vONTSS: vMF based semi-supervised neural topic modeling with optimal transport

Weijie Xu, Xiaoyu Jiang, Srinivasan H. Sengamedu, Francis Iannacci, Jinjin Zhao
Amazon
weijiexu@amazon.com

Abstract

Recently, Neural Topic Models (NTM), inspired by variational autoencoders, have attracted a lot of research interest; however, these methods have limited applications in the real world due to the challenge of incorporating human knowledge. This work presents a semi-supervised neural topic modeling method, vONTSS, which uses von Mises-Fisher (vMF) based variational autoencoders and optimal transport. When a few keywords per topic are provided, vONTSS in the semi-supervised setting generates potential topics and optimizes topic-keyword quality and topic classification. Experiments show that vONTSS outperforms existing semi-supervised topic modeling methods in classification accuracy and diversity. vONTSS also supports unsupervised topic modeling. Quantitative and qualitative experiments show that vONTSS in the unsupervised setting outperforms recent NTMs on multiple aspects: vONTSS discovers highly clustered and coherent topics on benchmark datasets. It is also much faster than the state-of-the-art weakly supervised text classification method while achieving similar classification performance. We further prove the equivalence of optimal transport loss and cross-entropy loss at the global minimum.

1 Introduction

Topic modeling methods such as (Blei et al., 2003) is an unsupervised approach for discovering latent structure in documents and achieving great performance (Blei et al., 2009). Topic modeling methods take a list of documents as input. It generates the defined number of topics. It can further produce keywords and related documents for each topic. In recent years, topic modeling methods have been widely used in many fields such as finance (Aziz et al.), healthcare (Bhattacharya et al., 2017), education (Zhao et al., 2020b), marketing (Reisenbichler, 2019) and social science (Roberts et al., 2013). With the development of Variational Autoencoder (VAE) (Kingma and Welling, 2013), Neural Topic Model (Miao et al., 2018; Dieng et al., 2020) has attracted attention as it enjoys better flexibility and scalability. However, recent research (Hoyle et al., 2021) shows that the topics generated by these methods are not aligned with human perceptions.

To incorporate users' domain knowledge into the model, semi-supervised topic modeling methods become an active area of research (Mao et al., 2012; Jagarlamudi et al., 2012; Gallagher et al., 2018) and applications (Choi et al., 2017; Cao et al., 2019; Kim et al., 2013). Semi-supervised topic modeling methods take a few keywords as input and generate topics based on these keywords. Poeple use semi-supervised topic modeling methods because they want each topic include certain keywords and incorporate their domain expertise in their generated topics. Traditional semi-supervised topic modeling methods fail to utilize semantic information of the corpus, causing low classification accuracy and high variance (Chiu et al., 2022a).

To solve these problems, we propose a von Mises-Fisher(vMF) based semi-supervised neural topic modeling method using optimal transport (vONTSS). We use the encoder-decoder framework for our model. The encoder uses modified vMF priors for latent distributions. The decoder uses a word-topic similarity matrix based on spherical embeddings. We use optimal transport to extend it to a semi-supervised version. vONTSS has the following enhancements:

1. We introduce the notion of temperature and make the spread of vMF distribution (κ) learnable, which leads to strong coherence and cluster-inducing properties.

2. vONT (In the rest of the paper, we use vONT to refer to the unsupervised topic model and vONTSS to semi-supervised version.) achieves the best coherence and clusterability compared to the state-of-the-art approaches on benchmark datasets.

3. We perform the human evaluation of the re-

Findings of the Association for Computational Linguistics: ACL 2023, pages 4433–4457
July 9-14, 2023 ©2023 Association for Computational Linguistics

Table 1: Description of the notations used in this work

Notion	Description	Dimension
M	topic dimension	
V	vocabulary dimension	
Z	topic proportions	R^M
X	bow of a document	R^V
x	represent word	R
θ	decoder	
ϕ	encoder	
$L_{\theta,\phi}(X)$	loss function for NTM	
e_V	word embedding matrix	$R^{V \times D}$
e_T	topic embedding matrix	$R^{M \times D}$
E	topic-word matrix	$R^{M \times V}$
μ	vmf direction parameter	R^M
κ	vmf concentration parameter	R
P	weight matrix for OT	$R^{\lvert T\rvert \times \lvert S\rvert}$
C	cost matrix for OT	$R^{\lvert T\rvert \times \lvert S\rvert}$
η	sample from vMF	R^M
s	keywords set	
S	group of keywords set	
t	topic	
T	group of topics	
(s,t)	topic keywords set pair	
L_{OT}	optimal transport loss	
λ	entropy penalty weights	
L_{ce}	cross-entropy loss	
$L_{X,T}$	loss function for vONTSS	
α, δ	parameters $L_{X,T}$	

sults for intrusion and rating tasks, and vONT outperforms other techniques.

4. Use of optimal transport to extend the stability of the model in the semi-supervised setting. The semi-supervised version is fast to train and achieves good alignment between keywords sets and topics. We also prove its theoretical properties.

5. In the semi-supervised scenario, we demonstrate the vONTSS achieves the best classification accuracy and lowest variance compared to other semi-supervised topic modeling methods.

6. We also show that vONTSS achieves similar performance as the state-of-the-art weakly text classification method while being much more efficient.

2 Related Methods and Challenges

NTM Variational Autoencoders (VAE) (Kingma and Welling, 2013) enable efficient variational inference. NTM (Miao et al., 2015) uses $Z \in R^M$ as topic proportions over M topics and $X \in R^V$ to represent word count for the dataset with V unique words. NTM assumes that for any document, Z is generated from a document satisfying the prior distribution $p(Z)$ and X is generated by the conditional distribution $p_\theta(X|Z)$ where θ denotes a decoder. Ideally, we want to optimize the marginal likelihood $p_\theta(X) = \int p(Z)p_\theta(X|Z)dZ$. Due to the intractability of integration, NTM introduces $q_\phi(Z|X)$, a variational approximation to the posterior $p(Z|X)$. The loss function of NTM is:

$$\begin{aligned} L_{\theta,\phi} = (&-E_{q_\phi(Z|X)}[\log p_\theta(X|Z)] \\ &+ KL[q_\phi(Z|X)||p(Z)]) \end{aligned} \tag{1}$$

NTM usually utilizes a neural network with softmax to approximate $p_\theta(X|Z) := softmax(Wz)$ (Srivastava and Sutton, 2017). NTM selects Gaussian (Miao et al., 2016), Gamma (Zhang et al., 2020) and Dirichlet distribution (Burkhardt and Kramer, 2019) to approximate $p(Z)$. The second term Kullback-Leibler (KL) divergence regularizes $q_\phi(Z|X)$ to be close to $p(Z)$. *NTM has several problems in practice.* Firstly, it does not capture the semantic relationship between words. Secondly, the generated topics are not aligned with human interpretations. (Hoyle et al., 2021). Thirdly, using Gaussian prior may risk gravitating latent space toward the center and produce tangled representations among classes of documents. This is due to the fact that gaussian density presents a concentrated mass around the origin in low dimensional settings (Dümbgen and Del Conte-Zerial, 2013) and resembles a uniform distribution in high dimensional settings.

Extending NTM to semi-supervised version is also challenging. $L_{\theta,\phi}$ is not always aligned with classification-related loss such as cross-entropy loss as identified by existing research (Chiu et al., 2022b). To be specific, cross-entropy makes keywords sets align with assigned topics, while reconstruction loss($-E_{q_\phi(Z|X)}[\log p_\theta(X|Z)]$) makes latent space as representative as possible. Thus, existing semi-supervised NTM methods either are not stable (Wang et al., 2021a; Harandizadeh et al., 2022) or need certain adaptions (Gemp et al., 2019).

Embedding Topic Model (ETM) Pre-trained word embeddings such as Glove (Pennington et al., 2014a) and word2vec (Mikolov et al., 2013) have the ability to capture semantic information, which is missing from basic bag-of-word (BoW) representations. They can serve as additional information to guide topic discovery. Dieng (Dieng et al., 2020) proposes ETM to use a vocabulary embedding matrix $e_V \in R^{V \times D}$ where D represents the dimension of word embeddings. The decoder ϕ learns a topic embedding matrix $e_T \in R^{M \times D}$. We denote topic to word distribution $softmax(e_T e_V^T)$ as E

$$p_\theta(X|Z) := Z \times E \tag{2}$$

However since there exists some common words that are related to many other words, these common words' embeddings may be highly correlated with few topics' embeddings. Thus, *ETM does not produce diverse topics* (Zhao et al., 2020a). Besides, using pre-trained embeddings cannot help the model identify domain-specific topics. For example, topics related to COVID-19 are more likely to be expressed by a few topics instead of one single topic using pre-trained Glove embeddings (Pennington et al., 2014b) since COVID-19 is not in the embeddings.

von Mises-Fisher 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 $z \in R^D$ is defined as:

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

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

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). However, vMF distribution has limited expressibility 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. *Most of vMF-based topic modeling methods are not VAE based and very slow to train as summarized in Appendix M.*

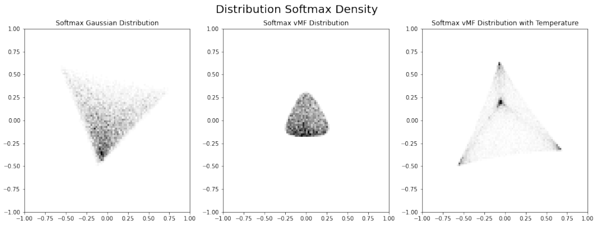


Figure 1: 2-D PCA projection of empirical CDF $softmax(\eta)$ where from left to right $\eta \sim \mathcal{N}(0, I)$, $\eta \sim vMF(.,0)$ and $\eta \sim 10 * vMF(.,0))$ respectively. From the heatmap, we observe a white hole in the middle, which denotes the unreachable probability vector from each distribution. Gaussian is mean-centered, while basic vMF tends to cluster around a small rounded triangular area due to its unity constraint. vMF with radius equals to 10 is even more expressive than Gaussian while still retaining edge weights, inducing separability among different topics.

