GINopic: Topic Modeling with Graph Isomorphism Network

Suman Adhya and **Debarshi Kumar Sanyal**
Indian Association for the Cultivation of Science, Jadavpur, Kolkata-700032, India
adhyasuman30@gmail.com, debarshi.sanyal@iacs.res.in

Abstract

Topic modeling is a widely used approach for analyzing and exploring large document collections. Recent research efforts have incorporated pre-trained contextualized language models, such as BERT embeddings, into topic modeling. However, they often neglect the intrinsic informational value conveyed by mutual dependencies between words. In this study, we introduce GINopic, a topic modeling framework based on graph isomorphism networks to capture the correlation between words. By conducting intrinsic (quantitative as well as qualitative) and extrinsic evaluations on diverse benchmark datasets, we demonstrate the effectiveness of GINopic compared to existing topic models and highlight its potential for advancing topic modeling.

https://github.com/AdhyaSuman/GINopic

1 Introduction

The rise in digital text data makes organizing them manually by theme increasingly difficult. Topic modeling plays a significant role here (Newman et al., 2010; Boyd-Graber et al., 2017; Adhya and Sanyal, 2022), as it can uncover the underlying topics in documents in an unsupervised manner. In topic modeling, we assume that each document is a mixture of topics and these latent topics are also defined as distribution over the words.

Motivation: Recent approaches to neural topic modeling (Bianchi et al., 2021a,b; Grootendorst, 2022) focus on the representation of the document as a sequence of words, which captures the contextual information. However, words in a document may be correlated to each other in a much more complex manner. So, why not explicitly consider these word dependency patterns while learning the topics? Several studies in the field of topic modeling delve into the representation of documents using graphs. In this context, nodes signify words, and edges depict relationships between words, such as syntax or semantic relations. For instance, in the case of short texts, the Graph Biterm Topic Model (GraphBTM) (Zhu et al., 2018), an extension of the Biterm Topic Model (BTM) (Yan et al., 2013), represents word co-occurrence as a graph, with nodes representing words and weighted edges reflecting the counts of corresponding biterms. Despite GraphBTM's emphasis on capturing word dependencies, it has been reported to exhibit poor performance (Shen et al., 2021). Additionally, its computational cost escalates with an expanding vocabulary, as it constructs a single graph using the entire vocabulary. In contrast, the Graph Neural Topic Model (GNTM) (Shen et al., 2021) employs a directed graph with word dependencies as edges between word nodes to incorporate semantic information from words in documents. However, GNTM considers word dependency solely by linking words within a small sliding window for a given document. This limitation makes it impossible to account for word dependencies that fall outside of that specific window. Furthermore, the computational complexity of generating document graphs increases with the length of the window.

Approach: To model the mutual dependency between words while addressing the existing issues of incorporation of document graphs into topic modeling, we developed a neural topic model that takes the word similarity graphs for each document, where the word similarity graph is constructed using word embeddings to capture the complex correlations between the words. These document graphs along with their unordered frequency-based text representation are then used as input. We have also used the Graph Isomorphism Network (GIN) to obtain the representation for each document graph. We have used GIN as it is provably the maximally powerful GNN under the neighborhood aggregation framework. It is as powerful as the Weisfeiler-Lehman graph isomorphism test (Xu et al., 2019).

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 6171–6183

Contributions: In summary, our work presents the following key contributions:

- We introduce GINopic, a neural topic model that leverages a graph isomorphism network to enhance word correlations in topic modeling.
- We perform a comprehensive analysis through quantitative, qualitative, and task-specific evaluations. Additionally, we visualize the latent spaces generated by our model to assess its capability to disentangle the latent representations of documents.
- We also conducted a sensitivity analysis for the selection of GIN among the GNNs and the choice of our graph construction methodology.

2 Related Work

Topic modeling processes extensive document collections efficiently, preserving key statistical relationships for tasks like classification, novelty detection, summarization, and similarity judgments. Traditional models like Latent Dirichlet Allocation (LDA) (Blei et al., 2003), Probabilistic Latent Semantic Index (pLSI) (Hofmann, 2013), and Correlated Topic Model (CTM) (Lafferty and Blei, 2005) use sampling-based algorithms or variational inference, but their design is limited by the need for careful selection, which limits the flexibility and scalability of model design.

Recent advancements in neural variational inference, particularly Auto Encoding Variational Bayes (AEVB) (Kingma and Welling, 2014), simplify posterior computation. Neural Variational Document Model (NVDM) (Miao et al., 2016) is the pioneer VAE-based topic model. However, following the traditional topic models of applying Dirichlet-prior to the document-topic distribution becomes challenging due to the limitations of the reparametrization trick. Autoencoding Variational Inference For Topic Models (AVITM) (Srivastava and Sutton, 2017) resolves this by using Laplacian approximation of the Dirichlet parameter with Gaussian parameters. CombinedTM (Bianchi et al., 2021a) extends AVITM by incorporating sentence BERT embeddings alongside Bag-of-Words (BoW) representations. ZeroShotTM (Bianchi et al., 2021b) further extends this approach, relying solely on SBERT embeddings, ignoring word co-occurrence relations in input documents.

Numerous contemporary methodologies incorporate Graph Neural Networks (GNNs) for topic modeling. In terms of the graph construction task, the Graph Biterm Topic Model (GraphBTM) (Zhu et al., 2018) and the Graph Neural Topic Model (GNTM) (Shen et al., 2021) employ a moving window-based approach with a specified window length to model word co-occurrence relationships, necessitating careful window length selection. The *graph topic model* (Zhou et al., 2020) constructs document graphs based on TF-IDF scores, capturing relationships with graph convolutions. *Topic modeling with knowledge graph embedding* (Li et al., 2019) incorporates external knowledge graphs. The *graph attention topic network* (Yang et al., 2020) addresses overfitting in probabilistic latent semantic indexing with amortized inference and word embeddings. The *graph relational topic model* (Xie et al., 2021) explores document relationships using higher-order graph attention networks.

3 Proposed Methodology

Recognizing the challenges in topic modeling, we acknowledge the necessity of capturing semantic similarity among words in a document. Additionally, we note the importance of addressing the graph construction issue and obtaining unique representations for dissimilar document graphs. In response to these challenges, we have introduced the Graph Isomorphism Network-based neural topic model, abbreviated as GINopic. The following subsections provide a detailed explanation of the graph construction methodology, model framework, and objective function.

