Knowledge-Aware Bayesian Deep Topic Model

Dongsheng Wang, Yishi Xu, Miaoge Li, Zhibin Duan, Chaojie Wang, Bo Chen*
National Laboratory of Radar Signal Processing
Xidian University, Xi'an, Shanxi 710071, China
{wds,xuyishi,limiaoge,zhibinduan}@stu.xidian.edu.cn
xd_silly@163.com, bchen@mail.xidian.edu.cn

Mingyuan Zhou
McCombs School of Business
The University of Texas at Austin, Austin, TX 78712, USA
mingyuan.zhou@mccombs.utexas.edu

Abstract

We propose a Bayesian generative model for incorporating prior domain knowledge into hierarchical topic modeling. Although embedded topic models (ETMs) and its variants have gained promising performance in text analysis, they mainly focus on mining word co-occurrence patterns, ignoring potentially easy-to-obtain prior topic hierarchies that could help enhance topic coherence. While several knowledge-based topic models have recently been proposed, they are either only applicable to shallow hierarchies or sensitive to the quality of the provided prior knowledge. To this end, we develop a novel deep ETM that jointly models the documents and the given prior knowledge by embedding the words and topics into the same space. Guided by the provided domain knowledge, the proposed model tends to discover topic hierarchies that are organized into interpretable taxonomies. Moreover, with a technique for adapting a given graph, our extended version allows the structure of the prior knowledge to be finetuned to match the target corpus. Extensive experiments show that our proposed model efficiently integrates the prior knowledge and improves both hierarchical topic discovery and document representation.

1 Introduction

Topic models (TMs) such as latent Dirichlet allocation (LDA) have enjoyed success in text mining and analysis in an unsupervised manner [5]. Typically, the goal of TMs is to infer per-document topic proportions and a set of latent topics from the target corpus using word co-occurrences within each document. The extracted topics are widely used in various machine learning tasks [22, 21, 35, 42]. However, the objective function of most TMs is to maximize the likelihood of the observed data, which causes an over-concentration on high-frequency words. Moreover, the infrequent words might be assigned to irrelevant topics due to the lack of side information. Those two drawbacks could lead to human-unfriendly topics that fail to make sense to end users in practice. This issue is further exacerbated in hierarchical cases where a large number of topics and their relevance need to be modeled [29, 14].

In many cases, users with prior domain knowledge are concerned with a specific topic structure, as shown in Fig. 1. Such a topic hierarchy provides semantic common sense among topics and words, which can guide high-quality topic discovery. Therefore, several researches have attempted to exploit

*Corresponding author

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

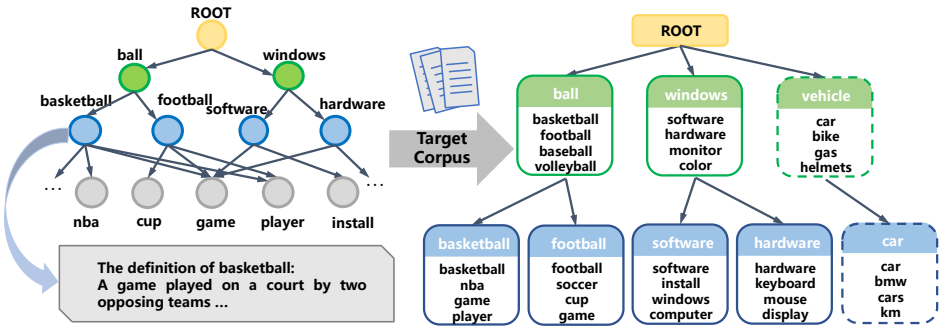

Figure 1: Motivation of the proposed model. We use the pre-specified topic hierarchy as the knowledge graph (left), where topics (colored nodes) and words (gray nodes) are organized into a tree-like taxonomy, and each topic is associated with a brief definition. Given the topic tree and the target corpus, our proposed model aims to ***1)*** retrieve a set of coherent words for each topic node (right); ***2)*** explore new topic hierarchies that are missed in the provided topic tree, e.g., the dashed boxes (right).

various prior knowledge into topic modeling to improve topic representation. For example, word correlation-based topic models [1, 32] use must-links and cannot-links to improve the interpretability of learned topics. Word semantic methods [20, 13] allow users to predefine a set of seed words that could appear together in a certain topic to improve topic modeling. Furthermore, knowledge graph-based topic models [19, 37] combine LDA with entity vectors to inject multi-relational knowledge, achieving better semantic coherence. Although the above mentioned methods improve topic coherence in different ways, they only work with shallow topic models and ignore the relationships between topics.

More recently, embedded topic models (ETMs) and their hierarchical variants [12, 41, 34, 14] employ distributed embedding vectors to encode semantic relationship, which have gained growing research interest due to their effectiveness and better flexibility. Viewing words and topics as embeddings in a shared latent space makes it possible for ETMs to integrate prior knowledge in the forms of pretrained word embeddings or semantic regularities [12, 29, 15]. The main idea behind those models is to constrain the topic embeddings under the predefined topic structure. Those models however are built on a strong assumption that the given topic hierarchy is well defined and matched to the target corpus, which is often unavailable in practice. Such mismatched structure might hinder the learning process and lead to sub-optimal results.

To address the above shortcomings, in this paper, we first propose TopicKG, a knowledge graph-guided deep ETM that views topics and words as nodes in the predefined topic tree. As a Bayesian probabilistic model, TopicKG aims to model the document Bag-of-Word (BoW) vector and the given topic tree jointly by sharing the embedding vectors. Specifically, we first adopt a deep document generation model to discover multi-layer document representations and topics. To incorporate the human knowledge in the topic tree and guide the learning of topic hierarchy, a graph generative model is employed, which refines the node embeddings by exploring the belonging relations hidden in topic tree, resulting in more interpretable topics. Besides, to address the mismatch between the given topic tree and target corpus, we further extend TopicKG to TopicKGA, which adopts the graph **A**daptive technique to explore the missing links in topic tree from the target corpora, achieving better document representation. The final revised topic tree is obtained by combining the prior structure of the given topic tree with the added structures learned from the current corpus. Thus the topic tree in our TopicKGA has the ability to reinforce itself constantly according to the given corpus. Finally, both the proposed models are based on variational autoencoders (VAEs) [23] and can be trained by maximizing the evidence lower bound (ELBO), enjoying promising flexibility and scalability.

The main contributions of this paper can be summarized as follows: ***1)***, A novel knowledge-aware deep ETM named TopicKG is proposed to incorporate prior domain knowledge into hierarchical topic modeling by accounting for the document and the given topic tree in a Bayesian generative framework. ***2)***, To overcome the drawback of the mismatch issue between the prior structure of the provided topic tree and the current corpus, TopicKG is further extended to TopicKGA, which allows to revise the prior structure to better represent the current corpus. ***3)***, Besides achieving promising performance on hierarchical topic discovery and document classification task, the proposed models give good flexibility and efficiency, providing a strong baseline for knowledge-based deep TMs.