3 Proposed Methods

The architecture of vONTSS is shown in Figure 3. At a high level, our encoder network ϕ transforms the BoW representation of the document X_d into a latent vector generated by vmf distribution and generates a sample η_d. We then apply a temperature function τ and softmax on this sample to get a probabilistic topic distribution z_d. Lastly, our decoder uses a modified topic-word matrix E to reconstruct X_d's BoW representation. To extend into semi-supervised setting, we leverage optimal transport to match keywords' set with topics. The encoder network ϕ and generative model parameter θ are learned jointly during the training process.

To overcome entangled topic latent space introduced by Gaussian distribution and limited expressibility of vMF distribution, we make two improvements: 1. Introduce a temperature function $\tau(\eta_i)$ prior to $softmax()$ to modify the radius of vMF distribution. 2. Set κ to a learnable parameter to flexibly infer the confidence of particular topics during training.

Encoder Network Temperature Function To alleviate concerns regarding expressibility while inducing separability among topics, we modify the radius of vMF distribution. We use a temperature function to represent the radius. As shown in Figure 1, unmodified vMF distribution has limited expressiveness. For instance, Gaussian posteriors can express a topic probability vector of [0.98, 0.01, 0.005, 0.0003, 0.0002], while vMF can't due to the unity constraint. In practice, if we change the ra-

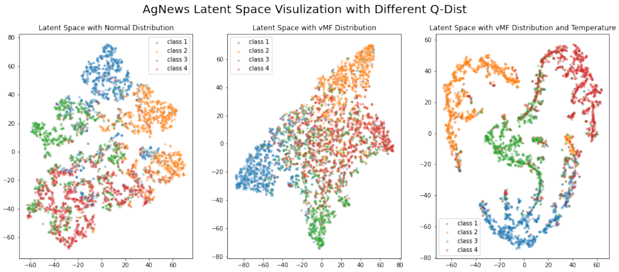


Figure 2: 2-D TSNE projection of randomly sampled η from latent spaces under different posterior-distributions. From left to right are Gaussian, vMF with fixed k, vONT Each color represents a different topic. All encoders are trained on AgNews dataset with the same network structure.

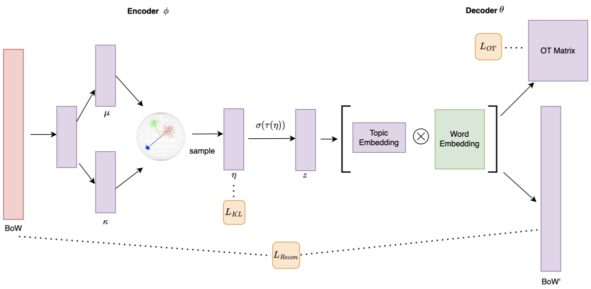


Figure 3: Architecture of the model. Purple represents the part of the network that can be trained. L_{recon} represents reconstruction loss. L_{KL} represents KL divergence. L_{OT} represents optimal transport loss

dius to 10, the network can learn more polarized topics distribution as shown in the right plot in Figure 1. The influence of different radii is analyzed in Appendix N. Given equally powerful learning networks of distributions' parameters, vMF with different radii learns richer and more nuanced structures in their latent representations than a Gaussian counterpart (Appendix I).

Learnable κ To further improve the clusterability, we convert κ from a fixed value to a learnable parameter. The KL divergence of vMF distribution makes the distribution more concentrated while not influencing the direction of latent distribution. This makes the result more clustered. For Gaussian distribution, KL divergence penalizes the polarization of latent distribution (Appendix K). This makes the Gaussian distribution less clustered. To illustrate this, we randomly sampled encoded documents' latent distributions from AgNews Dataset (Zhang et al., 2016) after training with both latent distributions, as shown in Figure 2. For the Gaussian distribution, we see that documents belonging to different topics are entangled around the center, causing the inseparability of topics during both the training and inference stage. vMF distribution, on the hand, repels four document classes into different quadrants, presents more structures when compared to Gaussian distribution, and creates better separable clusters. Detailed ablation study can be found in Appendix O

Decoder Network Our decoder follows ETM's construction and uses the embedding e_V and e_T to generate a topic-word matrix E. One distinction between our decoder and ETM's decoder is that we generate the word embeddings by training a spherical embedding on the dataset. Spherical embeddings perform well in word similarity evaluation and document clustering (Meng et al., 2019), which further improves the clusterability of the topic modeling methods.

We also keep word embeddings fixed during the topic modeling training process for two reasons. Firstly, keeping word embeddings fixed can alleviate sparsity issues (Zhao et al., 2018). Additionally, vMF based VAE tends to be less expressive in high dimensions due to limited variance freedom (Davidson et al., 2018). Keeping the embedding fixed can make topics more separable in higher-dimension settings and improve topic diversity.

Loss Function for vONTSS In semi-supervised settings, the user specifies sets of keywords S associated with topics T. Let (s, t) represent a keyword set and a topic pair, where each keyword $x \in s$ is labeled by topic t. Instead of training a separate neural network for the semi-supervised extension of NTM, we use the topic-word matrix (decoder θ) to represent the probability of a word x given topic t.

M1 + M2 is a semi-supervised model used in VAE. We adapt the M1 + M2 model framework (Kingma et al., 2014). Under the assumption that $p_\theta(x, t, z) = p_\theta(x|z)p_\theta(t|x)p(z)$, our loss function can be approximated as

$$L(X, T) = L_{\theta,\phi}(X) - \alpha H[q_\phi(X|T)] + \delta L_{ce} \quad (3)$$

$$L_{ce} = - \sum_{(s,t)\in(S,T)} E_{x\in s} \log q_\theta(x|t) \quad (4)$$

For topic i and word j, we let $q_\theta(x_j|t_i) = E_{i,j}$ where E is the topic-word matrix. $H[q_\phi(X|T)]$ is entropy of $q_\theta(X|T)$. We can consider it as a regularization term.

Optimizing the current model is hard because we have 3 objectives to minimize(cross-entropy, KL Divergence, and reconstruction loss) and they are not aligned with each other. To validate our point, we find out that if we make radius parameters learnable, the classification metric performs worse even if it decreases the reconstruction loss(Appendix D). If we apply cross-entropy at the beginning, topic embeddings get stuck into the center of selected keywords' embeddings, which makes the model overfitting. If we first train an unsupervised vONT, we need to find a way to match keywords and trained topics. If we match them based on their cosine similarity, different keywords may match to the same topics. This makes performance unstable. To deal with these challenges, we decide to use a two-stage training process and do not specify labeled keywords to topics at the beginning. vONTSS first optimizes $L_{\theta,\phi}(X) - \alpha H[q_\phi(X|T)]$ till convergence, then jointly optimizes $L(X,T)$ for few epochs. This makes our method easier to optimize, less time-consuming, and suitable for interactive topic modeling (Hu et al., 2014). To optimize L_{ce} after stage 1, we need to pair topics and keyword sets. Existing methods such as Gumbel softmax prior (Jang et al., 2016) often lead to instability, while naive matching by $q_\phi(x|t)$ may give us redundant topics.

Optimal Transport for vONTSS Optimal Transport (OT) distances (Chen et al., 2019; Torres et al., 2021) have been widely used for comparing the distribution of probabilities. Specifically, let $U(r,c)$ be the set of positive $m \times n$ matrices for which the rows sum to r and the sum of the column to c: $U(r,c) = \{P \in R_{>0}^{m\times n} | P1_t = r, P^T 1_s = c\}$. For each position t, s in the matrix, it comes with a cost $C_{t,s}$. Our goal is to solve $d_C(r,c) = \min_{P\in U(r,c)} \sum_{t,s} P_{t,s}C_{t,s}$. To make distribution homogeneous (Cuturi, 2013), we let

$$d_C^\lambda(r,c) = \min_{P\in U(r,c)} \sum_{t,s} P_{t,s}C_{t,s} - \frac{1}{\lambda}h(P) \quad (5)$$

$$h(P) = -\sum_{t,s} P_{t,s}\log P_{t,s} \quad (6)$$

OT has achieved good robustness and semantic invariance in NLP related tasks (Chen et al., 2019). Optimal transport has been used in topic modeling to replace KL divergence (Zhao et al., 2020a; Huynh et al., 2020; Wang et al., 2022) or create topic embeddings (Xu et al., 2018) as discussed in Appendix M. It has not been used for extending topic modeling to semi-supervised cases.

To better match topic and keywords set, we approximate L_{ce} using optimal transport. We choose sinkhorn distance since it has an entropy term, which makes our trained topics more coherent and stable. Our goal is to design the loss function that is aligned with derived cross-entropy loss at the global minimum. To be specific, the raw dimension of our cost matrix is equal to the dimension of topics and the column dimension of the cost matrix equals to the dimension of keywords group. We denote each entry in the M matrix in optimal transport as,

$$C_{t,s} = -E_{x\in s}\log(q_\theta(x|t)) \quad (7)$$

where t is the topic and x is the word in a keywords group s. The model uses sinkhorn distance and restricts the sum of each column and row of P to 1. We give the model an entropy penalty term to make sure each topic is only related to one group of keywords. Thus,

$$L_{OT} = \min_{P\in U(|T|,|S|)} \sum_{t,s} P_{t,s}C_{t,s} - \frac{1}{\lambda}h(P) \quad (8)$$

where λ controls the entropy penalty. The first term is similar to L_{ce} approximation, and the second term makes the result homogeneous. To lower the second term, each keyword should be highly correlated to one topic while not/negatively correlated with others. This further separates the topics and improves the topic diversity. We further show that $L_{OT} = L_{ce}$ when $L(X,T)$ is minimized.

Lemma 3.1 *When L(X, T) reaches the global minimal. For any (s,t), $(s',t') \in (S,T)$:*

$$E_{x\in s}\log q_\phi(x|t) + E_{x\in s'}\log q_\phi(x|t') \\ -(E_{x\in s'}\log q_\phi(x|t)) + E_{x\in s}\log q_\phi(x|t')) >= 0 \quad (9)$$

Theorem 3.2 *When L(X, T) reaches the global minimal,*

$$L_{OT} = L_{ce}$$

Appendix B contains the proof.

4 Experiment

Dataset Our experiments are conducted on four widely-used benchmark datasets for topic modeling and semi-supervised text classification with varied length: **DBLP** (Pan et al., 2016), **AgNews** (Zhang et al., 2016) and **20News** (Lang, 1995). All these

Table 2: Clusterability metrics for vONT. The number of topics is 20. The best and second-best scores of each dataset are highlighted in boldface and with an underline, respectively. Figure 5 in the appendix shows the variation of the metrics as a function of a number of topics. It is hard to get Km-Purity for ProdLDA. Since it does not perform well for Top-purity, we do not think it will perform well on Km-purity. Thus, we ignore that result.

