy and Macro F1 but S2vNTM_bn_Drop0.5 has higher aucroc and diversity and less variance. In general, adding bn with dropout stabilizes the model performance which was expected.

E.6 Effect of learnable distribution temperature

In Appendix E.3 we discuss effect of increasing distribution temperature. In this study, we make it a learnable parameter and implement it in two ways. The first way is setting temperature variable as one parameter that can be learned (1-p model). All topics share the same parameter. The second way is setting temperature variable as a vector with dimension equal to the number of topics (n-p model). This means each topic has its own temperature. The initialization value for both the vectors is 10.

After training, the 1-p model has value 4.99 and n-p model has values [-0.45,4.88,5.91,3.47,4.19] (values are rounded to 2 decimals). The accuracy for 1-p model is 78.9 and n-p model is 80.5. This means that our method can further improve with learnable temperature.

In Appendix E.3 we found that distribution temperature values between 15 to 20 gave highest accuracy (81) but on the contrary the learned values in 1-p is 4.99 (accuracy 78.9). This means that our loss function is not fully aligned with accuracy metric. This is due to the fact that we optimize reconstruction loss as well as KL divergence during the training procedure. This makes our objective less aligned with cross entropy loss.

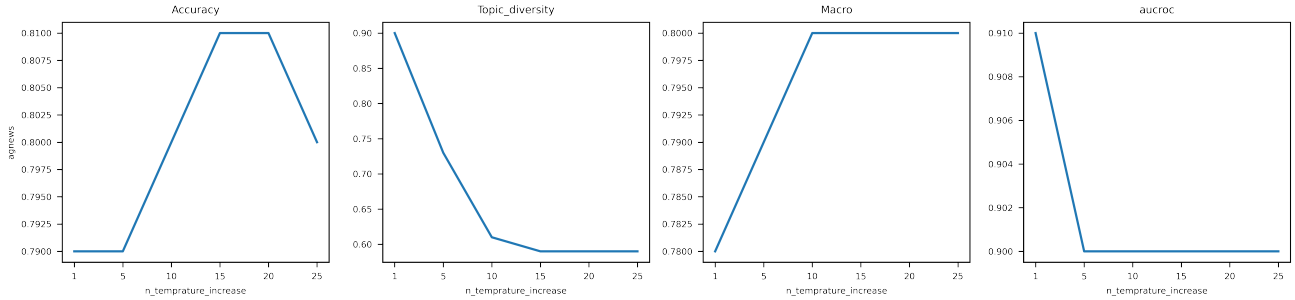

Figure 3: Impact of increasing temperature of vMF VS various metrics on AG News for S2vNTM model.

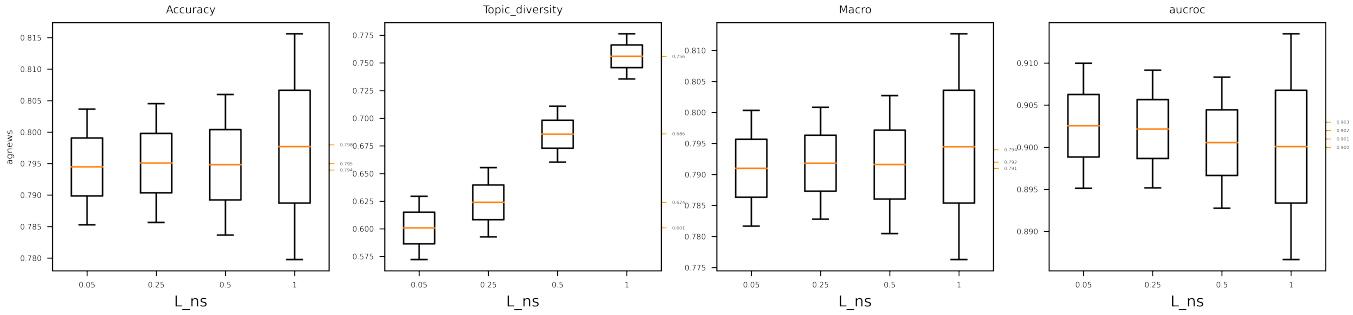

Figure 4: Effect of gamma. (y-axis on right shows mean.)

GuidedLDA	CorEx
iraq, kill, reuters, president, minister	**government**, **war**, **military**, iraq, kill
reuters, **stock**, oil, price, profit	**stock**, **market**, **industry**, price, oil
microsoft, company, **software**, service, internet	**software**, **computer**, microsoft, internet, service
win, game, team, season, lead	**footable**, **basketball**, game, win, season
space, reuters, win, quot, world	court, executive, chief, commission, union
quot, year, company, million, plan	inc, corp, people, bush, million

Table 3: Compare top 5 keywords from each topics for GuidedLDA and CorEx using Dataset AG News. The keywords that are given is [government,military,war], [stock,market,industry], [computer,telescope,software], [basketball,football,athlete]. CorEx provided diverse keywords but they are not similar in meaning which can make users confused.

