Multi-source Neural Topic Modeling in Multi-view Embedding Spaces

***Pankaj Gupta[1], *Yatin Chaudhary[1,2], Hinrich Schütze[2]**
[1]DRIMCo GmbH Munich, Germany
[2]CIS, University of Munich (LMU) Munich, Germany
`info@drimco.net` | `yatin.chaudhary@drimco.net`

Abstract

Though word embeddings and topics are complementary representations, several past works have only used pretrained word embeddings in (neural) topic modeling to address data sparsity in short-text or small collection of documents. This work presents a novel neural topic modeling framework using multi-view embedding spaces: (1) pretrained topic-embeddings, and (2) pretrained word-embeddings (context-insensitive from Glove and context-sensitive from BERT models) *jointly* from one or *many sources* to improve topic quality and better deal with polysemy. In doing so, we first build respective pools of pretrained topic (i.e., `TopicPool`) and word embeddings (i.e., `WordPool`). We then identify one or more relevant source domain(s) and transfer knowledge to guide meaningful learning in the sparse target domain. Within neural topic modeling, we quantify the quality of topics and document representations via generalization (perplexity), interpretability (topic coherence) and information retrieval (IR) using short-text, long-text, small and large document collections from news and medical domains. Introducing the multi-source multi-view embedding spaces, we have shown state-of-the-art neural topic modeling using 6 source (high-resource) and 5 target (low-resource) corpora.

1 Introduction

Probabilistic topic models, such as LDA (Blei et al., 2003), Replicated Softmax (RSM) (Salakhutdinov and Hinton, 2009) and Document Neural Autoregressive Distribution Estimator (DocNADE) (Larochelle and Lauly, 2012) are often used to extract topics from text collections and learn latent document representations to perform natural language processing tasks, such as information retrieval (IR). Though they have been shown to be powerful in modeling large text corpora, the topic modeling (TM) still remains challenging especially in the sparse-data setting, especially for the cases where word co-occurrence data is insufficient, e.g., on short-text or a corpus of few documents. It leads to a poor quality of topics and representations.

Topic	Topic Words	Topic Label
Z_1 ($\mathcal{S}^1$)	profit, growth, stocks, **apple**, fall, consumer, buy, billion, shares	*Trading*
Z_2($\mathcal{S}^2$)	smartphone, ipad, **apple**, app, iphone, devices, phone, tablet	*Product Line*
Z_3 ($\mathcal{S}^3$)	microsoft, mac, linux, ibm, ios, **apple**, xp, windows, software	*Operating System*
Z_4 ($\mathcal{T}$)	**apple**, talk, computers, shares, disease, driver, electronics, profit, ios	?

Table 1: Coherent (Z_1-Z_3) vs Incoherent (Z_4) topics from high-resource ($\mathcal{S}^1$-$\mathcal{S}^3$) and low-resource ($\mathcal{T}$) texts

To address data sparsity issues, several works (Das et al., 2015; Nguyen et al., 2015; Gupta et al., 2019a, 2020) have introduced external knowledge in traditional topic models, e.g., incorporating word embeddings obtained from Glove (Pennington et al., 2014) or word2vec (Mikolov et al., 2013a). However, no prior work in topic modeling has employed multi-view embedding spaces: (1) *pretrained topics*, i.e., topical embeddings obtained from large document collections, and (2) *pretrained contextualized word embeddings* from large-scale language models like BERT (Devlin et al., 2019).

Though topics and word embeddings are complementary in how they represent the meaning, they are distinctive in how they learn from word occurrences observed in text corpora. A topic model (Blei et al., 2003) is a statistical tool to infers topic distributions across a collection of documents and assigns a topic to each word occurrence, where the assignment is equally dependent on all other words appearing in the same document. Therefore, a topic has a *global view* representing semantic structures hidden in document collection. On other hand, word embeddings have primarily *local view* in the

* : equal contribution

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 4205–4217
June 6–11, 2021. ©2021 Association for Computational Linguistics

sense that they are learned based on local collocation pattern in a text corpus, where the representation of each word often depends on a local context window (Mikolov et al., 2013b) or is a function of its sentence(s) (Peters et al., 2018). Consequently, they are not aware of the thematic structures underlying the document collection. Additionally, recent studies (Peters et al., 2018; Devlin et al., 2019; Liu et al., 2019) have shown a reasonable success in several NLP applications by employing pretrained contextualized word embeddings, where the representation of a word is different in different contexts (i.e., context-sensitive). In context of this work, the representations due to global and local (context-sensitive or context-insensitive) views together are referred as *multi-view* embeddings.

For example in Table 1, consider four topics (Z_1-Z_4) of different domains where the topics (Z_1-Z_3) are respectively obtained from three different high-resource source ($\mathcal{S}^1$-$\mathcal{S}^3$) domains whereas Z_4 from a low-resource target domain $\mathcal{T}$ (especially in the data-sparsity settings). Observe that the topics about *Trading* (Z_1), *Product Line* (Z_2) and *Operating System* (Z_3) are coherent and and represent meaningful semantics at document-level via lists of topic words. However in sparse-data settings, the topic Z_4 discovered is incoherent (noisy) and it is difficult to infer meaningful document semantics.

Unlike the topics, word embeddings (context-insensitive) encode syntactic and semantic relatedness in fine-granularity and therefore, do not capture thematic structures. For instance, the top-5 nearest neighbors (NN) of *apple* (below) in word embedding (Mikolov et al., 2013b) space suggest that it refers to a *fruit* and do not express any topical information (e.g., *Trading*, *Product Line* or *Health*) in the corpora. Similarly given the NN of the word *fall*, it is difficult to infer its association with document-level semantics, e.g., *Trading* as expressed by Z_1 in topic-embedding space.

apple $\overset{\text{NN}}{\Longrightarrow}$ *apples, pear, fruit, berry, pears, strawberry*

fall $\overset{\text{NN}}{\Longrightarrow}$ *falling, falls, drop, tumble, rise, plummet*

Therefore, topic and word embedding spaces encode complementary semantics. Different to context-insensitive word embeddings, the word *apple* is referring to an organization and contextualized by different topical semantics respectively in the three sources $\mathcal{S}^1$-$\mathcal{S}^3$. Thus, it arises the need for context-sensitive embeddings in topic modeling.

Contribution (1) Multi-view Neural Topic Modeling using pretrained word and topic embeddings: To alleviate the data sparsity issues, it is the **first work** in unsupervised neural topic modeling (NTM) within transfer learning paradigm that employs *multi-view embedding spaces* via: (a) *Global-view Transfer* (*GVT*): Pretrained topic embeddings instead of using word embeddings exclusively, and (b) *Multi-view Transfer* (*MVT*): Pretrained topic and word embeddings (context-insensitive from Glove (Pennington et al., 2014) and context-sensitive from large-scale language models such as BERT (Devlin et al., 2019) *jointly* to address data sparsity and polysemy issues.

Notation	Description
LVT, GVT	Local-view Transfer, Global-view Transfer
MVT, MST	Multi-view Transfer, Multi-source Transfer
$\mathcal{T}$, $\mathcal{S}$	A target domain, a set of source domains
$\mathbf{v}$, k, $\mathcal{L}$	An input document, kth source, loss
K, D	Vocabulary size, document size
E, H	Word embedding dimension, #topics
$\mathbf{W} \in \mathbb{R}^{H\times K}$	Encoding matrix of DocNADE in $\mathcal{T}$
$\mathbf{U} \in \mathbb{R}^{K\times H}$	Decoding matrix of DocNADE
λ^k	Degree of relevance of $\mathbf{E}^k$ in $\mathcal{T}$
γ^k	Degree of imitation of $\mathbf{Z}^k$ by $\mathbf{W}$
$\mathbf{E}^k \in \mathbb{R}^{E\times K}$	Word embeddings of kth source
$\mathbf{Z}^k \in \mathbb{R}^{H\times K}$	Topic embeddings of kth source
$\mathbf{A}^k \in \mathbb{R}^{H\times H}$	Topic-alignment in $\mathcal{T}$ and $\mathbf{Z}^k$
$\mathbf{b} \in \mathbb{R}^K$, $\mathbf{c} \in \mathbb{R}^H$	Visible-bias, hidden-bias
DC	Document Collection

Table 2: Description of the notations used in this work

Contribution (2) Multi-source Multi-view Neural Topic Modeling: A single source of prior knowledge is often insufficient due to incomplete and non-overlapping domain information required by a target domain. Therefore, there is a need to leverage multiple sources of prior knowledge, dealing with domain-shifts (Cao et al., 2010) among the target and sources. In doing so, we first learn word and topic representations on multiple source domains to build `WordPool` and `TopicPool`, respectively and then perform *multi-view* and *multi-source* transfer learning in neural topic modeling by jointly using the complementary representations.