	AgNews				20News				DBLP			
Method	**Km-Purity**	**Km-NMI**	**Top-Purity**	**Top-NMI**	**Km-Purity**	**Km-NMI**	**Top-Purity**	**Top-NMI**	**Km-Purity**	**Km-NMI**	**Top-Purity**	**Top-NMI**
ETM	0.570±0.023	0.160±0.021	0.556±0.036	0.126±0.024	0.689±0.028	0.332±0.027	0.731±0.037	0.369±0.051	0.217±0.023	0.268±0.022	0.208±0.034	0.251±0.033
GSM	0.716±0.016	0.313±0.008	0.719±0.014	0.359±0.021	0.709±0.013	0.366±0.008	**0.829±0.019**	**0.470±0.017**	0.272±0.016	0.333±0.013	0.304±0.028	0.358±0.018
NSTM	0.728±0.012	0.288±0.007	0.755±0.026	0.304±0.020	0.518±0.013	0.221±0.013	0.670±0.019	0.292±0.009	0.272±0.010	0.322±0.013	0.340±0.021	0.375±0.016
vNVDM	0.814±0.009	0.372±0.012	0.810±0.011	0.397±0.016	0.793±0.014	0.368±0.010	0.788±0.016	0.392±0.010	0.389±0.020	0.413±0.014	0.371±0.024	0.425±0.015
prodLDA			0.562±0.051	0.117±0.065			0.355±0.105	0.105±0.105			0.074±0.015	0.038±0.029
vONT	**0.822±0.025**	**0.404±0.025**	**0.810±0.030**	**0.423±0.036**	**0.819±0.016**	**0.411±0.012**	0.820±0.015	0.433±0.014	**0.456±0.031**	**0.504±0.020**	**0.443±0.033**	**0.519±0.018**

Table 3: Classification performance and Topic Diversity Result for vONTSS. Number of topics equal to 20. Figure 6 provides box plots for the metrics. CatE does not produce topics, so we do not have a diversity score. CarEx has diversity equal to 1 by design.

	AgNews			20News			DBLP		
Method	accuracy	micro F1	diversity	accuracy	micro F1	diversity	accuracy	micro F1	diversity
gONTSS	0.793 ± 0.017	0.788 ± 0.017	**0.79 ± 0.025**	0.385 ± 0.129	0.327 ± 0.121	0.859 ± 0.029	0.509 ± 0.040	0.457 ± 0.077	0.804 ± 0.043
vONTSS with CE	0.754 ± 0.009	0.717 ± 0.115	0.683 ± 0.009	0.468 ± 0.137	0.417 ± 0.13	0.764 ± 0.046	0.547 ± 0.009	0.493 ± 0.076	0.521 ± 0.064
vONTSS with all loss	0.741 ± 0.072	0.702 ± 0.125	0.652 ± 0.051	0.473 ± 0.095	0.416 ± 0.112	0.729 ± 0.050	0.590 ± 0.014	0.541 ± 0.048	0.716 ± 0.12
CarEx	0.778 ± 0.003	0.778 ± 0.003		0.44 ± 0.039	0.443 ± 0.062		0.530 ± 0.009	0.491 ± 0.010	
CatE	0.820 ± 0.001	**0.822 ± 0.001**		**0.596 ± 0.002**	**0.621 ± 0.002**		0.518 ± 0.001	0.536 ± 0.001	
GuidedLDA	0.733 ± 0.037	0.735 ±0.039	0.561 ± 0.036	0.554 ± 0.024	0.474 ± 0.026	0.584 ± 0.021	0.493 ± 0.009	0.47 ± 0.008	0.314 ± 0.025
Best Unsupervised	0.799 ± 0.014	0.797 ± 0.015	0.573 ±0.049	0.501 ± 0.047	0.429 ± 0.042	**0.952 ± 0.026**	0.517 ± 0.037	0.377 ± 0.031	0.781 ± 0.12
Guided BERTopic	0.666 ± 0.023	0.573 ±0.049	0.487 ± 0.041	0.591 ± 0.011	0.407 ± 0.016	0.617 ± 0.031	0.486 ± 0.112	0.301 ± 0.076	0.717 ± 0.07
vONTSS	**0.823 ± 0.003**	0.821 ± 0.017	0.71 ± 0.024	0.590 ± 0.014	0.554 ± 0.013	0.92 ± 0.027	**0.606 ± 0.032**	**0.576 ± 0.026**	**0.871 ± 0.036**

Table 4: Coherence metrics for vONT. Number of topics is 20. Figure 6 in the appendix shows details of the result.

	AgNews		R8		20News	
Method	C_v	NPMI	C_v	NPMI	C_v	NPMI
GSM	0.41 ± 0.01	0.03 ± 0.01	0.61 ± 0.05	0.05 ± 0.01	0.55 ± 0.04	0.07 ± 0.03
ETM	0.41 ± 0.04	0.02 ± 0.002	0.35 ± 0.02	-0.04 ± 0.01	0.51 ± 0.02	0.06 ± 0.01
vNVDM	0.44 ± 0.02	0.028 ± 0.008	**0.74** ± 0.02	0.08 ± 0.007	0.52 ± 0.01	0.03 ± 0.01
ProdLDA	0.32 ± 0.04	-0.22 ± 0.04	0.59 ± 0.06	0.01 ± 0.003	0.35 ± 0.02	-0.18 ± 0.03
NSTM	0.37 ± 0.02	-0.04 ± 0.02	0.61 ± 0.01	-0.08 ± 0.007	0.38 ± 0.01	0.06 ± 0.04
vONT	**0.49** ± 0.02	**0.054** ± 0.02	0.70 ± 0.03	**0.10** ± 0.03	**0.69** ± 0.03	**0.16** ± 0.02

datasets have ground truth labels. Average document length varies from 5.4 to 155. We preprocess all the datasets by cleaning and tokenizing texts. We remove stop words, words that appear more than 15 percent of all documents and words that appear less than 20 time. For semi-supervised experiments, we use the same labels in DBLP and AgNews. We sample 4 similar classes from 20News to see how our method performs in datasets with similar labels. For unsupervised settings, we keep the number of topics equal to the number of classes plus one. I keep the unit of the length to 10 for all experiments. For semi-supervised settings, we set the number of topics equal to the number of classes in semi-supervised cases, and we provide 3 keywords for each class. We use 20% as the training set to get our keywords with the top tfidf score for each class. We use 80% data as the test set. Additional details and provided keywords on the dataset are available in Appendix H

Settings In our experiment setting, we do not utilize any external information beyond the dataset itself. The embedding is trained on the test set. We do not compare methods that rely on transfer learning or language models such as (Bianchi et al., 2021; Yu et al., 2021; Wang et al., 2021b) because of reasons mentioned in appendix Q. The hyperparameter setting used for all baseline models and vONT is 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 the 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 spherical embeddings (Meng et al., 2019) trained on the dataset for

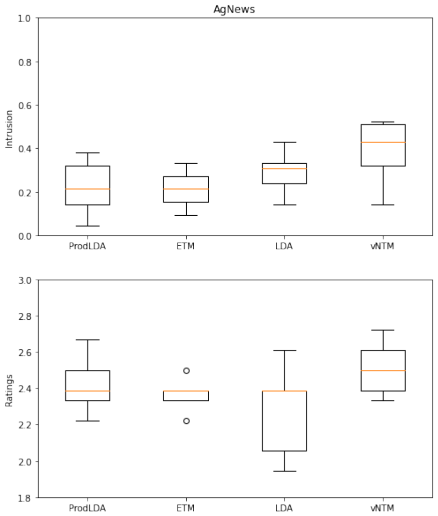


Figure 4: Comparison of intrusion and rating task performance on AgNews.

NVTM, ETM, GSM and NSTM. For vONT, we set the radius of vMF distribution equal to 10. We fix $\alpha = \delta = 1$ in $L(X, T)$. We keep $\lambda = 0.01$ in L_{OT}. Our code is written in PyTorch and all the models are trained on AWS using ml.p2.8xlarge (NVIDIA K80).[1]

4.1 Unsupervised vONT experiments

Evaluation Metrics We measure the topic coherence and clusterability of the model. Most of unsupervised topic coherence metrics are inconsistent with human judgment, based on a recent study (Hoyle et al., 2021). Thus, we have done a qualitative study where we ask crowdsource to perform rating and intrusion task on 4 models trained on AgNews. In rating task(Aletras and Stevenson, 2013; Newman et al., 2010; Mimno et al., 2011), raters see a topic and then give the topic a quality score on a three-point scale. The rating score is between 1 and 3. A rating score close to 3 means that users can see a topic from provided words. Chang(Chang et al., 2009) devise the intrusion task, where each topic is represented as its top words plus one intruder word which has a low probability belonging to that topic. Topic coherence is then judged by how well human annotators detect the intruder word. The intrusion score is between 0 and 1. An intrusion score close to 1 means that users can easily identify the intruder word. We use mechanical turk and sagemaker groundtruth to do the labeling work. To measure clusterability, we assign every document the topic with the highest probability as the clustering label and compute **Top-Purity** and Normalized Mutual Information(**Top-NMI**) as metrics(Nguyen et al., 2018) to evaluate alignment. Both of them range from 0 to 1. A higher score reflects better clustering performance. We further apply the KMeans algorithm to topic proportions z and use the clustered documents to report purity(**Km-Purity**) and NMI **Km-NMI** (Zhao et al., 2020a). We varied the number of topics from 10 to 50. We set the number of clusters to be the number of topics for KMeans algorithm. Models with higher clusterability are more likely to perform well in semi-supervised extension. Furthermore, we run all these metrics 10 times. We report mean and standard deviation. Detailed metric implementations are in Appendix G. We also analyze topic diversity in P and unsupervised topic coherence in F.

