Topic Discovery via Latent Space Clustering of Pretrained Language Model Representations

Yu Meng, Yunyi Zhang, Jiaxin Huang, Yu Zhang, Jiawei Han
Department of Computer Science, University of Illinois at Urbana-Champaign, IL, USA
{yumeng5, yzhan238, jiaxinh3, yuz9, hanj}@illinois.edu

ABSTRACT

Topic models have been the prominent tools for automatic topic discovery from text corpora. Despite their effectiveness, topic models suffer from several limitations including the inability of modeling word ordering information in documents, the difficulty of incorporating external linguistic knowledge, and the lack of both accurate and efficient inference methods for approximating the intractable posterior. Recently, pretrained language models (PLMs) have brought astonishing performance improvements to a wide variety of tasks due to their superior representations of text. Interestingly, there have not been standard approaches to deploy PLMs for topic discovery as better alternatives to topic models. In this paper, we begin by analyzing the challenges of using PLM representations for topic discovery, and then propose a joint latent space learning and clustering framework built upon PLM embeddings. In the latent space, topic-word and document-topic distributions are jointly modeled so that the discovered topics can be interpreted by coherent and distinctive terms and meanwhile serve as meaningful summaries of the documents. Our model effectively leverages the strong representation power and superb linguistic features brought by PLMs for topic discovery, and is conceptually simpler than topic models. On two benchmark datasets in different domains, our model generates significantly more coherent and diverse topics than strong topic models, and offers better topic-wise document representations, based on both automatic and human evaluations.[1]

CCS CONCEPTS

• **Information systems** → **Clustering**; **Document topic models**; • **Computing methodologies** → **Natural language processing**.

KEYWORDS

Topic Discovery, Pretrained Language Models, Clustering

ACM Reference Format:
Yu Meng, Yunyi Zhang, Jiaxin Huang, Yu Zhang, Jiawei Han. 2022. Topic Discovery via Latent Space Clustering of Pretrained Language Model Representations. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3485447.3512034

[1]Code and data can be found at https://github.com/yumeng5/TopClus.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
WWW '22, April 25–29, 2022, Virtual Event, Lyon, France
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3512034

1 INTRODUCTION

Automatically discovering coherent and meaningful topics from text corpora is intuitively appealing for web-scale content analyses, as it facilitates many web applications including document analysis [9], text summarization [63] and ad-hoc information retrieval [65]. Decades of research efforts have been dedicated to the development of such algorithms, among which topic models [11, 26] are the most prominent methods. The success of topic models can be largely credited to their proposed generative process: By maximizing the likelihood of a probabilistic process that models how documents are generated conditioned on the hidden topics, topic models are able to uncover the latent topic structures in the corpus.

Despite the success of topic models, the generative process incurs several notable limitations: (1) The "bag-of-words" generative assumption completely ignores word ordering information in text, which is essential for defining word meanings [18]. (2) The generative process cannot leverage external knowledge to learn word semantics, which may miss important topic-indicating words if they are not sufficiently reflected by the co-occurrence statistics of the given corpus, as is likely the case for small-scale/short-text corpora. (3) The generative process induces an intractable posterior that requires approximation algorithms like Monte Carlo simulation [50] or variational inference [1]. Unfortunately, there is always a trade-off between accuracy and efficiency with these approximations since they can only be asymptotically exact [57]. Later variants of topic models attempt to overcome some of these limitations by either replacing the analytic approximation of the posterior with deep neural networks [47, 60, 64] to improve the effectiveness and efficiency of the inference process, or incorporating word embeddings [15, 17, 52] to make up for the representation deficiency of the "bag-of-words" generative assumption. Nevertheless, without fundamental changes of the topic modeling framework, none of these approaches address the limitations of topic models all at once.

Along another line of text representation learning research, text embeddings have achieved enormous success in a wide spectrum of downstream tasks. The effectiveness of text embeddings stems from the learning of distributed representations of words and documents from contexts. Early models like Word2Vec [48] learn context-free word semantics based on a local context window of the center word. Recently, pretrained language models (PLMs) like BERT [16], RoBERTa [36] and XLNet [67] have revolutionized text processing via learning contextualized word embeddings. They employ Transformer [62] as the backbone architecture for capturing the long-range, high-order semantic dependency in text sequences, yielding superior representations to previous context-free embeddings. Since these PLMs are pretrained on large-scale text corpora like Wikipedia, they carry superb linguistic features that can be generalized to almost any text-related applications.

Motivated by the strong representation power of the contextualized embeddings that accurately capture word semantics, a few

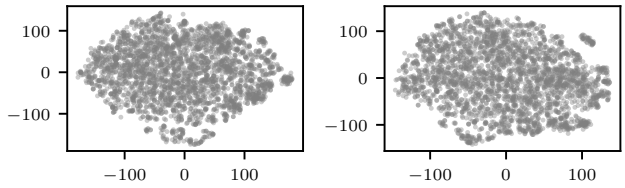
recent studies have attempted to utilize PLMs for topic discovery. Sia et al. [59] directly cluster averaged BERT word embeddings to obtain word clusters as topics. The resulting topic quality relies significantly on heuristic tricks like frequency-based weighting/re-ranking and barely reaches the performance of LDA, the most basic topic model. Instead of clustering word embeddings, BERTopic [23] clusters document embeddings and then uses TF-IDF metrics to extract representative terms from each notable document cluster as topics. However, as the document embeddings in BERTopic are obtained from Sentence-BERT [56], which is trained on natural language inference datasets with manually annotated sentence labels, the performance of BERTopic may suffer from domain shift when the target corpus is semantically different from the Sentence-BERT training set, and when manually annotated labels for re-training the sentence embeddings are absent. Moreover, BERTopic constructs topics via TF-IDF metrics and fails to take advantage of the distributed representations of PLMs, which are known to better capture word semantics than frequency-based statistics.

In this work, we study topic discovery with PLM embeddings as a potential alternative to topic models. We first analyze the challenges of directly operating on the PLM embedding space by investigating its structure. Motivated by the challenges, we propose TopClus, a joint latent space learning and clustering approach that derives a *lower-dimensional*, *spherical* latent embedding space with topic structures. Such latent space mitigates the "curse of dimensionality" issue and uses angular similarity to model semantic correlations among words, documents and topics, thus is better suited for clustering than the high-dimensional Euclidean embedding space of PLMs. Unlike traditional clustering algorithms that work with fixed data representations, TopClus jointly adjusts the latent space representations and performs clustering. Topic-word and document-topic distributions are jointly modeled in the latent space to derive topics that (1) are interpretable by coherent and distinctive words and (2) serve as meaningful summaries of documents.

TopClus enjoys the following advantages over topic models: (1) TopClus works with PLM contextualized embeddings obtained by modeling the entire text sequences with positional information, which are expected to provide better representations than the "bag-of-words" assumption of topic models. (2) TopClus employs PLMs to bring in general linguistic knowledge which helps generate more accurate and stable word representations on the target corpus than training topic models from scratch on it. (3) The training algorithm of TopClus does not involve any probabilistic approximations, and is computationally and conceptually simpler than variational inference in topic models. With these advantageous properties, TopClus simultaneously addresses the major limitations of topic models.

Our contributions are summarized as follows:

(1) We explore using PLM embeddings for topic discovery. We first identify the challenges with an in-depth analysis of the original PLM embedding space's structure.
(2) We propose a new framework TopClus which jointly learns a lower-dimensional, spherical latent space with cluster structures based on word and document embeddings from PLMs. High-quality topic clusters are derived by simultaneously modeling topic-word and document-topic distributions. TopClus can be integrated with any PLMs for unsupervised topic discovery.
(3) We propose three objectives for training TopClus to induce distinctive and balanced cluster structures in the latent space which result in diverse and coherent topics.



(a) **New York Times**. (b) **Yelp Review**.

Figure 1: Visualization using t-SNE of $3,000$ randomly sampled contextualized word embeddings of BERT on (a) NYT and (b) Yelp datasets, respectively. The embedding spaces do not have clearly separated clusters.