We evaluate the effectiveness of multi-source neural topic modeling in multi-view embedding spaces using 7 (5 low-resource and 2 high-resource) target and 5 (high-resource) source corpora from news and medical domains, consisting of short-text, long-text, small and large document collections. We have shown state-of-the-art results with significant gains quantified by generalization (perplexity), interpretability (topic coherence) and text retrieval. The code is available at https://github.com/YatinChaudhary/Multi-view-Multi-source-Topic-Modeling.

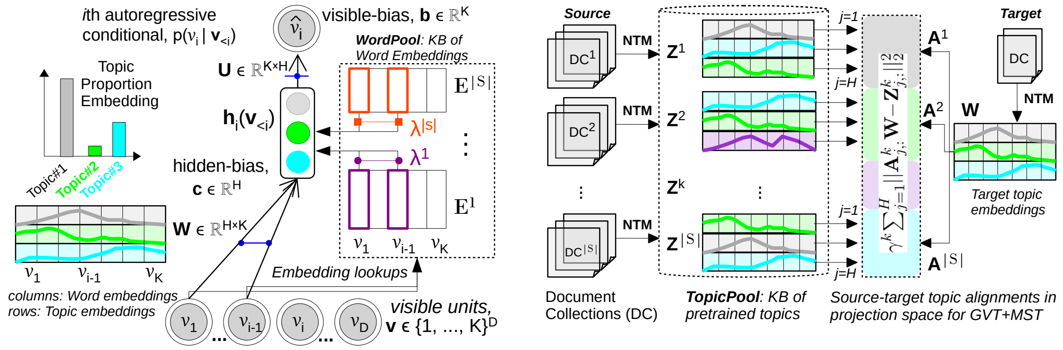


Figure 1: (Left) DocNADE (LVT+MST): Multi-source transfer learning in NTM for a document $\mathbf{v}$ by introducing pretrained word embeddings from a `WordPool` at each autoregressive step i. Double circle $\rightarrow$ multinomial (softmax) unit (Larochelle and Lauly, 2012). (Right) DocNADE (GVT+MST): Multi-source transfer learning in NTM by introducing pretrained (latent) topic embeddings from a `TopicPool`, illustrating topic alignments between source and target corpora. Each outgoing row from $\mathbf{Z}^k \in \mathbb{R}^{H \times K}$ signify a topic embedding of corresponding kth source corpus, DC^k. Here, NTM refers to a DocNADE (Larochelle and Lauly, 2012) based Neural Topic Model.

2 Knowledge-Aware Topic Modeling

Consider a sparse target domain $\mathcal{T}$ and a set of source domains $\mathcal{S}$, we first prepare two knowledge bases (KBs) of representations (or embeddings) from document collections of each of the $|\mathcal{S}|$ sources: (1) `WordPool`: a KB of pretrained word embeddings matrices $\{\mathbf{E}^1, ..., \mathbf{E}^{|\mathcal{S}|}\}$, where $\mathbf{E}^k \in \mathbb{R}^{E \times K}$, and (2) `TopicPool`: a KB of pretrained latent topic embeddings $\{\mathbf{Z}^1, ..., \mathbf{Z}^{|\mathcal{S}|}\}$, where $\mathbf{Z}^k \in \mathbb{R}^{H \times K}$ encodes a distribution over a vocabulary of K words. Here, $k \in [1, ..., |\mathcal{S}|]$ in superscript indicates knowledge of kth source, and E and H are word embedding and latent topic dimensions, respectively. While topic modeling on $\mathcal{T}$, we introduce the two types of knowledge transfers from one or many sources: *Local* (`LVT`) and *Global* (`GVT`) *View Transfer* using the two KBs of pretrained word (i.e., `WordPool`) and topic (i.e., `TopicPool`) embeddings, respectively. Specially, we employ a neural autoregressive topic model, i.e., DocNADE as backbone in building the pools and realizing the multi-source multi-view framework.

Table 2 describes the notations used. *Notice* that the superscript used in notations indicates a source.

2.1 Neural Autoregressive Topic Models

DocNADE (Larochelle and Lauly, 2012) is an unsupervised neural-network based generative topic model that is inspired by the benefits of NADE (Larochelle and Murray, 2011) and RSM (Salakhutdinov and Hinton, 2009) architectures. Specifically, DocNADE factorizes the joint probability distribution of words in a document as a product of conditional distributions and efficiently models each conditional via a feed-forward neural network (ff-net), following reconstruction mechanism.

DocNADE Formulation: For a document $\mathbf{v} = (v_1, ..., v_D)$ of size D, each word index v_i takes value in $\{1, ..., K\}$ of vocabulary size K. DocNADE learns topics in a language modeling fashion (Bengio et al., 2003) and decomposes the joint distribution $p(\mathbf{v}) = \prod_{i=1}^{D} p(v_i | \mathbf{v}_{<i})$ such that each autoregressive conditional $p(v_i | \mathbf{v}_{<i})$ is modeled by a ff-net using preceding words $\mathbf{v}_{<i}$ in the sequence:

$$\mathbf{h}_i(\mathbf{v}_{<i}) = g(\mathbf{c} + \sum_{q<i} \mathbf{W}_{:,v_q}) \text{ and } g = \{\text{sigmoid}, \tanh\}$$

$$p(v_i = w | \mathbf{v}_{<i}) = \frac{\exp(b_w + \mathbf{U}_{w,:}\mathbf{h}_i(\mathbf{v}_{<i}))}{\sum_{w'} \exp(b_{w'} + \mathbf{U}_{w',:}\mathbf{h}_i(\mathbf{v}_{<i}))}$$

for each word $i \in \{1, ..., D\}$ where $\mathbf{v}_{<i}$ is the subvector consisting of all v_q such that $q < i$ i.e., $\mathbf{v}_{<i} \in \{v_1, ..., v_{i-1}\}$, $g(\cdot)$ is a non-linear activation function, $\mathbf{W} \in \mathbb{R}^{H \times K}$ and $\mathbf{U} \in \mathbb{R}^{K \times H}$ are weight matrices, $\mathbf{c} \in \mathbb{R}^H$ and $\mathbf{b} \in \mathbb{R}^K$ are bias parameter vectors. H is the number of hidden units (the number of topics to be discovered).

Figure 1 (left) (except `WordPool`) describes the DocNADE architecture for the ith autoregressive step, where the parameter $\mathbf{W}$ is shared in the feed-forward networks and $\mathbf{h}_i$ encodes latent document-topic proportion. The value of each unit j in the hidden vector signifies contribution of the jth topic in the proportion. Importantly, the topic-word matrix $\mathbf{W}$ has a property that the column vector $\mathbf{W}_{:,v_i}$ corresponds to embedding of the word v_i, whereas the row vector $\mathbf{W}_{j,:}$ encodes latent features for the jth topic (i.e., topic-word distribution). We leverage this property to introduce external knowledge via word and topic embeddings.

Algorithm 1 Computation of $\log p(\mathbf{v})$ and Loss $\mathcal{L}(\mathbf{v})$

Input: Source domains $\mathcal{S}$, a target domain $\mathcal{T}$
Input: A training document $\mathbf{v}$ from $\mathcal{T}$
Input: `WordPool`: A KB of pretrained word embedding matrices $\{\mathbf{E}^1, ..., \mathbf{E}^{|\mathcal{S}|}\}$ from $\mathcal{S}$ domains
Input: `TopicPool`: A KB of pretrained latent topics $\{\mathbf{Z}^1, ..., \mathbf{Z}^{|\mathcal{S}|}\}$ from $\mathcal{S}$ domains
Parameters: $\mathbf{\Theta} = \{\mathbf{b}, \mathbf{c}, \mathbf{W}, \mathbf{U}, \mathbf{A}^1, ..., \mathbf{A}^{|\mathcal{S}|}, \mathbf{P}\}$
Hyper-params: $\mathbf{\Phi} = \{\lambda^1, ..., \lambda^{|\mathcal{S}|}, \gamma^1, ..., \gamma^{|\mathcal{S}|}, H\}$