Baseline Methods We compare with the state-of-the-art NTM methods that do not rely on a large neural networks to train. These methods include: **GSM** (Miao et al., 2018), an NTM replaces the Dirichlet-Multinomial parameterization in LDA with Gaussian Softmax; **ProdLDA** (Srivastava and Sutton, 2017), an NTM model which keeps the Dirichlet Multinomial parameterization with a Laplace approximation; **ETM** (Dieng et al., 2020), an NTM model which incorporates word embedding to model topics; **vNVDM** (Xu and Durrett, 2018), a vMF based NTM as mentioned in section 2. **NSTM** (Zhao et al., 2020a), optimal transport based NTM, as mentioned in section 3. All baselines are implemented carefully with the guidance of their official code.[2] For qualitative study, we choose **ProdLDA**, **ETM** and **LDA** as a comparison to align with previous study (Hoyle et al., 2021).

Results i) In Table 2, vONT performs significantly better than other methods in all datasets for cluster quality metrics. This means vMF distribution induces good clusterability. ii) vONT has

[1] Details on codebases used for baselines and fine-tuning are provided in Appendix E

[2] Some methods we tested had lower TC scores compared to other benchmarks. This may be because we have less complicated layers, small epochs to train, and we keep fewer words. The ranking of these metrics is mostly in alignment with the paper that has a benchmark. We exclude methods that need to rely on large neural networks and a lot of finetune such as (Duan et al., 2021a,b). We also exclude methods similar to existing methods such as (Wang et al., 2022). We exclude methods that do not perform well in previous papers' experiments (Duan et al., 2021a) such as (Burkhardt and Kramer, 2019). We also exclude methods that are relevant but work on different use cases, such as short text.(Wu et al., 2020)

the lowest variance in clusterability-related metrics. (iii) In Appendix F, vONT outperforms other models in TC metrics C_v and NPMI. This means that our model is coherent. We believe the introduction of the temperature function helps our method perform better than the existed method in coherence. iv) In Appendix P, vONT performs well on diversity and has the lowest variance.

Human Evaluation To evaluate human interpretability, we use intrusion test and ratings test. Details of the experiment are provided in Appendix J. We select AgNews as our dataset, we generate 10 topics each from 4 models. In the word intrusion task, we sample five of the ten topic words plus one intruder randomly sampled from the dataset; for the rating task, we present the top ten words in order. Figure 4 summarizes the results.

vONT performs significantly better than ProdLDA, ETM, and LDA qualitatively. In *intrusion test*, vONT has the highest score 0.4. The second-best method is LDA, which has score 0.29. The two sample test between the two methods has the p-value equal to 0.014. In *rating test*, vONT has the highest score 2.51 while ProdLDA has the second-highest score 2.42. The two sample test between the two methods has a p-value equal to 0.036. *Based on this study, we conclude that humans find it easier to interpret topics produced by vONT.*

4.2 Semi-Supervised vONTSS experiments

Evaluation Metric diversity aims to measure how diverse the discovered topic is. **diversity** is defined as the percentage of unique words in the top 25 words from all topics.(Dieng et al., 2020) **diversity** close to 0 means redundant and TD close to 1 means varied topics. We measure the classification accuracy of the model. Thus, we measure **accuracy**. Similar to other semi-supervised paper(Meng et al., 2018a), we also measure **micro f1** score, since this metric gives more information in semi-supervised cases with unbalanced data. We do not include any coherence metric since we already have ground truth.

Baseline methods CatE (Meng et al., 2020) retrieves category representative terms according to both embedding similarity and distributional specificity. It uses WeSTClass(Meng et al., 2018b) for all other steps in weakly-supervised classification. If we do not consider methods with transfer learning or external knowledge, it achieves the best classification performance. **GuidedLDA** (Jagarlamudi et al., 2012): incorporates keywords by combining the topics as a mixture of a seed topic and associating each group of keywords with a multinomial distribution over the regular topics. Correlation Explanation **CorEx** (Gallagher et al., 2018) is an information theoretic approach to learning latent topics over documents by searching for topics that are "maximally informative" about a set of documents. We fine-tune on the training set and choose the best anchor strength parameters for our reporting. We also created semi-supervised ETM by using gaussian distribution and adding the same optimal transport loss as vONTSS. We call it **gONTSS**. We also train all objectives instead of using two-stage training and call it **vONTSS with all loss**. Instead of applying optimal transport, we apply cross entropy directly after stage 1 and match topics by keywords set with the highest similarity. We call this method **vONTSS with CE**. To get **Best Unsupervised** method, we train the unsupervised models(ETM, vNVDM, vONT, ProdLDA) and consider all potential matching between topics and seed words. We report the method with the highest accuracy for each dataset across all different matching. **Guided BERTopic** We evaluate the guided version of BERTopic (Grootendorst, 2022) method. They create seeded embeddings to find the most similar document. It then takes seed words and assigns them a multiplier larger than 1 to increase the IDF value. [3]

Results Table 3 shows that i) vONTSS outperforms all other semi-supervised topic modeling methods in classification accuracy and micro F1 score, especially for large datasets with lengthy texts such as AgNews. ii) vONTSS has a lower standard deviation compared to other models. This advantage makes our model more stable and practical in real-world applications. iii) To compare methods with/without optimal transport, methods with optimal transport vONTSS achieve much better accuracy, diversity, and lower variance compared to vONTSS with CE and vONTSS with all loss. This means optimal transport does increase the classification accuracy, stability, and diversity of generated topics. iv) In benchmark datasets,

[3] We do not find code for other neural-based semi-supervised topic modeling methods (Gemp et al., 2019; Wang et al., 2021a; Harandizadeh et al., 2022), but based on their experiments, the best one is (Harandizadeh et al., 2022) which is almost the same as vONTSS with CE which means it has similar variance and lower performance compare to vONTSS with CE

vONTSS is comparable to CatE in quality metrics. As can be seen in Table 5 in the appendix, vONTSS is 15 times faster than CatE. v) Unsupervised methods cannot produce comparable results even if we use the best topic seed word matching. This shows that semi-supervised topic modeling methods are necessary. vi) Guided Bertopic does not produce good results. It is also not very stable. In Guided Bertopic, the assigned multiplier is increased across all topics, which makes their probability less representative. vi) If we change vONTSS to gONTSS,

5 Conclusions

In this paper, we propose a new semi-supervised neural topic modeling method vONTSS, which leverages vMF, the temperature function, optimal transport, and VAEs. Its unsupervised version exceeds state-of-the-art in topic coherence through both unsupervised and human evaluations while inducing high clusterability among topics. We show that optimal transport loss is equivalent to cross-entropy loss under the optimal condition and induces one-to-one mapping between keywords sets and topics. vONTSS achieves competitive classification performance, maintains top topic diversity, trains fast, and possesses the least variance among diverse datasets.

References

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.

Saqib Aziz, Michael Dowling, Helmi Hammami, and Anke Piepenbrink. Machine learning in finance: A topic modeling approach. *European Financial Management*, n/a(n/a).

Kayhan Batmanghelich, Ardavan Saeedi, Karthik Narasimhan, and Sam Gershman. 2016. Nonparametric spherical topic modeling with word embeddings.

Moumita Bhattacharya, Claudine Jurkovitz, and Hagit Shatkay. 2017. Identifying patterns of associated-conditions through topic models of electronic medical records. *CoRR*, abs/1711.10960.

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, Thomas L. Griffiths, and Michael I. Jordan. 2009. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies.

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

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.

Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-Graber, and David Blei. 2009. Reading tea leaves: How humans interpret topic models. *Advances in neural information processing systems*, 22.

Liqun Chen, Yizhe Zhang, Ruiyi Zhang, Chenyang Tao, Zhe Gan, Haichao Zhang, Bai Li, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019. Improving sequence-to-sequence learning via optimal transport.

Jeffrey Chiu, Rajat Mittal, Neehal Tumma, Abhishek Sharma, and Finale Doshi-Velez. 2022a. A joint learning approach for semi-supervised neural topic modeling. *arXiv preprint arXiv:2204.03208*.

Jeffrey Chiu, Rajat Mittal, Neehal Tumma, Abhishek Sharma, and Finale Doshi-Velez. 2022b. A joint learning approach for semi-supervised neural topic modeling.

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.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transportation distances.

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

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.

Zhibin Duan, Dongsheng Wang, Bo Chen, Chaojie Wang, Wenchao Chen, Yewen Li, Jie Ren, and Mingyuan Zhou. 2021a. Sawtooth factorial topic embeddings guided gamma belief network. In *International Conference on Machine Learning*, pages 2903–2913. PMLR.

Zhibin Duan, Yishi Xu, Bo Chen, Dongsheng Wang, Chaojie Wang, and Mingyuan Zhou. 2021b. Topicnet: Semantic graph-guided topic discovery.

Lutz Dümbgen and Perla Del Conte-Zerial. 2013. On low-dimensional projections of high-dimensional distributions. *From Probability to Statistics and Back: High-Dimensional Models and Processes – A Festschrift in Honor of Jon A. Wellner*, page 91–104.

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.

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

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.

Alexander Hoyle, Pranav Goel, Denis Peskov, Andrew Hian-Cheong, Jordan Boyd-Graber, and Philip Resnik. 2021. Is automated topic model evaluation broken?: The incoherence of coherence.

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

Viet Huynh, He Zhao, and Dinh Q. Phung. 2020. Otlda: A geometry-aware optimal transport approach for topic modeling. In *NeurIPS*.

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.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes.

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

P. Langley. 2000. Crafting papers on machine learning. In *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pages 1207–1216, Stanford, CA. Morgan Kaufmann.

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.

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.

Yu Meng, Jiaxin Huang, Guangyuan Wang, Zihan Wang, Chao Zhang, Yu Zhang, and Jiawei Han. 2020. 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. 2018a. Weakly-supervised neural text classification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 983–992. ACM.

Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018b. Weakly-supervised neural text classification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM.

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

Yishu Miao, Lei Yu, and Phil Blunsom. 2015. Neural variational inference for text processing.

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.

David Mimno, Hanna Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. Optimizing semantic coherence in topic models. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 262–272.

David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*, pages 100–108.

Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. 2018. Improving topic models with latent feature word representations.

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. 2014a. 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.

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

Reutterer Reisenbichler, M. 2019. Topic modeling in marketing: recent advances and research opportunities. *J Bus Econ*, 89.

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

M. Roberts, B. Stewart, D. Tingley, and E. Airoldi. 2013. The structural topic model and applied social science. *Neural Information Processing Society*.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, page 399–408, New York, NY, USA. Association for Computing Machinery.