3.1 Graph Construction

Let $\mathcal{D}$ be defined as the set of all documents, $\mathcal{V}$ as the set of all words in the corpus such that $|\mathcal{V}| = V$, and $\mathcal{E} \in \mathbb{R}^{V \times \tau}$ as the word embeddings matrix such that its i-th row $\mathcal{E}_i \in \mathbb{R}^{\tau}$, corresponds to the word $w_i \in \mathcal{V}$. Now for a document $d \in \mathcal{D}$, which contains a subset of words from $\mathcal{V}$, specifically V' words, we define its weighted undirected document graph G_d as the adjacency matrix $A = (a_{ij})_{1 \leq i,j \leq V'}$, where the elements a_{ij} are determined as follows:

$$a_{ij} = \begin{cases} 0 & \text{if } \mathrm{Sim}(\mathcal{E}_i, \mathcal{E}_j) < \delta \\ \mathrm{Sim}(\mathcal{E}_i, \mathcal{E}_j) & \text{otherwise} \end{cases} \quad (1)$$

Here, $\mathrm{Sim}(\mathcal{E}_i, \mathcal{E}_i)$ represents the cosine similarity between the word embedding vectors $\mathcal{E}_i$ and

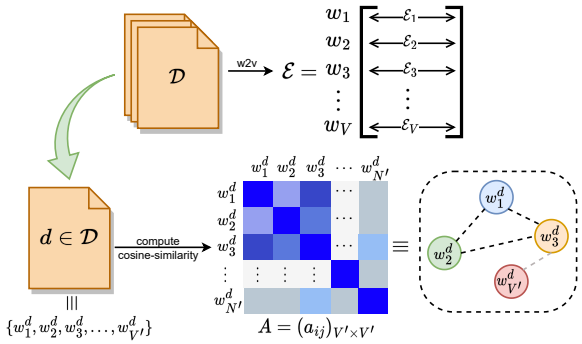

Figure 1: Graph construction methodology.

$\mathcal{E}_j$. In Eq. (1), δ is a threshold that indicates if the similarity score between two words is less than δ then there should not be any edge between them. The choice of this threshold is crucial, as opting for a lower value makes the connections in G_d denser, consequently elevating computational complexity. Conversely, opting for a higher threshold value leads to a sparse document graph, a scenario also undesired. The optimal choice of δ depends on the type of corpus. To balance these factors, we consider δ as a hyperparameter to be tuned.

3.2 Model Architecture

The proposed model GINopic comprises a document graph representation learning network followed by an encoder which is followed by a decoder. The output of the graph representation learning network is concatenated with the TF-IDF representation of the input document before feeding into the encoder. The framework is shown in Fig. 2 and a detailed description of these networks is described in the following.

3.2.1 Graph Representation Learning

The Weisfeiler-Lehman (WL) test serves as a means to evaluate the isomorphism of two provided graphs. The graph representation learning module within the proposed model is designed to process document graphs as its input and produce a unique representation for each topologically distinct document graph, identified through the WL test. To model this injective mapping we have used the *Graph Isomorphism Network* (GIN), known for its equivalent expressive power to the WL graph kernel (Shervashidze et al., 2011). GIN is theoretically proven as the most powerful GNN (Xu et al., 2019). Mathematically, the layer-wise propagation rule for GIN at layer $l+1$ is defined as follows:

$$h_i^{(l+1)} = \text{MLP}^{(l+1)}\Big((1+\epsilon)h_i^{(l)} + \text{AGG}\Big(\{\omega_{ji}h_j^{(l)}, j \in N(i)\}\Big)\Big) \quad (2)$$

Here, $h_i^{(l)}$ represents the feature vector for the i-th node at layer l, $N(i)$ denotes the set of all neighbors for node i, ω_{ji} signifies the edge weight between the i-th and j-th nodes. The operator $\text{AGG}(\cdot)$ stands for aggregation, and ϵ is a parameter that can be learned or a fixed scalar value close to zero. Furthermore, $\text{MLP}^{(l+1)}$ represents the multi-layer perceptron for the $(l+1)$-th layer. After applying the L number of GIN layers, the encoding of a node essentially captures its L-th order neighborhood's information. The detailed transformations are: $\big[\text{GINConv}(\tau, H) \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow [\text{GINConv}(H, H) \rightarrow \text{BN} \rightarrow \text{ReLU}]^{L-2} \rightarrow \text{GINConv}(H, \tau') \rightarrow \text{BN}\big]$, where $\text{GINConv}(I, J)$ represents GIN layer with a MLP of input dimension I and output dimension J, H is the number of hidden units, BN is the batch normalization, and ReLU is the activation function. The final node embeddings of dimension τ' are then summed up to obtain the representation of the document graph as follows: $h_G = \sum_i h_i^{(L)}$.

3.2.2 VAE framework

Encoder Network: The encoder network of GINopic, takes the combination of graph representation ($h_G \in \mathbb{R}^{\tau'}$) and TF-IDF representation ($x_{\text{TFIDF}} \in \mathbb{R}^V$) of the input document. For this concatenation, h_G is first scaled to the dimension same as of x_{TFIDF} and then concatenated with x_{TFIDF}. Therefore, the resultant representation is $x = \text{CONCAT}\left(f_W(h_G), x_{\text{TFIDF}}\right)$, where $W \in \mathbb{R}^{V \times \tau'}$ is a matrix, representing linear transformation $f_W : \mathbb{R}^{\tau'} \rightarrow \mathbb{R}^V$ whose weights are to be learned.

Careful selection of the prior for our modeling assumption is crucial. In topic modeling, the Dirichlet distribution has been demonstrated (Wallach et al., 2009) as effective in assigning topic proportions for a given document. However, the reparametrization trick is limited to Gaussian distributions. To integrate the Dirichlet assumption into a VAE following the method proposed by (Srivastava and Sutton, 2017), we used the Laplacian

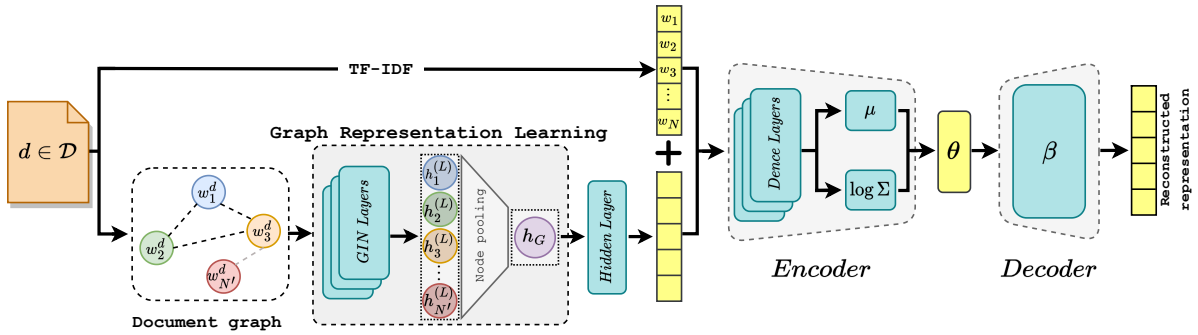

Figure 2: Proposed framework for GINopic model.