(4) We evaluate TopClus on two benchmark datasets in different domains. TopClus significantly outperforms strong topic discovery methods by generating more coherent and diverse topics and providing better document topic representations judged from both automatic and human evaluations.

2 CHALLENGES OF TOPIC DISCOVERY WITH PRETRAINED LANGUAGE MODELS

We first identify three major challenges of using PLM embeddings for topic discovery, which motivate our proposed model in Section 3.

Unsuitability of PLM Embedding Space for Clustering. One straightforward way of obtaining K topics with PLM embeddings (*e.g.*, from BERT [16]) is to simply apply clustering algorithms like K-means [37] to group correlated terms that form topics. To provide empirical evidence that such direct clustering may not work well, we visualize $3,000$ randomly sampled contextualized word embeddings obtained by running BERT on the New York Times and Yelp Review datasets in Figure 1. The embedding spaces do not exhibit clearly separated clusters, and applying clustering algorithms like K-means with a typical K (*e.g.*, $K = 100$) to these spaces leads to low-quality and unstable clusters. We show theoretically that such a phenomenon is due to too many clusters in the embedding space. Below, we study the effect of the Masked Language Modeling (MLM) pretraining objective of BERT on the embedding space.

Theorem 2.1. *The MLM pretraining objective of BERT assumes that the learned contextualized embeddings are generated from a Gaussian Mixture Model (GMM) with $|V|$ mixture components where $|V|$ is the vocabulary size of BERT.*

Proof. See Appendix B. □

Theorem 2.1 applies to many PLMs (*e.g.*, BERT [16], RoBERTa [35], XLNet [67]) that use MLM-like pretraining objectives. It reveals that the optimal number of cluster K to apply K-means like algorithm is $|V|$ ($|V| \approx 30,000$ in the BERT base model). In other words, the PLM embedding space is partitioned into extremely fine-grained clusters and lacks topic structures inherently. If a typical K for topic discovery is used ($K \ll |V|$), the partition will not fit the original data well, resulting in unstable and low-quality clusters. If a very big K is used ($K \approx |V|$), most clusters will contain only one unique term, which is meaningless for topic discovery.

Curse of Dimensionality. PLM embeddings are usually high-dimensional (*e.g.*, number of dimensions $r = 768$ in the BERT base model), while distance functions can become meaningless and unreliable in high-dimensional spaces [5], rendering Euclidean distance based clustering algorithms ineffective for high-dimensional

cases, known as the "curse of dimensionality". From another perspective, the high-dimensional PLM embeddings encode linguistic information of multiple aspects for the generic language modeling purpose, but some features are not necessary for or may even interfere with topic discovery. For example, some syntactic features in the PLM embeddings should not be considered when grouping semantically similar concepts (*e.g.*, "play", "plays" and "playing" should not represent different topics).

Lack of Good Document Representations from PLMs. Topic discovery usually requires jointly modeling documents with words to derive latent topics. Although PLMs are famous for their superior contextualized word representations, obtaining quality document embeddings from PLMs has been a big challenge. Sentence-BERT [56] reports that the inherent BERT sequence embeddings (*i.e.*, obtained from the `[CLS]` token) are of rather bad quality without fine-tuning, even worse than averaged GloVe context-free embeddings. To obtain meaningful sentence embeddings, Sentence-BERT fine-tunes pretrained BERT model on natural language inference (NLI) tasks with manually annotated sentences. However, using Sentence-BERT for topic discovery raises two concerns: (1) When the given corpus has a big domain shift from the Sentence-BERT training set (*e.g.*, the documents are much longer than the sentences in NLI, or are very different semantically from the NLI dataset), the document embeddings need to be re-trained from target corpus document labels, which contradicts the unsupervised nature of topic discovery. (2) The sentence embeddings are in a different space from word embeddings as they are not jointly trained, and cannot be simultaneously used to model both words and documents. This is why BERTopic [23] relies on TF-IDF for topic word selection.

3 METHOD

We first introduce the two major components in our TopClus model: (1) attention-based document embedding learning module and (2) latent space generative module, and then we introduce three training objectives for model learning. Figure 2 provides an overview of TopClus. We assume BERT is used as the PLM, but TopClus can be seamlessly integrated with any other PLMs.

3.1 Attention-Based Document Embeddings

As the prerequisite of topic discovery is the joint modeling of words and documents, we first propose a simple attention mechanism to learn document embeddings. Previous studies [33] show that a simple average of word embeddings from PLMs can serve as decent generic sequence representations. In this work, we assume that not all words in a document are equally topic-indicative, so we learn attention weights of each token to derive document embeddings as a weighted average of contextualized word embeddings which are expected to be better tailored for topic discovery than an unweighted average of word embeddings. This also allows the learned document embeddings to share the same space with word embeddings which enables joint modeling of words and documents.

For each text document $\boldsymbol{d} = [w_1, w_2, \ldots, w_n]$, we obtain the BERT contextualized word representations $[\boldsymbol{h}_1^{(w)}, \boldsymbol{h}_2^{(w)}, \ldots, \boldsymbol{h}_n^{(w)}]$ where $\boldsymbol{h}_i^{(w)} \in \mathbb{R}^r$ ($r = 768$ in the BERT base model). The attention weights $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_n]$ are learned for each token as follows:

$$\boldsymbol{l}_i = \tanh\left(\boldsymbol{W}\boldsymbol{h}_i^{(w)} + \boldsymbol{b}\right), \quad \alpha_i = \frac{\exp(\boldsymbol{l}_i^\top \boldsymbol{v})}{\sum_{j=1}^n \exp(\boldsymbol{l}_j^\top \boldsymbol{v})},$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ are learnable parameters of a linear layer with the $\tanh(\cdot)$ activation. Each word embedding $\boldsymbol{h}_i^{(w)}$ is transformed to a new representation $\boldsymbol{l}_i$ whose dot product with another learnable vector $\boldsymbol{v}$ reflects how topic-indicative the token is. Finally, the document embedding $\boldsymbol{h}^{(d)}$ is obtained as the combination of all word embeddings in the document weighted by the attention values:

$$\boldsymbol{h}^{(d)} = \sum_{i=1}^n \alpha_i \boldsymbol{h}_i^{(w)}.$$

We note that the contextualized word embeddings from BERT $\{\boldsymbol{h}_i^{(w)}\}_{i=1}^n$ are not updated during topic discovery since they already capture word semantics reliably and accurately through pretraining. The learnable parameters associated with the attention mechanism $\boldsymbol{A} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{v}\}$ are randomly initialized and trained via the unsupervised objectives to be introduced in Section 3.3.

3.2 The Latent Space Generative Model

Motivation and Assumptions. As we have shown in Section 2, the original embedding space $\boldsymbol{H}$ of PLMs is unsuitable for direct clustering to generate topic clusters. To address the challenges, we propose to project the original embedding space $\boldsymbol{H}$ into a latent space $\boldsymbol{Z}$ with K soft clusters of words corresponding to K latent topics. We assume that $\boldsymbol{Z}$ is *spherical* (*i.e.*, $\boldsymbol{Z} \subset \mathbb{S}^{r'-1}$; $\mathbb{S}^{r'-1} = \{\boldsymbol{z} \in \mathbb{R}^{r'} : \|\boldsymbol{z}\| = 1\}$ is the unit $r'-1$ sphere) and *lower-dimensional* (*i.e.*, $r' < r$). Such a latent space has the following preferable properties: (1) In the spherical latent space, angular similarity (*i.e.*, without considering vector norms) between vectors is employed to capture word semantic correlations, which works better than Euclidean metrics (*e.g.*, cosine similarity between embeddings is more effective for measuring word similarity [40, 46]). (2) The lower-dimensional space mitigates the "curse of dimensionality" of the original high-dimensional space and better suits the clustering task. (3) Projecting high-dimensional embeddings to the lower-dimensional space forces the model to discard the information that does not help form topic clusters (*e.g.*, syntactic features).