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

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

Luis Caicedo Torres, Luiz Manella Pereira, and M. Hadi Amini. 2021. A survey on optimal transport for machine learning: Theory and applications.

Dongsheng Wang, Dandan Guo, He Zhao, Huangjie Zheng, Korawat Tanwisuth, Bo Chen, and Mingyuan Zhou. 2022. Representing mixtures of word embeddings with mixtures of topic embeddings. *ArXiv*, abs/2203.01570.

Rui Wang, Xuemeng Hu, Deyu Zhou, Yulan He, Yuxuan Xiong, Chenchen Ye, and Haiyang Xu. 2020. Neural topic modeling with bidirectional adversarial training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 340–350, Online. Association for Computational Linguistics.

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.

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.

Hongteng Xu, Wenlin Wang, Wei Liu, and Lawrence Carin. 2018. Distilled wasserstein learning for word embedding and topic modeling. *arXiv preprint arXiv:1809.04705*.

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

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*.

Hao Zhang, Bo Chen, Dandan Guo, and Mingyuan Zhou. 2020. Whai: Weibull hybrid autoencoding inference for deep topic modeling.

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

He Zhao, Lan Du, Wray Buntine, and Mingyuan Zhou. 2018. Inter and intra topic structure learning with word embeddings. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5892–5901. PMLR.

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.

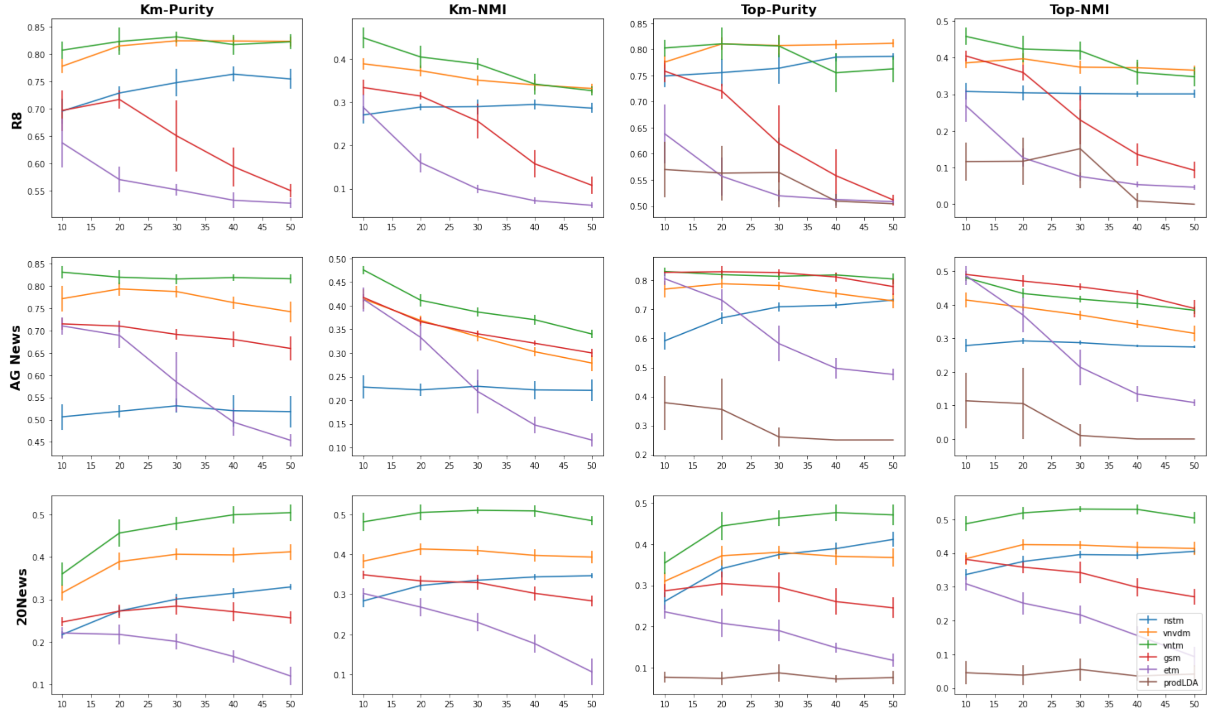


Figure 5: Each column represents a metric and each row represents a dataset. The error bar represents the standard deviation that is created by running the same model for 10 times with different random seeds.

Appendix

A Additional Experimental Results

Figure 5 shows the variation of cluster purity as the number of topics changes. This expands the information provided in Figure 2.

Figure 6 provides box plots for the metrics in Table 3.

B Proof of Lemma 3.1

Lemma B.1 *When L(X, T) reaches the global minimum. For any* (s, t), $(s', t') \in (S, T)$*:*

$$\begin{aligned} &E_{x\in s}\log q_\phi(x|t) + E_{x\in s'}\log q_\phi(x|t') \\ &-(E_{x\in s'}\log q_\phi(x|t)) + E_{x\in s}\log q_\phi(x|t')) >= 0 \end{aligned} \tag{10}$$

If the reverse is true, then, we can just switch position of topic t and t' in the topic-word matrix and also switch the position on latent space z using temperature function. This will not change reconstruction process, since for every input, get the same reconstruction. Thus, reconstruction loss does not change. Assume this new neural network structure has loss $L^{'}(X, T)$ and cross entropy loss is $L^{'}_{ce}$ L$^{'}(X, T) - L(X, T) = L^{'}_{ce} - L_{ce} = -(E_{x\in s'}\log q_\phi(x|t)) + E_{x\in s}\log q_\phi(x|t')) + E_{x\in s}\log q_\phi(x|t) + E_{x\in s'}\log q_\phi(x|t') < 0$ The last step is based on (9). This contradicts that $L(X, T)$ is global minimal. Thus, lemma holds.

C Proof of Theorem 3.2

Theorem C.1 *When L(X, T) reaches the global minimal,*

$$L_{OT} = L_{ce}$$

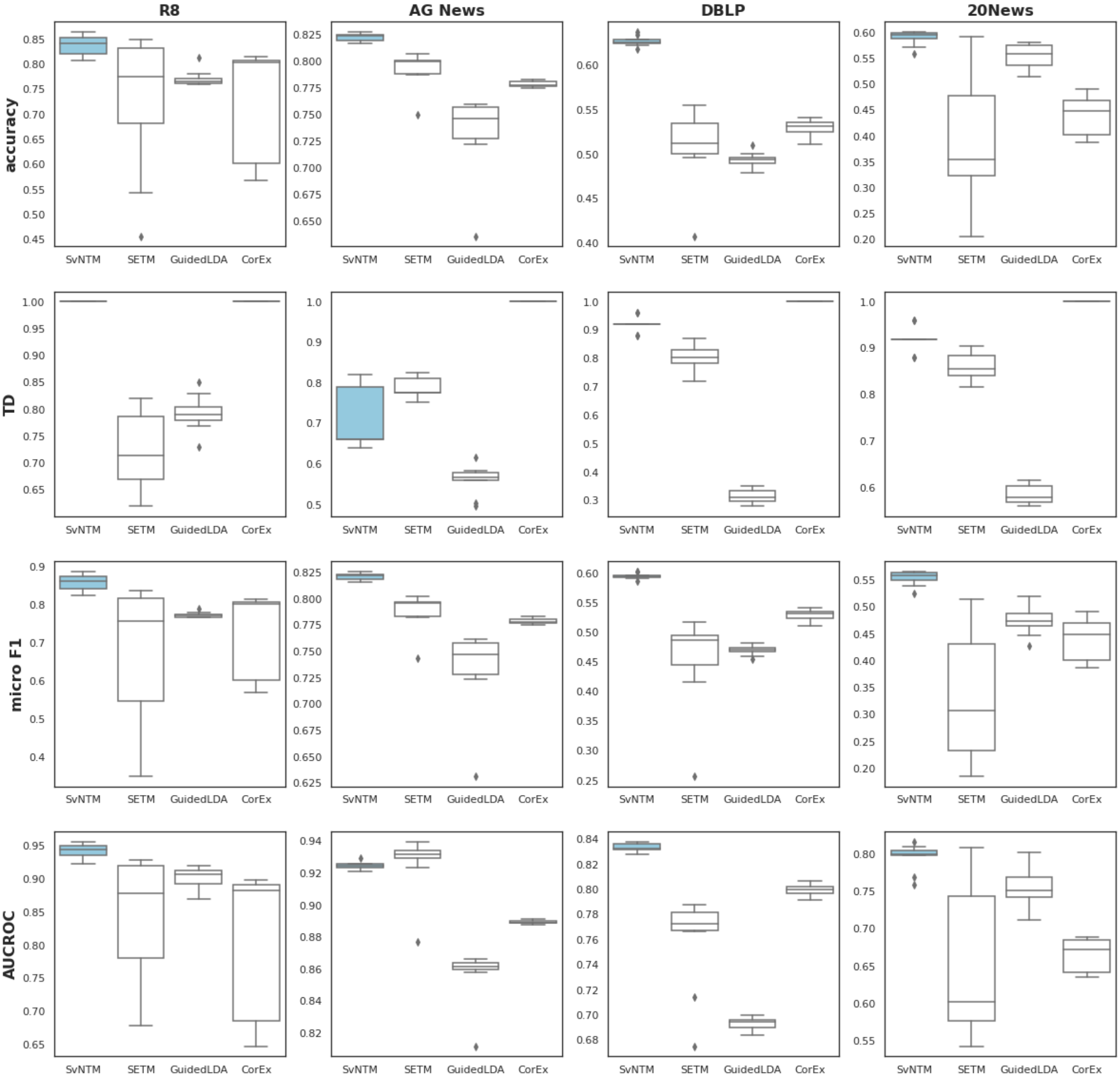


Figure 6: Each row represents a metric and each column represent a dataset. The boxplot is created by running the same model for 10 times with different random seeds. Mean and variance values are presented in the boxplot. vONT is the left most. We mark its performance in skyblue.