approximation to the $\mathrm{Dir}(\alpha)$ distribution:

$$\mu_{1\,k} = \log \alpha_k - \frac{1}{K} \sum_i \log \alpha_i$$

$$\Sigma_{1\,kk} = \frac{1}{\alpha_k} \left(1 - \frac{2}{K}\right) + \frac{1}{K^2} \sum_i^K \frac{1}{\alpha_i}$$

where, α_i is the i-th component of the K-dimensional Dirichlet's parameter, $\mu_{1\,k}$ is the k-th component of the vector $\mu_1 \in \mathbb{R}^K$ and $\Sigma_{1\,kk}$ is the k-th component of $\Sigma_1 \in \mathbb{R}^{K \times K}$, the diagonal covariance matrix. Given a prior distribution and the resultant input document representation vector x, the encoder outputs the posterior distribution $q_\phi(z|x) \equiv \mathcal{N}(\mu_0, \Sigma_0)$, where ϕ represents the weights of the encoder. The transformations in the encoder are: $\big[\mathrm{Linear}(2V, H') \rightarrow \mathrm{Softplus} \rightarrow [\mathrm{Linear}(H', H') \rightarrow \mathrm{Softplus}]^{L'-1} \rightarrow \mathrm{Dropout}(0.2)\big]$. This is followed by the two separate and similar transformations as follows: $\big[\mathrm{Linear}(H', K) \rightarrow \mathrm{BN}\big]$ for μ_0 and Σ_0 respectively. In these expressions, V represents the vocabulary size, H' and L' represent the number of hidden units and hidden layers respectively, Softplus is an activation function, and Dropout is a regularizer.

Sampling Procedure: A latent representation z is stochastically sampled from the posterior distribution $q_\phi(z|x)$ using the reparameterization trick (Kingma and Welling, 2014) as $z = \mu_0 + \Sigma_0^{1/2} \odot \epsilon$. The symbol $\odot$ denotes the Frobenius inner product and $\epsilon \sim \mathcal{N}(0, 1)$. The obtained latent representation z is then used as logit to a softmax function $\sigma(\cdot)$ in order to generate the document-topic distribution θ such that, $\theta = \sigma(z)$.

Decoder Network: In the decoder, the topic-word matrix β refers to the learnable weights of the decoder network. This matrix is utilized to reconstruct the word distribution $\hat{x}$ as: $\hat{x} = \sigma\left(\beta^\top \theta\right)$ Following (Srivastava and Sutton, 2017), we relaxed the simplex constraint on β, which is empirically shown to produce better topic quality. The transformations of the decoder network are, $\big[\mathrm{Linear}(K, V) \rightarrow \mathrm{BN} \rightarrow \mathrm{Softmax}\big]$, with σ employed in the output layer to generate the word distribution.

3.3 Training Objective

The objective function for GINopic is the same as ELBO which needs to be maximized in order to maximize the log-likelihood of the input data distribution. The loss function we seek to minimize is defined as:

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{\mathrm{RL}} + \mathcal{L}_{\mathrm{KL}} \\ &\equiv -\mathbb{E}_{z \sim q_\phi(z|x)}[p_\beta(x|z)] + \mathrm{D}_{\mathrm{KL}}\left(q_\phi(z|x) \| p(z)\right) \end{aligned} \tag{3}$$

In the above expression, the first term ($\mathcal{L}_{\mathrm{RL}}$) represents the reconstruction loss, quantified by the cross-entropy between the predicted output distribution $\hat{x}$ and the input vector x_{TFIDF}. On the other hand, the second term ($\mathcal{L}_{\mathrm{KL}}$) is the Kullback-Leibler (KL) divergence of the learned latent space distribution $q_\phi(z|x)$ from the prior $p(z)$ of the latent space.

4 Experimental Settings

We have conducted the experiments using OCTIS[1] (Terragni et al., 2021a), a comprehensive framework for comparing and optimizing topic models, available under the MIT License.

4.1 Datsets

In the experiments, we utilized five publicly available datasets. Among these, 20NewsGroups (**20NG**) and BBC News (**BBC**) (Greene and Cunningham, 2006) datasets were already included in

[1] https://github.com/MIND-Lab/OCTIS

Dataset	#Total Docs	#Tr Docs	#Ts/Va Docs	Avg. Doc. length	Labels
20NG	16309	11415	2447	48.020	20
BBC	2225	1557	334	120.116	5
SS	12270	8588	1841	13.104	8
Bio	18686	13080	2803	7.022	20
SO	15696	10986	2355	5.106	20

Table 1: Statistics of the used datasets.

Hyperpramerts	20NG	BBC	SS	Bio	SO
Graph construction threshold (δ):	0.4	0.3	0.2	0.05	0.1
Dim. of input node feature (τ):	2048	256	1024	1024	64
#GIN layers (L):	2	3	2	2	2
#Hidden layers in MLP:	1	1	1	1	1
Dim. of Hidden layers in MLP:	200	50	50	200	300
Dim. of output node feature (τ'):	768	512	256	256	512

Table 2: Value of the hyperparameters of GINopic for each dataset.

OCTIS in pre-processed formats. Additionally, we incorporated the SearchSnippets (**SS**), Biomedicine (**Bio**), and StackOverflow (**SO**) datasets (Qiang et al., 2022) and pre-processed them. A detailed description of these datasets is mentioned in Appendix A.1 and the pre-processing steps are mentioned in Appendix A.2. Statistical descriptions of these datasets can be found in Table 1. Each of these corpora was divided into training, validation, and testing sets, with a distribution ratio of 70% for training, 15% for validation, and 15% for testing, where the training part is used to train the model, the validation part is only used for the **GNTM** to modify the learning rate accordingly and the test part is used to conduct the extrinsic evaluation of the models.

4.2 Baselines

We conducted a comparative analysis of the proposed model **GINopic** with the graph-based topic models, namely **GraphBTM** (Zhu et al., 2018) and **GNTM** (Shen et al., 2021). Unfortunately, for other graph-based topic models, we could not access their source code, making it impossible to include them in our comparison. Beyond the graph-based models, our evaluation extended to various well-known neural and traditional topic models, including **ECRTM** (Wu et al., 2023), **CombinedTM** (Bianchi et al., 2021a), **ZeroShotTM** (Bianchi et al., 2021b), **ProdLDA** (Srivastava and Sutton, 2017), **NeuralLDA** (Srivastava and Sutton, 2017), **ETM** (Dieng et al., 2020), **LDA** (Blei et al., 2003), **LSI** (Dumais, 2004) and **NMF** (Zhao and Tan, 2017). A detailed description of the configurations of these baselines together with their implementation details can be found in Appendix B.

4.3 Hyperparameter Tuning

In GINopic, for a given dataset the hyperparameters that are tuned are mentioned in Table 2. Here, the hyperparameter tuning was conducted on each dataset, maintaining a topic count equal to the number of labels for 50 epochs. To ensure a fair comparison we have also tuned the hyperparameters for GNTM. However, due to computational limitations, we are unable to fine-tune the hyperparameters for GraphBTM.

5 Results and Discussions

We categorize our findings into the following sections: (1) quantitative evaluation (Section 5.1), (2) extrinsic evaluation (Section 5.2), (3) qualitative evaluation (Section 5.4), (4) latent space visualization (Section 5.3), and (5) sensitivity analysis (Section 5.5).

