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.

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

| Topic | Topic Words | Topic Label |
|-------|-------------|-------------|
| Z₁(S₁) | profit, growth, stocks, apple, fall, consumer, buy, billion, shares | Trading |
| Z₂(S₂) | smartphone, ipad, apple, app, iphone, devices, phone, tablet | Product Line |
| Z₃(S₃) | microsoft, mae, linx, ibm, ios, apple, sp, windows, software | Operating System |
| Z₄(T) | apple, talk, computers, shares, disease, driver, electronics, profit, ios | ? |

Table 1: Coherent (Z₁–Z₃) vs Incoherent (Z₄) topics from high-resource (S₁–S₃) and low-resource (T) texts
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 ($S^1$-$S^3$) domains whereas $Z_4$ from a low-resource target domain $T$ (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 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.

\[
\text{apple} \Rightarrow \text{apples, pear, fruit, berry, pears, strawberry}
\]

\[
\text{fall} \Rightarrow \text{falling, falls, drop, tumble, rise, plummet}
\]

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 $S^1$-$S^3$. 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.

**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](https://github.com/YatinChaudhary/Multi-view-Multi-source-Topic-Modeling).

**Table 2: Description of the notations used in this work**

| Notation | Description |
|----------|-------------|
| LVT, GVT | Local-view Transfer, Global-view Transfer |
| MVT, MST | Multi-view Transfer, Multi-source Transfer |
| $T, S$  | A target domain, a set of source domains |
| $V, \mathcal{K}, \mathcal{L}$ | An input document, 4th 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 $T$ |
| $\mathbf{U} \in \mathbb{R}^{K \times H}$ | Decoding matrix of DocNADE |
| $\mathbf{A}^k \in \mathbb{R}^{E \times K}$ | Degree of relevance of $E^k$ in $T$ |
| $\mathbf{Z}^k \in \mathbb{R}^{H \times K}$ | Degree of imitation of $Z^k$ by $W$ |
| $\mathbf{E}^k \in \mathbb{R}^{E \times K}$ | Word embeddings of 4th source |
| $\mathbf{A}^k \in \mathbb{R}^{H \times H}$ | Topic embeddings of 4th source |
| $b \in \mathbb{R}^k$, $c \in \mathbb{R}^H$ | Topic-alignment in $T$ and $Z^k$ |
| $DC$    | Visible-bias, hidden-bias |
| $\lambda_k$ | Degree of imitation of $Z_k$ |
| $\gamma_k$ | Degree of relevance of $Z_k$ |
| $R^k$   | Degree of imitation of $Z_k$ |

*Note: $\mathcal{K}$, $\mathcal{L}$, $\mathbf{E}$, $\mathbf{H}$, $\mathbf{A}$, $\mathbf{Z}$, $\lambda$, $\gamma$, $R$, $b$, $c$, $DC$ are representing the notations used in this work. The meanings of the notations are described in Table 2.*
2 Knowledge-Aware Topic Modeling

Consider a sparse target domain $T$ and a set of source domains $S$, we first prepare two knowledge bases (KBs) of representations (or embeddings) from document collections of each of the $|S|$ sources: (1) WordPool: a KB of pretrained word embeddings matrices $\{E^1, ..., E^{|S|}\}$, where $E^k \in \mathbb{R}^{K \times |V|}$, and (2) TopicPool: a KB of pretrained latent topic embeddings $\{Z^1, ..., Z^{|S|}\}$, where $Z^k \in \mathbb{R}^{H \times K}$ encodes a distribution over a vocabulary of $K$ words. Here, $k \in [1, ..., |S|]$ in superscript indicates knowledge of $k$th source, and $E$ and $H$ are word embedding and latent topic dimensions, respectively. While topic modeling on $T$, 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 $v = (v_1, ..., v_D)$ 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 $p(v) = \prod_{i=1}^{D} p(v_i | v_{<i})$ such that each autoregressive conditional $p(v_i | v_{<i})$ is modeled by a ff-net using preceding words $v_{<i}$ in the sequence:

$$h_i(v_{<i}) = g(c + \sum_{q<i} W_{v_i}. w_q) \text{ and } g = \{\text{sigmoid}, \text{tanh}\}$$