Step 1 show that $p_{t,s} = 1$ when $(t, s) \in (T, S)$ and equal to 0 in all other cases.
$\exists p_{t,s} = \gamma < 1$ when $(t, s) \in (T, S)$. Without loss of generality, we assume $p_{t,s'} = 1 - \gamma$, $p_{t',s'} = \gamma$ and $p_{t',s} = 1 - \gamma$. Consider related term in L_{OT}, for the first term:

$$\begin{aligned}
&\gamma(C_{t,s} + C_{t',s'}) + (1 - \gamma)(C_{t,s'} + C_{t',s}) \\
&= (C_{t,s} + C_{t',s'}) \\
&- (1 - \gamma)(C_{t,s} + C_{t',s'} - (C_{t,s'} + C_{t',s})) \\
&\geq C_{t,s} + C_{t',s'} \text{ using Lemma 3.1 and Equation (7)}
\end{aligned}$$

For the second term in L_{OT}, $-p_{t,s} \log p_{t,s} = 0$ when $p_{t,s} = 1$ or 0. Otherwise, it is larger than 0. This means that $p_{t,s} = p_{t',s'} = 1$ achieve smaller L_{OT} compare to current settings. This contradicts the definition of L_{OT} which is the min in the space. Thus, $p_{t,s} = 1$ when $(t, s) \in (T, S)$. Since the raw sum and column sum equal to $|T|$. This means $p_{t,s} = 0$ when $(t, s) \notin (T, S)$

Step 2: h(P) = - $\sum_{t,s} P_{t,s} \log P_{t,s}$
$= -(\sum_{(t,s)\in(T,S)} 1 * \log 1 + \sum_{(t,s)\notin(T,S)} 0 * \log 0) = 0$

$\sum_{t,s} P_{t,s}C_{t,s} = \sum_{(t,s)\in(T,S)} C_{t,s} = -\sum_{(t,s)\in(T,S)} E_{x\in s} \log q_\phi(x|t(x))$
Combine (10) and (11), we have L$_{OT} = \sum_{(t,s)\in(S,T)} C_{t,s} - h(P) = -\sum_{(t,s)\in(T,S)} E_{x\in s} \log q_\phi(x|t_x) = L_{ce}$

D Effect of learn-able 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 the 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 vONTSS cannot further improve with learnable temperature. 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.

E Code

Code we used to implement GSM is https://github.com/YongfeiYan/Neural-Document-Modeling Code we used to implement ETM is https://github.com/adjidieng/ETM Code we used to implement vNVDM is https://github.com/jiacheng-xu/vmf_vae_nlp with kl weight = 1 and default scaling item for auxiliary objective term equal to 0.0001 Code we used to implement NSTM is https://github.com/ethanhezhao/NeuralSinkhornTopicModel We use same parameters suggested by paper for optimal transport reclossweight = 0.07 and epsilon = 0.001. Code we used to implement ProdLDA is https://github.com/vlukiyanov/pt-avitm Code we used to implement GSM is https://github.com/YongfeiYan/Neural-Document-Modeling with topic covariance penalty equals to 1. Code we used to implement GuidedLDA is https://github.com/vi3k6i5/GuidedLDA We fine tune best seed confidence from 0 to 1 with step equal to 0.05. We simply report the best performance on average of 10 results. Code we used to implement CorEx is https://github.com/gregversteeg/corex_topic CorEx are fine-tuned by anchor strength from 1 to 7 with step equal to 1. We simply report the best performance on average of 10 results. Code we used to implement Spherical Embeddings is https://github.com/yumeng5/Spherical-Text-Embedding. We set word dimension equals 100, window size equals 10, minimum word count equals 20 and number of threads to be run in parallel equals to 20.The pretrained embedding of all datasets is at the attached data file. Code we used to implement LDA is https://scikit-learn.org/stable/modules/generated/sklearn.discriminant_analysis.LinearDiscriminantAnalysis.html with solver = SVD and tol = 0.00001

Code we used to implement CatE is https://github.com/yumeng5/WeSTClass and https://github.com/yumeng5/CatE with number of terms per topic = 10 and text embeddings dimension = 50.

F Coherence

Topic coherence **TC** metric (Mimno et al., 2011) is used to check if topic will include words that tend to co-occur in the same documents. TC (Lau et al., 2014) is the average point wise mutual information (NPMI) of two words drawn randomly from the same documents. We use both NPMI and C_v(Röder et al., 2015) by using top 10 words from each topic as suggested in (Röder et al., 2015).

G Diversity Metric

diversity is implemented using scripts: https://github.com/adjidieng/ETM/blob/master/utils.py line 4. C_v is implemented using gensim.models.coherencemodel where coherence = 'C_v', **NPMI** is implemented using gensim.models.coherencemodel where coherence = 'c_npmi'. **Top-NMI** is implemented using metrics.$normalized_mutual_info_score$ from sklearn. **Top-Purity** is implemented by definitions. **km** based is implemented by sklearn package kmeans.

H Datasets

We store the datasets and related embeddings in the attached data file. Overall, we use 4 datasets from different domain to evaluate the performance of our 2 methods.

(1) **AgNews** We use the same AG's News dataset from (Zhang et al., 2016).Overall it has 4 classes and, 30000 documents per class. Classes categories include World, Sports, Business, and Sci/Tech. for evaluation; Keywords we use: group1: government,military,war; group2:basketball,football,athletes; group3:stocks,markets,industries; group4:computer,telescope,software

(2) **R8** is a subset of the Reuters 21578 dataset, which consists of 7674 documents from 8 different reviews groups. We use class acq, earn, and we group all other data in one class. Keywords we use: group1:['acquir', 'acquisit', 'stake'], group2:['avg', 'mth', 'earn'], group3:['japan', 'offici', 'export']]

(3) **20News** (Lang, 1995) is a collection of newsgroup posts. We only select 4 categories here. Compare to previous 2 datasets, 4 categories newsgroup is small so that we can check the performance of our methods on small datasets. Keywords we use: group1: faith,accept,world; group2:evidence,religion,belief; group3:algorithm,information,problem; group4:earth,solar,satellite

(4) **DBLP** (Pan et al., 2016) dataset consists of bibliography data in computer science. DBLP selects a list of conferences from 4 research areas, database (SIGMOD, ICDE, VLDB, EDBT, PODS, ICDT, DASFAA, SSDBM, CIKM), data mining (KDD, ICDM, SDM, PKDD, PAKDD), artificial intelligent (IJCAI, AAAI, NIPS, ICML, ECML, ACML, IJCNN, UAI, ECAI,COLT, ACL, KR), and computer vision (CVPR, ICCV, ECCV, ACCV, MM, ICPR, ICIP, ICME). With a total 60,744 papers averaging 5.4 words in each title, DBLP tests the performance on small text corpus. keywords we have: group1: 'system', 'database','query'; group2: 'density', 'nonparametric', 'kernel'; group3: 'image', 'neural', 'recognition'; group4: 'partition', 'group', 'cluster'

I Analysis on vMF and Gaussian

In this section, we show empirically, vMF encourages topic separation naturally when comparing to Gaussian priors, especially in low dimensions. In the VAE training setting, we have the encoder network θ learning to transform document inputs x into distribution parameters. Without loss of generality, we denote learned parameters ϑ_i which is updated in the training process and corresponds to latent space $\eta_i \sim q(\vartheta_i)$.

Theoretically, the best q should be able to approximate the posterior distribution $p(\eta_i|x)$; however, our choice of parametric distribution family in practice will always associate with our intentions, whether to reduce training time or increase expressability. The choice of prior and posterior distribution can be viewed as a form of regularization on our decoder network, which is arbitrarily powerful. Intuitively,

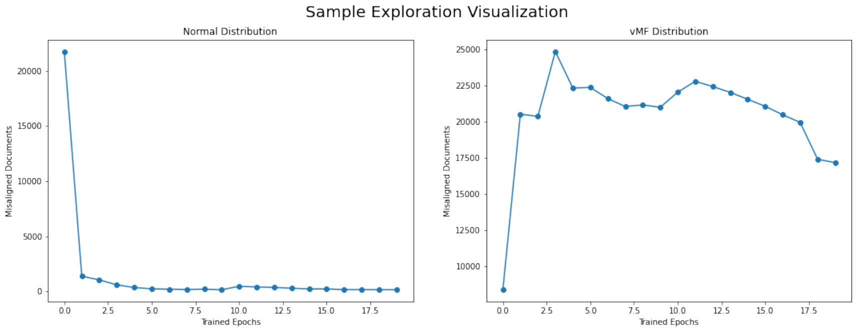


Figure 7: Sample Exploration

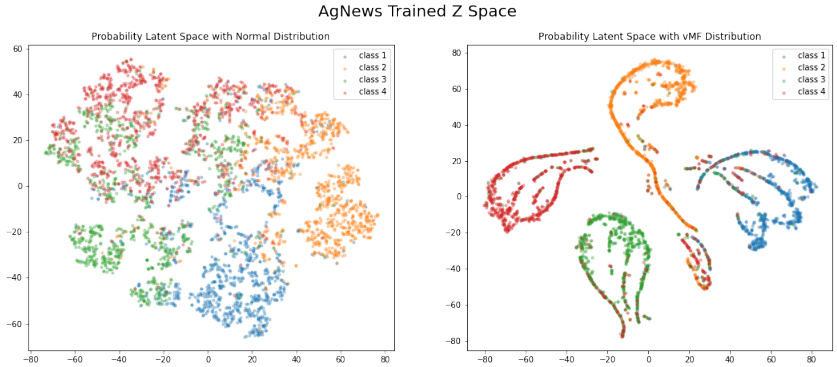


Figure 8: Z Space

distributions with fewer parameters will introduce more regularization at the cost of less flexibility, analog to bias variance trade off.