5.1 Quantitative Evaluation

In the quantitative evaluation, we have evaluated the topic models based on the generated topic quality measured by coherence and diversity metrics. To measure the topic coherence we have used **Normalized Pointwise Mutual Information** (**NPMI**) (Lau et al., 2014) and **Coherence Value** (**CV**) (Röder et al., 2015). NPMI is commonly utilized (Adhya et al., 2022) as a surrogate for human judgment of topic coherence, although some researchers also employ CV, despite its known issues. We measure the diversity of topics using **Inverted Rank-Biased Overlap** (**IRBO**) (Bianchi et al., 2021a), **Word Embedding-based Inverted Rank-Biased Overlap - Match** (**wI-M**), and **Word Embedding-based Inverted Rank-Biased Overlap - Centroid** (**wI-C**) (Terragni et al., 2021b). Higher values of NPMI, CV, IRBO, wI-C, and wI-M indicate better performance. These metrics are elaborately discussed in Appendix C.

Experimental Setup: For a given dataset we run all the models by varying the topic count in $\{20, 50, 100\} \cup \{k_{gold}\}$ where k_{gold} stands for the *golden topic count* which is the number of ground-truth labels in the dataset (since they are available for the datasets we used). The values of k_{gold} for **20NG**, **BBC**, and **M10** are 20, 5, and 8, respectively. For the robustness of the results, we have reported the mean value over 5 random runs for

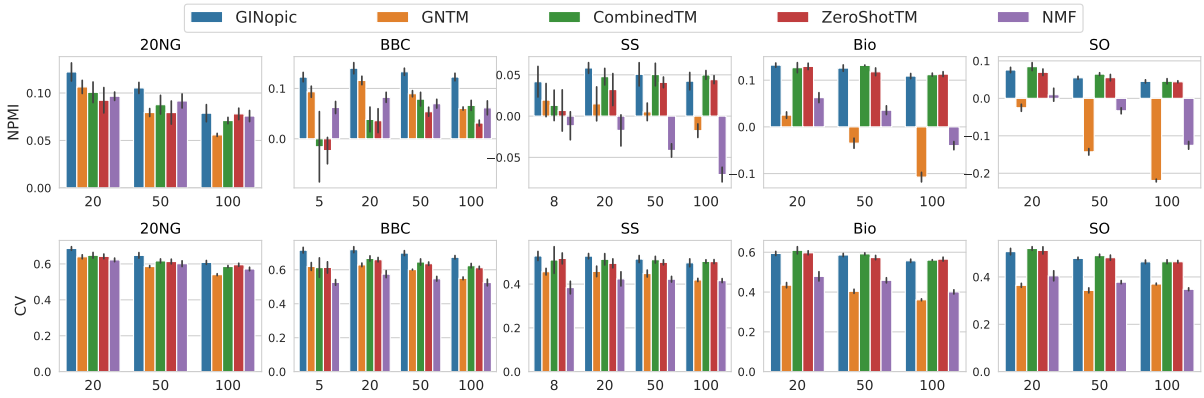

Figure 3: Topic coherence (NPMI and CV) scores for each topic count for top-5 topic models on five datasets.

Model	20NG		BBC		SS		Bio		SO	
	NPMI	CV	NPMI	CV	NPMI	CV	NPMI	CV	NPMI	CV
ECRTM	-0.145	0.363	-0.041	0.625	-0.388	0.474	-0.435	0.529	-0.416	0.526
CombinedTM	0.086	0.617	0.042	0.637	0.040	0.510	**0.123**	0.587	**0.065**	0.491
ZeroShotTM	0.083	0.617	0.024	0.630	0.031	0.504	0.120	0.579	0.056	0.486
ProdLDA	0.071	0.593	0.035	0.628	-0.001	0.486	0.105	0.571	0.042	0.473
NeuralLDA	0.045	0.500	-0.065	0.472	-0.114	0.400	-0.061	0.435	-0.177	0.407
ETM	0.050	0.528	0.030	0.452	-0.099	0.309	-0.136	0.140	-0.332	0.441
LDA	0.069	0.562	0.049	0.518	-0.165	0.376	-0.118	0.392	-0.174	0.345
LSI	-0.019	0.400	-0.042	0.406	-0.122	0.280	-0.118	0.392	-0.129	0.303
NMF	0.088	0.599	0.069	0.543	-0.035	0.412	0.019	0.446	-0.050	0.377
GraphBTM	0.017	0.605	-0.173	0.484	-0.322	0.444	-0.398	0.519	-0.451	0.558
GNTM	0.081	0.588	0.090	0.600	0.005	0.445	-0.039	0.400	-0.129	0.359
GINopic	**0.102**	**0.647**	**0.130**	**0.701**	**0.048**	**0.517**	**0.123**	**0.589**	0.059	**0.493**

Table 3: Comparison of topic models on five datasets. For each metric and each topic model, we mention the mean scores over topic counts $\{20, 50, 100\} \cup \{k_{gold}\}$.

a given model, a given dataset, and a given topic count.

Findings: We present coherence scores for all models across datasets in Table 3. Notably, GINopic achieves the highest coherence scores (both NPMI and CV) across most datasets, except for the **SO** dataset where it ranks second in NPMI score, following CombinedTM. However, GINopic still leads in CV score for the **SO** dataset. To provide a comprehensive comparison, we focus on the top 5 models based on their NPMI scores across all datasets. Figure 3 shows the mean and standard deviation of NPMI and CV scores for each topic count. The results establish the consistent superior performance of GINopic compared to existing models. In terms of diversity, Table 3 displays all three diversity scores. GINopic achieves the highest wI-M and wI-C diversity scores across most datasets, except for the **20NG** dataset where its wI-M score is comparable to ECRTM's highest score. ECRTM exhibits the highest IRBO scores across all datasets due to its embedding clustering regularization approach, despite its poor coherence scores indicating ineffective topic representation learning. IRBO scores of GINopic are also competitive, being close to the highest score across all datasets.

5.2 Extrinsic Evaluation

We have also incorporated an extrinsic task to assess the performance of the topic models, specifically by evaluating their predictive capabilities in a document classification task.