Why Not Naive Approach? A straightforward way is to first apply a dimensionality reduction technique to the original embedding space $\boldsymbol{H}$ to obtain the aforementioned latent space $\boldsymbol{Z}$, and subsequently apply clustering algorithms to $\boldsymbol{Z}$ for obtaining the latent space clusters representing topics. However, such a naive approach cannot guarantee that the reduced-dimension embeddings will be naturally suited for clustering, given that no clustering-promoting objective is incorporated in the dimensionality reduction step. Therefore, we propose to *jointly* learn the latent space projection and cluster in the latent space instead of conducting them one after another, so that the latent representation learning is guided by the clustering objective, and the cluster quality benefits from the well-separated structure of the latent space, achieving a mutually-enhanced effect. Such joint learning is realized by training a generative model that connects the latent topic structure with the original space representations.

Our Generative Model. We introduce our latent space generative model as follows. With the number of topics K as the input to the model, we assume that there exists a latent space $\boldsymbol{Z} \subset \mathbb{S}^{r'-1}$ with K topics reflecting the latent structure of the original embedding space $\boldsymbol{H}$. Each topic is associated with a spherical distribution called the von Mises-Fisher (vMF) distribution [2, 21] that characterizes the topic-word and document-topic distributions in the latent space.

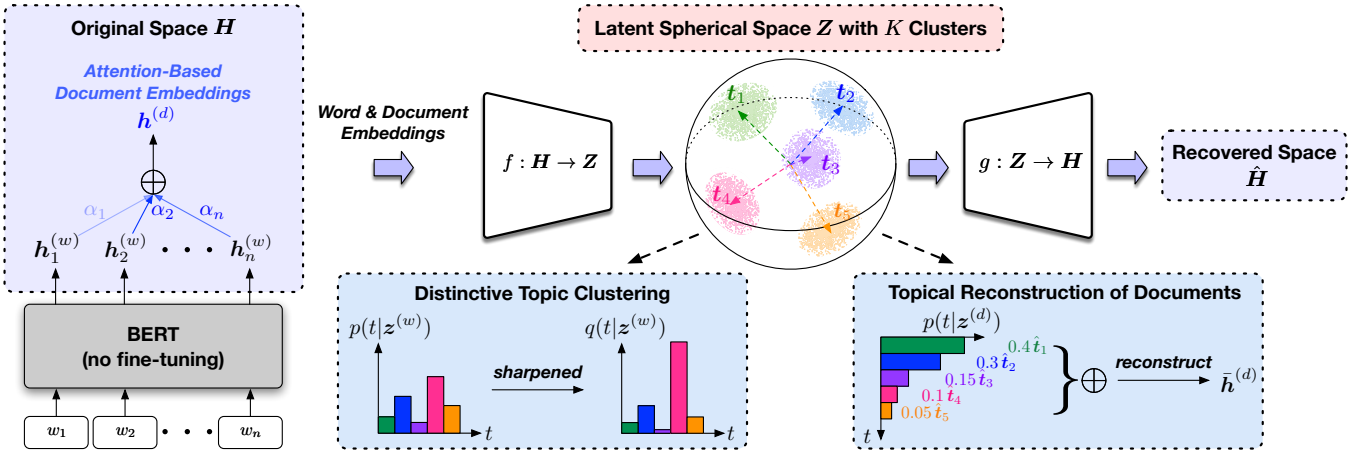


Figure 2: Overview of TopClus**. We assume that the K-topic structure exists in a latent spherical space Z. We jointly learn the attention weights for document embeddings and the latent space generation model via three objectives: (1) a clustering loss that encourages distinctive topic learning in the latent space, (2) a topical reconstruction loss of documents that promotes meaningful topic representations for summarizing document semantics and (3) an embedding space preserving loss that maintains the semantics of the original embedding space. The PLM is not fine-tuned.**

Specifically, the vMF distribution (can be seen as the spherical counterpart of the Gaussian distribution) of a topic t is parameterized by a mean vector $\boldsymbol{t}$ and a concentration parameter κ. The probability density closer to $\boldsymbol{t}$ is greater and the spread is controlled by κ. Intuitively, words and documents are more likely to be correlated with a topic t if their latent space representations are closer to the topic vector $\boldsymbol{t}$. Formally, a unit random vector $\boldsymbol{z} \in \mathbb{S}^{r'-1}$ has the r'-variate vMF distribution $\text{vMF}_{r'}(\boldsymbol{t}, \kappa)$ if its probability density function is

$$p(\boldsymbol{z}; \boldsymbol{t}, \kappa) = n_{r'}(\kappa) \exp\left(\kappa \cdot \cos(\boldsymbol{z}, \boldsymbol{t})\right),$$

where $\|\boldsymbol{t}\| = 1$ is the center direction, $\kappa \geq 0$ is the concentration parameter, $\cos(\boldsymbol{z}, \boldsymbol{t})$ is the cosine similarity between $\boldsymbol{z}$ and $\boldsymbol{t}$, and the normalization constant $n_{r'}(\kappa)$ is given by

$$n_{r'}(\kappa) = \frac{\kappa^{r'/2-1}}{(2\pi)^{r'/2} I_{r'/2-1}(\kappa)},$$

where $I_{r'/2-1}(\cdot)$ represents the modified Bessel function of the first kind at order $r'/2-1$. We assume all topics' vMF distributions share the same concentration parameter κ (*i.e.*, the topic terms are equally concentrated around the topic center for all topics) which can be set as a hyperparameter.

Every word embedding $\boldsymbol{h}_i^{(w)} \in \boldsymbol{H}$ from the original space is assumed to be generated through the following process : (1) A topic t_k is sampled from a uniform distribution over the K topics. (2) A latent embedding $\boldsymbol{z}_i^{(w)}$ is generated from the vMF distribution associated with topic t_k. (3) A function $g : Z \to \boldsymbol{H}$ maps the latent embedding $\boldsymbol{z}_i^{(w)}$ to the original embedding $\boldsymbol{h}_i^{(w)}$ corresponding to word w_i. The generative process is summarized as follows:

$$t_k \sim \text{Uniform}(K),\ \boldsymbol{z}_i^{(w)} \sim \text{vMF}_{r'}(\boldsymbol{t}_k, \kappa),\ \boldsymbol{h}_i^{(w)} = g(\boldsymbol{z}_i^{(w)}). \quad (1)$$

The generative process of document embedding $\boldsymbol{h}^{(d)} \in \boldsymbol{H}$ is similar since it resides in the same word embedding space:

$$t_k \sim \text{Uniform}(K),\ \boldsymbol{z}^{(d)} \sim \text{vMF}_{r'}(\boldsymbol{t}_k, \kappa),\ \boldsymbol{h}^{(d)} = g(\boldsymbol{z}^{(d)}). \quad (2)$$

We assume that the mapping function g can be nonlinear to model arbitrary transformations, and we parameterize g as a deep neural network (DNN) since DNNs can approximate any nonlinear function [27]. Each layer l in the DNN is a linear layer with ReLU activation function, taking $\boldsymbol{x}_l$ as input and outputting $\boldsymbol{y}_l$:

$$\boldsymbol{y}_l = \text{ReLU}(\boldsymbol{W}_l \boldsymbol{x}_l + \boldsymbol{b}_l),$$

where $\boldsymbol{W}_l$ and $\boldsymbol{b}_l$ are the learnable parameters in the layer. We also jointly learn the mapping $f : \boldsymbol{H} \to Z$ from the original space to the latent space (*i.e.*, the inverse function of g, also parameterized by a DNN) to map unseen word/document embeddings to the latent space. Such joint learning of two nonlinear functions follows an autoencoder [25] setup where an encoding network maps data points from the original space to the latent space, and a decoding network converts latent space data back to an approximate reconstruction of the original data.

