Topic Taxonomy Expansion via Hierarchy-Aware Topic Phrase Generation

Dongha Lee[1], Jiaming Shen[2], Seonghyeon Lee[3], Susik Yoon[1], Hwanjo Yu[3]*, and Jiawei Han[1]

[1]University of Illinois at Urbana-Champaign (UIUC), Urbana, IL, United States
[2]Google Research, New York, NY, United States
[3]Pohang University of Science and Technology (POSTECH), Pohang, Republic of Korea
{donghal,susik,hanj}@illinois.edu, jmshen@google.com, {sh0416,hwanjoyu}@postech.ac.kr

Abstract

Topic taxonomies display hierarchical topic structures of a text corpus and provide topical knowledge to enhance various NLP applications. To dynamically incorporate new topic information, several recent studies have tried to expand (or complete) a topic taxonomy by inserting emerging topics identified in a set of new documents. However, existing methods focus only on frequent terms in documents and the local topic-subtopic relations in a taxonomy, which leads to limited topic term coverage and fails to model the global topic hierarchy. In this work, we propose a novel framework for topic taxonomy expansion, named Topic-Expan, which directly generates topic-related terms belonging to new topics. Specifically, TopicExpan leverages the hierarchical relation structure surrounding a new topic and the textual content of an input document for topic term generation. This approach encourages newly-inserted topics to further cover important but less frequent terms as well as to keep their relation consistency within the taxonomy. Experimental results on two real-world text corpora show that TopicExpan significantly outperforms other baseline methods in terms of the quality of output taxonomies.

1 Introduction

Topic taxonomy is a tree-structured representation of hierarchical relationship among multiple topics found in a text corpus (Zhang et al., 2018; Shang et al., 2020; Meng et al., 2020). Each topic node is defined by a set of semantically coherent terms related to a specific topic (i.e., topic term cluster), and each edge implies the "general-specific" relation between two topics (i.e., topic-subtopic). With the knowledge of hierarchical topic structures, topic taxonomies have been successfully utilized in many text mining applications, such as text summarization (Petinot et al., 2011; Bairi et al., 2015) and categorization (Meng et al., 2019; Shen et al., 2021).
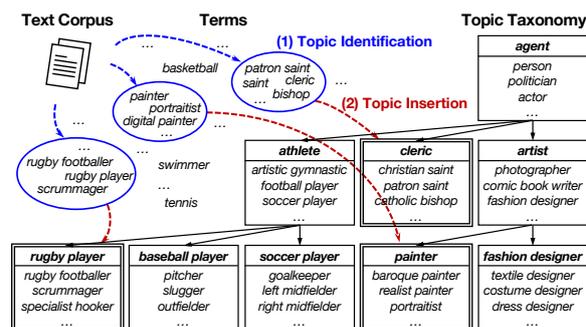

Figure 1: An example of topic taxonomy expansion. The known (i.e., existing) topics and novel topics are in single-line and double-line boxes, respectively.

Recently, automated expansion (or completion) of an existing topic taxonomy has been studied (Huang et al., 2020; Lee et al., 2022), which helps people to incrementally manage the topic knowledge within fast-growing document collections. This task has two technical challenges: (1) identifying new topics by collecting topic-related terms that have novel semantics, and (2) inserting the new topics at the right position in the hierarchy. In Figure 1, for example, a new topic node *painter* that consists of its topic-related terms [*baroque painter*, *realist painter*, *portraitist*, ...] is inserted at the child position (i.e., subtopic) of the existing topic node *artist*, without breaking the consistency of topic relations with the neighbor nodes.

The existing methods for topic taxonomy expansion, however, suffer from two major limitations: (1) *Limited term coverage* – They identify new topics from a set of candidate terms, while relying on entity extraction tools (Zeng et al., 2020) or phrase mining techniques (Liu et al., 2015; Shang et al., 2018; Gu et al., 2021) to obtain the high-frequency candidate terms in a corpus. Such extraction techniques will miss a lot of topic-related terms that have low frequency, and thus lead to an incomplete set of candidate terms (Zeng et al., 2021). (2) *Inconsistent topic relation* – As they insert new topics by considering only the first-order relation

---

*Corresponding author

between two topics (i.e., a topic and its subtopic), the newly-inserted topics are likely to have inconsistent relations with other existing topics. The expansion strategy based on the first-order topic relation is inadequate to capture the holistic structure information of the existing topic taxonomy.