2 Related Work

Embedded Topic Models: The recent developed ETM is a new topic-modeling framework in embedding space. Dieng et al. [12] first proposed ETM with the goal of marrying traditional topic models with word embeddings. In particular, ETM regards words and topics as continue embedding vectors and the topic's distribution over words is calculated using the inner product of the topic's embedding and each word's embedding, which provides a natural way to incorporate word meanings into TMs. In constrast to LDA, ETM employs the logistic-normal distribution to estimate the posterior of per-document topic proportion, making it is easier to reparameterization in the inference step. All parameters in ETM are optimized by maximizing its evidence lower bound (ELBO). To explore hierarchical document topical representations, ETM is extended to SawETM [14] with the gamma belief network [45]. SawETM designs the sawtooth connection (SC) to capture the relations between topics at two adjacent layers in the embedding space, resulting in more efficient algorithm for hierarchical topic mining. Note that our proposed TopicKG is built upon SawETM, but different from SawETM that only considers the BoW representation, which may fail to discover high quality topic hierarchies when a large of topics and their relations need to be modeled [29, 15], TopicKG introduces a novel Bayesian generative model where both document and the pre-specific domain knowledge are considered to integrate human knowledge into NTMs, resulting in a new knowledge-aware topic modeling framework.

Topic Models With Various Knowledge: Pre-trained language models such as BERT [11] and GPT [8] have been successful in various natural language processing tasks. Pre-trained on huge text, such models can serve as a powerful encoder that outputs contextual semantic embeddings. Behind this idea, combining a TM with BERT via knowledge distillation [18] treats BERT as the teacher model to improve the representation of topic models. CombinedTM [4] concatenates the Bag-of-Word (BoW) vector with the BERT contextual embedding as the final input of a TM, and performs a consistent increase in topic coherence. Despite their improvement, those BERT-based TMs focus on the document latent representation, they ignore the relationship between words and topics. Moreover they require a large amount of memory to load the pre-trained BERT. On the other hand, incorporating knowledge graph ($e.g.$, WordNet [30]) to improve existing NTMs becomes an interesting direction recently. For example, [29] use the category tree (usually with simple and shallow structure) as the supervised information and preserve such relative hierarchical structure in the spherical embedding space. to explore user interested topic structure. [15] view words and topics as the Gaussian distributions and employ the asymmetrical Kullback–Leibler (KL) divergence to measure the directional similarity in the given topic hierarchy. However, computing such KL divergence for each pair is time-consuming that limits its application to large-scale knowledge. Moreover, those knowledge-based models ignore the mismatch problem between the given topic tree and the current corpus, leading to sub-optimal results in practice.

3 The Proposed Model

To incorporate human knowledge into deep TMs, we first propose TopicKG that generates the target corpus and the given topic tree in the Bayesian manner. We further extend TopicKG to its adaptive version TopicKGA that allows the topic structure to be refined according to the current corpus, resulting in a better document representation. Below we introduce the details of our proposed model.

3.1 Problem Formulation

Consider a corpus containing J documents with a vocabulary of V unique tokens. Each document is represented by a BoW vector $\boldsymbol{x} \in R_+^V$, where x_v denotes the number of occurrences of the vth word. Unlike other TMs that are purely data-driven, TopicKG intends to inject the knowledge graph to improve topic quality. Specifically, we introduce a topic tree $\mathcal{T}$ as our prior knowledge, where each node $e_i \in \mathcal{T}$ denotes a word or a topic. For each topic node, there is a corresponding definition as shown in Fig. 1 . Suppose there are $L+1$ layers in the topic tree with the bottom word layer and L topic layers, and there are K_l nodes $\mathcal{T}_l : \{e_1^{(l)}, ..., e_{K_l}^{(l)}\}$ at the l-th layer, where $K_0 = V$. Generally, for the k-th topic node at the l-th layer $e_k^{(l)}$, we use $\mathcal{S}(e_k^{(l)})$ to denote the set of its child nodes, and use $\mathcal{C}(e_k^{(l)})$ to denote its key words extracted from the definition sentence. Mathematically, we adopt the binary matrix $\boldsymbol{S}^{(l)} \in \{0,1\}^{K_{l-1} \times K_l}$ and $\boldsymbol{C}^{(l)} \in \{0,1\}^{V \times K_l}$, $l = 1, ..., L$, to denote

the belonging relations between two adjacent layers and the links between topics and their concept words, respectively. Guided by this pre-specified topic tree, TopicKG expects to retrieve hierarchical topics from the target corpus that provide a clear taxonomy for the end users.

3.2 Knowledge-Aware Bayesian Deep Topic Model

Given the corpus $\mathcal{D}$ and topic tree $\mathcal{T}$, we aim to discover hierarchical document representations and topic hierarchy by jointly modeling the BoW vector $\boldsymbol{x}$, $\boldsymbol{S}^{(l)}$ and $\boldsymbol{C}^{(l)}$, $l = 1, ...L$ in a Bayesian framework. Specifically, we employ the gamma belief network (GBN) of [43] to model the count-value vector $\boldsymbol{x}$, and use the semantic similarity between two node embeddings to generate the topic tree $\mathcal{T}$. The whole generative model with L layers can be expressed as:

$$\begin{aligned}
&\boldsymbol{\theta}_j^{(L)} \sim \text{Gam}(\boldsymbol{\gamma}, 1/c_j^{(L+1)}), \quad \left\{\boldsymbol{\theta}_j^{(l)} \sim \text{Gam}(\boldsymbol{\Phi}^{(l+1)}\boldsymbol{\theta}_j^{(l+1)}, 1/c_j^{(l+1)})\right\}_{l=1}^{L-1}, \\
&\boldsymbol{x}_j \sim \text{Pois}(\boldsymbol{\Phi}^{(1)}\boldsymbol{\theta}_j^{(1)}) \quad \left\{\boldsymbol{\Phi}_k^{(l)} = \text{Softmax}(\boldsymbol{\Psi}_k^{(l)})\right\}_{l=1}^{L}, \quad \left\{\Psi_{k_1k_2}^{(l)} = \boldsymbol{e}_{k_1}^{(l-1)^T}\boldsymbol{e}_{k_2}^{(l)}\right\}_{k_1=1,k_2=1,l=1}^{K_{l-1},K_l,L}, \\
&\left\{S_{k_1,k_2}^{(l)} \sim \text{Bern}(\sigma(\boldsymbol{e}_{k_1}^{(l-1)^T}\boldsymbol{W}\boldsymbol{e}_{k_2}^{(l)}))\right\}_{k_1=1,k_2=1,l=1}^{K_{l-1},K_l,L}, \left\{C_{vk}^{(l)} \sim \text{Bern}(\sigma(\boldsymbol{e}_v^{(0)^T}\boldsymbol{e}_k^{(l)}))\right\}_{v=1,k=1,l=1}^{V,K_l,L},
\end{aligned} \tag{1}$$