1: Initialize: $\mathbf{a} \leftarrow \mathbf{c}$ and $p(\mathbf{v}) \leftarrow 1$
2: **for** word i from 1 to D **do**
3: Compute i^{th} position-dependent hidden: $\mathbf{h}_i(\mathbf{v}_{<i}) \leftarrow g(\mathbf{a})$, where g = {sigmoid, tanh}
4: Compute i^{th} autoregressive conditional: $p(v_i = w|\mathbf{v}_{<i}) \leftarrow \frac{\exp(b_w + \mathbf{U}_{w,:}\mathbf{h}_i(\mathbf{v}_{<i}))}{\sum_{w'} \exp(b_{w'} + \mathbf{U}_{w',:}\mathbf{h}_i(\mathbf{v}_{<i}))}$
5: Memorize: $p(\mathbf{v}) \leftarrow p(\mathbf{v})p(v_i|\mathbf{v}_{<i})$
6: Compute pre-activation for word i : $\mathbf{a} \leftarrow \mathbf{a} + \mathbf{W}_{:,v_i}$
7: **if** `LVT` **then**
8: Get word-embeddings $\mathbf{E}$ from `WordPool`
9: Introduce prior knowledge $\mathbf{E}$ for word i:
scheme (i): $\mathbf{a} \leftarrow \mathbf{a} + \sum_{k=1}^{|\mathcal{S}|} \lambda^k \, \mathbf{E}^k_{:,v_i}$
scheme (ii): $\hat{\mathbf{e}}_i \leftarrow$ concat$(\mathbf{E}^1_{:,v_i}, ..., \mathbf{E}^k_{:,v_i})$
$\mathbf{a} \leftarrow \mathbf{a} + \mathbf{P} \cdot \hat{\mathbf{e}}_i$
10: Loss (negative log-likelihood): $\mathcal{L}(\mathbf{v}) \leftarrow -\log p(\mathbf{v})$
11: **if** `GVT` **then**
12: Topic-embedding transfer using `TopicPool`: $\Delta \leftarrow \sum_{k=1}^{|\mathcal{S}|} \gamma^k \sum_{j=1}^{H} ||\mathbf{A}^k_{j,:}\mathbf{W} - \mathbf{Z}^k_{j,:}||_2^2$
13: Overall loss with controlled topic-imitation: $\mathcal{L}(\mathbf{v}) \leftarrow \mathcal{L}(\mathbf{v}) + \Delta$
14: Minimize $\mathcal{L}(\mathbf{v})$ using stochastic gradient descent

Algorithm 1 (for DocNADE, set both `LVT` and `GVT` to *False*) demonstrates the computation of $\log p(\mathbf{v})$ and loss (i.e., negative log-likelihood) $\mathcal{L}(\mathbf{v})$ that is minimized using stochastic gradient descent. Moreover, computing each $\mathbf{h}_i$ is efficient (linear complexity) due to NADE architecture that leverages the pre-activation $\mathbf{a}_{i-1}$ of $(i-1)$th step in computing $\mathbf{a}_i$ for the ith step (line #6). See Larochelle and Lauly (2012) for further details.

Why DocNADE backbone: It has shown outperforming traditional models such as LDA and RSM. Additionally, Gupta et al. (2019a,b) have extended DocNADE on short texts by introducing context-insensitive word embeddings; however, based on a single-source transfer. Thus, we adopt DocNADE.

2.2 MVT and MST in Neural Topic Modeling

We describe our transfer learning framework in topic modeling that jointly exploits the complementary prior knowledge accumulated in (`WordPool`, `TopicPool`), obtained from large document collections (DCs) from several sources. In doing so, we first apply the DocNADE to generate a topic-word matrix for each of the DCs, where its column-vector and row-vector generate $\mathbf{E}^k$ and $\mathbf{Z}^k$, respectively for the kth source. See *appendix* for the mechanics of extracting word and topic embeddings from the topic-word matrix of a source.

LVT+MST Formulation for Multi-source Word Embedding Transfer: As illustrated in Figure 1 (left) and Algorithm 1 (with `LVT` being *True*, line #7), we perform transfer learning on a target $\mathcal{T}$ using the `WordPool` of pretrained word embeddings $\{\mathbf{E}^1, ..., \mathbf{E}^{|\mathcal{S}|}\}$ from several sources $\mathcal{S}$ (i.e., multi-source) under the two schemes:

***scheme* (i)**: Using a domain-relevance factor λ for every source in the `WordPool` such that the hidden vector $\mathbf{h}_i$ encodes document-topic distribution, augmented with prior knowledge in form of pretrained word embeddings from several sources:

$$\mathbf{h}_i(\mathbf{v}_{<i}) = g(\mathbf{c} + \sum_{q<i} \mathbf{W}_{:,v_q} + \sum_{q<i} \sum_{k=1}^{|\mathcal{S}|} \lambda^k \, \mathbf{E}^k_{:,v_q})$$

Here, k refers to the kth source and λ^k is a weight for $\mathbf{E}^k$ that controls the amount of knowledge transferred in $\mathcal{T}$, based on cross-domain overlap.

***scheme* (ii)**: Using a projection matrix $\mathbf{P} \in \mathbb{R}^{H \times P}$ with $P = E \times |\mathcal{S}|$ in order to align word-embedding spaces of the target and all source domains for all D words in the document $\mathbf{v}$ such that:

$$\text{For } q \in \{i, ...D\} : \hat{\mathbf{e}}_q = \text{concat}(\mathbf{E}^1_{:,v_q}, ..., \mathbf{E}^k_{:,v_q})$$
$$\mathbf{h}_i(\mathbf{v}_{<i}) = g(\mathbf{c} + \sum_{q<i} \mathbf{W}_{:,v_q} + \sum_{q<i} \mathbf{P} \cdot \hat{\mathbf{e}}_q)$$

Unlike scheme (i), the second schema allows us to automatically determine shifts in the target and source domains, identify and transfer relevant prior knowledge from many sources without configuring λ for every source. To better guide TM, we also introduce pre-trained contextualized word embedding from BERT, concatenating with $\hat{\mathbf{e}}_q$.

GVT+MST Formulation for Multi-source Topic Embedding Transfer: Next, we perform knowledge transfer exclusively using the `TopicPool` of pretrained topic embeddings (e.g., $\mathbf{Z}^k$) from one or several sources, $\mathcal{S}$. In doing so, we add a regularization term to the loss function $\mathcal{L}(\mathbf{v})$ and require DocNADE to minimize the overall loss in a way that the (latent) topic features in $\mathbf{W}$ simultaneously inherit relevant topical features from each of the source domains $\mathcal{S}$, and thus, it generates meaningful representations for the target $\mathcal{T}$ in order to address data-sparsity. The overall loss $\mathcal{L}(\mathbf{v})$ due to GVT+MST configuration in DocNADE is:

$$\mathcal{L}(\mathbf{v}) = -\log p(\mathbf{v}) + \sum_{k=1}^{|\mathcal{S}|} \gamma^k \sum_{j=1}^{H} ||\mathbf{A}^k_{j,:}\mathbf{W} - \mathbf{Z}^k_{j,:}||_2^2$$

Target Domain Corpora								Source Domain Corpora							
ID	**Data**	**Train**	**Val**	**Test**	K	**L**	**C**	**ID**	**Data**	**Train**	**Val**	**Test**	K	**L**	**C**
$\mathcal{T}^1$	20NSshort	1.3k	0.1k	0.5k	1.4k	13.5	20	$\mathcal{S}^1$	20NS	7.9k	1.6k	5.2k	2k	107.5	20
$\mathcal{T}^2$	20NSsmall	0.4k	0.2k	0.2k	2k	187.5	20	$\mathcal{S}^2$	R21578	7.3k	0.5k	3.0k	2k	128	90
$\mathcal{T}^3$	TMNtitle	22.8k	2.0k	7.8k	2k	4.9	7	$\mathcal{S}^3$	TMN	22.8k	2.0k	7.8k	2k	19	7
$\mathcal{T}^4$	R21578title	7.3k	0.5k	3.0k	2k	7.3	90	$\mathcal{S}^4$	AGNews	118k	2.0k	7.6k	5k	38	4
$\mathcal{T}^5$	Ohsumedtitle	8.3k	2.1k	12.7k	2k	11.9	23	$\mathcal{S}^5$	PubMed	15.0k	2.5k	2.5k	3k	254.8	-
$\mathcal{T}^6$	Ohsumed	8.3k	2.1k	12.7k	3k	159.1	23								

Table 3: Data statistics: Short/long texts and/or small/large corpora in target and source domains. Symbols- K: vocabulary size, L: average text length (#words), C: #classes and k: thousand. For short-text, L<15. $\mathcal{S}^3$ is also used in target. '-': unlabeled data.

	$\mathcal{T}^1$	$\mathcal{T}^2$	$\mathcal{T}^3$	$\mathcal{T}^4$	$\mathcal{T}^5$	$\mathcal{T}^6$
$\mathcal{S}^1$	$\mathcal{I}$	$\mathcal{I}$	$\mathcal{R}$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{D}$
$\mathcal{S}^2$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{I}$	$\mathcal{D}$	$\mathcal{D}$
$\mathcal{S}^3$	$\mathcal{R}$	$\mathcal{R}$	$\mathcal{I}$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{D}$
$\mathcal{S}^4$	$\mathcal{R}$	$\mathcal{R}$	$\mathcal{R}$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{D}$
$\mathcal{S}^5$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{D}$	-	-

Table 4: Domain overlap in source-target corpora. $\mathcal{I}$: Identical, $\mathcal{R}$: Related and $\mathcal{D}$: Distant domains.