As a solution to both challenges, we present TopicExpan, a new framework that expands the topic taxonomy via *hierarchy-aware topic term generation*. The key idea is to directly generate topic-related terms from documents by taking the topic hierarchy into consideration. From the perspective of term coverage, this generation-based approach can identify more multi-word terms even if they have low frequency in the given corpus (Zeng et al., 2021), compared to the extraction-based approach only working on the extracted candidate terms that frequently appear in the corpus. To combat the challenge of relation inconsistency, we utilize graph neural networks (GNNs) to encode the relation structure surrounding each topic (Kipf and Welling, 2017; Shen et al., 2021) and generate topic-related terms conditioned on these relation structure encodings. This allows us to accurately capture a hierarchical structure beyond the first-order relation between two topics.

To be specific, TopicExpan consists of *the training step* and *the expansion step*. The training step is for optimizing a neural model that topic-conditionally generates a term from an input document. Technically, for topic-conditional term generation, the model utilizes the relation structure of a topic node as well as the textual content of an input document. The expansion step is for discovering novel topics and inserting them into the topic taxonomy. To this end, TopicExpan places a *virtual* topic node underneath each existing topic node, and then it generates the topic terms conditioned on the virtual topic by utilizing the trained model. In the end, it performs clustering on the generated terms to identify multiple novel topics, which are inserted at the position of the virtual topic node.

Contributions. The main contributions of this paper can be summarized as follows: (1) We propose a novel framework for topic taxonomy expansion, which tackles the challenges in topic term coverage and topic relation consistency via hierarchy-aware topic term generation. (2) We present a neural model to generate a topic-related term from an input document *topic-conditionally* by capturing the hierarchical relation structure surrounding

each topic based on GNNs. (3) Our comprehensive evaluation on two real-world datasets demonstrates that output taxonomies of TopicExpan show better relation consistency as well as term coverage, compared to that of other baseline methods.

2 Related Work

Topic Taxonomy Construction. To build a topic taxonomy of a given corpus from scratch, the state-of-the-art methods have focused on finding out discriminative term clusters in a hierarchical manner (Zhang et al., 2018; Meng et al., 2020; Shang et al., 2020). Several recent studies have started to enrich and expand an existing topic taxonomy by discovering novel topics from a corpus and inserting them into the taxonomy (Huang et al., 2020; Lee et al., 2022). They leverage the initial topic taxonomy as supervision for learning the hierarchical relation among topics. To be specific, they discover new subtopics that should be inserted at the child of each topic, by using a relation classifier trained on (parent, child) topic pairs (Huang et al., 2020) or performing novel subtopic clustering (Lee et al., 2022). However, all the methods rely on candidate terms extracted from a corpus and also consider only the first-order relation between two topics, which degrades the term coverage and relation consistency of output topic taxonomies.

GNN-based Taxonomy Expansion. Recently, there have been several attempts to employ GNNs for expanding a given entity taxonomy (Mao et al., 2020; Shen et al., 2020; Zeng et al., 2021). Their goal is to figure out the correct position where a new entity should be inserted, by capturing structural information of the taxonomy based on GNNs. They mainly focus on an entity taxonomy that shows the hierarchical semantic relation among fine-grained entities (or terms), requiring plenty of nodes and edges in a given taxonomy to effectively learn the inter-entity relation. In contrast, a topic taxonomy represents coarse-grained topics (or high-level concepts) that encode *discriminative term meanings* as well as *term co-occurrences* in documents (Figure 1), which allows its node to correspond to a topic class of documents. That is, it is not straightforward to apply such methods to a topic taxonomy with much fewer nodes and edges, and thus how to enrich a topic taxonomy with GNNs remains an important research question.