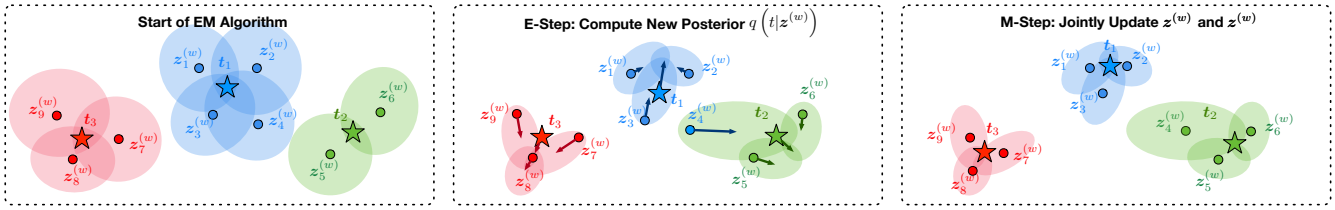
3.3 Model Training

To jointly train the attention module for document embeddings in Section 3.1 and the latent generative model in Section 3.2, we introduce three objectives: (1) a clustering loss that enforces separable cluster structures in the latent space for distinctive topic learning, (2) a topical reconstruction loss of documents to ensure the discovered topics are meaningful summaries of document semantics, and (3) an embedding space preserving loss to maintain the semantic information in the original space.

Distinctive Topic Clustering. The first clustering objective induces a latent space with K well-separated clusters by gradually sharpening the posterior topic-word distributions via an expectation–maximization (EM) algorithm. In the E-step, we estimate a new (soft) cluster assignment of each word based on the current parameters; in the M-step, we update the model parameters given the cluster assignments. The process is illustrated in Figure 3.

E-Step. To estimate the cluster assignment of each word, we compute the posterior topic distribution obtained via the Bayes rule:

$$p\left(t_k | \boldsymbol{z}_i^{(w)}\right) = \frac{p\left(\boldsymbol{z}_i^{(w)} | t_k\right) p(t_k)}{\sum_{k'=1}^{K} p\left(\boldsymbol{z}_i^{(w)} | t_{k'}\right) p(t_{k'})},$$



(a) Start of EM Algorithm. (b) E-Step. (c) M-Step.

Figure 3: One iteration of EM algorithm. During the E-Step, we compute new posterior topic-word distribution $q(t|z^{(w)})$ that sharpens the original posterior $p(t|z^{(w)})$ (resulting in lower entropy of $p(t|z^{(w)})$ denoted by the smaller colored area around $z^{(w)}$) and meanwhile encourage balanced cluster distribution (resulting in some cluster assignment change). During the M-Step, we update topic embeddings t and word embeddings $z^{(w)} = f(\boldsymbol{h}^{(w)})$ according to the new posteriors.

where $p\left(\boldsymbol{z}_i^{(w)}|\boldsymbol{t}_k\right) = \text{vMF}_{r'}(\boldsymbol{t}_k, \kappa) = n_{r'}(\kappa)\exp\left(\kappa \cdot \cos(\boldsymbol{z}_i^{(w)}, \boldsymbol{t}_k)\right)$ and $p(t_k) = 1/K$ according to Eq. (1). The posterior is simplified as

$$p\left(t_k|\boldsymbol{z}_i^{(w)}\right) = \frac{\exp\left(\kappa \cdot \cos\left(\boldsymbol{z}_i^{(w)}, \boldsymbol{t}_k\right)\right)}{\sum_{k'=1}^{K}\exp\left(\kappa \cdot \cos\left(\boldsymbol{z}_i^{(w)}, \boldsymbol{t}_{k'}\right)\right)}.$$

Then we compute a new estimate of the cluster assignments $q(t_k|\boldsymbol{z}_i^{(w)})$ to be used for updating the model in the M-Step following [66]:

$$q\left(t_k|\boldsymbol{z}_i^{(w)}\right) = \frac{p\left(t_k|\boldsymbol{z}_i^{(w)}\right)^2/s_k}{\sum_{k'=1}^{K}p\left(t_{k'}|\boldsymbol{z}_i^{(w)}\right)^2/s_{k'}},\ s_k = \sum_{i=1}^{N}p\left(t_k|\boldsymbol{z}_i^{(w)}\right), \quad (3)$$

where N is the total number of tokens in the corpus. Using Eq. (3) to obtain the target cluster assignment has the following two favorable effects: (1) *Distinctive topic learning*. Squaring-then-normalizing the posterior distribution $p(t_k|\boldsymbol{z}_i^{(w)})$ has a sharpening effect that skews the distribution towards its most confident cluster assignment, and the so learned latent space will have gradually well-separated clusters for distinctive topic interpretation. This is similar in spirit to the Dirichlet prior used in LDA that promotes sparse topic distributions. (2) *Topic prior regularization*. The soft cluster frequency s_k should encode the uniform topic prior assumed in Eq. (1), and dividing the sharpened $p(t_k|\boldsymbol{z}_i^{(w)})^2$ by s_k encourages balanced clusters.

M-Step. We update the model parameters to maximize the expected log-probability of the current cluster assignment under the new cluster assignment estimate $\mathbb{E}_q[\log p]$, which is equivalent to minimizing the following cross entropy loss:

$$\mathcal{L}_{\text{clus}} = -\sum_{i=1}^{N}\sum_{k=1}^{K}q\left(t_k|\boldsymbol{z}_i^{(w)}\right)\log p\left(t_k|\boldsymbol{z}_i^{(w)}\right), \quad (4)$$

where p is updated to approximate q which is a fixed target. Using Eq. (4) to update the model parameters has a notable difference from standard clustering algorithms: Since $p(t_k|\boldsymbol{z}_i^{(w)})$ is jointly determined by the topic center vector $\boldsymbol{t}_k$ and latent representation $\boldsymbol{z}_i^{(w)}$, both of them will be updated to fit the new estimate $q(t_k|\boldsymbol{z}_i^{(w)})$ which encourages distinctive cluster distribution. Therefore, the mapping function f will be adjusted accordingly to induce a latent space with a K-cluster structure and the topic center vectors will become K anchoring points surrounded by topic-representative words. In contrast, standard clustering algorithms only update the cluster parameters without changing the data representations.

Topical Reconstruction of Documents. The second objective aims to reconstruct document semantics with topic representations so that the learned latent topics are meaningful summaries of the documents. Specifically, the reconstructed document embedding $\hat{\boldsymbol{h}}^{(d)}$ is obtained by combining all projected topic vectors $\hat{\boldsymbol{t}}_k$ weighted by the document-topic distribution $p(t_k|\boldsymbol{z}^{(d)})$:

$$\hat{\boldsymbol{h}}^{(d)} = \sum_{k=1}^{K}p\left(t_k|\boldsymbol{z}^{(d)}\right)\hat{\boldsymbol{t}}_k, \quad \hat{\boldsymbol{t}}_k = g(\boldsymbol{t}_k),$$

where $p(t_k|\boldsymbol{z}^{(d)})$ is obtained according to Eq. (2):

$$p\left(t_k|\boldsymbol{z}^{(d)}\right) = \frac{\exp\left(\kappa \cdot \cos\left(\boldsymbol{z}^{(d)}, \boldsymbol{t}_k\right)\right)}{\sum_{k'=1}^{K}\exp\left(\kappa \cdot \cos\left(\boldsymbol{z}^{(d)}, \boldsymbol{t}_{k'}\right)\right)}.$$

We require the reconstructed document embedding to be a good approximation of the original content by minimizing the following reconstruction loss:

$$\mathcal{L}_{\text{rec}} = \sum_{d\in\mathcal{D}}\left\|\hat{\boldsymbol{h}}^{(d)} - \bar{\boldsymbol{h}}^{(d)}\right\|^2, \quad (5)$$

where $\bar{\boldsymbol{h}}^{(d)}$ is the average of word embeddings in the document serving as the generic document embedding.

Preservation of Original PLM Embeddings. We need to ensure the latent space preserves the important semantic information of the original embedding space, and the third objective encourages the output of the autoencoder to faithfully recover the structure of the original embedding space by minimizing the the following loss:

$$\mathcal{L}_{\text{pre}} = \sum_{i=1}^{N}\left\|\boldsymbol{h}_i^{(w)} - g\left(f\left(\boldsymbol{h}_i^{(w)}\right)\right)\right\|^2. \quad (6)$$

Overall Algorithm. We summarize the training of TopClus in Algorithm 1. We first pretrain the mapping functions f and g only using the preservation loss in Eq. (6) as it provides a stable initialization of the latent space [66]. During training, we apply the EM algorithm to iteratively update all model parameters with the summed objectives (the clustering loss is weighed by λ).