where $\boldsymbol{x}_j$ is generated as in Poisson factor analysis [44], in which $\boldsymbol{x}_j$ is factorized as the product of the factor loading matrix (topics) $\boldsymbol{\Phi}^{(1)} \in \mathbb{R}_+^{K_0 \times K_1}$ and the gamma distributed factor scores (topic proportions) $\boldsymbol{\theta}_j^{(1)} \in \mathbb{R}_+^{K_1}$ under the Poisson likelihood; Then the GBN [45] is applied to explore multi-layer document representations by stacking hierarchical prior in the topic proportions $\boldsymbol{\theta}_j$. The topics at each layer $\boldsymbol{\Phi}^{(l)}$, $l = 1, ..., L$ is calculated by the semantic similarity ($e.g.$, the inner product) between corresponding topic embeddings, followed by a Softmax layer to satisfy the probability simplex: $\sum_{k_{l-1}=1}^{K_{l-1}} \Phi_{k_{l-1}k}^{(l)} = 1$. $\boldsymbol{e}_k^{(l)} \in R^d$ is the embedding vector of k-th topic at l-th layer (k-th node at l-th layer e_k^l in $\mathcal{T}$), d is the embedding dimension. $\boldsymbol{e}_v^{(0)}$ is the embedding vector for v-th word in vocabulary. $\sigma(\cdot)$ is the sigmoid function and Bern$(\cdot)$ denotes the Bernoulli distribution that is employed to model the two edge types $\boldsymbol{S}^{(l)}$ and $\boldsymbol{C}^{(l)}$. As suggested in [29] and [15], we use asymmetry to capture the directed relations in $\boldsymbol{S}^{(l)}$, while use symmetry to capture the semantic similarity between topics and their concept words. $\boldsymbol{W}$ is a learnable parameter to guarantee the directed structure. Under the Bernoulli likelihood, our TopicKG has the ability to incorporate the topic hierarchical relations in $\mathcal{T}$ into topic modeling by sharing word and topic embeddings with $\boldsymbol{\Phi}$. In particular, for a node pair in $\mathcal{T}$, the embedding semantics of the source and destination nodes need to be similar enough to determine whether there is an edge, which will guide the learning of $\boldsymbol{\Phi}$, resulting in more coherent topics and some appealing model properties as described below.

The Flexibility of Human Knowledge Incorporation: As mentioned above, guided by the pre-defined topic tree, TopicKG models $\boldsymbol{x}$, $\boldsymbol{S}^{(l)}$ and $\boldsymbol{C}^{(l)}$, $l = 1, ...L$ jointly, making it possible for the learned hierarchical topics that not only satisfy the interpretable taxonomy structure but also prevent incoherent issues with the help of side information. Besides, the incorporation of topic tree $\mathcal{T}$ is simple and flexible. Firstly, as shown in Eq. 1 and Sec. 4, the plug-and-play module of $\boldsymbol{S}^{(l)}$ and $\boldsymbol{C}^{(lk)}$ can be applied to most of ETMs without changing their model structures, which provides a convenient alternative for introducing side information to TMs. Secondly, the pre-defined $\mathcal{T}$ can be constructed in various ways to encode our beliefs about the graph structure, $e.g.$, the knowledge graph, taxonomy, and hierarchical label tree [37, 19, 29].

The Relationship Between $\{\boldsymbol{\Phi}^{(l)}\}_{l=1}^L$ and $\{\boldsymbol{S}^{(l)}\}_{l=1}^L$: The topic distribution matrix $\boldsymbol{\Phi}^{(l)}$ and the symmetry matrix $\boldsymbol{S}^{(l)}$ are related but play different roles. The sparse structure matrix $\boldsymbol{S}^{(l)}$ is constructed from the topic tree which will be used to guide the learning of the hierarchy of topics, while the learned topics $\boldsymbol{\Phi}^{(l)}$ contains more specific knowledge extracted from the text corpus, and can be viewed as the dense version of $\boldsymbol{S}^{(l)}$. In other words, the topics learned from TopicKG retains the semantic structure of $\boldsymbol{S}^{(l)}$ and enriches itself by the current corpora.

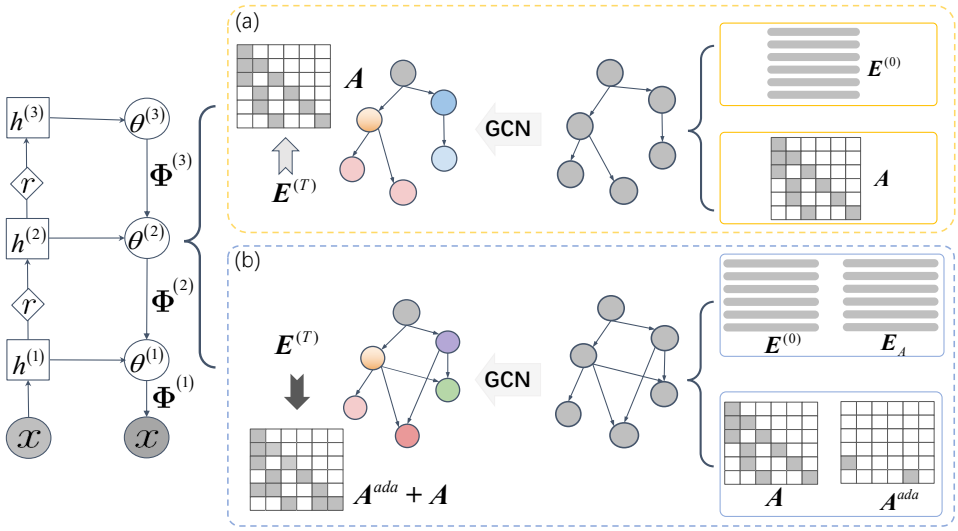

Figure 2: Overview of the proposed framework. It jointly models the topic tree and document via the shared node embeddings $\boldsymbol{E}^{(T)}$. The proposed two models TopicKG (a) and TopicKGA (b) share the same document generative module (left part) while with different graph modelings.

3.3 Inference Network of TopicKG

The inference network of the proposed model is built around two main components: Weibull upward-downward variational encoder and the GCN-based topic aggregation module. The former aims to infer the topic proportions given the document, and the latter updates the node embedding by aggregating the neighbor information via the GCN layer.

Weibull Upward-Downward Variational Encoder: To approximate the posterior of the topic proportions $\{\boldsymbol{\theta}_j^{(l)}\}$, like most of VAE-based methods, we define the variational distribution $q(\boldsymbol{\theta}_j|\boldsymbol{x}_j)$, which can be further factorized as [33]:

$$q(\boldsymbol{\theta}_j|\boldsymbol{x}_j) = q(\boldsymbol{\theta}_j^{(L)}|\boldsymbol{x}_j)\prod_{l=1}^{L-1} q(\boldsymbol{\theta}_j^{(l)}|\boldsymbol{\theta}_j^{(l+1)}, \boldsymbol{x}_j).$$

In practice, it first obtains the latent feature by feeding the input $\boldsymbol{x}_j$ into a residual upward neural networks: $\boldsymbol{h}_j^{(l)} = \boldsymbol{h}_j^{(l-1)} + f^{(l)}_{\boldsymbol{W}_h^{(l)}}(\boldsymbol{h}_j^{(l-1)})$, where $l = 1, ..., L$, $\boldsymbol{h}_j^{(0)} = \boldsymbol{x}_j$, $f^{(l)}_{\boldsymbol{W}_h^{(l)}}(\cdot)$ is a two layer fully connected network parameterized by $\boldsymbol{W}_h^{(l)}$. To complete the variational distribution, we adopt the Weibull downward stochastic path [40]:

$$\begin{gathered} q(\boldsymbol{\theta}_j^{(l)}|\boldsymbol{\Phi}^{(l+1)}, \boldsymbol{h}_j^{(l)}, \boldsymbol{\theta}_j^{(l+1)}) = \text{Weibull}(\boldsymbol{k}_j^{(l)}, \boldsymbol{\lambda}_j^{(l)}) \\ \boldsymbol{k}_j^{(l)} = \text{Softplus}(f_k^{(l)}(\boldsymbol{\Phi}^{(l+1)}\boldsymbol{\theta}_j^{(l+1)} \oplus \hat{\boldsymbol{k}}_j^{(l)})), \quad \boldsymbol{\lambda}_j^{(l)} = \text{Softplus}(f_\lambda^{(l)}(\boldsymbol{\Phi}^{(l+1)}\boldsymbol{\theta}_j^{(l+1)} \oplus \hat{\boldsymbol{\lambda}}_j^{(l)})) \\ \hat{\boldsymbol{k}_j}^{(l)} = \text{Relu}(\boldsymbol{W}_k^{(l)}\boldsymbol{h}_j + \boldsymbol{b}_k^{(l)}), \quad \hat{\boldsymbol{\lambda}_j}^{(l)} = \text{Relu}(\boldsymbol{W}_k^{(l)}\boldsymbol{h}_j + \boldsymbol{b}_k^{(l)}), \end{gathered} \tag{2}$$

where $\oplus$ denotes the concatenation at topic dimension, and we use the Softplus to make sure the positive Weibull shape and scale parameters. f is a single layer fully connected network, correspondingly. The Weibull distribution is chosen mainly because it is reparameterizable and the KL divergence from the gamma to Weibull distributions has an analytic expression [40, 17].