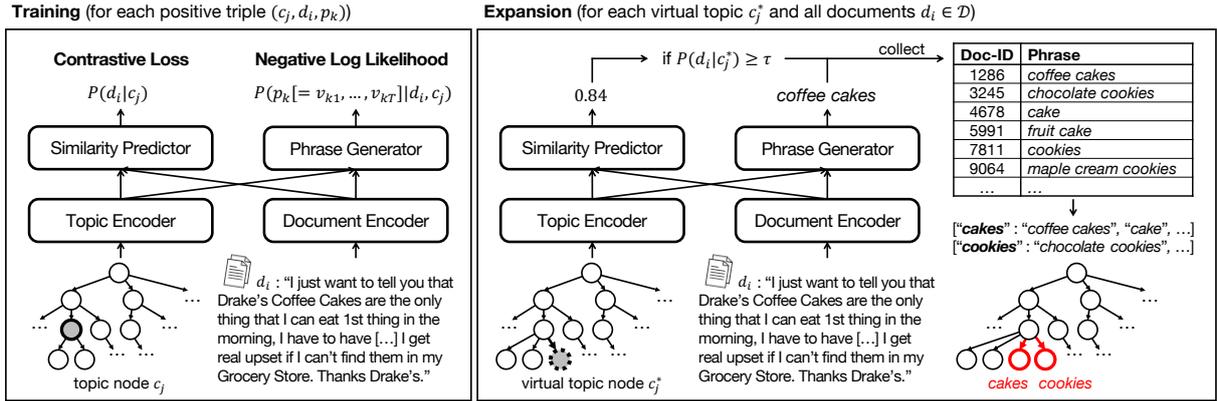**Keyphrase Generation.** The task of keyphrase prediction aims to find condensed terms that con-

Figure 2: The overall process of TopicExpan. (Left) It trains a unified model via multi-task learning of topic-document similarity prediction and topic-conditional phrase generation. (Right) It selectively collects the phrases conditionally-generated for a virtual topic node, and then it identifies multiple novel topics from phrase clusters.

cisely summarize the primary information of an input document (Liu et al., 2020). The state-of-the-art approach to this problem is modeling it as the text generation task, which sequentially generates word tokens of a keyphrase (Meng et al., 2017; Zhou et al., 2021). They adopt neural architectures as a text encoder and decoder, such as an RNN/GRU (Meng et al., 2017; Wang et al., 2019) and a transformer (Zhou et al., 2021). Furthermore, several methods have incorporated a neural topic model into the generation process (Wang et al., 2019; Zhou et al., 2021) to fully utilize the topic information extracted in an unsupervised way. Despite their effectiveness, none of them has focused on *topic-conditional* generation of keyphrases from a document, as well as *hierarchical* modeling of topic relations.

3 Problem Formulation

Notations. A *topic taxonomy* $\mathcal{T} = (\mathcal{C}, \mathcal{R})$ is a tree structure about topics, where each node ($\in \mathcal{C}$) represents a single conceptual topic and each edge ($\in \mathcal{R}$) implies the hierarchical relation between a topic and its subtopic. A topic node $c_j \in \mathcal{C}$ is described by the set of topic-related terms, denoted by $\mathcal{P}_j$ (i.e., term cluster for the topic c_j), where the most representative term (i.e., center term) serves as the topic name. Each document $d_i = [v_{i1}, \ldots, v_{iL}]$ and each term[1] $p_k = [v_{k1}, \ldots, v_{kT}]$ in a given corpus $\mathcal{D}$ is the sequence of L and T word tokens in the vocabulary set $v \in \mathcal{V}$, respectively. Here, each term is regarded as a phrase that consists of one or

---

[1] Note that our proposed approach considers each term as a sequence of word tokens, and this enables us to handle a much larger number of multi-word terms, compared to the case of using a precomputed set of candidate terms.

more word tokens, so the terms "phrase" and "term" are used interchangeably in this paper.

Problem Definition. Given a text corpus $\mathcal{D}$ and an initial topic taxonomy $\mathcal{T}$, the task of topic taxonomy expansion aims to discover novel topics by collecting the topic-related terms from $\mathcal{D}$ and insert them at the right position in $\mathcal{T}$ (Figure 1).