Here, $\mathbf{A}^k \in \mathbb{R}^{H \times H}$ aligns latent topics in the target $\mathcal{T}$ and kth source, and γ^k governs the degree of imitation of topic features $\mathbf{Z}^k$ by $\mathbf{W}$ in $\mathcal{T}$. Consequently, the generative process of learning meaningful topics in $\mathbf{W}$ of the target domain $\mathcal{T}$ is guided by relevant topic features $\{\mathbf{Z}\}_1^{|\mathcal{S}|} \in$ `TopicPool`. Algorithm 1 (line #11) describes the computation of the loss, when `GVT` = *True* and `LVT` = *False*.

Moreover, Figure 1 (right) illustrates the need for topic alignments between target and source(s). Here, j indicates the topic (i.e., row) index in a topic matrix, e.g., $\mathbf{Z}^k$. Observe that the first topic (gray curve), i.e., $Z^1_{j=1} \in \mathbf{Z}^1$ of the first source aligns with the first row-vector (i.e., topic) of $\mathbf{W}$ (of target). However, the other two topics $Z^1_{j=2}, Z^1_{j=3} \in \mathbf{Z}^1$ need alignment with the target.

MVT+MST Formulation for Multi-source Word and Topic Embeddings Transfer: When `LVT` and `GVT` are *True* (Algorithm 1) for many sources, the two complementary representations are jointly used in transfer learning using `WordPool` and `TopicPool`, and therefore, the name *multi-view* and *multi-source* transfers.

Computational complexity of NTM: For DocNADE, the complexity of computing all hidden layers $\mathbf{h}_i(\mathbf{v}_{<i})$ is in $O(DH)$ and all $p(\mathbf{v}|\mathbf{v}_{<i})$ in $O(KDH)$. Thus, the overall complexity of DocNADE is in $O(DH + KDH)$.

Within the proposed transfer learning framework, the complexity of computing all hidden layers (`LVT+MST` in *scheme* (i)) and topic-embedding transfer term (`GVT+MST`) is in $O(DH + |\mathcal{S}|DH)$ and $O(|\mathcal{S}|KH)$, respectively. Since $|\mathcal{S}| << H$, thus the overall complexity of DocNADE with `MVT+MST` is in $O(DH + KDH + KH)$.

3 Evaluation and Analysis

Datasets: Table 3 describes the datasets used in high-resource source and low-and high-resource target domains for our experiments. The target domain $\mathcal{T}$ consists of four short-text corpora (`20NSshort`, `TMNtitle`, `R21578title` and `Ohsumedtitle`), one small corpus (`20NSsmall`) and two large corpora (`TMN` and `Ohsumed`). However in source $\mathcal{S}$, we use five large corpora (`20NS`, `R21578`, `TMN`, `AGnews` and `PubMed`) in different label spaces (i.e, domains). Here, the corpora ($\mathcal{T}^5$, $\mathcal{T}^6$ and $\mathcal{S}^5$) belong to *medical* and others to *news*.

Baselines (Related Works)	Features					
	NTM	*AuR*	*LVT*	*GVT*	*MVT*	*MST*
LDA						
RSM	✓					
DocNADE	✓	✓				
NVDM	✓					
ProdLDA						
Gauss-LDA			✓			
glove-DMM			✓			
DocNADEe	✓	✓	✓			
EmbSum-Glove, EmbSum-BERT						
doc2vec						
this work	✓	✓	✓	✓	✓	✓

Table 5: Baselines (related works) vs this work. Here, *NTM* and *AuR* refer to neural network-based TM and autoregressive assumption, respectively. DocNADEe → DocNADE+Glove embeddings.

Additionally, Table 4 suggests domain overlap (label match) in the target and source corpora, where we define 3 types of overlap: $\mathcal{I}$ (identical) if all labels match, $\mathcal{R}$ (related) if some labels match, and $\mathcal{D}$ (distant) if a very few or no labels match. Note, our approaches are completely unsupervised and do not use the data labels (*appendix*).

Reproducibility: We follow the experimental setup similar to DocNADE (Larochelle and Lauly, 2012) and DocNADEe (Gupta et al., 2019a), where the number of topics (H) is set to 200. While DocNADEe requires the dimension (i.e., E) of word embeddings be the same as the latent topic (i.e., H), we follow *scheme (ii)* (Algorithm 1) to introduce

	KBs from Source Corpus	Model or Transfer Type	Scores on Target Corpus (*in sparse-data and sufficient-data settings*)														
			`20NSshort`			`TMNtitle`			`R21578title`			`20NSsmall`			`TMN`	
			PPL	*COH*	*IR*	*PPL*	*COH*	*IR*	*PPL*	*COH*	*IR*	*PPL*	*COH*	*IR*	*PPL*	*COH*
Baselines	**Baseline** ***TM*** **without** ***Word-Embeddings***	NVDM	1047	.736	.076	973	.740	.190	372	.735	.271	957	.515	.090	833	.673
		ProdLDA	923	.689	.062	1527	.744	.170	480	.742	.200	1181	.394	.062	1519	.577
		DocNADE	646	.667	.290	706	.709	.521	192	.713	.657	594	.462	.270	584	.636
Proposed	*20NS*	LVT	**630**	.673	.298	705	.709	.523	194	.708	.656	594	.455	.288	582	.649
		GVT	646	.690	.303	718	.720	.527	184	.698	.660	594	.500	.310	590	.652
		MVT	638	.690	**.314**	714	.718	.528	188	.715	.655	600	.499	.311	588	.650
	TMN	LVT	649	.668	.296	**655**	.731	.548	187	.703	.659	593	.460	.273	-	-
		GVT	661	.692	.294	689	.728	.555	191	.709	.660	596	.521	.276	-	-
		MVT	658	.687	.297	663	.747	.553	195	.720	.660	599	.507	.292	-	-
	R21578	LVT	656	.667	.292	704	.715	.522	186	.715	.676	593	.458	.267	581	.636
		GVT	654	.672	.293	716	.719	.526	194	.706	.672	595	.485	.279	591	.646
		MVT	650	.670	.296	716	.720	.528	194	.724	.676	599	.490	.280	589	.650
	AGnews	LVT	650	.677	.297	682	.723	.533	185	.710	.659	**592**	.458	.260	**564**	.668
		GVT	667	.695	.300	728	.735	.534	190	.717	.663	598	.563	.282	601	.684
		MVT	659	.696	.290	718	.740	.533	189	.727	.659	599	.566	.279	592	.686
	MST	LVT	640	.678	.308	663	.732	.547	**182**	.739	.673	594	.542	.277	568	.674
		GVT	658	.705	.305	704	.746	.550	192	.727	.673	599	.585	**.326**	602	.680
		MVT	656	**.740**	**.314**	680	**.752**	**.569**	188	**.745**	**.685**	600	**.637**	.285	600	**.690**
	Gain%(vs DocNADE)		↑2.48	↑10.9	↑8.28	↑7.22	↑6.06	↑9.21	↑5.20	↑4.49	↑4.26	↑0.34	↑37.9	↑20.7	↑3.42	↑8.50

Table 6: State-of-the-art comparisons with TMs: Perplexity (PPL), topic coherence (COH) and precision@recall (IR) at retrieval fraction 0.02. Scores reported on each of the target, given KBs from several sources. LVT and GVT employ `WordPool` and `TopicPool`, respectively. MVT employs both. LVT+MST scores using *scheme (i)*. Here, Bold → Best score (in column) and Gain% → Bold vs DocNADE.

pre-trained word embeddings from Glove, FastText (E=300) (Bojanowski et al., 2017) and BERT-base (E=768) models. See *appendix* for the experimental setup, hyperparameters and optimal values of $\lambda^k \in [0.1, 0.5, 1.0]$ and $\gamma^k \in [0.1, 0.01, 0.001]$.

Baselines (Related Works): (1) *Topic Models without Transfer Learning* that learn topics in isolation using the given target corpus only. We employ LDA-based variant, i.e., ProdLDA (Srivastava and Sutton, 2017) and neural network-based variants, i.e., DocNADE (autoregressive) and NVDM (non-autoregressive) (Miao et al., 2016).

(2) *Topic Models with Transfer Learning* that leverages pre-trained context-insensitive word embeddings (Pennington et al., 2014). We consider topic models based on both LDA, i.e., Gauss-LDA (Das et al., 2015) and glove-GMM (Nguyen et al., 2015), and neural networks, i.e., DocNADEe (Gupta et al., 2019a). They do not leverage pre-trained topic-embeddings (i.e., *GVT*), contextualized word-embedding and MST-MVT techniques.