GCN-Based Topic Aggregation: Attracted by the excellent ability of graph neural network (GCN) [24, 25] in propagating graph structure information, and the predefined topic tree can be viewed as a directed graph. We thus construct a deterministic aggregating module for node embeddings with GCN:

$$\boldsymbol{E}^{(t)} = \text{GCN}(\tilde{\boldsymbol{A}}, \boldsymbol{E}^{(t-1)}), t = 1, ..., T \tag{3}$$

where T is the number of GCN layer, $\boldsymbol{E}^{(0)} \in R^{d\times N}$ is the node embedding matrix, and $N = \sum_{l=0}^{L} K_l$. $\tilde{\boldsymbol{A}} = \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}}$ is the normalized adjacent matrix with degree matrix $\boldsymbol{D}$, $e.g.$, $\boldsymbol{D}_{ii} = \sum_{j=1}^{N} \boldsymbol{A}_{ij}$, $\boldsymbol{A} \in \{0,1\}^{N\times N}$ is the adjacent matrix of the topic tree $\mathcal{T}$. Unlike previous ETMs that view embeddings as independent learnable parameters, the GCN module in TopicKG updates the node embeddings by considering their child-level topics and related words, resulting in more meaningful topic embeddings. We calculate $\{\boldsymbol{\Phi}^{(l)}\}_{l=1}^{L}$ via the updated node embeddings:

$$\boldsymbol{\Phi}_k^{(l)} = \text{Softmax}({\boldsymbol{e}^{(T)}}^{(l-1)^T} {\boldsymbol{e}_k^{(T)}}^{(l)}) \tag{4}$$

3.4 TopicKG With Adaptive Structure

One of the main assumptions in the previous section is that the pre-defined topic tree is helpful for the current corpus and the corresponding edges are highly reliable. However, this is generally unrealistic in practical applications, as $\mathcal{T}$ may be (i) noisy, (ii) built on an ad hoc basis, (iii) not closely related to the topic discovering task [16, 10]. Consequently, we further propose TopicKGA that overcomes the above mismatching issue and revises the structure of the given topic tree based on the corpus at hand.

In detail, TopicKGA first randomly initializes a learnable node embedding dictionaries $\boldsymbol{E}_A \in R^{d\times N}$ for all nodes in $\mathcal{T}$. Then an adaptive graph $\boldsymbol{A}^{\text{ada}}$ is generated based on $\boldsymbol{E}_A$ using the certain kernel function $k : R^d \times R^d \longrightarrow R$:

$$\tilde{\boldsymbol{A}}^{\text{ada}} = \text{Softmax}(k(\boldsymbol{E}_A, \boldsymbol{E}_A)) \tag{5}$$

Here, we choose the consine similarity to define our kernel function: $k(\boldsymbol{E}_A, \boldsymbol{E}_A) = \frac{\boldsymbol{E}_A \boldsymbol{E}_A^T}{||\boldsymbol{E}_A||||\boldsymbol{E}_A||}$, and Softmax function is used to normalize the adaptive matrix. Note that, instead of generating $\boldsymbol{A}^{\text{ada}}$, we directly generate its normalized version to avoid unnecessary calculations [3]. During training, $\boldsymbol{E}_A$ will be updated automatically to learn the hidden dependencies which are ignored by $\boldsymbol{A}$. Following [39], the final revised adjacency matrix is formed as (here we still use $\tilde{\boldsymbol{A}}$ to denote the revised graph for convenience):

$$\tilde{\boldsymbol{A}} = \tilde{\boldsymbol{A}} + \tilde{\boldsymbol{A}}^{\text{ada}} \tag{6}$$

which will enhance the GCN in Eq. 3 by replacing the adjacency matrix with the revised one.

4 Training

As a deep ETM, TopicKG intends to learn the latent document-topic distribution and the deterministic hierarchical topic embeddings. Like other ETMs, the posterior of the topic proportions and topic embeddings are intractable to compute. We thus derive an efficient algorithm for approximating the posterior with amortized variational inference [6], which makes the proposed model flexible for downstream task. The resulting algorithm can either use pre-trained word embeddings, or train them from scratch. The ELBO of the proposed model can be expressed as:

$$\begin{aligned}\mathcal{L} = &\sum_{j=1}^{J} \mathrm{E}_{q(\boldsymbol{\theta}|\boldsymbol{x}_j)}[\log p(\boldsymbol{x}_j|\boldsymbol{\Phi}^{(1)}, \boldsymbol{\theta}_j^{(1)})] + \beta \sum_{l=1}^{L}\sum_{k_2=1}^{K_l}(\sum_{k_1=1}^{K_{l-1}} \log p(S_{k_1k_2}^{(l)})|\boldsymbol{e}_{k_1}^{(l-1)}, \boldsymbol{e}_{k_2}^{(l)}) \\ &+ \sum_{v=1}^{V} \log p(C_{vk_2}^{(l)}|\boldsymbol{e}_v^{(0)}, \boldsymbol{e}_{k_2}^{(l)})) - \sum_{j=1}^{J}\sum_{l=1}^{L} \mathrm{KL}(q(\boldsymbol{\theta}_j^{(l)})||p(\boldsymbol{\theta}_j^{(l)}|\boldsymbol{\Phi}^{(l+1)}, \boldsymbol{\theta}_j^{(l+1)})))\end{aligned} \tag{7}$$

It consists of three main parts: the expected log-likelihood of $\boldsymbol{x}$ (first term), the concept structure log-likelihood (the two middle terms), and the KL divergence from prior $p(\boldsymbol{\theta}_j^{(l)})$ to $q(\boldsymbol{\theta}_j^{(l)})$. The graph weight β denotes the belief about the predefined topic tree. Notably, the two middle terms distinguish the proposed models from previous deep ETMs. On the one hand, it acts as a regularization that constrains the embedding vectors to conform to the provided prior structure, and on the other hand, it provides an alternative for ETM to introduce side information to improve its interpretability.