4 TopicExpan: Proposed Framework

4.1 Overview

TopicExpan consists of (1) *the training step* that trains a neural model for generating phrases topic-conditionally from documents (Figure 2 Left) and (2) *the expansion step* that identifies novel topics for each new position in the taxonomy by using the trained model (Figure 2 Right). The detailed algorithm is described in Section A.1.

Training Step. TopicExpan optimizes parameters of its neural model to maximize the total likelihood of the initial taxonomy $\mathcal{T}$ given the corpus $\mathcal{D}$.

$$
\begin{aligned}
P(\mathcal{T}; \mathcal{D}) &= \prod_{c_j \in \mathcal{C}} \prod_{p_k \in \mathcal{P}_j} P(p_k | c_j; \mathcal{D}) \\
&= \prod_{c_j \in \mathcal{C}} \prod_{p_k \in \mathcal{P}_j} \sum_{d_i \in \mathcal{D}} P(p_k, d_i | c_j) \quad (1) \\
&\approx \prod_{c_j \in \mathcal{C}} \prod_{d_i \in \mathcal{D}} \prod_{p_k \in \mathcal{P}_j \cap d_i} P(p_k | d_i, c_j) P(d_i | c_j).
\end{aligned}
$$

In the end, the total likelihood is factorized into the topic-conditional likelihoods of a document and a phrase, i.e., $P(d_i | c_j)$ and $P(p_k | d_i, c_j)$, for all the positive triples (c_j, d_i, p_k) collected from $\mathcal{T}$ and $\mathcal{D}$. That is, each triple satisfies the condition that its phrase p_k belongs to the topic c_j (i.e., $p_k \in \mathcal{P}_j$) and also appears in the document d_i.

To maximize Equation (1), we propose a unified model for estimating $P(d_i | c_j)$ and $P(p_k | d_i, c_j)$ via

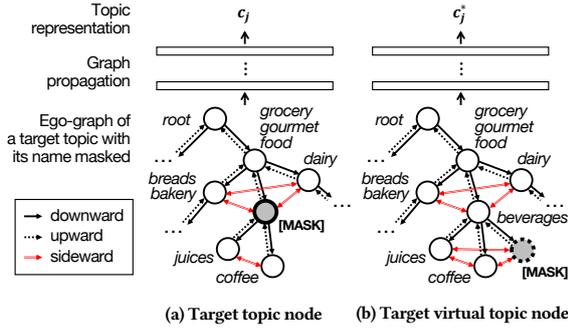**(a) Target topic node** **(b) Target virtual topic node**

Figure 3: The topic encoder architecture. It computes topic representations by encoding a topic relation graph.

the tasks of *topic-document similarity prediction* and *topic-conditional phrase generation*, respectively. In Figure 2 Left, for each positive triple (c_j, d_i, p_k), the former task increases the similarity between the topic c_j and the document d_i. This similarity indicates how confidently the document d_i includes any sentences or mentions about the topic c_j. At the same time, the latter task maximizes the decoding probability of the phrase p_k (i.e., generates the phrase) conditioned on the topic c_j and the document d_i. The model parameters are jointly optimized for the two tasks, and each of them will be discussed in Section 4.3.

Expansion Step. TopicExpan expands the topic taxonomy by discovering novel topics and inserting them into the taxonomy. To this end, it utilizes the trained model to generate the phrases p that have a high topic-conditional likelihood $P(p|c^*; \mathcal{D})$ for a new topic c^* from a given corpus $\mathcal{D}$. In Figure 2 Right, it first places a virtual topic node c_j^* at a *valid* insertion position in the hierarchy (i.e., a child position of a topic node c_j), and then it collects the phrases relevant to the virtual topic by generating them from documents $d_i \in \mathcal{D}$. Finally, it identifies multiple novel topics by clustering the collected phrases into semantically coherent but distinguishable clusters, which are inserted as the new topic nodes at the position of the virtual node. The details will be presented in Section 4.4.