$$p(v_i = w | v_{<i}) = \frac{\exp(b_w + U_{w,.} h_i(v_{<i}))}{\sum_{w'} \exp(b_{w'} + U_{w,.} h_i(v_{<i}))}$$

for each word $i \in \{1, ..., D\}$ where $v_{<i}$ is the subvector consisting of all $v_q$ such that $q < i$ i.e., $v_{<i} \in \{v_1, ..., v_{i-1}\}$. $g(.)$ is a non-linear activation function, $W \in \mathbb{R}^{H \times K}$ and $U \in \mathbb{R}^{K \times H}$ are weight matrices, $c \in \mathbb{R}^H$ and $b \in \mathbb{R}^K$ 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 $i$th autoregressive step, where the parameter $W$ is shared in the feed-forward networks and $h_i$ encodes latent document-topic proportion. The value of each unit $j$ in the hidden vector signifies contribution of the $j$th topic in the proportion. Importantly, the topic-word matrix $W$ has a property that the column vector $W_{v_i}$ corresponds to embedding of the word $v_i$, whereas the row vector $W_j$ encodes latent features for the $j$th topic (i.e., topic-word distribution). We leverage this property to introduce external knowledge via word and topic embeddings.
We describe our transfer learning framework in word matrix for each of the DCs, where its column-Algorithm 1

Computation of (linear complexity) due to NADE architecture that

E

\[ \sum_{v} \exp (p_{vw} + W_{vw} h_{i}(v_{ci})) \]

1: Initialize: \( a \leftarrow c \) and \( p(v) \leftarrow 1 \)

2: for word \( i \) from 1 to \( D \) do

3: Compute \( i^{th} \) position-dependent hidden:

\( h_{i}(v_{ci}) \leftarrow g(a) \), where \( g \in \{ \text{sigmoid, tanh} \} \)

4: Compute \( i^{th} \) autoregressive conditional:

\( p(v_{ci} | v_{<ci}) \leftarrow \sum_{w} \exp (p_{vw} + W_{vw} h_{i}(v_{ci})) \)

5: Memorize: \( p(v) \leftarrow p(v)p(v_{ci}) \)

6: Compute pre-activation for word \( i \):

\( a \leftarrow a + W_{v,i} \)

7: if LVT then

8: Get word-embeddings \( E \) from WordPool

9: Introduce prior knowledge \( E \) for word \( i \):

\[ \text{scheme (i): } a \leftarrow a + \sum_{k=1}^{N} \lambda^{k} E_{k}^{v} \]

\[ \text{scheme (ii): } \hat{e}_{i} \leftarrow \text{concat}(E_{1,v}, \ldots, E_{N,v}) \]

10: Loss (negative log-likelihood): \( L(v) \leftarrow - \log p(v) \)

11: if GVT then

12: Topic-embedding with controlled topic-imitation:

\[ \Delta \leftarrow \sum_{k=1}^{N} \gamma^{k} \sum_{j=1}^{H} \| A_{k}^{j} W - Z_{k} \|^{2} \]

13: Overall loss with controlled topic-imitation:

\( L(v) \leftarrow L(v) + \Delta \)

14: Minimize \( L(v) \) using stochastic gradient descent

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

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

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 \( \text{WordPool, TopicPool} \), 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 \( E^{k} \) and \( Z^{k} \), respectively for the \( k^{th} \) source. See \textit{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 \textit{True}, line #7), we perform transfer learning on a target \( T \) using the WordPool of pretrained word embeddings \( \{E^{1}, \ldots, E^{|S|}\} \) from several sources \( S \) (i.e., multi-source) under the two schemes:

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

\[ h_{i}(v_{ci}) = g(c + \sum_{q<i} W_{i,v_{q}} + \sum_{q<i} \sum_{k=1}^{N} \lambda^{k} E_{k}^{v_{q}}) \]

Here, \( k \) refers to the \( k^{th} \) source and \( \lambda^{k} \) is a weight for \( E^{k} \) that controls the amount of knowledge transferred in \( T \), based on cross-domain overlap.