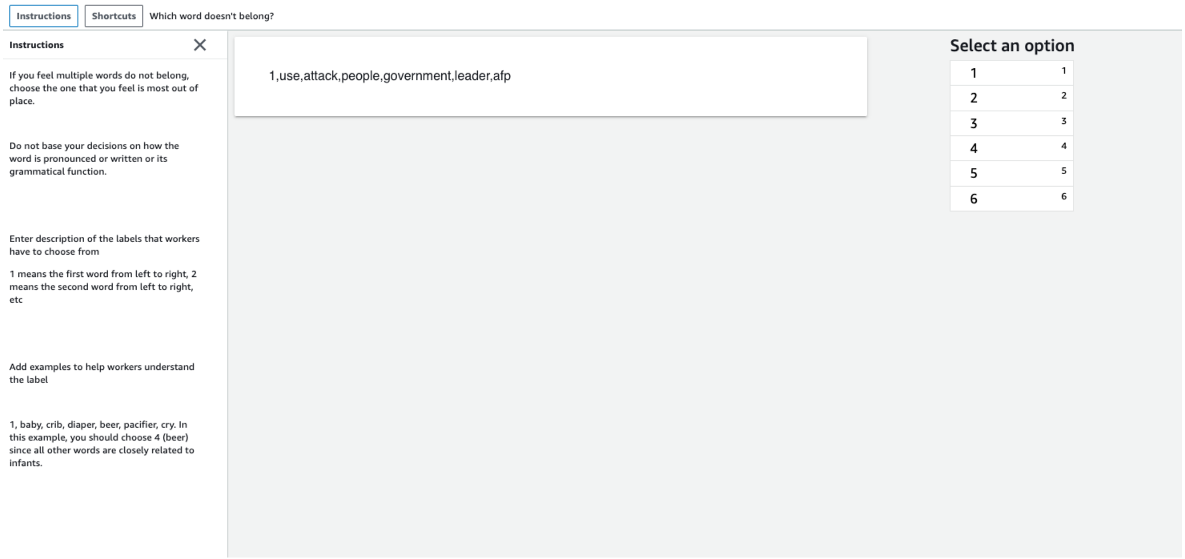
For p dimensional latent space, vMF is parameterized by p+1 variables while Gaussian is parameterized by 2*p variables assuming conditional independence or up to p(p+1)/2 + p variables assuming interdependence. In the extreme setting when labelled documents are less than $O(p^2)$, our encoder and decoder may overfit, learning identity mapping.

In the topic modelling space, a softmax transformation σ is applied to η to extract a probabilistic mixture of topics. In the independent Gaussian posterior case, we view affinity and confidence of the document to topic 1 is encoded in the first entry of μ and, σ^2 respectively. Ideally, we would want the encoder to offer variability in the sampling process to regularize, defined as difference in topic probability with initial training epochs; however, we will show through an example 7, that Gaussian may learn identity mapping by predicting variance to be near 0.

In the figure below, we define misaligned document as those documents such $argmax(\varsigma)! = argmax(\eta)$. This can be viewed as a measure of regularization. In the Gaussian case, our encoder network learns identity mapping within the first epoch. Out of 120000 documents, only 200 or so documents were able to explore different spaces. vMF allows 1/6th of documents to vary and stabilizes after KL divergence kicks in. In trained latent spaces representation, we clearly see vMF learning more nuanced and structured data when comparing to Gaussian as you can see in 8

J Human Evaluation

We use the ratings and word intrusion tasks as human evaluations of topic quality. We recruit crowdworkers using Amazon Mechanical Turk inside Amazon Sagemaker. We pay workers 0.024 per ratings task and 0.048 per intrusion tasks. We select enough crowdworkers per task so that p value for two sample t test between the best method and the second-best method is less than 0.05, resulting in a minimal of 18 crowd workers per topic for both tasks. Overall, we ask crowdsources to perform 1641 tasks and create 223



Which word doesn't belong?

Instructions

If you feel multiple words do not belong, choose the one that you feel is most out of place.

Do not base your decisions on how the word is pronounced or written or its grammatical function.

Enter description of the labels that workers have to choose from

1 means the first word from left to right, 2 means the second word from left to right, etc

Add examples to help workers understand the label

1, baby, crib, diaper, beer, pacifier, cry. In this example, you should choose 4 (beer) since all other words are closely related to infants.

1,use,attack,people,government,leader,afp

Select an option

1	1
2	2
3	3
4	4
5	5
6	6

Figure 9: User interface of intrusion task

objects. It costs 77.89 for the whole labeling job(Internal price). The user interfaces are shown in Figure 9 and Figure 10.

We select AgNews as our dataset, we generate 10 topics each from 4 models. In the word intrusion task, we sample five of the ten topic words plus one intruder randomly sampled from the dataset; for the ratings task, we present the top ten words in order.

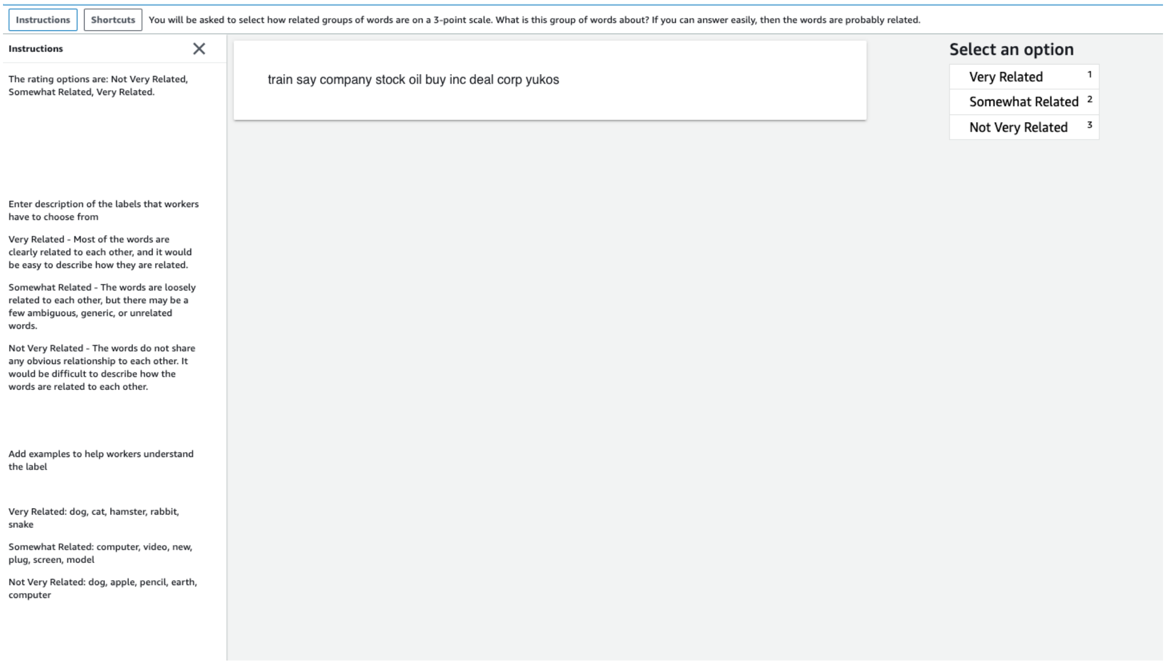
We also document the confidence per task generated by Amazon Mechanical Turk tool and average time per task for each task as can be seen below. For time spent, crowdsources spend 100 115 seconds per intrusion task and 70 80 seconds per rating task. Crowdsources spent 102.7 seconds on intrusion task generated by vONT which is lower than all other tasks. This means that it is easier for users to find intrusion word for topics generated by vONT. The confidence per rating task is between 0.88 to 0.94, where vONT has highest confidence 0.938 while LDA has lowest confidence 0.886. The confidence per intrusion task is between 0.74 to 0.86, where vONT has highest confidence 0.858 while ETM has lowest confidence 0.747. This means the crowdsources are in general more confident in their answer to questions that is generated by vONT.

K Theoretical Analysis of vMF clusterability

In this section, we present theoretical intuition behind cluster inducing property of vMF distribution comparing to the normal distribution.

In the normal VAE set up, the encoder network learns mean parameter μ_i and variance parameter σ_i for each document i. During the training process, we sample one data point, η_i from the learned distribution and pass into the softmax function to represent a probability distribution of topics. To introduce high clusterability, we need sampled η to have the ability to induce high confidence assignment to a topic under some form of regularization. In other words, with p number of topics, model can increase $argmax(softmax(\eta)) \in (1/p, 1)$ without additional penalty.

We prove that under normal distribution and in the two dimensional case, it is impossible to increase $argmax(softmax(\eta))$ without increase KL divergence loss with respect to the prior $N(0, I)$. The KL divergence with p = 2 is $KL_{normal} = -\frac{1}{2}[2 + \log \sigma_1^2 + \log \sigma_2^2 - \mu_1^2 - \mu_2^2 - \sigma_1^2 - \sigma_2^2]$ If we denote p_1 and p_2 to be expected distribution of topics, then $p_1 = \frac{e^{\mu_1}}{e^{\mu_1}+e^{\mu_2}}$ and $p_2 = \frac{e^{\mu_2}}{e^{\mu_1}+e^{\mu_2}}$. Without loss of generality, we assume that the document i is more aligned with the first topic, the model will learn and output $\mu_1 > \mu_2$. To minimize KL defined above, μ_1 and μ_2 will be centered be around 0 with $\mu_1 = -\mu_2$; however, in order to increase propensity of $argmax(softmax(\eta))$ or p_1, μ_1 and μ_2 have to increase and decrease respectively, forcing the KL divergence penalty to increase.



You will be asked to select how related groups of words are on a 3-point scale. What is this group of words about? If you can answer easily, then the words are probably related.

Instructions

The rating options are: Not Very Related, Somewhat Related, Very Related.

Enter description of the labels that workers have to choose from

Very Related - Most of the words are clearly related to each other, and it would be easy to describe how they are related.

Somewhat Related - The words are loosely related to each other, but there may be a few ambiguous, generic, or unrelated words.

Not Very Related - The words do not share any obvious relationship to each other. It would be difficult to describe how the words are related to each other.