4.2 Encoder Architectures

For modeling the two likelihoods $P(d_i|c_j)$ and $P(p_k|d_i, c_j)$, we introduce a topic encoder and a document encoder, which respectively computes the representation of a topic c_j and a document d_i.

4.2.1 Topic Encoder

There are two important challenges of designing the architecture of a topic encoder: (1) The topic

encoder should be *hierarchy-aware* so that the representation of each topic can accurately encode the hierarchical relation with its neighbor topics, and (2) the representation of each topic needs to be *discriminative* so that it can encode semantics distinguishable from that of the sibling topics. Hence, we adopt graph convolutional networks (GCNs) (Kipf and Welling, 2017) to capture the semantic relation structure surrounding each topic.

We first construct a topic relation graph $\mathcal{G}$ by enriching the edges of the given hierarchy $\mathcal{T}$ to model heterogeneous relations between topics, as shown in Figure 3. The graph contains three different types of inter-topic relations: (1) downward, (2) upward, and (3) sideward. The downward and upward edges respectively capture the top-down and bottom-up relations (i.e., hierarchy-awareness). We additionally insert the sideward edges between sibling nodes that have the same parent node. Unlike the downward and upward edges, the sideward edges pass the information in a negative way to make topic representations discriminative among the sibling topics. The topic representation of c_j at the m-th GCN layer is computed by

$$\boldsymbol{h}_j^{(m)} = \phi\left(\sum\nolimits_{(i,j)\in\mathcal{G}} \alpha_{r(i,j)} \cdot \boldsymbol{W}_{r(i,j)}^{(m-1)} \cdot \boldsymbol{h}_i^{(m-1)}\right), \quad (2)$$

where ϕ is the activation function, $r(i,j) \in \{\text{down}, \text{up}, \text{side}\}$ represents the relation type of an edge (i,j), and α indicates either positive or negative aggregation according to its relation type; i.e., $\alpha_{\text{down}} = \alpha_{\text{up}} = +1$ and $\alpha_{\text{side}} = -1$. The GloVe word vectors (Pennington et al., 2014) for each topic name are used as the base node features (i.e., $\boldsymbol{h}_j^{(0)}$) after being averaged for all tokens in the topic name. Using a stack of M GCN layers, we finally obtain the representation of a *target* topic node c_j (i.e., the topic node that we want to obtain its representation) by $\boldsymbol{c}_j = \boldsymbol{h}_j^{(M)}$.

The topic encoder should also be able to obtain the representation of a virtual topic node, whose topic name is not determined yet, during the expansion step. For this reason, we mask the base node features of a target topic node regardless of whether the node is virtual or not, as depicted in Figures 3(a) and (b). In other words, with the name of a target topic masked, the topic representation encodes the relation structure of its M-hop neighbor topics.

4.2.2 Document Encoder

For the document encoder, we employ a pretrained language model, BERT (Devlin et al., 2019). It

models the interaction among the tokens based on the self-attention mechanism, thereby obtaining each token's contextualized representation, denoted by $[\boldsymbol{v}_{i1}, \ldots, \boldsymbol{v}_{iL}]$. A document representation $\boldsymbol{d}_i$ is obtained by mean pooling in the end.

4.3 Learning Topic Taxonomy

In the training step, TopicExpan optimizes model parameters by using positive triples as training data $\mathcal{X} = \{(c_j, d_i, p_k)|p_k \in \mathcal{P}_j \cap d_i, \forall c_j \in \mathcal{C}, \forall d_i \in \mathcal{D}\}$ via multi-task learning of topic-document similarity prediction and topic-conditional phrase generation (Sections 4.3.1 and 4.3.2).

4.3.1 Topic-Document Similarity Prediction

The first task is to learn the similarity between a topic and a document. We define the topic-document similarity score by bilinear interaction between their representations, i.e., $\boldsymbol{c}_j^\top \boldsymbol{M} \boldsymbol{d}_i$ where $\boldsymbol{M}$ is the trainable interaction matrix. The topic-conditional likelihood of a document in Equation (1) is optimized by using this topic-document similarity score, $P(d_i|c_j) \propto \exp(\boldsymbol{c}_j^\top \boldsymbol{M} \boldsymbol{d}_i)$.