(3) *Unsupervised Document Representation* to quantify the quality of document representations. We use 3 strategies: doc2vec (Le and Mikolov, 2014), EmbSum-Glove and EmbSum-BERT (represent a document by summing the pre-trained embeddings of it's words from Glove and BERT).

(4) *Zero-shot Topic Modeling* to demonstrate transfer learning capabilities of the proposed framework, where we build (train) a TM using all source corpora and evaluate on the target corpus $\mathcal{T}$, and

(5) *Data-augmentation* that first augments the target corpus with all the source corpora and then builds a TM to evaluate transfer learning on $\mathcal{T}$.

Table 5 summarizes the comparison of this work with the aforementioned baselines. Tables 6 and 7 employ baseline TMs without and with transfer learning, respectively.

3.1 Generalization: Perplexity (PPL)

To evaluate generative performance of DocNADE-based NTM, we compute average held-out perplexity per word: $PPL = \exp\big(-\frac{1}{N}\sum_{t=1}^{N}\frac{1}{|\mathbf{v}_t|}\log p(\mathbf{v}_t)\big)$, where N and $|\mathbf{v}_t|$ are the number of documents and words in a document $\mathbf{v}_t$, respectively.

Tables 6 and 7 quantitatively show PPL scores on the five target corpora using one or four sources. In Table 6 using `TMN` (as a single source) for LVT, GVT and MVT transfer types on the target `TMNtitle`, we see improved (reduced) PPL scores: (655 vs 706), (689 vs 706) and (663 vs 706) respectively in comparison to DocNADE. We also observe gains due to MST+LVT, MST+GVT and MST+MVT configurations on `TMNtitle`. Similarly in MST+LVT for `R21578title`, we observe a gain of 5.2% (182 vs 192), suggesting that multi-source transfer learning using pretrained

	KBs from Source Corpus	Model or Transfer Type	Scores on Target Corpus (*in sparse-data and sufficient-data settings*)														
			20NSshort			TMNtitle			R21578title			20NSsmall			TMN	
			PPL	*COH*	*IR*	*PPL*	*COH*	*IR*	*PPL*	*COH*	*IR*	*PPL*	*COH*	*IR*	*PPL*	*COH*
Baselines		doc2vec	-	-	.090	-	-	.190	-	-	.518	-	-	.200	-	-
		EmbSum-Glove	-	-	.236	-	-	.513	-	-	.587	-	-	.214	-	-
		EmbSum-BERT	-	-	.261	-	-	.499	-	-	.594	-	-	.262	-	-
	Baseline TM	Gauss-LDA	-	-	.080	-	-	.408	-	-	.367	-	-	.090	-	-
	with *Word-*	glove-DMM	-	.512	.183	-	.633	.445	-	.364	.273	-	.578	.090	-	.705
	Embeddings	→ DocNADEe	**629**	.674	.294	680	.719	.540	187	.721	.663	**590**	.455	.274	572	.664
Proposed	*20NS*	MVT+Glove	<u>630</u>	.721	.320	688	.741	.565	**183**	.724	.667	597	.561	**.306**	**570**	.693
	TMN	MVT+Glove	640	.731	.295	**673**	.750	.576	184	.716	.672	599	.594	.261	-	-
	R21578	MVT+Glove	633	.705	.295	689	.738	.540	185	.737	**.691**	<u>595</u>	.485	.255	577	.697
	AGnews	MVT+Glove	642	.734	.302	706	.748	.565	190	.734	.675	598	.573	.284	585	.703
		MVT+Glove	644	.739	.304	**673**	**.752**	.570	**183**	.742	.684	598	.631	.282	582	.710
	MST	+ FastText	654	.741	.313	**673**	.751	.578	**183**	.744	.684	599	.634	.254	582	**.711**
		+ BERT	-	**.744**	**.322**	-	**.752**	**.604**	-	**.745**	.680	-	**.640**	.282	-	.709
		Gain% (vs DocNADEe)	↓0.16	↑10.4	↑9.5	↑1.03	↑4.60	↑11.9	↑3.33	↑3.20	↑4.22	↓0.85	↑40.7	↑2.92	↑.35	↑7.08

Table 7: State-of-the-art comparisons against baseline TMs using context-insensitive word embeddings: PPL, COH and IR at retrieval fraction 0.02. Scores are reported on each of the target, given the KBs. Here, MVT → LVT+GVT, DocNADEe → DocNADE+Glove, Bold → Best score (in column), Underline → Second best score (in column) and Gain% → Bold vs DocNADEe. For all the configurations, we apply a projection on ([non-]contextualized) word embeddings from several sources, i.e., *scheme (ii)*.

word and topic embeddings (jointly) helps improving TM, and it also verifies domain relatedness (e.g., in TMN-TMNtitle and AGnews-TMN). Similarly, Table 7 reports gains in PPL (e.g., on TMNtitle, R21578title, etc.) compared to the baseline DocNADEe. PPL scores due to BERT can be not computed since its embeddings are aware of both preceding and following contexts.

In Table 8, we show PPL scores on 2 medical target corpora: Ohsumtitle and Ohsumed using 2 sources: AGnews (*news*) and PubMed (*medical*) to perform cross-domain and in-domain transfers. We see that using PubMed for LVT on both the targets improves generalization. Overall, we report a gain of 17.3% (1268 vs 1534) on Ohsumedtitle and 8.55% (1497 vs 1637) on Ohsumed datasets, compared to DocNADEe.

3.2 Interpretabilty: Topic Coherence (COH)

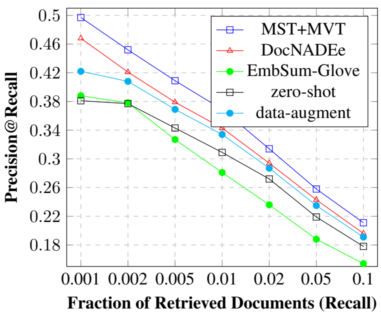
While PPL is used for model selection, Chang et al. (2009) showed in some cases humans preferred TMs (based on the semantic quality of topics) with higher (worse) perplexities. Therefore, we also estimate the quality of topics. We follow Röder et al. (2015) and Gupta et al. (2019a) to compute COH of the top 10 words in each topic. Essentially, the higher scores imply the coherent topics.

Tables 6 and 7 (under COH column) demonstrate that our approaches (GVT, MVT and MST) show noticeable gains and thus improve topic quality. For instance in Table 6, when AGnews is used as a single source for 20NSsmall datatset, we observe a gain in COH due to GVT (.563 vs .462) and MVT (.566 vs .462). Additionally, noticeable gains are reported due to MST+LVT (.542 vs .462), MST+GVT (.585 vs .462) and MST+MVT (.637 vs .462), compared to DocNADE. Importantly, we find a trend MVT>GVT>LVT in COH scores for both the single-source and multi-source transfers. Similarly, Table 7 show noticeable gains (e.g., 40.7%, 10.4%, 7.08%, etc.) in COH due to MST+MVT+Glove +FastText+BERT setting. Moreover, Table 8 shows gains in COH due to GVT on Ohsumedtitle and Ohsumed, using pretrained knowledge from PubMed. Overall, the GVT, MVT and MST boost COH for all the five target corpora compared to the baseline TMs (i.e., DocNADE and DocNADEe). The improvements suggest that the approaches scale across domains.

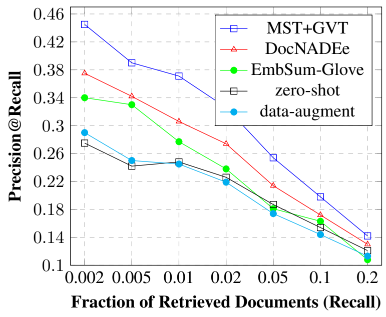
3.3 Applicability: Information Retrieval (IR)

We further evaluate the quality of document representations and perform an IR task using the label information only to compute precision. We follow the experimental setup similar to Gupta et al. (2019a). See the details in *appendix*.