Add examples to help workers understand the label

Very Related: dog, cat, hamster, rabbit, snake

Somewhat Related: computer, video, new, plug, screen, model

Not Very Related: dog, apple, pencil, earth, computer

train say company stock oil buy inc deal corp yukos

Select an option

Very Related 1

Somewhat Related 2

Not Very Related 3

Figure 10: User interface of rating task

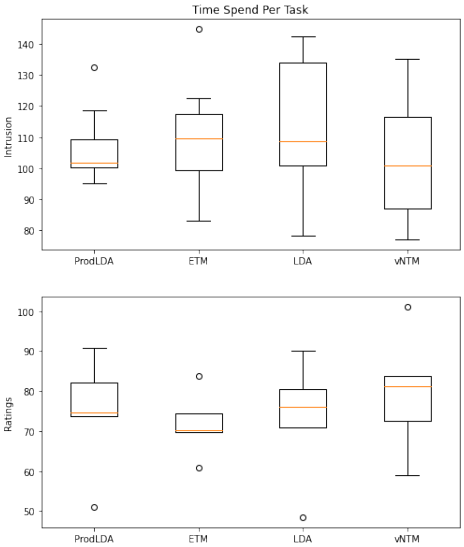


Figure 11: Compare different methods' time spend per task

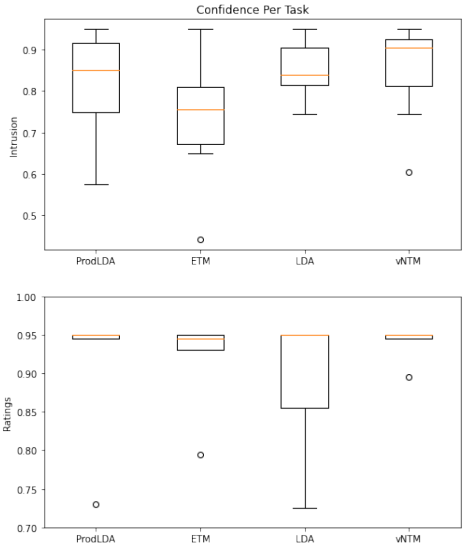


Figure 12: Compare different methods' confidence per task

For vMF distribution, the KL divergence is

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

$$+\frac{M}{2}\log\pi + \log 2 + \log\Gamma(\frac{M}{2})$$

We note that the KL penalty under vMF case is not associated with μ, thus the model can increase the propensity without increasing regularization penalties. The KL divergence of vMF distribution also makes κ small, inducing the generated topic distribution to be localized. If a data point is far different from any direction parameter μ, the reconstruction loss will be high as κ is small. Thus, μ should be as representative as possible which makes it more clustered.

L Speed

We run each model 10 times with different seeds to evaluate how long it takes to finetune the model by modifying 20 percent of keywords set.

	20News	AgNEWS
CorEX	2.18	94.98
vONTSS	0.51	51.33
GuidedLDA	1.66	24.6
CatE	104.30	888.61

Table 5: Fine Tuning in Seconds

M Related Works

Most of vMF based topic modeling methods does not incorporate variational autoencoders. Spherical Admixture Model (SAM) (Reisinger et al., 2010) is the first topic modeling method that uses vMF distribution to model corpus μ, topics and reconstructed documents. Kayhan (Batmanghelich et al., 2016) combines vMF distribution with word embeddings and uses vMF to regenerate the center of topics. It is based on Dirichlet Process to get the proportion of topics for a certain document. Hafsa (Ennajari et al., 2021) combines knowledge graph and word embeddings for spherical topic modeling. They use vMF

temperature	Top-Purity	Top-NMI	KM-purity	KM-NIM	NPMI	C_v
1	0.70735	0.33626	0.71006	0.3392	0.07476	0.53402
2	0.7636	0.39615	0.76408	0.39653	**0.10407**	0.60342
3	0.79176	0.42084	0.79174	0.42093	0.09666	0.61272
4	0.80763	0.43793	0.80762	0.43774	0.10233	0.62054
5	0.82157	0.45221	0.82128	0.45177	0.10288	**0.6225**
6	0.82232	0.45522	0.82221	0.45498	0.09325	0.60377
7	0.81163	0.45198	0.81151	0.45124	0.08936	0.58558
8	0.83201	0.46658	0.83202	0.46466	0.09089	0.57832
9	0.83013	0.47905	0.8301	0.47413	0.07968	0.55148
10	**0.8353**	0.47956	**0.83524**	0.47599	0.08726	0.5656
11	0.82824	0.48175	0.82812	0.47662	0.08048	0.55159
12	0.82555	0.47746	0.82604	0.47259	0.07785	0.54873
13	0.8268	0.49485	0.82719	0.4852	0.06195	0.50852
14	0.83481	0.49908	0.83552	0.49035	0.05528	0.50998
15	0.83189	0.50736	0.83336	0.49562	0.03945	0.48949
16	0.83049	0.51134	0.83171	0.49854	0.02964	0.48032
17	0.83023	0.50305	0.83103	0.49246	0.05862	0.50878
18	0.82232	0.50624	0.8247	0.49461	0.03647	0.48372
19	0.82955	**0.51175**	0.83167	**0.49915**	0.03496	0.48405

Table 6: Evaluate the influence of radius on coherence and clusterability related metric in Dataset AgNews. Temperature is from 1 to 20. The best scores of is highlighted in boldface. The number of topis is 10

distribution to model corpus μ, word embeddings and entity embeddings. To compare, we use modified vMF to generate topic distributions over documents and adapt spherical word embeddings instead of modeling it using vMF. Our method scales well, optimizes fast and offers highly stable performance. The choice of spherical word embeddings also alleviates the sparsity issue among words. vNVDM (Xu and Durrett, 2018) is the only other method that combines vMF with variational autoencoders. (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.* Firstly, fixing κ causes KL divergence to be constant, which reduces the regularization effect and increases the variance of the encoder. 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. We also have a different decoder.

NSTM (Zhao et al., 2020a) uses optimal transport to replace KL divergence. Row and column represent topics and words. Instead, our method represents row and column as topics and keywords with M matrix also defined differently. (Xu et al., 2018) uses optimal transport for topic embeddings, but with wasserstein distances as metric and jointly learns word embeddings. Instead, our algorithm keeps word embedding fixed during the training process to maintain stability.

N Ablation Study on Radius

Ablation study for radius parameter on AG-News where we set topics equal to 10: as we sweep temperature from 1 to 20, nmi increases and diversity decreases. Radius=10 has the best average rank over coherence based metrics in this temperature range. It has good diversity while has good coherence based metric. Temperature = 10 also has the best pruity score which make it useful for semi-supervised learning

kappa	diversity	Top-Purity	Top-Nmi	Km-Purity	Km-Nmi	NPMI	C_v
10	0.87	0.78211	0.4333	0.78428	**0.42829**	0.04752	0.51337
50	0.9904	0.77143	0.41847	0.77266	0.4183	0.05275	**0.52367**
100	0.9896	0.7832	0.42627	0.78528	0.42654	0.05031	0.51347
500	**0.9912**	0.78176	0.42274	0.78325	0.42291	0.05113	0.51655
1000	0.9896	0.76249	0.40888	0.76466	0.40932	0.04678	0.5096
varied	0.9902	**0.81**	**0.423**	**0.822**	0.404	**0.054**	0.49

Table 7: Evaluate the influence of learnable on coherence and clusterability related metric in Dataset AgNews. The best scores is highlighted in boldface. The number of topic is 20.

method	dataset	diversity	std
NSTM	R8	0.3672	0.02692
NSTM	20News	0.55636	0.04306
NSTM	AgNews	0.974	0.00806
vONT	R8	0.9224	0.01613
vONT	20News	0.9592	0.0257
vONT	AgNews	0.99022	0.01185
vNVDM	R8	0.52875	0.0868
vNVDM	20News	0.9044	0.06152
vNVDM	AgNews	0.6224	0.03772
GSM	R8	0.3868	0.05126
GSM	20News	0.6648	0.04766
GSM	AgNews	0.576	0.02091
ETM	R8	0.1224	0.01961
ETM	20News	0.5024	0.03267
ETM	AgNews	0.4896	0.04975
ProdLDA	R8	0.87429	0.05746
ProdLDA	20News	0.97143	0.04467
ProdLDA	AgNews	0.88286	0.08379

Table 8: Evaluate the diversity of vONT compare to other methods in all 3 datasets. The number of topic is 20.

O Ablation Study on κ

Ablation study for Kappa on AG-News: we check kappa = 10, 50, 100, 500, 1000. Kappa=100 has highest purity and nmi, kappa = 50 has highest NPMI and C_v. Kappa = 500 has highest diversity. Our version of kappa has highest diversity, purity and NPMI compare to all fixed kappa.

P Diversity Evaluation on vONT

vONTSS has high diversity by design. As you can see in the table, vONT achieves the best diversity on R8 and AgNews. vONT is the second best on 20News dataset. It also has the lowest standard deviation compare to other methods.

Q Why not use language modeling based methods?

Most language modeling methods are time-consuming to train and need a lot of transfer learning. They also need finetune in most of our use cases. Without fine-tuning, (Bianchi et al., 2021) makes it harder to be used in domain-specific datasets. We have tried (Yu et al., 2021; Wang et al., 2021b) to compare, but both takes too much time to run. On AG-News, (Yu et al., 2021) takes 108 minutes to run, while (Wang et al., 2021b) takes more than 2.5 hours. It also occurs in other models in footnote 2. vONTSS takes 8 minutes to run and 50 seconds to fine-tune.

We also tried some methods which only leverage embeddings of language modeling such as On AgNews and we set topics equal to 20, For (Wang et al., 2020), diversity 0.71, C_v 0.396, NPMI:-0.1089. For (Bianchi et al., 2021), diversity 1, C_v 0.435, NPMI:-0.1073. Except diversity in (Bianchi et al., 2021), all other metric perform worse than vONT.

For semi-spervised cases, we take keywords as input. It is really different from other weakly supervised learning formulations, and how to incorporate keywords into a language model is not straight forward. We have tried few methods, but it does take a lot of time to run and change their code is not easy since their effectiveness do rely on the specific version of language model. Thus, we exclude language modeling methods in our paper. Also, in our use case, each topic model is designed for a specific user or use case. It will be very hard to be interactive or store the model on user's side when the number of parameters is too large for every single model.

R Limitations and Risks

vMF distribution has a unit constraint. This limits the variability of latent space, which in turn reduces the gains as the number of topics increase. We can try other distributions with richer variability, such as Bivariate von Mises distribution and Kent distribution.

Also, in weakly supervised cases, vONTSS may not perform as well as those methods that leverage pretraining language models in classification. In the future, we can combine the structure of this model with existed language modeling to further improve its classification performance.

Lastly, in semi-supervised cases version, our formulation of vONTSS requires each topic to have at least one keyword. This limits its practical usage to some extent. To solve it, we can first preselect topics before doing the topics and keywords mapping, or we can modify the optimal transport loss using Gumbel distributions.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Appendix R

☑ A2. Did you discuss any potential risks of your work?
Appendix R

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

3 Proposed Methods

□ B1. Did you cite the creators of artifacts you used?
Not applicable. I provided the artifacts

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Not applicable. Left blank.

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4 Appendix L

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 4 Appendix NOP

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix E

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Section 4 Appendix J

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Appendix J

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Appendix J

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Appendix J

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Appendix J

☒ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
It is anouymous