Annealed Training for TopicKGA: In TopicKGA, the structure of the revised graph changes during the training. To incorporate the new edges that are inferred from the target corpus, we develop an

annealed training algorithm for TopicKGA. Specifically, we update $\boldsymbol{S}^{(l)}$ and $\boldsymbol{C}^{(l)}$ in Eq. 7 every M iterations:

$$S_{k_1k_2}^{(l)} = \begin{cases} 1, & \text{if} \quad \tilde{A}_{k_1k_2}^{(l)} > s; \\ 0, & else \end{cases} \qquad C_{vk_2}^{(l)} = \begin{cases} 1, & \text{if} \tilde{A}_{vk_2}^{(l)} > s; \\ 0, & else \end{cases} \tag{8}$$

where s is the threshold, and $\tilde{\boldsymbol{A}}^{(l)}$ is the corresponding sub-block in $\tilde{\boldsymbol{A}}$. As $\boldsymbol{E}_A$ becomes stable, the structure of the graph will converge to a good blueprint that balances the prior graph and the current corpus. We summarize all training algorithm in Appendix.

5 Experiment

In this section, we conduct extensive experiments on several benchmark text datasets to evaluate the performance of the proposed models against other knowledge-based TMs, in terms of topic interpretability and document representations. The code is available at https://github.com/wds2014/TopicKG.

5.1 Corpora

Our experiments are conducted on four widely used benchmark text datasets, varying in scale. The datasets include 20 Newsgroups (20NG) [26], Reuters extracted from the Reuters-21578 dataset, R8, and Reuters Corpus Volume 2 (RCV2) [27]. R8 is a subset of Reuters that collected from 8 different classes. For the multi-label RCV2 dataset, we follow previous works [31] in which documents with a single label at the second level topics are left, resulting in 0.1M documents totally. Both R8 and RCV2 are already pre-processed. For other two datasets, we tokenize and clean text by excluding standard stop words and low-frequency words appearing less than 20 times [38]. The statistics of the preprocessed datasets are summarized in Appendix.

5.2 Baselines

To demonstrate the effectiveness of incorporating human knowledge into deep TMs, we consider several baselines for a fair comparison, including representative ETMs and recent knowledge-based topic models, described as follows: ***1)***, **ETM** [12], the first neural embedded topic model that assumes topics and words live in the same embedding space. The topic distributions in ETM are calculated by the inner product of the word embedding matrix and the topic embedding. ***2)***, **CombinedTM** [4], a BERT-based neural topic model that uses the pre-trained BERT as its contextual encoder. We use it as our BERT baseline. ***3)***, **SawETM** [14], a hierarchical ETM that employs the Poisson and gamma distributions to model the BoW vector and the latent representation, respectively. ***4)***, **JoSH** [29], a knowledge-based hierarchical topic mining method that uses category hierarchy as the side information and employs the EM algorithm to learn the spherical tree and text embedding. ***5)***, **TopicNet** [15], a semantic graph guided topic model that views the words and topics as the Gaussian distributions and uses the KL divergence to regularize the structure of the pre-defined tree. For all baselines, we employ their official codes and default settings obtained from their release repositories.

5.3 External Knowledge

Since the learning of TopicKG involves a pre-defined topic tree, some generic knowledge graphs such as WordNet [30] provide a convenient way to bring in external knowledge. WordNet is a large lexical database that groups semantically similar words into synonym sets, these sets are further linked by the hyponymy relation. For example, in WordNet the category furniture includes bed, which in turn includes bunkbed. With these relations, the topic structure can be easily defined according to certain heuristic rules. Specifically, for each dataset we first get the word intersection of the vocabulary and WordNet, the chosen words are then considered as leaf nodes of the topic hierarchy. Afterwards, each leaf node can continuously find its ancestor nodes based on hyponymy relations and finally all leaf nodes converge to the same root node, resulting in an adaptive topic structure.

5.4 Settings

For all experiments, we set the embedding dimension as $d = 50$, the knowledge confidence hyper-parameter as $\beta = 50.0$, the threshold as $s = 0.4$. Empirically, we find that the above settings work

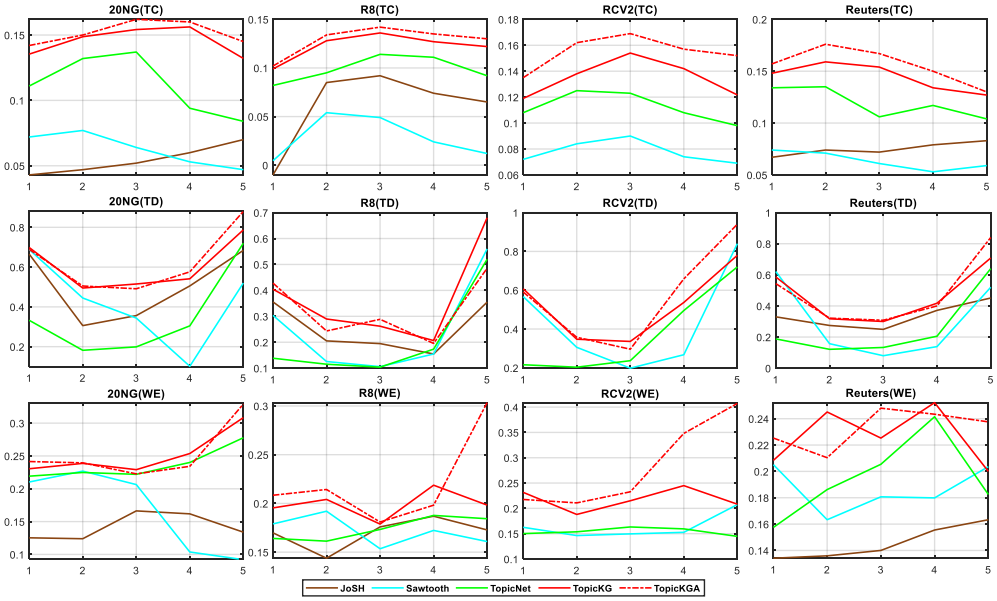

Figure 3: Topic coherence (TC, top row), topic diversity (TD, middle row), and word embedding coherence (WE, bottom row) results for various deep topic models on four datasets. In each subfigure, the horizontal axis indicates the layer index of the topics. For all metrics, higher is better.

well for all datasets, we refer readers to the appendix for more detailed analysis. We initialize the node embedding from the Gaussian distribution $\mathcal{N}(0, 0.02)$. We set the batch size as 200 and use the AdamW [28] optimizer with learning rate 0.01. We generate a 7-layer topic hierarchy for each dataset with the method described in Sec. 5.3. From the top to bottom, the number of topics at each layer are: 20NG: [1,2,11,66,185,277,270]; R8: [1,2,11,77,287,501,547]; RCV2: [1,2,11,68,263,469,562]; Reuters: [1,2,11,78,300,526,584]. For the single-layer methods (ETM and CombinedTM), we set the number of topics as the same of the first layer of deep models. For all methods, we run the algorithms in comparison five times by modifying only the random seeds and calculate the mean and standard deviation. All experiments are performed on an Nvidia RTX 3090-Ti GPU and our proposed models are implemented with PyTorch.

Table 1: Micro F1 and Macro F1 score of different models on three datasets. The digits in brackets indicate the number of layers. Micro F1 /Macro F1.