The loss function is defined based on InfoNCE (Oord et al., 2018), which pulls positively-related documents into the topic while pushing away negatively-related documents from the topic.

$$\mathcal{L}_{sim} = - \sum_{(c_j, d_i, p_k) \in \mathcal{X}} \log \frac{\exp(\boldsymbol{c}_j^\top \boldsymbol{M} \boldsymbol{d}_i / \gamma)}{\sum_{i'} \exp(\boldsymbol{c}_j^\top \boldsymbol{M} \boldsymbol{d}_{i'} / \gamma)}, \quad (3)$$

where γ is the temperature parameter. For each triple (c_j, d_i, p_k), we use its document d_i as positive and regard documents from all the other triples in the current mini-batch as negatives.

4.3.2 Topic-Conditional Phrase Generation

The second task is to generate phrases from a document being conditioned on a topic. For the phrase generator, we employ the architecture of the transformer decoder (Vaswani et al., 2017).

For topic-conditional phrase generation, the *context* representation, $\boldsymbol{Q}(c_j, d_i)$, needs to be modeled by fusing the textual content of a document d_i as well as the relation structure of a topic c_j. To leverage the textual features while focusing on the topic-relevant tokens, we compute *topic-attentive* token representations and pass them as the input context of the transformer decoder. Precisely, the topic-attention score of the l-th token in the document d_i, $\beta_l(c_j, d_i)$, is defined by its similarity with the topic.

$$\beta_l(c_j, d_i) = \exp(\boldsymbol{c}_j^\top \boldsymbol{M} \boldsymbol{v}_{il}) / \sum_{l'=1}^{L} \exp(\boldsymbol{c}_j^\top \boldsymbol{M} \boldsymbol{v}_{il'})$$
$$\boldsymbol{Q}(c_j, d_i) = [\beta_1(c_j, d_i) \cdot \boldsymbol{v}_{i1}, \ldots, \beta_L(c_j, d_i) \cdot \boldsymbol{v}_{iL}], \quad (4)$$

where the interaction matrix $\boldsymbol{M}$ is weight-shared with the one in Equation (3). Then, the sequential generation process of a token $\hat{v}_t$ is described by

$$\boldsymbol{s}_t = \text{Decoder}(\hat{v}_{<t}; \boldsymbol{Q}(c_j, d_i))$$
$$\hat{v}_t \sim \text{Softmax}(\text{FFN}(\boldsymbol{s}_t)). \quad (5)$$

FFN is the feed-forward networks for mapping a state vector $\boldsymbol{s}_t$ into vocabulary logits. Starting from the first token [BOP], the phrase is acquired by sequentially decoding a next token $\hat{v}_t$ until the last token [EOP] is obtained; the two special tokens indicate the begin and the end of the phrase.

The loss function is defined by the negative log-likelihood, where the phrase $p_k = [v_{k1}, \ldots, v_{kT}]$ in a positive triple (c_j, d_i, p_k) is used as the target sequence of word tokens.

$$\mathcal{L}_{gen} = - \sum_{(c_j, d_i, p_k) \in \mathcal{X}} \sum_{t=1}^{T} \log P(v_{kt}|v_{k(<t)}, c_j, d_i). \quad (6)$$

To sum up, the joint optimization of Equations (3) and (6) updates all the model parameters in an end-to-end manner, including the similarity predictor, the phrase generator, and both encoders.

4.4 Expanding Topic Taxonomy

In the expansion step, TopicExpan expands the topic taxonomy by utilizing the trained model to generate the phrases for a virtual topic, which is assumed to be located at a valid insertion position in the hierarchy. For thorough expansion, it considers a child position of every existing topic node as the valid position. That is, for each virtual topic node c_j^* (referring to a new child of a topic node c_j) one at a time, it performs topic phrase generation and clustering (Sections 4.4.1 and 4.4.2) to discover multiple novel topic nodes at the position.