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

For \( q \in \{i, \ldots, D\} : \hat{e}_{q} = \text{concat}(E_{1,v_{q}}, \ldots, E_{N,v_{q}}) \)

\[ h_{i}(v_{ci}) = g(c + \sum_{q<i} W_{i,v_{q}} + \sum_{q<i} P \cdot \hat{e}_{q}) \]

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 \( \lambda \) for every source. To better guide TM, we also introduce pre-trained contextualized word embedding from BERT, concatenating with \( \hat{e}_{q} \).

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

\[ L(v) = - \log p(v) + \sum_{k=1}^{N} \gamma^{k} \sum_{j=1}^{H} \| A_{k}^{j} W - Z_{k}^{j} \|^{2} \]
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, $L<15$. $S^3$ is also used in target. '-' : unlabeled data.

Here, $A^k \in \mathbb{R}^{H \times H}$ aligns latent topics in the target $T$ and $k$th source, and $\gamma^k$ governs the degree of imitation of topic features $Z^k$ by $W$ in $T$. Consequently, the generative process of learning meaningful topics in $W$ of the target domain $T$ is guided by relevant topic features $\{Z^k\}_{k=1}^{S} \in$ TopicPool.

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

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

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 $T$ 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 $S$, we use five large corpora (20NS, R21578, TMN, AGnews and PubMed) in different label spaces (i.e, domains). Here, the corpora ($T^5$, $T^6$ and $S^5$) belong to medical and others to news.

Additionally, Table 4 suggests domain overlap (label match) in the target and source corpora, where we define 3 types of overlap: $I$ (identical) if all labels match, $R$ (related) if some labels match, and $D$ (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

\[Z_{ij} \sim \text{Gauss-LDA}, \quad \text{for} \quad 0 \leq i < |S|, \quad j = 1, \ldots, K,\]

\[\gamma_{jk} \sim \text{Gauss-LDA}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<k}) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]

\[p(v_{<j} | v_{<j} = \text{ArgMax}(\gamma_{jk})) = \frac{1}{Z_{ij}^k}, \quad \text{for} \quad 0 \leq j < K, \quad k = 1, \ldots, S,\]
We use 3 strategies: doc2vec (Le and Mikolov, Gupta et al., 2019a). They do not leverage pre-trained word embeddings of it's words from Glove and BERT). Here, Bold vs DocNADE.

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 corporea and evaluate on the target corpus $T$, and

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

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:

$$PPL = \exp \left( \frac{1}{N} \sum_{t=1}^{N} \frac{1}{|v_t|} \log p(v_t) \right),$$

where $N$ and $|v_t|$ are the number of documents and words in a document $v_t$, respectively.

Tables 6 and 7 quantitatively show PPL scores on the five target corporea 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
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 DocNADE. 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: Ohsumedtitle 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 20NSmall dataset, 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 20NSmall, 9.21% (.569 vs .521) on
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.

### 3.4 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.

For topic level inspection, we first extract topics using the rows of 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 W. Table 10
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.

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.
Rajarshi Das, Manzil Zaheer, and Chris Dyer. 2015. **Gaussian Ida 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.** *TAACL*, 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, Matt 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 \( T \) following the similar experimental setup as in DocNADE: (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 \( S \): (1) 20NS: 20NewsGroups corpus, a collection of new 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 selection. 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 \( W_j \) gives a word vector of the word \( v_i \), therefore the dimension of word embeddings in each of the \( E^k \) is same (i.e., \( H = 200 \)). Thus, we prepare the knowledge base of word representations \( E^k \) from \( k \)th source using DocNADE, where each word vector is of \( H = 200 \) dimension.

Since the row vector of \( W_j \) in DocNADE encodes \( j \)th topic feature, therefore each latent topic (i.e., row) in feature matrix \( W \) 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. \( W^k \in \mathbb{R}^{H \times K} \) of \( k \)th source.

For a target corpus of vocabulary size \( K' \), the DocNADE learns a feature matrix \( W^T \in \mathbb{R}^{H \times K'} \). Similarly, \( W^k \in \mathbb{R}^{H \times K} \) for \( k \)th 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 \( W^k \in \mathbb{R}^{H \times K} \) of the \( k \)th source for which there is no corresponding word in the vocabulary of the target domain. As a result, we obtain \( Z^k \) 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 \( Zs \) such that each latent topic feature matrix from a source domain gets the same topic feature dimension as the target.