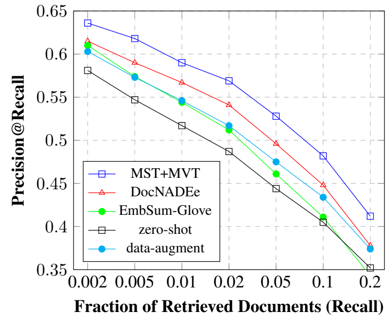
Tables 6 and 7 report precision scores at retrieval fraction 0.02 where the configuration MST+MVT outperforms both the DocNADE and DocNADEe for all 4 targets. We observe *large gains* in precision: (a) Table 6: 20.7% (.326 vs .270) on 20NSsmall, 9.21% (.569 vs .521) on



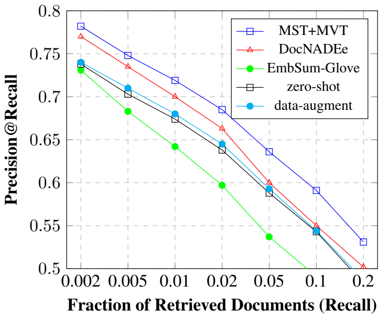
(a) **IR:** 20NSshort



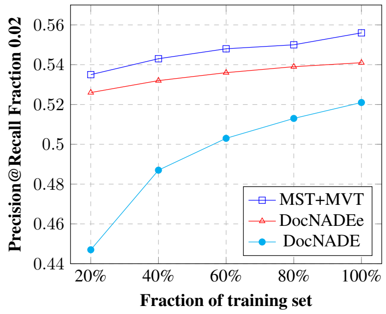
(b) **IR:** 20NSsmall



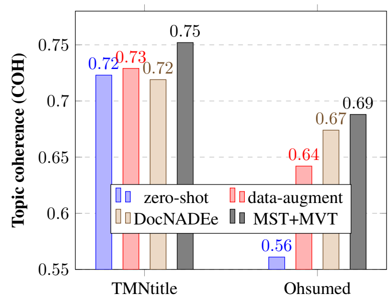
(c) **IR:** TMNtitle



(d) **IR:** R21578title



(e) **IR:** TMNtitle



(f) **COH**: Zero-shot and DA

Figure 2: (a, b, c, d) Retrieval performance (precision@recall) on 4 datasets: 20NSshort, 20NSsmall, TMNtitle and R21578title. (e) Precision at recall fraction 0.02, each for a fraction (20%, 40%, 60%, 80%, 100%) of the training set of TMNtitle. (f) Zero-shot and data-augmentation (DA) for COH on TMNtitle and Ohsumed.

`TMNtitle`, etc., (b) Table 7: 11.9% (.604 vs .540) on `TMNtitle` and 9.5% (.322 vs .294) on `20NSshort`, etc., (c) Table 8: 14.4% (.183 vs .160) on `Ohsumedtitle`. Additionally, Figures 2a, 2b, 2c and 2d illustrate precision-recall curves on `20NSshort`, `20NSsmall`, `TMNtitle` and `R21578title` respectively, where MST+MVT and MST+GVT consistently outperform the baselines at all fractions.

3.4 Zero/Few-shot and Data-augmentation

Figures 2a, 2b, 2c and 2d show precision in the *zero-shot* (source-only training) and *data-augmentation* (source+target training) configurations. Observe that the latter helps in learning meaningful representations and performs better than zero-shot; however, it is outperformed by MST+MVT, suggesting that a naive (data space) augmentation does not add sufficient prior or relevant information to the sparse target. Thus, we find that it is beneficial to augment training data in feature space (e.g., LVT, GVT and MVT) especially for unsupervised topic models using `WordPool` and `TopicPool`.

Moreover in the *few-shot* setting, we first split the training data of `TMNtitle` into several sets: 20%, 40%, 60%, 80% of the training set and then retrain DocNADE, DocNADEe and DocNADE+MST+MVT on each as a sparse target. We demonstrate transfer learning in such sparse-data settings using the KBs: `WordPool` and `TopicPool` jointly. Figure 2e plots precision at retrieval fraction 0.02 and validates that the proposed modeling consistently outperforms both the baselines: DocNADE and DocNADEe.

Beyond IR, we further investigate computing topic coherence (COH) for the zero-shot and data-augmentation baselines, where the COH scores in Figure 2f suggest that MST+MVT outperforms DocNADEe, zero-shot and data-augmentation.

3.5 Topics and Nearest Neighbors (NN)

For topic level inspection, we first extract topics using the rows of $\mathbf{W}$ of source and target corpora. Table 9 shows the topics (top-5 words) from source and target domains. Observe that the target topics become more coherent after transfer learning (i.e., +GVT) from one or more sources. The blue color signifies that a target topic has imitated certain topic words from the source. We also show a topic (the last) improved due to multi-source transfer.

For word level inspection, we extract word representations using the columns of $\mathbf{W}$. Table 10

KBs from Source Corpus	Model or Transfer Type	Ohsumedtitle			Ohsumed		
		PPL	*COH*	*IR*	*PPL*	*COH*	*IR*
baselines	ProdLDA	1121	.734	.080	1677	.646	.080
	DocNADE	1321	.728	.160	1706	.662	.184
	EmbSum-BioEmb	-	-	.150	-	-	.148
	EmbSum-SciBERT	-	-	.160	-	-	.165
	DocNADEe	1534	.738	.175	1637	.674	.183
AGnews	LVT	1587	.732	.160	1717	.657	.184
	GVT	1529	.732	.160	1594	.665	.185
	MVT	1528	.734	.160	1598	.666	.184
	+ BioEmb	1488	.747	.176	1595	.681	.187
PubMed	LVT	**1268**	.732	.172	1535	.669	.190
	GVT	1392	.740	.173	1718	.671	**.192**
	MVT	1408	.743	.178	1514	.674	.191
	+ BioEmb	1364	**.753**	.182	1633	**.689**	.191
MST	LVT	**1268**	.733	.172	1536	.668	.190
	GVT	1391	.740	.172	1504	.666	**.192**
	MVT	1399	.744	.177	1607	.679	.191
	+ BioEmb	1375	.751	.180	**1497**	.693	.190
	+ BioFastText	1350	**.753**	.178	1641	.688	.187
	+ SciBERT	-	**.753**	**.183**	-	.682	.182
Gain% (vs DocNADE)		↑4.01	↑3.43	↑14.4	↑12.3	↑4.08	↑4.35
Gain% (vs DocNADEe)		↑17.3	↑2.03	↑4.60	↑8.55	↑2.22	↑4.91

Table 8: PPL, COH, and IR at fraction 0.02. BioEmb and BioFastText (Moen and Ananiadou, 2013): 200-dimension; SciBERT: Pretrained BERT-variant (Beltagy et al., 2019). + BioEmb: MVT+BioEmb

$\mathcal{T}$	$\mathcal{S}$	Model	Topic-words (Top 5)
`20NSshort`	`20NS`	DocNADE	shipping, sale, prices, expensive, price
		-GVT	sale, price, monitor, site, setup
		+GVT	shipping, sale, price, expensive, subscribe
	`AGnews`	DocNADE	microsoft, software, ibm, linux, computer
		-GVT	apple, modem, side, baud, perform
		+GVT	microsoft, software, desktop, computer, apple
`TMNtitle`	`AGnews`	DocNADE	miners, earthquake, explosion, stormed, quake
	`TMN`	DocNADE	tsunami, quake, japan, earthquake, radiation
		-GVT	strike, jackson, kill, earthquake, injures
		+GVT	earthquake, radiation, explosion, wildfire

Table 9: Source $\mathcal{S}$ and target $\mathcal{T}$ topics before (-) and after (+) topic transfer (GVT) from one/more source(s)

shows nearest neighbors (NNs) of the word *chip* in `20NSshort` (target) corpus, before (-) and after (+) topic knowledge transfer via GVT using three sources (i.e., MST+GVT). Observe that the NNs in the target become more meaningful by gaining knowledge mainly from `20NS` source.

4 Conclusion

We have presented a state-of-the-art neural topic modeling framework using multi-view embedding spaces: pretrained topic-embeddings and word-embeddings (context-sensitive and context-insensitive) from one or many sources to improve quality of topics and document representations.

source corpora			target corpus	
`20NS`	`R21578`	`AGnews`	`20NSshort`	
			-GVT	+GVT
key	chips	chips	virus	chips
encrypted	semiconductor	chipmaker	intel	technology
encryption	miti	processors	gosh	intel
clipper	makers	semiconductor	crash	encryption
keys	semiconductors	intel	chips	clipper

Table 10: Five nearest neighbors of the word *chip* in a target and three source semantic spaces before (-) and after (+) transfer via MST+GVT configuration

Acknowledgments

This research was supported by the Federal Ministry for Economic Affairs and Energy (Bundeswirtschaftsministerium: bmwi.de), grant 01MD19003E (PLASS: Platform for Analytical Supply Chain Mangement Services, plass.io) at Siemens AG (Technology- Machine Intelligence), Munich Germany.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

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

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *TACL*, 5:135–146.

Bin Cao, Sinno Jialin Pan, Yu Zhang, Dit-Yan Yeung, and Qiang Yang. 2010. Adaptive transfer learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. AAAI Press.