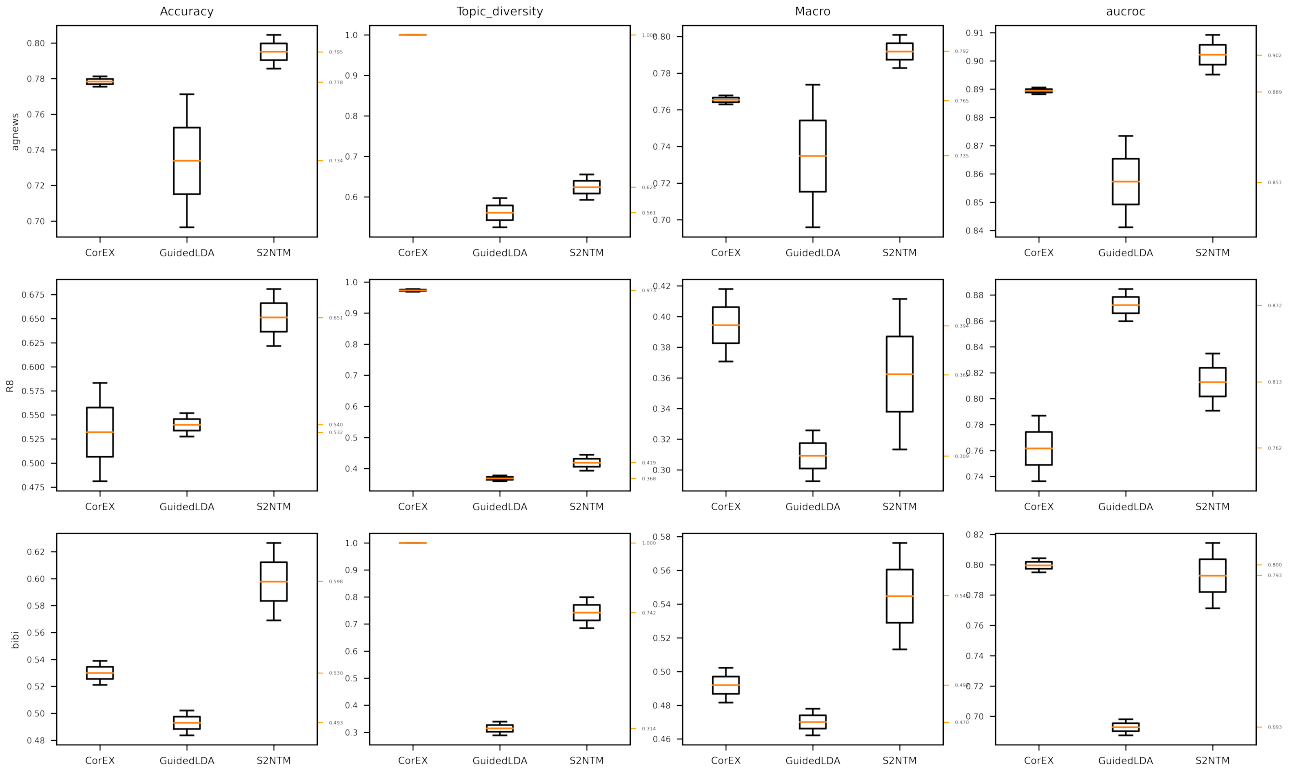

Figure 5: Results for Accuracy, Topic Diversity, Macro F1 and aucroc for GuidedLDA, CoreEx and S2vNTM. (Right y-axis shows mean).