Complexity. In the E-Step of the algorithm, $q(t_k|\boldsymbol{z}_i^{(w)})$ is computed for every latent representation over each topic, resulting in an $O(NKr')$ complexity per iteration. The M-Step updates DNN parameters whose complexity is related to the number of parameters in the model and the optimization method.

Algorithm 1: TopClus Training.

Input: $\mathcal{D}$: Text corpus; M: PLM; K: Number of topics.
Parameter: $\boldsymbol{A}$: Attention mechanism parameters; f, g: Encoding/decoding functions; $\boldsymbol{T}$: Topic embeddings.
Hyperparameter: E: Training epochs; λ: Clustering loss weight.
Output: Topic-word distributions $p\left(z_i^{(w)}|t_k\right)$; document-topic distributions $p\left(t_k|z^{(d)}\right)$.

$f, g \leftarrow \arg\min_{f,g} \mathcal{L}_{\text{pre}}$; // Pretrain f, g via Eq. (6);
$\boldsymbol{T} = \boldsymbol{t}_k|_{k=1}^{K} \leftarrow$ Initialize with K-means on $\mathbb{S}^{r'-1}$;
for $j \in [1, 2, \ldots, E]$ **do**
  // E-Step: Update cluster assignment estimation;
  $q\left(t_k|\boldsymbol{z}_i^{(w)}\right) \leftarrow$ Eq. (3);
  // M-Step: Update model parameters;
  $\boldsymbol{A}, f, g, \boldsymbol{T} \leftarrow \arg\min_{\boldsymbol{A},f,g,\boldsymbol{T}} \left(\lambda\mathcal{L}_{\text{clus}} + \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{pre}}\right)$;
return $p\left(z_i^{(w)}|t_k\right), p\left(t_k|z^{(d)}\right)$;

4 EXPERIMENTS

4.1 Experiment Setup

Settings. We use two benchmark datasets in different domains with long/short texts for evaluation: (1) The New York Times annotated corpus (**NYT**) [58]; and (2) The Yelp Review Challenge dataset (**Yelp**). The dataset statistics can be found in Table 4. The implementation details and parameters of TopClus are shown in Appendix C. For both datasets, we set the number of topics $K = 100$ for all compared methods.

Compared Methods. We compare TopClus with the following strong baselines:

- LDA [11]: LDA is the standard topic model that learns topic-word and document-topic distributions by modeling the generative process of the corpus.
- CorEx [19]: CorEx does not rely on generative assumptions and learns maximally informative topics measured by total correlation.
- ETM [17]: ETM models word topic correlations via distributed representations to improve the expressiveness of topic models.
- BERTopic [23]: BERTopic first clusters document embeddings from BERT and then uses TF-IDF to extract topic representative words, which does not leverage word embeddings from PLMs.

4.2 Topic Discovery Evaluation

Evaluation Metrics. We evaluate the quality of the topics from two aspects: *topic coherence* and *topic diversity*. Good topic results should be both coherent for humans to interpret and diverse to cover more information about the corpus. We evaluate the effectiveness of document-level topic modeling by document clustering.

For topic coherence, we use three metrics including both human and automatic evaluations:

- UMass [49]: UMass computes the log-conditional probability of every top word in each topic given every other top word that has a higher order in the ranking of that topic. The probability is computed based on document-level word co-occurrence.
- UCI [51]: UCI computes the average pointwise mutual information of all pairs of top words in each topic. The word co-occurrence counts are derived using a sliding window of size 10.
- Intrusion: Given the top terms of a topic, we inject an intrusion term that is randomly chosen from another topic. Then a human evaluator is asked to identify the intruded term. The more coherent the top terms are, the more likely an evaluator can correctly identify the fake term, and thus we compute the ratio of correctly identified intrusion instances as the topic coherence score given by the intrusion test. The topics from all compared methods are randomly shuffled during evaluation to avoid the bias of human evaluators.

For topic diversity, we report the percentage of unique words in the top words of all topics following the definition in [17].

Qualitative Evaluation. We randomly select several ground truth topics from both datasets, and manually match the most relevant topic generated by all methods. Table 1 shows the top-5 words per topic. All methods are able to generate relevant topics to the ground truth ones. LDA and CorEx results contain noises that are semantically irrelevant to the topic; ETM improves LDA by incorporating word embeddings, but still generates slightly off-topic terms; BERTopic also has noisy terms in the results, as it uses TF-IDF metrics without exploiting word representations from BERT for obtaining top words. TopClus consistently outputs coherent and meaningful topics.

Quantitative Evaluation. We report the performance of all methods under the four metrics in Table 2. Overall, the quantitative evaluation coincides with the previous qualitative results. TopClus generates not only the most coherent but also diverse topics, under both automatic and human evaluations.

4.3 Document Clustering Evaluation

Evaluation Metrics. We use the learned latent document embedding $\boldsymbol{z}^{(d)}$ as the feature to K-Means for obtaining document clusters, then we report the Normalized Mutual Information (NMI) score between the clustering results and the ground truth document labels.

We use the topic label set (*e.g.*, politics, sports) and location label set (*e.g.*, United States, China) on the **NYT** dataset. The detailed label statistics can be found in [39]. On the two label sets, the document-topic distribution learned by TopClus consistently yields the best clustering results among all methods as shown in Table 3.

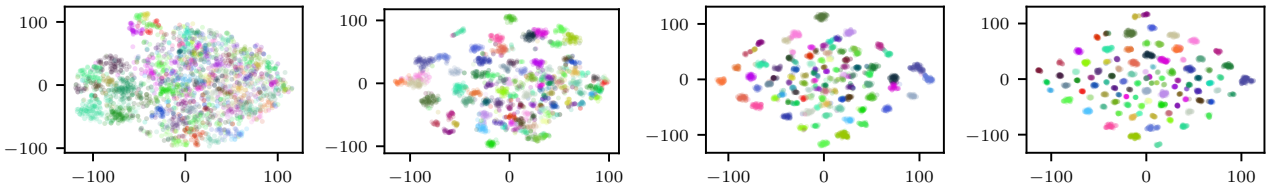
4.4 Study of TopClus Training

Joint Learning Latent Space and Clustering Improves Topic Quality. Figure 5 shows the improvement in topic quality (measured by both intrusion test score and topic diversity) and document clustering performance during TopClus training. At epoch 0, the result is equivalent to first applying dimensionality reduction (*i.e.*, pretraining autoencoder with $\mathcal{L}_{\text{pre}}$) and then clustering with K-means, the "naive approach" mentioned in the second paragraph of Section 3.2. Its inferior performance confirms that conducting the two steps separately does not generate satisfactory topics. Topic quality and document clustering performance improve when the model is trained longer, showing that joint latent space learning and clustering indeed helps generate coherent and distinctive topics.

Visualization. To intuitively understand how TopClus jointly learns the latent space structure and performs clustering, we visualize the learned latent embeddings at different training epochs in Figure 4.

Table 1: Qualitative evaluation of topic discovery. We select several ground truth topics and manually find the most relevant topic generated by all methods. Words not strictly belonging to the corresponding topic are italicized and underlined.