Jonathan Chang, Jordan L. Boyd-Graber, Sean Gerrish, Chong Wang, and David M. Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada.*, pages 288–296.

Rajarshi Das, Manzil Zaheer, and Chris Dyer. 2015. Gaussian lda for topic models with word embeddings. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 795–804. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Pankaj Gupta, Yatin Chaudhary, Florian Buettner, and Hinrich Schütze. 2019a. Document informed neural autoregressive topic models with distributional prior. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*.

Pankaj Gupta, Yatin Chaudhary, Florian Buettner, and Hinrich Schütze. 2019b. textTOvec: Deep contextualized neural autoregressive topic models of language with distributed compositional prior. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Pankaj Gupta, Yatin Chaudhary, Thomas A. Runkler, and Hinrich Schütze. 2020. Neural topic modeling with continual lifelong learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 3907–3917. PMLR.

Hugo Larochelle and Stanislas Lauly. 2012. A neural autoregressive topic model. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems*, pages 2717–2725.

Hugo Larochelle and Iain Murray. 2011. The neural autoregressive distribution estimator. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 15 of *JMLR Proceedings*, pages 29–37. JMLR.org.

Quoc V. Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31th International Conference on Machine Learning, ICML*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1188–1196. JMLR.org.

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. *CoRR*, abs/1907.11692.

Yishu Miao, Lei Yu, and Phil Blunsom. 2016. Neural variational inference for text processing. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1727–1736. JMLR.org.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems*, pages 3111–3119.

SPFGH Moen and Tapio Salakoski2 Sophia Ananiadou. 2013. Distributional semantics resources for biomedical text processing. *Proceedings of LBM*, pages 39–44.

Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. 2015. Improving topic models with latent feature word representations. *TACL*, 3:299–313.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

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 2015, Shanghai, China, February 2-6, 2015*, pages 399–408. ACM.

Ruslan Salakhutdinov and Geoffrey E. Hinton. 2009. Replicated softmax: an undirected topic model. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems*, pages 1607–1614. Curran Associates, Inc.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. In *5th International Conference on Learning Representations, ICLR*.

A Data Description

In order to evaluate knowledge transfer within unsupervised neural topic modeling, we use the following seven datasets in the target domain $\mathcal{T}$ following the similar experimental setup as in DocNADEe: (1) `20NSshort`: We take documents from 20NewsGroups data, with document size (number of words) less than 20. (2) `20NSsmall`: We sample 20 document (each having more than 200 words) for training from each class of the 20NS dataset. For validation and test, 10 document for each class. Therefore, it is a corpus of few (long) documents. (3) `TMNtitle`: Titles of the Tag My News (TMN) news dataset. (4) `R21578title`: Reuters corpus, a collection of new stories from `nltk.corpus`. We take titles of the documents. (5) `Ohsumedtitle`: Titles of Ohsumed abstracts. Source: `disi.unitn.it/moschitti/corpora.htm`. (6) `Ohsumed`: Ohsumed dataset, collection of medical abstracts. Source: `disi.unitn.it/moschitti/corpora.htm`. (7) `TMN`: The Tag My News (TMN) news dataset.

To **prepare knowledge base** of word embeddings (local semantics) and latent topics (global semantics) features, we use the following six datasets in the source $\mathcal{S}$: (1) `20NS`: 20NewsGroups corpus, a collection of news stories from `nltk.corpus`. (2) `TMN`: The Tag My News (TMN) news dataset. (3) `R21578`: Reuters corpus, a collection of new stories from `nltk.corpus`. (4) `AGnews`: AGnews data sellection. `PubMed`: Medical abstracts of randomized controlled trials. Source: `https://github.com/Franck-Dernoncourt/pubmed-rct`.

See Table 3 (in paper content) describes each of the datasets, where a short-text refers to a text document having less than 15 words. Notice that each of the datasets in the target and source domains, we see overlap in their label spaces. See Table 4 for the label information for each of the source and target corpora. Additionally in supplementary, we have also provided the code and pre-processed datasets used in our experiments.

B Getting Word and Latent Topic Representations from Source(s)

Since in DocNADE, the column of $\mathbf{W}_{:,v_i}$ gives a word vector of the word v_i, therefore the dimension of word embeddings in each of the $\mathbf{E}^k$ is same (i.e., $H = 200$). Thus, we prepare the knowledge base of word representations $\mathbf{E}^k$ from kth source using DocNADE, where each word vector is of $H = 200$ dimension.

Since the row vector of $\mathbf{W}_{j,:}$ in DocNADE encodes jth topic feature, therefore each latent topic (i.e., row) in feature matrix $\mathbf{W}$ is a vector of K dimension, corresponding the definition of topics that it is a distribution over vocabulary. H is the number of latent topics and K is the vocabulary size, where K varies across corpora. Thus, we train DocNADE to learn a feature matrix specific to each of the source corpora, e.g. $\mathbf{W}^k \in \mathbb{R}^{H \times K}$ of kth source.

For a target corpus of vocabulary size K', the DocNADE learns a feature matrix $\mathbf{W}^{\mathcal{T}} \in \mathbb{R}^{H \times K'}$. Similarly, $\mathbf{W}^k \in \mathbb{R}^{H \times K}$ for kth source of vocabulary size K. Since in the sparse-data setting for the target, $K' << K$ due to additional word in the source. To perform GVT, we need the same topic feature dimensions in the target and source, i.e., K' of the target. Therefore, we remove those column vectors from $\mathbf{W}^k \in \mathbb{R}^{H \times K}$ of the kth source for which there is no corresponding word in the vocabulary of the target domain. As a result, we obtain $\mathbf{Z}^k$ as a latent topic feature matrix to be used in knowledge transfer to the target domain. Following the similar steps, we prepare a KB of $\mathbf{Z}$s such that each latent topic feature matrix from a source domain gets the same topic feature dimension as the target.

data	labels / classes
TMN*	world, us, sport, business, sci_tech, entertainment, health
AGnews	business, sci_tech, sports, world
20NS 20NSshort, 20NSsmall,	misc.forsale, comp.graphics, rec.autos, comp.windows.x, rec.sport.baseball, sci.space, rec.sport.hockey, soc.religion.christian, rec.motorcycles, comp.sys.mac.hardware, talk.religion.misc, sci.electronics, comp.os.ms-windows.misc, sci.med, comp.sys.ibm.pc.hardware, talk.politics.mideast, talk.politics.guns, talk.politics.misc, alt.atheism, sci.crypt
R21578title R21578	trade, grain, crude, corn, rice, rubber, sugar, palm-oil, veg-oil, ship, coffee, wheat, gold, acq, interest, money-fx, carcass, livestock, oilseed, soybean, earn, bop, gas, lead, zinc, gnp, soy-oil, dlr, yen, nickel, groundnut, heat, sorghum, sunseed, cocoa, rapeseed, cotton, money-supply, iron-steel, palladium, platinum, strategic-metal, reserves, groundnut-oil, lin-oil, meal-feed, sun-meal, sun-oil, hog, barley, potato, orange, soy-meal, cotton-oil, fuel, silver, income, wpi, tea, lei, coconut, coconut-oil, copra-cake, propane, instal-debt, nzdlr, housing, nkr, rye, castor-oil, palmkernel, tin, copper, cpi, pet-chem, rape-oil, oat, naphtha, cpu, rand, alum

Table 11: Label space of the corpora. TMN*:TMN or TMNtitle

C Experimental Setup

For DocNADE and DocNADEe in different knowledge transfer configurations, we follow the same experimental setup as in DocNADE and DocNADEe. We rerun DocNADE and DocNADEe using the code released for DocNADEe. For all the hyperpa-

Hyperparameter	Search Space
retrieval fraction	[0.02]
learning rate	[0.001]
hidden units, H	[200]
activation function (g)	sigmoid
iterations	[100]
λ^k	[1.0, 0.5, 0.1]
γ^k	[0.1, 0.01, 0.001]

Table 12: Hyperparameters in Generalization experiments of DocNADE, DocNADEe, LVT, GVT and MVT

Hyperparameter	Search Space
retrieval fraction	[0.02]
learning rate	[0.001]
hidden units, H	[200]
activation function (g)	tanh
iterations	[100]
λ^k	[1.0, 0.5, 0.1]
γ^k	[0.1, 0.01, 0.001]

Table 13: Hyperparameters search in the IR task, where λ^k and γ^k are weights for kth source.