Experimental Setup: Our datasets include category labels for each document. We trained all models on the training subset of a particular dataset to generate k_{gold} topics. The resulting k_{gold}-dimensional *document-topic* vector serves as a representation of the document. A linear support vector machine is then trained on these representations, and model performance on the test subset is reported. We calculate the average accuracy over

Model	20NG			BBC			SS			Bio			SO		
	IRBO	wI-M	wI-C	IRBO	wI-M	wI-C	IRBO	wI-M	wI-C	IRBO	wI-M	wI-C	IRBO	wI-M	wI-C
ECRTM	**0.998**	**0.473**	0.852	**0.999**	0.454	0.848	**1.000**	0.442	0.839	**1.000**	0.433	0.838	**1.000**	0.382	0.825
CombinedTM	0.988	0.468	**0.895**	0.978	0.442	0.888	0.993	0.45	0.888	0.983	0.443	0.887	0.985	0.392	0.878
ZeroShotTM	0.986	0.467	0.894	0.964	0.435	0.887	0.99	0.448	0.888	0.983	0.445	0.885	0.985	0.393	0.879
ProdLDA	0.990	0.469	**0.895**	0.975	0.44	0.888	0.994	0.45	0.888	0.987	0.446	0.888	0.977	0.394	0.878
NeuralLDA	0.989	0.466	0.892	0.984	0.444	0.887	0.997	0.453	0.887	0.996	0.452	0.888	0.979	0.390	0.875
ETM	0.802	0.37	0.87	0.802	0.354	0.874	0.647	0.294	0.867	0.344	0.138	0.843	0.490	0.187	0.842
LDA	0.981	0.462	0.893	0.947	0.424	0.885	0.988	0.447	0.886	0.991	0.446	0.886	0.913	0.390	0.875
LSI	0.925	0.429	0.887	0.869	0.385	0.879	0.845	0.382	0.881	0.991	0.399	0.881	0.927	0.337	0.868
NMF	0.975	0.458	0.892	0.966	0.432	0.886	0.978	0.443	0.887	0.988	0.443	0.887	0.984	0.388	0.876
GraphBTM	0.971	0.462	0.852	0.986	0.448	0.846	0.947	0.421	0.836	0.924	0.427	0.837	0.958	0.374	0.821
GNTM	0.984	0.461	0.852	0.983	0.444	0.845	0.995	**0.454**	0.846	0.999	0.455	0.845	0.949	0.406	0.831
GINopic	0.989	0.468	**0.895**	0.992	**0.457**	**0.893**	0.998	**0.454**	**0.889**	0.983	**0.462**	**0.888**	0.986	**0.497**	**0.879**

Table 4: Comparison of topic models on five datasets. For each metric and each topic model, we mention the mean scores over topic counts $\{20, 50, 100\} \cup \{k_{gold}\}$.

five runs for each dataset and present the scores in Table 5.

Findings: Table 5 shows that GINopic attains the highest accuracy across all datasets, except for the 20NG dataset where it secures the second-highest accuracy, with the GNTM closely edging ahead.

Model	20NG	BBC	SS	Bio	SO
ECRTM	0.411	0.816	0.492	0.361	0.457
CombinedTM	0.397	0.796	0.706	0.493	0.715
ZeroShotTM	0.385	0.817	0.698	0.501	0.687
ProdLDA	0.385	0.752	0.662	0.489	0.674
NeuralLDA	0.297	0.575	0.464	0.376	0.403
ETM	0.370	0.754	0.496	0.083	0.072
LDA	0.428	0.798	0.440	0.364	0.412
LSI	0.329	0.337	0.343	0.402	0.660
NMF	0.350	0.785	0.415	0.437	0.708
GraphBTM	0.052	0.231	0.224	0.060	0.050
GNTM	**0.449**	0.806	0.222	0.049	0.053
GINopic	0.441	**0.888**	**0.713**	**0.566**	**0.785**

Table 5: Average accuracy scores in the document classification task for all the models trained with topic count k_{gold} for all five datasets.

5.3 Latent Space Visualization

We have further examined the latent space generated by GINopic. In topic modeling, documents are projected into a lower-dimensional latent (topic) space.

Experimental Setup: To visualize the latent space, we have trained GINopic for the topic count of k_{gold} associated with each of the five datasets. Following the training phase, we captured the document-topic distribution for each document. We applied the *Uniform Manifold Approximation and Projection* (UMAP) technique, a robust dimensionality reduction method (McInnes et al., 2018). UMAP transformed the k_{gold}-dimensional document-topic distribution into a two-dimensional representation, making it possible to visualize. Each document was assigned to a cluster based on its topic distribution vector $\boldsymbol{\theta}$, where the cluster was determined by selecting the topic with the highest probability. Figure 4 illustrates the clusters obtained for each dataset.

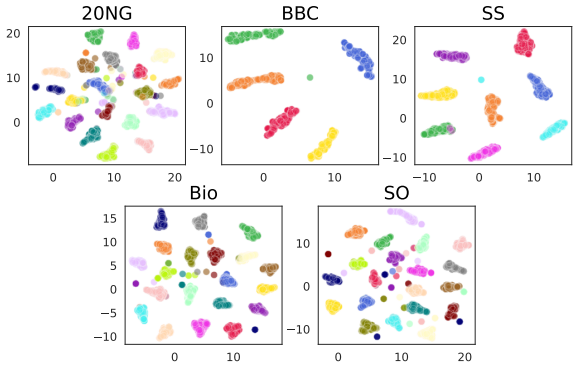

Figure 4: Latent space visualization for GINopic model across all five datasets.

Findings: The disentanglement of clusters is depicted in Figure 4 for each dataset. Notably, the clarity of disentanglement is more pronounced in the **BBC** and **SS** datasets compared to the other three datasets **20NG**, **Bio**, and **SO**. This difference can be attributed to the greater challenge of disentangling 20 different labels in the **20NG**, **Bio**, and **SO** datasets, as opposed to the **BBC** and **SS** datasets with fewer distinct labels.

5.4 Qualitative Evaluation

Since topic models operate as unsupervised methods, it is recommended to assess their performance not solely relying on automated estimates of topic coherence but also through manual evaluation of