Methods	NYT					Yelp				
	Topic 1 (sports)	Topic 2 (politics)	Topic 3 (research)	Topic 4 (france)	Topic 5 (japan)	Topic 1 (positive)	Topic 2 (negative)	Topic 3 (vegetables)	Topic 4 (fruits)	Topic 5 (seafood)
LDA	olympic	*mr*	*said*	french	japanese	amazing	loud	spinach	mango	fish
	year	bush	report	*union*	tokyo	*really*	awful	carrots	strawberry	*roll*
	said	president	evidence	*germany*	*year*	*place*	*sunday*	greens	*vanilla*	salmon
	games	white	findings	*workers*	matsui	phenomenal	*like*	salad	banana	*fresh*
	team	house	defense	paris	*said*	pleasant	slow	*dressing*	*peanut*	*good*
CorEx	baseball	house	possibility	french	japanese	great	*even*	garlic	strawberry	shrimp
	championship	white	challenge	*italy*	tokyo	friendly	bad	tomato	*caramel*	*beef*
	playing	support	reasons	paris	*index*	*atmosphere*	mean	onions	*sugar*	crab
	fans	*groups*	*give*	francs	osaka	love	cold	*toppings*	fruit	*dishes*
	league	*member*	planned	jacques	*electronics*	favorite	*literally*	*slices*	mango	*salt*
ETM	olympic	government	approach	french	japanese	nice	disappointed	avocado	strawberry	fish
	league	national	problems	*students*	*agreement*	worth	cold	*greek*	mango	shrimp
	national	*plan*	experts	paris	tokyo	*lunch*	*review*	salads	*sweet*	lobster
	basketball	public	*move*	*german*	*market*	recommend	*experience*	spinach	*soft*	crab
	athletes	support	*give*	*american*	*european*	friendly	bad	tomatoes	*flavors*	*chips*
BERTopic	swimming	bush	researchers	french	japanese	awesome	horrible	tomatoes	strawberry	lobster
	freestyle	democrats	scientists	paris	tokyo	*atmosphere*	*quality*	avocado	mango	crab
	popov	white	cases	lyon	ufj	friendly	disgusting	*soups*	*cup*	shrimp
	gold	bushs	*genetic*	*minister*	*company*	*night*	disappointing	kale	lemon	oysters
	olympic	house	study	*billion*	yen	good	*place*	cauliflower	banana	*amazing*
TopClus	athletes	government	hypothesis	french	japanese	good	tough	potatoes	strawberry	fish
	medalist	ministry	methodology	seine	tokyo	best	bad	onions	lemon	octopus
	olympics	bureaucracy	possibility	toulouse	osaka	friendly	painful	tomatoes	apples	shrimp
	tournaments	politicians	criteria	marseille	hokkaido	cozy	frustrating	cabbage	grape	lobster
	quarterfinal	electoral	assumptions	paris	yokohama	casual	brutal	mushrooms	peach	crab



(a) Epoch 0. (b) Epoch 2. (c) Epoch 4. (d) Epoch 8.

Figure 4: Visualization using t-SNE of 3,000 randomly sampled latent word embeddings during training. Embeddings assigned to the same cluster are in the same color. The latent space gradually exhibits distinctive and balanced cluster structure.

Before the training starts (epoch 0), the latent embedding space does not have clear cluster structures, just like the original space. During training, the latent embeddings are becoming well-separated and the cluster structure is gradually more distinctive and balanced, resulting in coherent and diverse topics.

5 RELATED WORK

5.1 Topic Models

Topic models aim to discover underlying topics and semantic structures from text corpora. Despite extensive studies of topic models following LDA, most approaches suffer from one or more of the following limitations: (1) *The "bag-of-words" assumption* that presumes words in the document are generated independently from each other. (2) *The reliance on local corpus statistics*, which could be improved by leveraging general knowledge such as pretrained language models [16]. (3) *The intractable posterior* that requires approximation techniques during model inference.

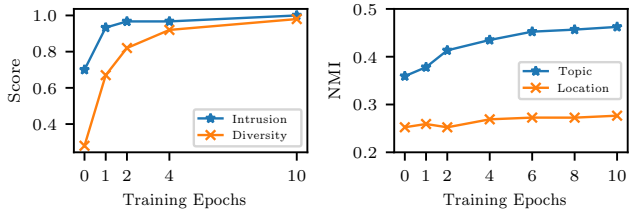
Topic modeling approaches can be divided into three major categories: (1) *LDA-based approaches* use pLSA [26] or LDA [11] as the backbone. The idea is to characterize documents as mixtures of latent topics and represent each topic as a distribution over words. Popular models in this category include Hierarchical LDA [22], Dynamic Topic Models [8], Correlated Topic Models [7], Pachinko

Table 2: Quantitative evaluation of topic discovery. We evaluate all methods with three topic coherence metrics UCI, UMAss and Intrusion (Int.) and a topic diversity (Div.) metric. Higher score means better for all metrics. We do not report Div. for CorEx because it requires topics to have non-overlapping words by design.

Methods	NYT				Yelp			
	UMass	UCI	Int.	Div.	UMass	UCI	Int.	Div.
LDA	-3.75	-1.76	0.53	0.78	-4.71	-2.47	0.47	0.65
CorEx	-3.83	-0.96	0.77	-	-4.75	-1.91	0.43	-
ETM	-2.98	-0.98	0.67	0.30	-3.04	-0.33	0.47	0.16
BERTopic	-3.78	-0.51	0.70	0.61	-6.37	-2.05	0.73	0.36
TopClus	**-2.67**	**-0.45**	**0.93**	**0.99**	**-1.35**	**-0.27**	**0.87**	**0.96**

Table 3: Document clustering NMI scores on NYT (Topic/Location label set).

LDA	CorEx	ETM	BERTopic	TopClus
0.39/0.20	0.29/0.20	0.41/0.21	0.26/0.22	**0.46/0.28**



(a) Topic Quality. (b) Document Clustering.

Figure 5: Study of TopClus training on NYT. We show (a) topic coherence measured by intrusion test and topic diversity and (b) document clustering NMI scores over training.

Allocation [34], Supervised Topic Models [10] and Labeled LDA [55]. Most of these models suffer from all three limitations mentioned above. (2) *Topic models with word embeddings* have been broadly studied after word2vec [48] came out. The common strategy is to convert the discrete text into continuous representations of embeddings, and then adapt LDA to generate real-valued data. Such kind of models include Gaussian LDA [15], Spherical Hierarchical Dirichlet Process [3] and WELDA [12]. There are some other strategies combining topic modeling and word embedding. For example, LFTM [52] models a mixture of the multinomial distribution and a link function between word and topic embeddings. TWE [35] uses pretrained topic structures to learn topic embeddings and improve word embeddings. Although these models consider word embeddings to make up for the "bag-of-words" assumption, they are not equipped with general knowledge from pretrained language models. (3) *Neural topic models* are inspired by deep generative models such as VAE [32]. NVDM [47] encodes documents with variational posteriors in the latent topic space. Instead, ProdLDA [60] proposes a Laplace approximation of Dirichlet distributions to enable reparameterization. Although these neural topic models improve the posterior approximation with neural networks, they still do not utilize general knowledge such as pretrained language models.

5.2 Pretrained Language Models

Bengio et al. [4] propose the Neural Network Language Model which pioneers the study of modern word embedding. Mikolov et al. [48] introduce two architectures, CBOW and Skip-Gram, to capture local context semantics of each word.

Although word embeddings have been shown effective in NLP tasks, they are context-independent. Meanwhile, most NLP tasks are beyond word-level, thus it is beneficial to derive word semantics based on specific contexts. Therefore, contextualized PLMs are widely studied recently. For example, BERT [16] and RoBERTa [36] adopt masked token prediction as the pretraining task to leverage bidirectional contexts. XLNet [67] proposes a new pretraining objective on a random permutation of input sequences. ELECTRA [14], COCO-LM [43] and AMOS [44] use a generator to replace some tokens of a sequence and predict whether a token is replaced given its surrounding context. For more related studies, one can refer to a recent survey [54]. There have been a few recent studies that attempt to incorporate PLM representations into the topic modeling framework for different purposes [6, 13, 24, 28, 61]. By contrast, our approach features a latent space clustering framework that leverages the inherent representations of PLMs for topic discovery without following the topic modeling setup.

6 CONCLUSION

We explore a new alternative to topic models via latent space clustering of PLM representations. We first analyze the challenges of using PLM embeddings to generate topic structures, and then propose a joint latent space learning and clustering approach TopClus to address the identified challenges. TopClus generates coherent and distinctive topics and outperforms strong topic modeling baselines in both topic quality and topical document representations. We also conduct studies to provide insights on how the joint learning setup in TopClus gradually improves the generated topic quality.

TopClus is conceptually simple which facilitates future extensions such as integrating with new PLMs and advanced clustering techniques. TopClus may also be extended to perform hierarchical topic discovery, perhaps via top-down clustering in the latent space. Other related tasks like taxonomy construction [30] and weakly-supervised text classification [29, 41, 42, 45, 68] may benefit from the coherent and distinctive topics generated by TopClus.