			Scores on Target Corpus (*in sparse-data setting*)								
			20NSshort			TMNtitle			20NSsmall		
		Type	*PPL*	*COH*	*IR*	*PPL*	*COH*	*IR*	*PPL*	*COH*	*IR*
$+$	*MST*	LVT	667	.661	.308	670	.730	.535	610	.440	.286
		GVT	651	.658	.285	701	.712	.523	602	.460	.273
		MVT	667	.660	.309	667	.730	.535	608	.441	.293
		+ Glove	662	.677	.296	672	.731	.540	634	.412	.207
$\times$	*MST*	LVT	**640**	.678	.308	**663**	.732	.547	**596**	.442	.277
		GVT	658	.705	.305	704	.746	.550	599	.585	**.326**
		MVT	656	**.721**	**.314**	680	**.752**	**.556**	600	**.600**	**.285**
		+ Glove	644	.719	.293	687	**.752**	.538	609	.586	.282

Table 14: $\{\lambda, \gamma\}$ as Parameter ($+$) vs Hyperparameters ($\times$): Perplexity (PPL), topic coherence (COH) and precision@recall (IR) at retrieval fraction 0.02, when λ and γ are (1) learned with backpropagation, and (2) treated as hyperparameters. Results suggest the superiority of the second configuration.

rameters, optimal values are selected based on the performance on development set.

C.1 Experimental Setup for Generalization

We set the maximum number of training passes to 100, topics to 200 and the learning rate to 0.001 with *sigmoid* hidden activation. Since the baseline DocNADE and DocNADEe reported better scores in PPL for $H = 200$ topics than using 50, therefore we use $H = 200$ in our experiments. See Table 12 for hyperparameters used in generalization task, i.e., computing PPL.

C.2 Experimental Setup for IR Task

We treat all test documents as queries to retrieve a fraction of the closest documents in the original training set using cosine similarity between their document vectors. To compute retrieval precision for each fraction (e.g., 0.02), we average the number of retrieved training documents with the same label as the query.

We set the maximum number of training passes to 100, topics to 200 and the learning rate to 0.001 with *tanh* hidden activation. Since the baseline DocNADE and DocNADEe reported better scores in precision for the retrieval task for $H = 200$ topics than using 50, therefore we use $H = 200$ in our experiments. We follow the similar experimental setup as in DocNADEe. For model selection, we used the validation set as the query set and used the average precision at 0.02 retrieved documents as the performance measure. Note that the labels are not used during training. The class labels are only used to check if the retrieved documents have the same class label as the query document. To perform document retrieval, we use the same train/development/test split of documents as for PPL setup.

Given DocNADE, the representation of a document of size D can be computed by taking the last hidden vector $\mathbf{h}_D$ at the autoregressive step D. Since, the RSM and DocNADE strictly outperformed LDA, therefore we only compare DocNADE and its recent extension DocNADEe. We use the same number of topic dimensions ($H = 200$) across all the source and target in training DocNADE.

See Table 13 for the hyperparameters in the document retrieval task, where λ^k and γ^k are weights for kth source. We use the same grid-search for all the source domains. We set γ^k smaller than λ^k to control the degree of imitation of the source domain(s) by the target domain. We use the development set of the target corpus to find the optimal setting in different configurations of knowledge transfers from several sources.

C.3 $\{\lambda, \gamma\}$ as Parameter vs Hyperparameters

Here, we treat λ and γ as parameters of the model, instead of hyperparameters and learn them with backpropagation. We initialize each $\lambda^k = 0.5$ and $\gamma^k = 0.01$ for each of the sources. We perform experiments on short-text datasets in MST+LVT, MST+GVT and MST+MVT configurations. We evaluate the topic modeling using PPL, topic coherence and retrieval accuracy. Table 14 reports the scores, when λ and γ are (1) learned with backpropagation, and (2) treated as hyperparameters. The experimental results suggest that the second configuration performs better the former. Thus, we have reported scores considering $\{\lambda, \gamma\}$ as hyperparameters.

C.4 Reproducibility: Optimal Configurations of λ and γ

As mentioned in Tables 12 and 13, the hyperparameter λ^k takes on values in [1.0, 0.5, 0.1] for each of the word embeddings matrix $\mathbf{E}^k$ and γ^k in [0.1, 0.01, 0.001] for each of the latent topic features $\mathbf{Z}^k$, respectively for the k^{th} source domain. To determine an optimal configuration, we perform grid-search over the values and use the scores on the development set to determine the best setting. We have a common model for PPL and COH scores due to generalization.

To **reproduce** scores (best/bold in Table 5, we mentioned the best settings of (λ^k, γ^k) in MST+MVT configuration for each of the target and source combinations:

1. Generalization (PPL and COH) in MST+MVT when **target** is `20NSshort`: ($\lambda^{20NS} = 1.0$, $\gamma^{20NS} = 0.001$, $\lambda^{TMN} = 0.1$, $\gamma^{TMN} = 0.001$, $\lambda^{R21578} = 0.5$, $\gamma^{R21578} = 0.001$, $\lambda^{AGnews} = 0.1$, $\gamma^{AGnews} = 0.001$

2. Generalization (PPL and COH) in MST+MVT when **target** is `TMNtitle`: ($\lambda^{20NS} = 0.1$, $\gamma^{20NS} = 0.001$, $\lambda^{TMN} = 1.0$, $\gamma^{TMN} = 0.001$, $\lambda^{R21578} = 0.5$, $\gamma^{R21578} = 0.001$, $\lambda^{AGnews} = 1.0$, $\gamma^{AGnews} = 0.001$

3. Generalization (PPL and COH) in MST+MVT when **target** is `R21578title`: ($\lambda^{20NS} = 0.1$, $\gamma^{20NS} = 0.001$, $\lambda^{TMN} = 0.5$, $\gamma^{TMN} = 0.001$, $\lambda^{R21578} = 1.0$, $\gamma^{R21578} = 0.001$, $\lambda^{AGnews} = 1.0$, $\gamma^{AGnews} = 0.001$

4. Generalization (PPL and COH) in MST+MVT when **target** is `20NSsmall`: ($\lambda^{20NS} = 0.5$, $\gamma^{20NS} = 0.001$, $\lambda^{TMN} = 0.1$, $\gamma^{TMN} = 0.001$, $\lambda^{R21578} = 0.1$, $\gamma^{R21578} = 0.001$, $\lambda^{AGnews} = 0.1$, $\gamma^{AGnews} = 0.001$

5. Generalization (PPL and COH) in MST+MVT when **target** is `Ohsumedtitle`: ($\lambda^{AGnews} = 0.1$, $\gamma^{AGnews} = 0.001$, $\lambda^{PubMed} = 1.0$, $\gamma^{PubMed} = 0.001$

6. Generalization (PPL and COH) in MST+MVT when **target** is `Ohsumed`: ($\lambda^{AGnews} = 0.1$, $\gamma^{AGnews} = 0.001$, $\lambda^{PubMed} = 1.0$, $\gamma^{PubMed} = 0.001$

7. IR in MST+MVT when **target** is `20NSshort`: ($\lambda^{20NS} = 1.0$, $\gamma^{20NS} = 0.1$, $\lambda^{TMN} = 0.5$, $\gamma^{TMN} = 0.01$, $\lambda^{R21578} = 0.1$, $\gamma^{R21578} = 0.001$, $\lambda^{AGnews} = 1.0$, $\gamma^{AGnews} = 0.01$

8. IR in MST+MVT when **target** is `TMNtitle`: ($\lambda^{20NS} = 0.1$, $\gamma^{20NS} = 0.01$, $\lambda^{TMN} = 1.0$, $\gamma^{TMN} = 0.01$, $\lambda^{R21578} = 0.1$, $\gamma^{R21578} = 0.01$, $\lambda^{AGnews} = 0.5$, $\gamma^{AGnews} = 0.001$

9. IR in MST+MVT when **target** is `R21578title`: ($\lambda^{20NS} = 0.1$, $\gamma^{20NS} = 0.01$, $\lambda^{TMN} = 1.0$, $\gamma^{TMN} = 0.01$, $\lambda^{R21578} = 1.0$, $\gamma^{R21578} = 0.01$, $\lambda^{AGnews} = 0.5$, $\gamma^{AGnews} = 0.001$

10. IR in MST+GVT when **target** is `20NSsmall`: ($\gamma^{20NS} = 0.01$, $\gamma^{TMN} = 0.01$, $\gamma^{R21578} = 0.1$, $\gamma^{AGnews} = 0.01$

11. IR in MST+MVT when **target** is `Ohsumedtitle`: ($\lambda^{AGnews} = 0.1$, $\gamma^{AGnews} = 0.001$, $\lambda^{PubMed} = 1.0$, $\gamma^{PubMed} = 0.1$

12. IR in MST+MVT when **target** is `Ohsumed`: ($\lambda^{AGnews} = 0.1$, $\gamma^{AGnews} = 0.001$, $\lambda^{PubMed} = 0.5$, $\gamma^{PubMed} = 0.1$

The hyper-parameters mentioned above also applies to a single source transfer configuration.