Model	Topics
ECRTM	**turkish**, **soviet**, **bullet**, **minority**, **population**, **burn**, **jewish**, cold, prepare, joke draft, **baseball**, **game**, **shot**, blue, luck, stupid, programming, basically, **score** **bike**, **car**, **controller**, button, camera, strategy, win, black, atheism, attribute
CombinedTM	**german**, publish, **genocide**, **turkish**, **muslim**, **armenian**, book, representative, **european**, century **team**, **hockey**, **season**, **game**, draft, expansion, **ticket**, **play**, year, ice **car**, **engine**, **tire**, **bike**, **ride**, **brake**, good, problem, buy, **mile**
ZeroShotTM	**greek**, **turkish**, **minority**, **genocide**, state, **muslim**, **soviet**, **armenian**, **israeli**, **struggle** ranger, **hockey**, **playoff**, **team**, **game**, devil, king, pen, wing, period **motorcycle**, **bike**, clean, wave, **ride**, **wheel**, tip, **mirror**, remove, replace
ProdLDA	**arab**, **israeli**, **religious**, **people**, **religion**, **jewish**, solution, **territory**, understanding, **land** year, fund, money, spend, program, **player**, private, **team**, job, good eat, food, **car**, problem, **engine**, **brake**, stone, weight, pain, day
NeuralLDA	**army**, **muslim**, **genocide**, **international**, **turkish**, **village**, **armenian**, **population**, organize, enter **goal**, **win**, **score**, **play**, **wing**, **penalty**, **playoff**, **team**, **pass**, **game** **front**, clean, **ride**, foot, **bike**, bar, **engine**, pull, weight, remove
ETM	**armenian**, **people**, **turkish**, **village**, **kill**, **genocide**, **woman**, live, **soldier**, **jewish** good, year, **win**, **game**, back, **play**, make, post, line, **goal** **bike**, **engine**, mission, orbit, temperature, **car**, earth, space, planet, solar
LDA	**war**, **jewish**, **israeli**, **land**, **country**, **arab**, **peace**, **territory**, **force**, **attack** double, trade, **game**, **hockey**, final, **team**, star, **playoff**, king, regular **bike**, **ride**, hate, advice, bank, **motorcycle**, weight, good, instruction, surrender
LSI	**turkish**, drive, **war**, **armenian**, **russian**, **government**, **secret**, **military**, **power**, **jewish** year, car, scsi, love, bit, client, **team**, server, call, **player** access, **engine**, **power**, kill, database, word, **bus**, attack, disk, card
NMF	**kill**, **woman**, time, **soldier**, start, **child**, back, leave, **armenian**, **man** power, **play**, government, constitution, **team**, control, level, individual, idea, zone **car**, **engine**, price, buy, **bike**, **mile**, **ride**, make, **driver**, **tire**
GraphBTM	**armenian**, **afraid**, neighbor, clock, **soldier**, **turkish**, floor, **soviet**, **beat**, arrive **game**, **score**, car, engine, **play**, **goal**, **season**, **playoff**, **shot**, **player** **tire**, **bike**, connector, ide, **brake**, scsi, cable, **car**, **rear**, **engine**
GNTM	**israeli**, **arab**, **jewish**, policy, **land**, **territory**, **area**, peace, **human**, **population** **team**, **game**, **play**, **player**, **win**, year, good, call, point, time **tire**, **oil**, **brake**, **bike**, **paint**, weight, corner, air, **lock**, **motorcycle**
GINopic	**genocide**, **muslim**, **armenian**, **massacre**, **turkish**, **population**, **kill**, **government**, **troop**, **war** **team**, **win**, **score**, **baseball**, **game**, **player**, **hockey**, **playoff**, **goal**, **play** **car**, **bike**, **ride**, **brake**, **light**, **tire**, **engine**, **lock**, side, **mile**

Table 6: Some representative topics extracted from the **20NG** dataset with a topic count of 100. Relevant terms within each topic are emphasized in **bold**.

the topics, as emphasized by (Hoyle et al., 2021; Adhya and Sanyal, 2023).

Experimental Setup: We conducted a qualitative analysis of the topics, utilizing the **20NG** dataset and training all models with the golden topic count i.e. $k_{gold} = 20$. The results appear in Table 6. Note that the table exhibits aligned topics, wherein the first topic listed for one model is similar to the first topic for every other model, and the same goes for the rest of the topics, following the alignment method proposed by (Adhya et al., 2023). Additionally, words closely associated with a given topic are highlighted in **bold**.

Findings: In Table 6, we showcase three topics: "Armenian genocide", "Sports", and "Automobile" related. Across these distinct topics, GINopic consistently generates more correlated words compared to other models. This observation is supported by the consistently higher number of **bold** words for each topic in GINopic, indicating stronger word correlations than the other models.

5.5 Sensitivity Analysis

5.5.1 Choice of the Graph Neural Network

To empirically check the effectiveness of GIN over other GNNs in our model, we substitute GIN with Graph Attention Network (GAT) (Veličković et al., 2018), Graph SAmple and aggreGatE (Graph-

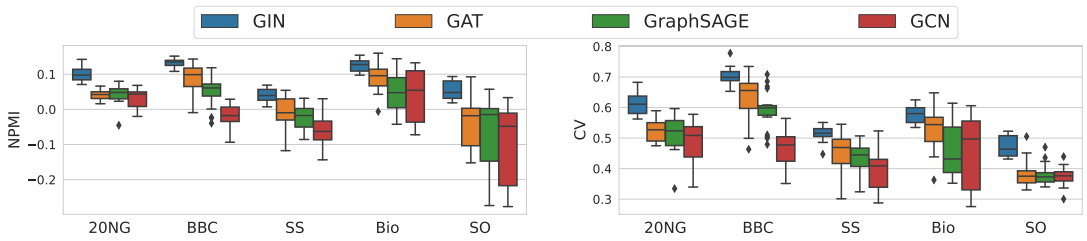

Figure 5: Box plot of topic coherence (NPMI and CV) scores incorporating GIN, GAT, GraphSAGE, and GCN in GINopic on five datasets.

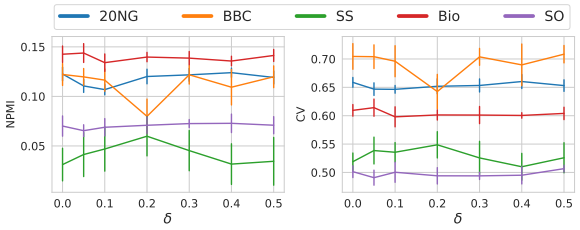

Figure 6: Coherence (NPMI and CV) scores for each dataset by varying the threshold (δ) value in $\{0.0, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$.

SAGE) (Hamilton et al., 2017) and Graph Convolutional Network (GCN) (Kipf and Welling, 2017).

Experimental Setup: We trained our proposed model on the five datasets, adjusting the topic count within the set $\{20, 50, 100\} \cup k_{gold}$. To ensure a fair comparison, we maintained consistent parameter values across all models, aligning them with those of GINopic.

Findings: In Figure 5, a box plot is presented, illustrating the NPMI and CV scores derived from five random runs for each model across the five datasets. The results indicate that the GIN-incorporated model consistently outperforms other GNN-based models across all datasets in terms of both the coherence measures.

5.5.2 Choice of the Graph Construction Threshold (δ)

We have examined how the graph construction threshold δ, as specified in Eq. (1), influences model performance and training time.

Experimental Setup: Given a dataset, we have trained our model for the corresponding k_{gold} number of topics by varying the value of δ over $\{0.0, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$. We reported the mean and standard deviation of the coherence scores (NPMI and CV) over 5 random runs in Figure 6.

Findings: Figure 6 illustrates the optimal threshold values that maximize coherence scores (NPMI and CV) for each dataset. This threshold signifies that if the similarity between two nodes in a document graph falls below it, no edge connects those nodes. Moreover, increasing the δ value results in a sparser document graph, leading to reduced training time. Table 7 provides details of the dataset-wise optimal threshold (δ) values and the corresponding percentage reductions in training time from that with the δ value of 0.0. Thus, by tuning δ, we improve the coherence scores and simultaneously reduce the training time.

Value	20NG	BBC	SS	Bio	SO
Optimal threshold (δ)	0.4	0.3	0.2	0.05	0.1
Reduction (%)	154.27%	266.72%	16.71%	0.29%	1.06%

Table 7: Optimal threshold (δ) value along with the percentage of training time reduction for all five datasets.

6 Conclusion

We have introduced GINopic, a neural topic model based on a graph isomorphism network, and evaluated its performance on five widely used datasets for assessing topic models. Across the majority of our experiments, GINopic consistently exhibits superior topic coherence and diversity compared to other competitive topic models from the literature. Manual evaluation of selected topics further confirms that GINopic generates more coherent topics than alternative models. In extrinsic evaluations, GINopic generally outperforms existing models across all the datasets, except for the **20NG** dataset. We utilized visualizations of the latent space generated by GINopic to assess its clustering disentanglement capability. Sensitivity analysis demonstrates the impact of graph construction threshold values on the performance and training time of GINopic. Additionally, we highlight the effectiveness of GIN over other graph neural networks in our topic model.