ACKNOWLEDGMENTS

Research was supported in part by US DARPA KAIROS Program No. FA8750-19-2-1004, SocialSim Program No. W911NF-17-C-0099, and INCAS Program No. HR001121C0165, National Science Foundation IIS-19-56151, IIS-17-41317, and IIS 17-04532, and the Molecule Maker Lab Institute: An AI Research Institutes program supported by NSF under Award No. 2019897. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and do not necessarily represent the views, either expressed or implied, of DARPA or the U.S. Government. Yu Meng is supported by the Google PhD Fellowship. We thank anonymous reviewers for valuable and insightful feedback.

REFERENCES

[1] Hagai Attias. 2000. A variational Bayesian framework for graphical models. *NIPS*.

[2] Arindam Banerjee, Inderjit S. Dhillon, Joydeep Ghosh, and Suvrit Sra. 2005. Clustering on the Unit Hypersphere using von Mises-Fisher Distributions. *J. Mach. Learn. Res.* (2005).

[3] Kayhan Batmanghelich, Ardavan Saeedi, Karthik Narasimhan, and Sam Gershman. 2016. Nonparametric spherical topic modeling with word embeddings. In *ACL*.

[4] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *JMLR* (2003).

[5] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. 1999. When is "nearest neighbor" meaningful?. In *ICDT*.

[6] Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2021. Pre-training is a Hot Topic: Contextualized Document Embeddings Improve Topic Coherence. In *ACL*.

[7] David Blei and John Lafferty. 2006. Correlated topic models. In *NIPS*.

[8] David M Blei and John D Lafferty. 2006. Dynamic topic models. In *ICML*.

[9] David M Blei, John D Lafferty, et al. 2007. A correlated topic model of science. *The annals of applied statistics* (2007).

[10] David M Blei and Jon D Mcauliffe. 2008. Supervised topic models. In *NIPS*.

[11] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. In *NIPS*.

[12] Stefan Bunk and Ralf Krestel. 2018. WELDA: Enhancing Topic Models by Incorporating Local Word Context. In *JCDL*.

[13] Yatin Chaudhary, Pankaj Gupta, Khushbu Saxena, Vivek Kulkarni, Thomas A. Runkler, and Hinrich Schütze. 2020. TopicBERT for Energy Efficient Document Classification. In *EMNLP Findings*.

[14] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555* (2020).

[15] Rajarshi Das, Manzil Zaheer, and Chris Dyer. 2015. Gaussian LDA for topic models with word embeddings. In *ACL*.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*.

[17] Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *TACL* (2020).

[18] J. R. Firth. 1957. A synopsis of linguistic theory 1930-55. (1957).

[19] Ryan J Gallagher, Kyle Reing, David Kale, and Greg Ver Steeg. 2017. Anchored correlation explanation: Topic modeling with minimal domain knowledge. *TACL* (2017).

[20] Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models. In *EMNLP Findings*.

[21] Siddharth Gopal and Yiming Yang. 2014. Von Mises-Fisher Clustering Models. In *ICML*.

[22] Thomas L Griffiths, Michael I Jordan, Joshua B Tenenbaum, and David M Blei. 2004. Hierarchical topic models and the nested Chinese restaurant process. In *NIPS*.

[23] Maarten Grootendorst. 2020. BERTopic: Leveraging BERT and c-TF-IDF to create easily interpretable topics. *Zenodo: 10.5281/zenodo.4430182* (2020).

[24] Pankaj Gupta, Yatin Chaudhary, and Hinrich Schütze. 2021. Multi-source Neural Topic Modeling in Multi-view Embedding Spaces. In *NAACL*.

[25] Geoffrey E Hinton and Richard S Zemel. 1993. Autoencoders, Minimum Description Length and Helmholtz Free Energy. In *NIPS*.

[26] Thomas Hofmann. 2004. Latent semantic models for collaborative filtering. *ACM TOIS* (2004).

[27] Kurt Hornik. 1991. Approximation capabilities of multilayer feedforward networks. *Neural networks* (1991).

[28] Alexander Miserlis Hoyle, Pranav Goel, and Philip Resnik. 2020. Improving Neural Topic Models Using Knowledge Distillation. In *EMNLP*.

[29] Jiaxin Huang, Yu Meng, Fang Guo, Heng Ji, and Jiawei Han. 2020. Weakly-Supervised Aspect-Based Sentiment Analysis via Joint Aspect-Sentiment Topic Embedding. In *EMNLP*.

[30] Jiaxin Huang, Yiqing Xie, Yu Meng, Yunyi Zhang, and Jiawei Han. 2020. CoRel: Seed-Guided Topical Taxonomy Construction by Concept Learning and Relation Transferring. In *KDD*.

[31] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

[32] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[33] Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the Sentence Embeddings from Pre-trained Language Models. In *EMNLP*.

[34] Wei Li and Andrew McCallum. 2006. Pachinko allocation: DAG-structured mixture models of topic correlations. In *ICML*.

[35] Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2015. Topical word embeddings. In *AAAI*.

[36] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[37] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* (1982).

[38] Xinyao Ma, Maarten Sap, Hannah Rashkin, and Yejin Choi. 2020. PowerTransformer: Unsupervised Controllable Revision for Biased Language Correction. In *EMNLP*.

[39] 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*.

[40] Yu Meng, Jiaxin Huang, Guangyuan Wang, Chao Zhang, Honglei Zhuang, Lance M. Kaplan, and Jiawei Han. 2019. Spherical Text Embedding. In *NeurIPS*.

[41] Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018. Weakly-Supervised Neural Text Classification. In *CIKM*.

[42] Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2019. Weakly-Supervised Hierarchical Text Classification. In *AAAI*.

[43] Yu Meng, Chenyan Xiong, Payal Bajaj, Saurabh Tiwary, Paul Bennett, Jiawei Han, and Xia Song. 2021. COCO-LM: Correcting and contrasting text sequences for language model pretraining. In *NeurIPS*.

[44] Yu Meng, Chenyan Xiong, Payal Bajaj, Saurabh Tiwary, Paul Bennett, Jiawei Han, and Xia Song. 2022. Pretraining Text Encoders with Adversarial Mixture of Training Signal Generators. In *ICLR*.

[45] Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. 2020. Text Classification Using Label Names Only: A Language Model Self-Training Approach. In *EMNLP*.

[46] Yu Meng, Yunyi Zhang, Jiaxin Huang, Yu Zhang, Chao Zhang, and Jiawei Han. 2020. Hierarchical Topic Mining via Joint Spherical Tree and Text Embedding. In *KDD*.

[47] Yishu Miao, Lei Yu, and Phil Blunsom. 2016. Neural variational inference for text processing. In *ICML*.

[48] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS*.

[49] David Mimno, Hanna Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. Optimizing semantic coherence in topic models. In *EMNLP*.

[50] Radford M Neal. 1993. *Probabilistic inference using Markov chain Monte Carlo methods*.

[51] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In *NAACL*.

[52] Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. 2015. Improving topic models with latent feature word representations. *TACL* (2015).

[53] Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W. Black. 2018. Style Transfer Through Back-Translation. In *ACL*.

[54] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained Models for Natural Language Processing: A Survey. *arXiv preprint arXiv:2003.08271* (2020).

[55] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D Manning. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *EMNLP*.

[56] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP*.

[57] Tim Salimans, Diederik Kingma, and Max Welling. 2015. Markov chain monte carlo and variational inference: Bridging the gap. In *ICML*.

[58] Evan Sandhaus. 2008. The New York Times Annotated Corpus.

[59] Suzanna Sia, Ayush Dalmia, and Sabrina J Mielke. 2020. Tired of Topic Models? Clusters of Pretrained Word Embeddings Make for Fast and Good Topics Too!. In *EMNLP*.

[60] Akash Srivastava and Charles Sutton. 2017. Autoencoding Variational Inference For Topic Models. In *ICLR*.

[61] Laure Thompson and David Mimno. 2020. Topic Modeling with Contextualized Word Representation Clusters. *ArXiv* abs/2010.12626 (2020).

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NIPS*.

[63] Dingding Wang, Shenghuo Zhu, Tao Li, and Yihong Gong. 2009. Multi-document summarization using sentence-based topic models. In *ACL*.