Model	20NG	R8	RCV2
ETM	50.25 ±0.42 / 47.44 ±0.21	88.10 ±0.45 / 59.67 ±0.24	68.63 ±0.15 / 24.40 ±0.11
CombinedTM	56.43 ±0.14 / 54.95 ±0.11	93.69 ±0.09 /84.14 ±0.10	84.85 ±0.11 / 51.47 ±0.21
Sawtooth(3)	52.41 ±0.08 / 51.53 ±0.10	90.04 ±0.15 / 78.84 ±0.21	82.54 ±0.11 / 49.25 ±0.10
TopicNet(3)	55.16 ±0.22 / 54.78 ±0.34	89.95 ±0.17 / 64.15 ±0.16	84.15 ±0.25 / 50.37 ±0.22
TopicKG(3)	55.73 ±0.15 / 54.48 ±0.08	93.6 ±0.05 / 83.32 ±0.07	84.75 ±0.16 / 50.51 ±0.41
TopicKGA(3)	**58.63** ±0.15 / **57.90** ±0.10	**93.70** ±0.52 / **84.50** ±0.11	**85.34** ±0.14 / **52.35** ±1.10
ETM	47.79 ±0.12 / 44.19 ±1.01	86.54 ±0.84 / 59.88 ±1.11	63.77 ±0.14 / 21.44 ±1.04
CombinedTM	58.16 ±0.15 / 58.10 ±0.10	93.50 ±0.13 /84.84 pm0.11	82.91 ±0.11 / 48.17 ±0.05
Sawtooth(7)	53.71 ±0.11 / 53.02 ±0.47	92.86 ±0.07 / 82.54 ±0.41	82.46 ±0.15 / 49.34 ±0.34
TopicNet(7)	56.13 ±0.19 / 55.41 ±0.39	90.65 ±0.00 / 66.57 ±0.00	82.81 ±0.00 / 49.44 ±0.00
TopicKG(7)	56.32 ±0.12 / 57.35 ±0.04	94.04 ±0.12 / 85.04 ±0.11	82.48 ±0.11 / 48.24 ±0.09
TopicKGA(7)	**60.04** ±0.34 / **59.12** ±0.13	**94.10** ±0.08 / **85.50** ±0.10	**83.08** ±0.23 / **50.50** ±0.08

5.5 Topic Interpretability

Generally, topic models are evaluated based on perplexity. However, perplexity on the held-out test is not an appropriate measure of the topic quality and sometimes can even be contrary to human judgements [9, 37]. To this end, we instead adopt three common metrics, including Topic coherence (**TC**), Topic diversity (**TD**) and word embedding topic coherence (**WE**), to evaluate the learned topics from various aspects [12, 4]. TC measures the average Normalized Pointwise Mutual Information

(NPMI) over the top 10 words of each topic, and a higher score indicates more interpretable topics. TD denotes the percentage of unique words in the top 25 words of the selected topics. WE, as its name implies, provides an embedded measure of how similar the words in a topic are. WE is calculated by the average pairwise cosine similarity of the word embeddings of the top-10 words in a topic. We report the topic quality results at the first five layer in Fig. 3 [2]. We here focus on topic hierarchy and ignore the single layer methods. We find that: ***1)***, Knowledge-based methods including JoSH, CombinedTM, TopicNet and our proposed models are generally better than other likelihood-based ETMs, which illustrates the benefit of incorporating side information; ***2)***, Our proposed TopicKG and TopicKGA achieve higher performance than other knowledge-based models in most cases, especially on TC and WE tasks, which means our proposed model prefer to mine more coherent topics while achieving comparable TD. We attribute this to the joint topic tree likelihood that provides an efficient alternative to integrate human knowledge into ETMs. ***3)***, Compared to TopicKG, its adaptive version TopicKGA discovers better topics at higher layers. It is not surprise that TopicKGA gives greater robustness and flexibility by refining the topic structure according to the current corpus.

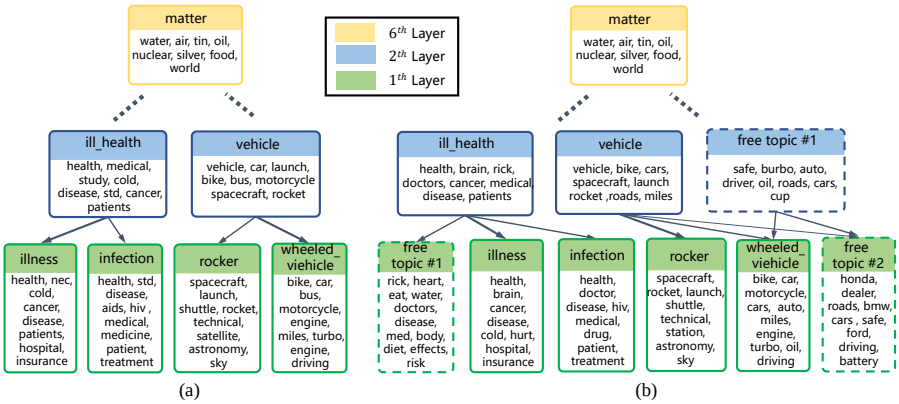

Figure 4: The learned hierarchical topic structure from TopicKG (a) and TopicKGA(b) on 20NG. Each topic box contains its concept name and the corresponding keywords. Topics in dashed box (free topics) are newly added by TopicKGA from the target corpus. The thickness of the arrow represents the relation weight and different colors denote different layers.

5.6 Document Classification

Besides the topic quality evaluation, we also use document classification to compare the extrinsic predictive performance. In detail, we collect the inferred topic proportions, $e.g.$, $\boldsymbol{\theta}^{(1)}$, and than apply logistic regression to predict the document label. We report the Micro F1 and Macro F1 score on 20NG, R8 and RCV2 datasets in Table. 1. Overall, the proposed models outperform the baselines on all corpora, which confirms the effectiveness of our innovation of combining human knowledge and ETMs in improving document latent representations. Moreover, with the revised topic structure fine-tuned to the current corpus, TopicKGA surpasses other knowledge graph fixation methods with significant gaps. This conclusion is consistent with one of our motivations that the predefined concept tree may not match the target corpus, leading to suboptimal document representations.

5.7 Qualitative Analysis

We visualize the learned topic hierarchies of TopicKG and TopicKGA on 20NG in Fig. 4(a-b), respectively. Each topic box consists of the pre-specified concept name on the top bar and its keywords listed in the bottom content. We can observe that ***1)***, The mined keywords are highly relevant to the corresponding topics, providing a clean description of their concepts. ***2)***, Guided by human knowledge, the connections between topics at two adjacent layers are highly interpretable, resulting in human-friendly topic taxonomies. ***3)***, More interestingly, to further verify the adaptive ability of our proposed TopicKGA, we manually added several free topics (dashed boxes in Fig. 4(b)) to each layer of the given topic tree, which can be viewed as the missing topics in the predefined

[2] We can't report the results of JoSH on RCV2 as JoSH requires the sequential text, but only the BoW form is available for RCV2.

knowledge and need to be learned from the target corpus. We find that our TopicKGA indeed has the ability to capture the missing concepts, and revise the prior graph to match the current corpus.

5.8 Time Efficiency

To demonstrate the time efficiency of incorporating human knowledge into TMs, we run TopicKG, TopicKGA and TopicNet (as they are implemented in PyTorch, JoSH is in C) on RCV2 and 20NG dataset with a 7-layer topic tree. Fig. 5 shows the Negative log-likelihood of documents $\boldsymbol{x}$, which shows that the proposed models not only have faster learning speed than TopicNet, but also achieve better reconstruction performance on both small and large corpus. This result illustrates the efficiency of the introduced graph likelihood which is important in real-time applications.