Limitations

This paper focuses solely on utilizing word similarity for constructing document graphs. However, there exist alternative methods for constructing document graphs, such as incorporating dependency parse graphs. Future extensions of this work could explore capturing diverse word dependencies and integrating them to construct a multifaceted document graph.

Ethics Statement

The topic words presented in Table 6 depict the output of the topic models trained on the **20NG** dataset. The authors have no intention to cause harm or offense to any community, religion, country, or individual.

References

Suman Adhya, Avishek Lahiri, Debarshi Kumar Sanyal, and Partha Pratim Das. 2022. Improving contextualized topic models with negative sampling. In *Proceedings of the 19th International Conference on Natural Language Processing (ICON)*, pages 128–138, New Delhi, India. Association for Computational Linguistics.

Suman Adhya, Avishek Lahiri, and Debarshi Kumar Sanyal. 2023. Do neural topic models really need dropout? analysis of the effect of dropout in topic modeling. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2220–2229, Dubrovnik, Croatia. Association for Computational Linguistics.

Suman Adhya and Debarshi Kumar Sanyal. 2022. What does the Indian Parliament discuss? an exploratory analysis of the question hour in the Lok Sabha. In *Proceedings of the LREC 2022 workshop on Natural Language Processing for Political Sciences*, pages 72–78, Marseille, France. European Language Resources Association.

Suman Adhya and Debarshi Kumar Sanyal. 2023. Improving neural topic models with Wasserstein knowledge distillation. In *Advances in Information Retrieval*, pages 321–330, Cham. Springer Nature Switzerland.

Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2021a. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 759–766, Online. Association for Computational Linguistics.

Federico Bianchi, Silvia Terragni, Dirk Hovy, Debora Nozza, and Elisabetta Fersini. 2021b. Cross-lingual contextualized topic models with zero-shot learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1676–1683, Online. Association for Computational Linguistics.

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

Jordan L. Boyd-Graber, Yuening Hu, and David M. Mimno. 2017. Applications of topic models. *Found. Trends Inf. Retr.*, 11(2-3):143–296.

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Susan T. Dumais. 2004. Latent semantic analysis. *Annual Review of Information Science and Technology*, 38(1):188–230.

Derek Greene and Pádraig Cunningham. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 377–384, New York, NY, USA. Association for Computing Machinery.

Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Thomas Hofmann. 2013. Probabilistic latent semantic analysis. *CoRR*, abs/1301.6705.

Alexander Hoyle, Pranav Goel, Andrew Hian-Cheong, Denis Peskov, Jordan Boyd-Graber, and Philip Resnik. 2021. Is automated topic model evaluation broken? the incoherence of coherence. *Advances in Neural Information Processing Systems*, 34:2018–2033.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations, ICLR 2014*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

John Lafferty and David Blei. 2005. Correlated topic models. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539, Gothenburg, Sweden. Association for Computational Linguistics.

Dingcheng Li, Siamak Zamani, Jingyuan Zhang, and Ping Li. 2019. Integration of knowledge graph embedding into topic modeling with hierarchical Dirichlet process. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 940–950, Minneapolis, Minnesota. Association for Computational Linguistics.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. 2018. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861.

Yishu Miao, Lei Yu, and Phil Blunsom. 2016. Neural variational inference for text processing. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1727–1736, New York, New York, USA. PMLR.

David Newman, Youn Noh, Edmund Talley, Sarvnaz Karimi, and Timothy Baldwin. 2010. Evaluating topic models for digital libraries. In *Proceedings of the 10th Annual Joint Conference on Digital Libraries*, JCDL '10, page 215–224, New York, NY, USA. Association for Computing Machinery.

J. Qiang, Z. Qian, Y. Li, Y. Yuan, and X. Wu. 2022. Short text topic modeling techniques, applications, and performance: A survey. *IEEE Transactions on Knowledge & Data Engineering*, 34(03):1427–1445.

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*, pages 399–408.

Dazhong Shen, Chuan Qin, Chao Wang, Zheng Dong, Hengshu Zhu, and Hui Xiong. 2021. Topic modeling revisited: A document graph-based neural network perspective. In *Advances in Neural Information Processing Systems*, volume 34, pages 14681–14693. Curran Associates, Inc.

Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn, and Karsten M. Borgwardt. 2011. Weisfeiler-Lehman graph kernels. *Journal of Machine Learning Research*, 12(77):2539–2561.

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

Silvia Terragni, Elisabetta Fersini, Bruno Giovanni Galuzzi, Pietro Tropeano, and Antonio Candelieri. 2021a. OCTIS: Comparing and optimizing topic models is simple! In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 263–270. Association for Computational Linguistics.

Silvia Terragni, Elisabetta Fersini, and Enza Messina. 2021b. Word embedding-based topic similarity measures. In *Natural Language Processing and Information Systems*, pages 33–45, Cham. Springer International Publishing.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Hanna Wallach, David Mimno, and Andrew McCallum. 2009. Rethinking LDA: Why priors matter. In *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc.

William Webber, Alistair Moffat, and Justin Zobel. 2010. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38.

Xiaobao Wu, Xinshuai Dong, Thong Nguyen, and Anh Tuan Luu. 2023. Effective neural topic modeling with embedding clustering regularization. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23.

Qianqian Xie, Jimin Huang, Pan Du, and Min Peng. 2021. Graph relational topic model with higher-order graph attention auto-encoders. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2604–2613, Online. Association for Computational Linguistics.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How powerful are graph neural networks? In *International Conference on Learning Representations*.

Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. 2013. A biterm topic model for short texts. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, page 1445–1456, New York, NY, USA. Association for Computing Machinery.

Liang Yang, Fan Wu, Junhua Gu, Chuan Wang, Xiaochun Cao, Di Jin, and Yuanfang Guo. 2020. Graph attention topic modeling network. In *Proceedings of The Web Conference 2020*, WWW '20, page 144–154, New York, NY, USA. Association for Computing Machinery.

Renbo Zhao and Vincent Y. F. Tan. 2017. Online nonnegative matrix factorization with outliers. *IEEE Transactions on Signal Processing*, 65(3):555–570.

Deyu Zhou, Xuemeng Hu, and Rui Wang. 2020. Neural topic modeling by incorporating document relationship graph. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3790–3796, Online. Association for Computational Linguistics.

Qile Zhu, Zheng Feng, and Xiaolin Li. 2018. GraphBTM: Graph enhanced autoencoded variational inference for biterm topic model. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4663–4672, Brussels, Belgium. Association for Computational Linguistics.

A Data Overview

A.1 Dataset Descriptions

Datasets used in experiments:

1. **20NewsGroups (20NG)** dataset comprising 16,309 pre-processed documents from 20 different newsgroups posts. Each document is labeled with its corresponding category type.