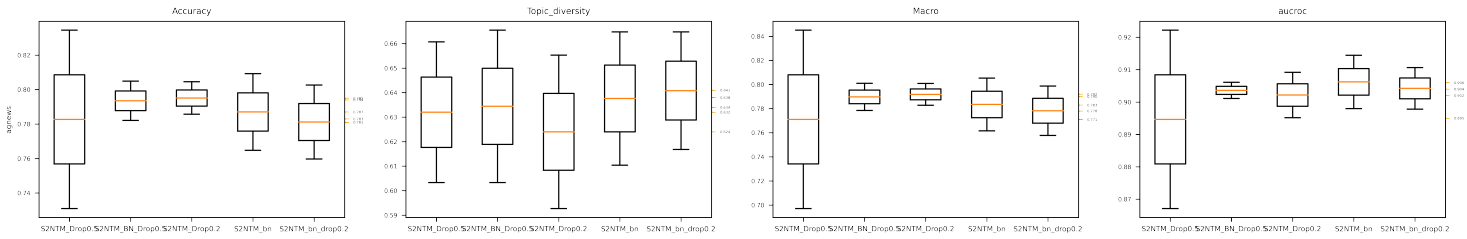

Figure 6: Effect of Batch Normalization and Dropout. (y-axis on right shows mean.)

```
IND 0: iraq kill president , game team season , stock price company , microsoft software internet ,
IND 1: palestinian leader official , victory lead year , percent year high , phone technology search ,
IND 2: people government bomb , sport champion sunday , report york quarter , quot security online ,
IND 3: troop police hostage , quot beat saturday , market million bank , user world window ,
IND 4: country security year , yankee title round , large friday plan , market million wireless ,
IND 5: darfur israel city , match york state , target week airline , version open offer ,
IND 6: soldier sunday wednesday , yesterday race take , yesterday fall executive , report network week ,
```

Figure 7: New seed topic labels use. Figure 8 reports classifcation metrics for these seeds.

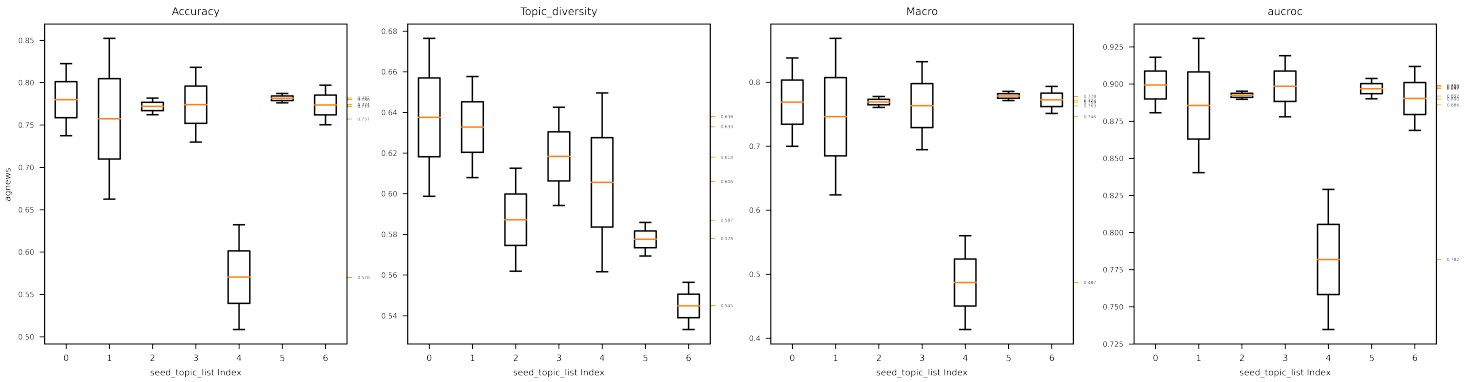

Figure 8: Effect of different keywords sets listed in Figure 7 on classification and diversity metrics. Y-axis on right shows mean.

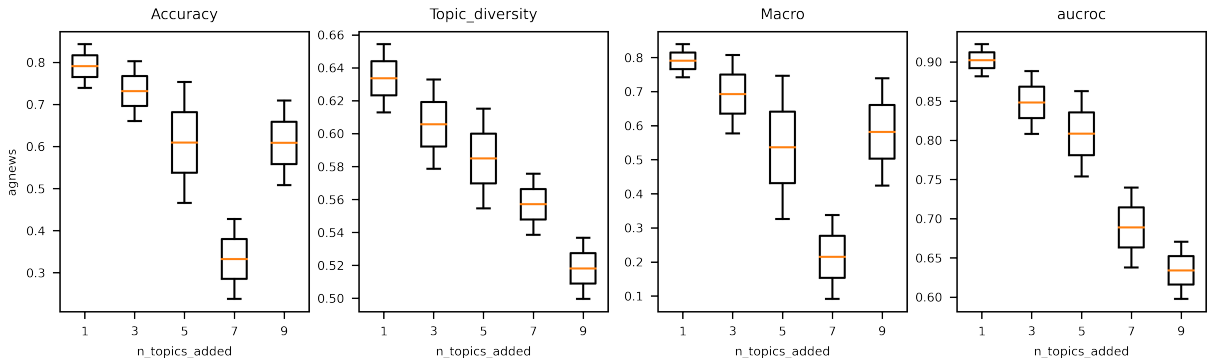

Figure 9: Number of topics added for diversity VS various metrics for S2vNTM model. The topics is increasing from 5 to 13 which is increasing 1 to 9 from 4 basic topics