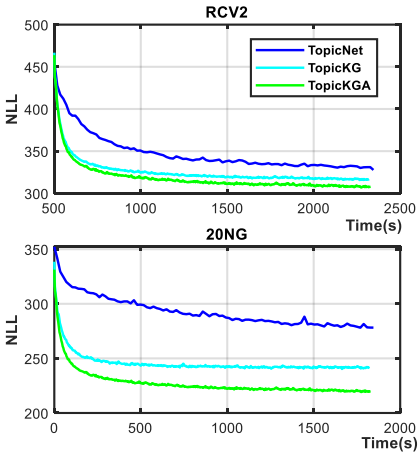

Figure 5: Negative log-likelihood (NLL) curves of various methods on RCV2 and 20NG datasets.

6 Conclusion

We develop an efficient Bayesian probabilistic model that integrates pre-specified domain knowledge into hierarchical ETMs. The core idea is the joint document and topic tree modeling with the shared word and topic embeddings. Besides, with the graph adaptive technique, TopicKGA has the ability to revise the given prior structure according to the target corpus, enjoying robustness and flexibility in practice. Extensive experiments show that the proposed models outperform competitive methods in term of both topic quality and document classification task. Moreover, thanks to the efficient knowledge incorporation algorithm, our proposed models achieve faster learning speed than other knowledge-base models.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant U21B2006; in part by Shaanxi Youth Innovation Team Project; in part by the 111 Project under Grant B18039; in part by the Fundamental Research Funds for the Central Universities QTZX22160.

References

[1] D. Andrzejewski, X. Zhu, and M. Craven. Incorporating domain knowledge into topic modeling via dirichlet forest priors. In *Proceedings of the 26th annual international conference on machine learning*, pages 25–32, 2009.

[2] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007.

[3] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang. Adaptive graph convolutional recurrent network for traffic forecasting. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[4] F. Bianchi, S. Terragni, and D. Hovy. 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, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 759–766.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[6] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

[7] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008.

[8] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

[9] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei. Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems*, pages 288–296, 2009.

[10] Y. Chen, L. Wu, and M. Zaki. Iterative deep graph learning for graph neural networks: Better and robust node embeddings. *Advances in Neural Information Processing Systems*, 33, 2020.

[11] J. Devlin, M. Chang, K. Lee, and K. Toutanova. 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.

[12] A. B. Dieng, F. J. Ruiz, and D. M. Blei. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453, 2020.

[13] F. Doshi-Velez, B. Wallace, and R. Adams. Graph-sparse lda: a topic model with structured sparsity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[14] Z. Duan, D. Wang, B. Chen, C. Wang, W. Chen, Y. Li, J. Ren, and M. Zhou. Sawtooth factorial topic embeddings guided gamma belief network. In *International Conference on Machine Learning*, pages 2903–2913. PMLR, 2021.

[15] Z. Duan, Y. Xu, B. Chen, dongsheng wang, C. Wang, and M. Zhou. Topicnet: Semantic graph-guided topic discovery. In *Advances in Neural Information Processing Systems*, 2021.

[16] P. Elinas, E. V. Bonilla, and L. C. Tiao. Variational inference for graph convolutional networks in the absence of graph data and adversarial settings. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[17] X. Fan, S. Zhang, B. Chen, and M. Zhou. Bayesian attention modules. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

[18] A. M. Hoyle, P. Goel, and P. Resnik. Improving neural topic models using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1752–1771.

[19] Z. Hu, G. Luo, M. Sachan, E. P. Xing, and Z. Nie. Grounding topic models with knowledge bases. In *IJCAI*, volume 16, pages 1578–1584, 2016.

[20] J. Jagarlamudi, H. Daumé III, and R. Udupa. Incorporating lexical priors into topic models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–213, 2012.

[21] M. Jin, X. Luo, H. Zhu, and H. H. Zhuo. Combining deep learning and topic modeling for review understanding in context-aware recommendation. 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 1605–1614, 2018.

[22] S. S. Kataria, K. S. Kumar, R. R. Rastogi, P. Sen, and S. H. Sengamedu. Entity disambiguation with hierarchical topic models. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1037–1045, 2011.

[23] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

[24] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[25] T. N. Kipf and M. Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[26] K. Lang. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*, pages 331–339. Elsevier, 1995.

[27] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. Rcv1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397, 2004.

[28] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

[29] Y. Meng, Y. Zhang, J. Huang, Y. Zhang, C. Zhang, and J. Han. Hierarchical topic mining via joint spherical tree and text embedding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020.

[30] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11): 39–41, 1995.

[31] T. Miyato, A. M. Dai, and I. J. Goodfellow. Adversarial training methods for semi-supervised text classification. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

[32] D. Newman, E. V. Bonilla, and W. Buntine. Improving topic coherence with regularized topic models. *Advances in neural information processing systems*, 24:496–504, 2011.

[33] A. Vahdat and J. Kautz. NVAE: A deep hierarchical variational autoencoder. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

[34] D. Wang, D. Guo, H. Zhao, H. Zheng, K. Tanwisuth, B. Chen, and M. Zhou. Representing mixtures of word embeddings with mixtures of topic embeddings. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*.

[35] H. Xu, W. Wang, W. Liu, and L. Carin. Distilled wasserstein learning for word embedding and topic modeling. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 1723–1732.

[36] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764, 2019.

[37] L. Yao, Y. Zhang, B. Wei, Z. Jin, R. Zhang, Y. Zhang, and Q. Chen. Incorporating knowledge graph embeddings into topic modeling. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

[38] L. Yao, C. Mao, and Y. Luo. Graph convolutional networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7370–7377, 2019.

[39] D. Yu, R. Zhang, Z. Jiang, Y. Wu, and Y. Yang. Graph-revised convolutional network. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2020, Ghent, Belgium, September 14-18, 2020, Proceedings, Part III*, volume 12459 of *Lecture Notes in Computer Science*, pages 378–393.

[40] H. Zhang, B. Chen, D. Guo, and M. Zhou. WHAI: weibull hybrid autoencoding inference for deep topic modeling. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

[41] H. Zhao, D. Phung, V. Huynh, T. Le, and W. L. Buntine. Neural topic model via optimal transport. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

[42] H. Zhao, L. Du, W. Buntine, and M. Zhou. Dirichlet belief networks for topic structure learning. *Advances in neural information processing systems*, 31, 2018.

[43] M. Zhou, Y. Cong, and B. Chen. The poisson gamma belief network. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3043–3051.

[44] M. Zhou, L. Hannah, D. Dunson, and L. Carin. Beta-negative binomial process and poisson factor analysis. In *Artificial Intelligence and Statistics*, pages 1462–1471. PMLR, 2012.

[45] M. Zhou, Y. Cong, and B. Chen. The poisson gamma belief network. *Advances in Neural Information Processing Systems*, 28, 2015.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Please refer to the appendix
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Please refer to the appendix
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] The code and data are included in the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you citep the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Detailed discussions of our work

A.1 Limitations

We in this paper propose a Bayesian generative model for incorporating prior human knowledge into deep topic models, named TopicKG. TopicKG jointly models the provided topic tree $\mathcal{T}$ and the current corpus $\mathcal{D}$ in a Bayesian framework by sharing the word embeddings and topic embeddings. Besides, TopicKG employs a Weibull upward-downward variational encoder to infer hierarchical document representations and a GCN-based topic aggregation module to account for the relations between nodes in the topic tree. To address the mismatch issues between the provided topic tree and the target corpus, TopicKG is further extended into TopicKGA which uses the graph adaptive technique to revise the tree structure according to the current learning state. Despite the promising results of those two proposed models, a main limitation of this work is the form of knowledge. TopicKG and TopicKGA are designed for the prior domain knowledge that contains the relations

of entities. While most existing knowledge graphs (e.g. WordNet[30], DBpedia[2], Freebase[7]) meet this requirement, the recent pre-trained language model, such as BERT[11], XLNET[36] can be viewed as another form of knowledge. How to combine deep topic models with such pre-trained language model remains a challenge in the topic modelling community. This is beyond the scope of this paper and we leave it for future research.