2. **BBC News (BBC)** (Greene and Cunningham, 2006) dataset consists of 2,225 news articles from BBC. Documents are categorized into 5 different classes: *tech*, *business*, *entertainment*, *sports*, and *politics*.

3. **SearchSnippets (SS)** (Qiang et al., 2022) is derived from predefined phrases across 8 domains, this dataset is constructed from web search transactions. The domains include business, computers, culture-arts, education-science, engineering, health, politics-society, and sports.

4. **Biomedicine (Bio)** (Qiang et al., 2022) makes use of the challenge data delivered on BioASQ's official website.

5. **StackOverflow (SO)** (Qiang et al., 2022) The dataset is released on Kaggle.com. The raw dataset contains 3,370,528 samples from July 31st, 2012 to August 14, 2012. Here, the dataset randomly selects 20,000 question titles from 20 different tags.

The initial two datasets, 20NG, and BBC, are available on OCTIS[2]. As for the remaining three datasets SS, Bio, and SO, we have pre-processed them using the method detailed in Section A.2.

[2] https://github.com/MIND-Lab/OCTIS

#No.	Label	#Docs	%Docs
1.	misc.forsale	861	5.28
2.	comp.windows.x	883	5.41
3.	soc.religion.christian	920	5.64
4.	talk.religion.misc	521	3.19
5.	rec.autos	822	5.04
6.	sci.med	866	5.31
7.	talk.politics.misc	689	4.22
8.	talk.politics.mideast	828	5.08
9.	sci.electronics	867	5.32
10.	rec.sport.hockey	843	5.17
11.	rec.sport.baseball	787	4.83
12.	talk.politics.guns	808	4.95
13.	sci.crypt	883	5.41
14.	comp.sys.mac.hardware	838	5.14
15.	comp.sys.ibm.pc.hardware	891	5.46
16.	comp.graphics	836	5.13
17.	comp.os.ms-windows.misc	828	5.08
18.	alt.atheism	689	4.22
19.	sci.space	856	5.25
20.	rec.motorcycles	793	4.86

Table 8: **20NG** labels with corresponding document counts and percentage of documents.

#No.	Label	#Docs	%Docs
1.	tech	401	18.02
2.	business	510	22.92
3.	entertainment	386	17.35
4.	sport	511	22.97
5.	politics	417	18.74

Table 9: **BBC** labels with corresponding document counts and percentage of documents.

#No.	Label	#Docs	%Docs
1.	business	2652	21.61
2.	computers	2177	17.74
3.	culture-arts	1499	12.22
4.	education-science	1498	3.01
5.	engineering	1491	12.15
6.	health	1411	12.21
7.	politics-society	1173	9.56
8.	sports	369	11.5

Table 10: **SS** labels with corresponding document counts and percentage of documents.

A.2 Preprocessing

Using OCTIS, we convert each document to lowercase, remove the punctuations, lemmatize it, filter the vocabulary with the most frequent 2000 terms, filter words with less than 3 characters, and filter documents with less than 3 words.

B Baseline Configurations

We reproduced all baseline models by following the guidance provided in their original papers and utilizing codes from either the original sources or from OCTIS. Specifically, for **CombinedTM** (Bianchi et al., 2021a), **ZeroShotTM** (Bianchi et al., 2021b), **ProdLDA** (Srivastava and Sutton, 2017), **NeuralLDA** (Srivastava and Sutton, 2017), **ETM** (Dieng et al., 2020), **LDA** (Blei et al., 2003), **LSI** (Dumais, 2004), **NMF** (Zhao and Tan, 2017), we employed the implementation from OCTIS with default parameter values. For **GraphBTM**[3] (Zhu et al., 2018), **GNTM**[4] (Shen et al., 2021), and ECRTM[5] (Wu et al., 2023) we utilized the official source codes. Hyperparameter optimization was performed for GNTM on each dataset, and the values are detailed in Table 11. However, hyperparameter optimization for GBTM is computationally intensive, likely due to its exhaustive consideration of all words in the vocabulary when constructing the graph.

Hyperpramerts	20NG	BBC	SS
Temperature for STGS:	0.6	0.6	0.7
Window size for graph construction:	3	2	10

Table 11: Hyperparameter values for GNTM on each dataset.

C Coherence Metrics

Coherence matrices are used to compute the relevance of the top words within topics. The NPMI topic coherence for a given topic β_k with n top words is calculated as follows:

$$\mathrm{NPMI}(\beta_k) = \frac{1}{\binom{n}{2}} \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{\log \frac{p(w_i,w_j)+\epsilon}{p(w_i)p(w_j)}}{-\log\left(p(w_i,w_j)+\epsilon\right)}$$

Here, $p(w_i, w_j)$ is the probability of co-occurrence of words w_i and w_j in a boolean sliding window in topic k, and $p(w_i)$ and $p(w_j)$ represent the probability of the individual words' occurrence in topic k. ϵ is a small positive constant to prevent zero in the $\log(\cdot)$ function. NPMI ranges from -1 (words never co-occur) to $+1$ (they always co-occur). CV is computed using an indirect cosine measure along with NPMI scores over a boolean sliding window. In our experiments, we consider the top 10 words for each topic (i.e., $n = 10$) to compute NPMI and CV scores.

D Diversity Metrics

Topic diversity quantifies the uniqueness of generated topics. To measure the topic diversity we have used three following metrics: (i) IRBO (Bianchi et al., 2021b), (ii) wI-M (Terragni et al., 2021b), (iii) wI-C (Terragni et al., 2021b). The IRBO gives 0 for identical topics and 1 for completely dissimilar topics. Suppose we are given a collection $\aleph$ of T topics where each topic is a list of words such that the words at the beginning of the list have a higher probability of occurrence (i.e., are more important or more highly ranked) in the topic. Then, the IRBO score of the topics is defined as,

$$\mathrm{IRBO}(\aleph) = 1 - \frac{\sum_{i=2}^{T} \sum_{j=1}^{i-1} \mathrm{RBO}(l_i, l_j)}{n}$$

where $n = \binom{T}{2}$ is the number of pairs of lists, and $\mathrm{RBO}(l_i, l_j)$ denotes the standard Rank-Biased Overlap between two ranked lists l_i and l_j (Webber et al., 2010). IRBO allows the comparison of lists that may not contain the same items, and in particular, may not cover all items in the domain. Two lists (topics) with overlapping words receive a smaller IRBO score when the overlap occurs at the highest ranks of the lists than when it occurs at lower ranks. IRBO is implemented in OCTIS.

E Computing Infrastructure

Our experiments were run on a workstation with Intel® Xeon® W-1350 @ 3.30GHz, 6 Cores, 12 Threads, 16.0 GB RAM, NVIDIA RTX A4000 GPU, CUDA Version: 12.2 and Ubuntu 22.04 operating system.

[3] https://github.com/valdersoul/GraphBTM
[4] https://github.com/SmilesDZgk/GNTM
[5] https://github.com/BobXWu/ECRTM