[64] Rui Wang, Xuemeng Hu, Deyu Zhou, Yulan He, Yuxuan Xiong, Chenchen Ye, and Haiyang Xu. 2020. Neural Topic Modeling with Bidirectional Adversarial Training. In *ACL*.

[65] Xuerui Wang, Andrew McCallum, and Xing Wei. 2007. Topical n-grams: Phrase and topic discovery, with an application to information retrieval. In *ICDM*.

[66] Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *ICML*.

[67] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *NeurIPS*.

[68] Yu Zhang, Shweta Garg, Yu Meng, Xiusi Chen, and Jiawei Han. 2022. MotifClass: Weakly Supervised Text Classification with Higher-order Metadata Information. In *WSDM*.

A ETHICAL CONSIDERATIONS

PLMs have been shown to contain potential biases [53] which may be carried to the downstream applications. Our work focuses on using representations from PLMs for discovery of topics in a target corpus, and the results will be related to both the PLMs and the corpus statistics. We suggest applying our method together with bias reduction and correction techniques for PLMs [20, 38] and filtering out biased contents in the target corpus to mitigate potential risks and harms.

B PROOF OF THEOREM 2.1

Proof. The MLM objective of BERT trains contextualized word embeddings to predict the masked tokens in a sequence. Formally, given an input sequence $\boldsymbol{d} = [w_1, w_2, \ldots, w_n]$, a random subset of tokens (*e.g.*, usually 15% from the original sequence) $\mathcal{M}$ is selected and replaced with `[MASK]` symbols. Then the BERT encoder maps the masked sequence $\hat{\boldsymbol{d}}$ to a sequence of contextualized representations $[\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_n]$ where $\boldsymbol{h}_i \in \mathbb{R}^r$ ($r = 768$ in the BERT base model). BERT is trained by maximizing the log-probability of correctly predicting every masked word with a Softmax layer over the vocabulary V:

$$\max_{\boldsymbol{e},\boldsymbol{h},\boldsymbol{b}} \sum_{w_i \in \mathcal{M}} \log \frac{\exp\left(\boldsymbol{e}_{w_i}^\top \boldsymbol{h}_i + b_{w_i}\right)}{\sum_{j=1}^{|V|} \exp\left(\boldsymbol{e}_{w_j}^\top \boldsymbol{h}_i + b_{w_j}\right)}, \tag{7}$$

where $\boldsymbol{e}_{w_i} \in \mathbb{R}^r$ is the token embedding; and $b_{w_i} \in \mathbb{R}$ is a bias value for token w_i.

Next, we construct a multivariate GMM parameterized by the learned token embeddings $\boldsymbol{e}$ and bias vector $\boldsymbol{b}$ of BERT, and we show that the MLM objective (Eq. (7)) optimizes the posterior probability of contextualized embeddings $\boldsymbol{h}$ generated from this GMM. We consider the following GMM with $|V|$ mixture components, where each component i is a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_i, \Sigma_i)$ with mean vector $\boldsymbol{\mu}_i \in \mathbb{R}^r$, covariance matrix $\Sigma_i \in \mathbb{R}^{r \times r}$ and mixture weight π_i (*i.e.*, the prior probability) defined as follows:

$$\boldsymbol{\mu}_i \coloneqq \Sigma\, \boldsymbol{e}_{w_i},\ \Sigma_i \coloneqq \Sigma,\ \pi_i \coloneqq \frac{\exp\left(\frac{1}{2}\boldsymbol{e}_{w_i}^\top \Sigma\, \boldsymbol{e}_{w_i} + b_{w_i}\right)}{\sum_{1 \le j \le |V|} \exp\left(\frac{1}{2}\boldsymbol{e}_{w_j}^\top \Sigma\, \boldsymbol{e}_{w_j} + b_{w_j}\right)},$$

where all components share the same covariance matrix Σ.

The contextualized embeddings $\boldsymbol{h}_i$ are generated by first sampling a token w_i according to the prior distribution, and then sampling from the Gaussian distribution corresponding to w_i, as follows:

$$w_i \sim \text{Categorical}(\boldsymbol{\pi}),\ \boldsymbol{h}_i \sim \mathcal{N}\left(\Sigma\, \boldsymbol{e}_{w_i},\ \Sigma\right).$$

Based on the above generative process, the prior probability of token w_i is

$$p(w_i) = \pi_i = \frac{\exp\left(\frac{1}{2}\boldsymbol{e}_{w_i}^\top \Sigma\, \boldsymbol{e}_{w_i} + b_{w_i}\right)}{\sum_{j=1}^{|V|} \exp\left(\frac{1}{2}\boldsymbol{e}_{w_j}^\top \Sigma\, \boldsymbol{e}_{w_j} + b_{w_j}\right)},$$

and the likelihood of generating $\boldsymbol{h}_i$ given w_i is

$$p\left(\boldsymbol{h}_i | w_i\right) = \frac{\exp\left(-\frac{1}{2}(\boldsymbol{h}_i - \Sigma\, \boldsymbol{e}_{w_i})^\top \Sigma^{-1} \left(\boldsymbol{h}_i - \Sigma\, \boldsymbol{e}_{w_i}\right)\right)}{(2\pi)^{r/2}|\Sigma|^{1/2}}.$$

The posterior probability can be obtained using the Bayes rule:

$$p(w_i | \boldsymbol{h}_i) = \frac{p\left(\boldsymbol{h}_i | w_i\right) p(w_i)}{\sum_{j=1}^{|V|} p\left(\boldsymbol{h}_i | w_j\right) p(w_j)},$$

where the numerator $p\left(\boldsymbol{h}_i | w_i\right) p(w_i)$ is

$$\frac{\exp\left(-\frac{1}{2}\boldsymbol{h}_i^\top \Sigma^{-1} \boldsymbol{h}_i + \boldsymbol{h}_i^\top \boldsymbol{e}_{w_i} - \cancel{\frac{1}{2}\boldsymbol{e}_{w_i}^\top \Sigma\, \boldsymbol{e}_{w_i}} + \cancel{\frac{1}{2}\boldsymbol{e}_{w_i}^\top \Sigma\, \boldsymbol{e}_{w_i}} + b_{w_i}\right)}{(2\pi)^{r/2}|\Sigma|^{1/2} \sum_{j=1}^{|V|} \exp\left(\frac{1}{2}\boldsymbol{e}_{w_j}^\top \Sigma\, \boldsymbol{e}_{w_j} + b_{w_j}\right)}.$$

The terms in the denominator are in a similar form and many common factors between the numerator and the denominator cancel out. Finally, the above posterior probability is simplified as:

$$p(w_i | \boldsymbol{h}_i) = \frac{\exp\left(\boldsymbol{e}_{w_i}^\top \boldsymbol{h}_i + b_{w_i}\right)}{\sum_{j=1}^{|V|} \exp\left(\boldsymbol{e}_{w_j}^\top \boldsymbol{h}_i + b_{w_j}\right)},$$

which is precisely the probability maximized by the MLM objective (Eq. (7)). Therefore, the MLM pretraining objective of BERT assumes that the contextualized representations are generated from a $|V|$-component GMM. □

C IMPLEMENTATION DETAILS AND PARAMETERS

We preprocess the corpora by discarding infrequent words that appear less than 5 times. We use the default hyperparameters of baseline methods. The hyperparameters of TopClus are set as follows: Latent space dimension $r' = 100$; training epochs $E = 20$; clustering loss weight $\lambda = 0.1$; DNN hidden dimensions are 500-500-1000 for learning f and 1000-500-500 for learning g; the shared concentration parameter of topic vMF distributions $\kappa = 10$. We use the BERT [16] base model to obtain pretrained embeddings, and use Adam [31] with $5e-4$ learning rate to optimize the DNNs with batch size 32. When computing the generic document as an average of word embeddings in Eq. (5), we only use the words that are nouns, verbs, or adjectives because they are usually the topic-indicative ones.

Table 4: Dataset statistics.

Corpus	# documents	# words/doc.	Vocabulary
NYT	31,997	690	25,903
Yelp	29,280	114	11,419