A.2 Negative societal impacts

The proposed model improves deep topic models by incorporating side information, It not only achieves state-of-the-art performance in document representation tasks (e.g., document classification), but also discovers coherent and diverse topics, which can be applied in many natural language processing (NLP) tasks, such as document generation, dialogue systems and text mining. Note that technology is neither inherently helpful nor harmful. It is simply a tool and the real effects of technology depend upon how it is wielded. For example some merchants or advertisers may use TopicKG to recommend articles with specific contents of interest to their users, which may induce users to buy some unnecessary items. This negative impact can be avoided by strengthening data regulation to avoid false advertising.

B Datasets

To evaluate the performance of the proposed models, we conduct experiment on four widely-used benchmark text datasets, varying in scale.

- **20NG**[3]: The 20 Newsgroups data set is a collection of 18,864 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups. We follow the default training/testing split and choose the top 2000 words after removing stop words and low-frequency words. The average length of document is 108.75.
- **R8**: R8 is a subset of Reuters with 8 different classes. There are 5,485 training samples, and 2,189 test samples. The average length of R8 is 65.72.
- **Reuters** [4]: The documents in the Reuters collection appeared on the Reuters newswire in 1987. Here we only use Reuters on topic quality task. the average length of Reuters is 74.14.
- **RCV2**[5]: Reuters Corpus Volume v2 is a benchmark dataset on text categorization. It is a collection of newswire articles produced by Reuters in 1996-1997. The average length of RCV2 is 52.82.

A summary of dataset statistics is shown in Table 2.

Dataset	J_{all}	J_{train}	J_{test}	C	V	$Avg.N$
20NG	18,864	11,314	7,532	20	2,000	108.75
Reuters	11,367	11,367	/	/	8,817	74.14
R8	7,674	5,485	2,189	8	7,688	65.72
RCV2	150,737	100,899	49,838	52	7,282	82.82

Table 2: Summary statistics of the datasets, where J denotes the number of documents, C the number of classes, V the vocabulary size and $Avg.N$ the average length of documents in the corpus, respectively.

C Training algorithm

We summary the training algorithm as bellow:

[3] http://qwone.com/ jason/20Newsgroups

[4] https://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html

[5] https://trec.nist.gov/data/reuters/reuters.html

Algorithm 1 Training algorithm for our proposed models

Input: training documents $\mathcal{D}$, topic tree $\mathcal{T}$, topic number of each layer K, hyperparameter β, s and M.
Initialize: node embedding $\boldsymbol{E}$ and $\boldsymbol{E}_A$, all learnable parameters.
for iter = 1,2, $\cdots$ **do**
  Sample a batch of J input documents; and feed them into the variational encoder to infer the latent representation $\boldsymbol{\theta}$ with Eq.2 in the manuscript;
  if TopicKGA **then**
    Calculate the normalized $\tilde{A}$ with Eq.5 in the manuscript;
  end if
  With the GCN-based topic aggregation module, update all node embeddings with Eq.3 and calculate the global parameters $\boldsymbol{\Phi}$;
  if TopicKGA **and** everay M iterations **then**
    Update $\boldsymbol{S}^{(l)}$ and $\boldsymbol{C}^{(l)}$ with Eq.8;
  end if
  Compute the loss with Eq.7, and update all learnable parameters.
end for

D Hypeparameter sensitivity

We fix the knowledge confidence weight $\beta = 50.0$ and the threshold $s = 0.4$ in the previous experiments. Bellow we analyze the performance of the proposed models at different hyperparameter setting and report the Micro F1 score and WE on 20NG, R8, and RCV2 datasets in Fig. 6 and Fig. D, respectively. We find that 1), Overall, by accounting for both concept structure and document likelihood, HFTM achieves better document representation and topic discovery than only considering the document ($e.g.$, $\beta = 0$). 2), TopicKGA is insensitive to graph generation shreshold s and has a wide tolerance. 3), One can obtain better results by finetuning β and s for each dataset.

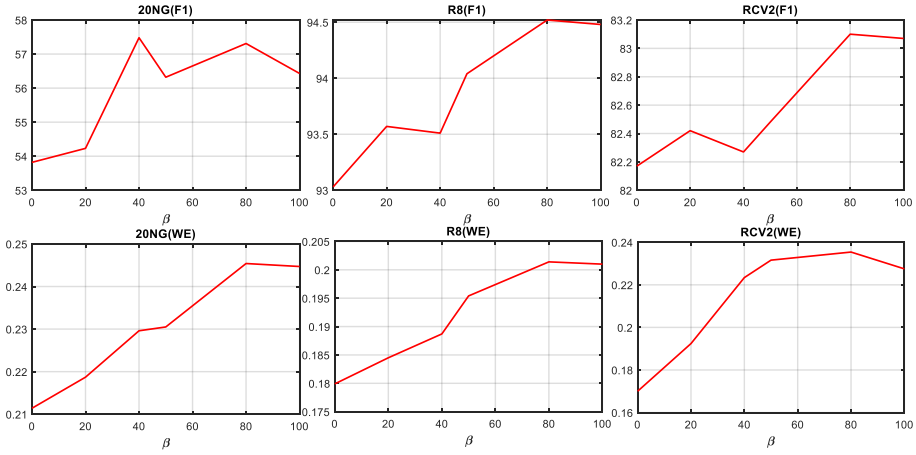

Figure 6: Micro F1 score (F1, top row) and word embedding coherence (WE, bottom row) on 20NG, R8 and RCV2 datasets with different β.

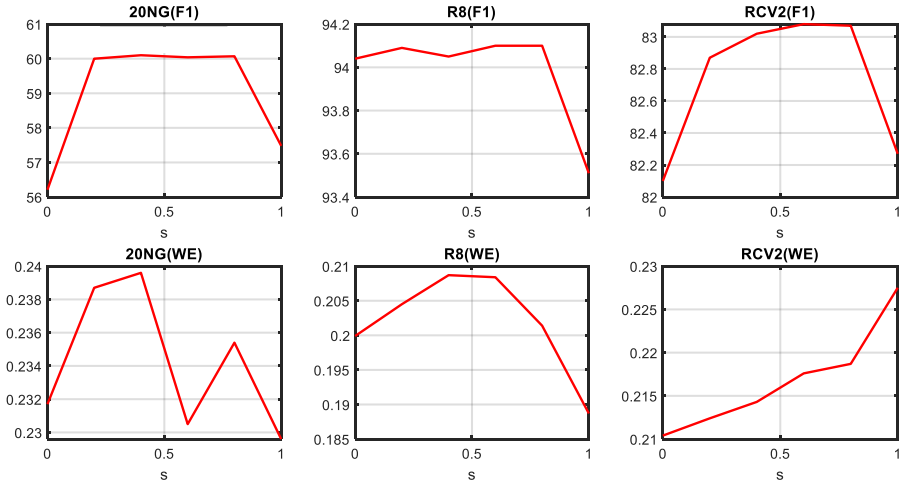

Figure 7: Micro F1 score (F1, top row) and word embedding coherence (WE, bottom row) on 20NG, R8 and RCV2 datasets with different s.