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-

| data | labels / classes |
|------|-----------------|
| 20NS | world, us, sport, business, sci_tech, entertainment, health |
| 20NSshort | 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.mideast, alt.atheism, sci.crypt |
| 20NSsmall | veg-oil, ship, coffee, wheat, gold, acq, interest, money-fin, caracas, livetstock, oilseed, soybean, earn, bop, gas, lead, zinc, gap, soy-oil, dte, 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, bre, coconut, coconnut-oil, copra-cake, propane, instal-debt, nzdlr, housing, nkr, rye, castor-oil, palmkernel, tin, copper, cpi, pet-chem, rape-seed, oat, naphtha, epu, rand, alum |

Table 11: Label space of the corpora. TMN*:TMN or TMNtitle
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 \( h_D \) 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 \( \lambda^k \) and \( \gamma^k \) are weights for \( k \)-th source.

we used the grid-search for all the source domains. We set \( \gamma^k \) smaller than \( \lambda^k \) 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 \( \lambda \) and \( \gamma \) 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

### 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 \( (g) \) | sigmoid          |
| iterations        | [100]                |
| \( \lambda^k \)   | [1.0, 0.5, 0.1]   |
| \( \gamma^k \)    | [0.1, 0.01, 0.001] |

### Table 13: Hyperparameters search in the IR task, where \( \lambda^k \) and \( \gamma^k \) are weights for \( k \)-th source.

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

### Table 14: \( \{\lambda, \gamma\} \) as Parameter (+) vs Hyperparameters (×): Perplexity (PPL), topic coherence (COH) and precision@recall (IR) at retrieval fraction 0.02, when \( \lambda \) and \( \gamma \) are (1) learned with backpropagation, and (2) treated as hyperparameters. Results suggest the superiority of the second configuration.
the scores, when \( \lambda \) and \( \gamma \) 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 \( \lambda \) and \( \gamma \)
As mentioned in Tables 12 and 13, the hyperparameter \( \lambda^k \) takes on values in \([1.0, 0.5, 0.1]\) for each of the word embeddings matrix \( E^k \) and \( \gamma^k \) in \([0.1, 0.01, 0.001]\) for each of the latent topic features \( Z^k \), 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 \( \{ \lambda^k, \gamma^k \} \) in MST+MVT configuration for each of the target and source combinations:

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

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

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

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

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

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

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

8. IR in MST+MVT when target is TMNtitle: \( \lambda^{20NS} = 0.1, \quad \gamma^{20NS} = 0.01, \)
   \[ \lambda^{TMN} = 0.1, \quad \gamma^{TMN} = 0.01, \quad \lambda^{R21578} = 0.1, \]
   \[ \lambda^{AGnews} = 0.5, \quad \gamma^{AGnews} = 0.001 \]

9. IR in MST+MVT when target is R21578title: \( \lambda^{20NS} = 0.1, \quad \gamma^{20NS} = 0.01, \)
   \[ \lambda^{TMN} = 1.0, \quad \gamma^{TMN} = 0.01, \]
   \[ \lambda^{AGnews} = 1.0, \quad \gamma^{AGnews} = 0.001 \]

10. IR in MST+GVT when target is 20NSsmall: \( \gamma^{20NS} = 0.01, \quad \gamma^{TMN} = 0.01, \quad \gamma^{R21578} = 0.1, \)
   \[ \lambda^{AGnews} = 0.01 \]

11. IR in MST+MVT when target is Ohsumed:
    \( \lambda^{AGnews} = 0.1, \quad \gamma^{AGnews} = 0.001, \)
    \[ \lambda^{PubMed} = 1.0, \quad \gamma^{PubMed} = 0.1 \]

12. IR in MST+MVT when target is Ohsumed:
    \( \lambda^{AGnews} = 0.1, \quad \gamma^{AGnews} = 0.001, \)
    \[ \lambda^{PubMed} = 0.5, \quad \gamma^{PubMed} = 0.1 \]

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