4.4.1 Novel Topic Phrase Generation

Given a virtual topic node c_j^* and each document $d_i \in \mathcal{D}$, the trained model computes the topic-document similarity score and generates a topic-conditional phrase $p^* = [\hat{v}_1, \ldots, \hat{v}_T]$ where $\hat{v}_t \sim P(v_t|\hat{v}_{<t}, c_j^*, d_i)$. Here, the generated phrase p^* is less likely to belong to the virtual topic c_j^* if its source document d_i is less relevant to the virtual topic. Thus, we utilize the topic-document similarity score as the *confidence* of the generated phrase. To collect only qualified topic phrases, we filter out non-confident phrases whose normalized topic-document similarity is smaller than a threshold, i.e., $P(d_i|c_j^*) \approx \text{Norm}_{d_i \in \mathcal{D}}(\exp(\boldsymbol{c}_j^{*\top} \boldsymbol{M} \boldsymbol{d}_i)) < \tau$.

In addition to the confidence-based filtering, we exclude phrases that do not appear in the corpus at all, since they are likely implausible phrases. This has substantially reduced the *hallucination* problem of a generation model.

4.4.2 Novel Topic Phrase Clustering

To identify multiple novel topics at the position of the virtual topic node c_j^*, we perform clustering on the phrases collected for the virtual topic. We acquire semantic features of each phrase by averaging the GloVe vectors (Pennington et al., 2014) of word tokens in the phrase, then run k-means clustering with the initial number of clusters k manually set. Among the clusters, we selectively identify the new topics based on their cluster size, and the center phrase of each cluster is used as the topic name.

5 Experiments

5.1 Experimental Settings

Datasets. We use two real-world document corpora with their three-level topic taxonomy: **Amazon** (McAuley and Leskovec, 2013) contains product reviews collected from Amazon, and **DBPedia** (Lehmann et al., 2015) contains Wikipedia articles. All the documents in both datasets are tokenized by the BERT tokenizer (Devlin et al., 2019) and truncated to have maximum 512 tokens. The statistics are listed in Table 1.

Baseline Methods. We consider methods for building a topic taxonomy from scratch, **hLDA** (Griffiths et al., 2003) and **TaxoGen** (Zhang et al., 2018). We also evaluate the state-of-the-art methods for topic taxonomy expansion, **CoRel** (Huang et al., 2020) and **TaxoCom** (Lee et al., 2022).[2] Both of them identify and insert new topic nodes based on term embedding and clustering, with the initial topic taxonomy leveraged as supervision.

Experimental Settings. To evaluate the performance for novel topic discovery, we follow the previous convention that randomly deletes half of leaf nodes from the original taxonomy and asks each expansion method to reproduce them (Shen et al., 2020; Lee et al., 2022). Considering the deleted topics as *ground-truth*, we measure how completely new topics are discovered and how accurately they are inserted into the taxonomy.

Table 1: The statistics of the datasets.

Corpus	Vocab. size	# Documents	# Topic nodes
Amazon	19,615	29,487	531
DBPedia	27,435	196,665	298

5.2 Quantitative Evaluation

5.2.1 Topic Taxonomy Expansion

First of all, we assess the quality of output topic taxonomies. Following previous topic taxonomy evaluations (Huang et al., 2020; Lee et al., 2022), we recruit 10 doctoral researchers and use their domain knowledge to examine three different aspects of a topic taxonomy. **Term coherence** indicates how strongly terms in a topic node are relevant to each other. **Relation accuracy** computes how accurately a topic node is inserted into the topic taxonomy (i.e., *precision* for novel topic discovery). **Subtopic integrity** measures the completeness of subtopics for a topic node (i.e., *recall* for novel topic discovery). For exhaustive evaluation, we divide the output taxonomy of each expansion method into three disjoint parts $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$ so that each of them covers some first-level topics (and their subtrees) in Table 6 in Section A.5.[3]

In Table 2, TopicExpan achieves the highest scores for all the aspects.[4] For all the baseline methods, the term coherence is not good enough because they assign candidate terms into a new topic according to the topic-term relevance mostly learned from term co-occurrences. In contrast, TopicExpan effectively collects coherent terms relevant to a new topic (i.e., term coherence ≥ 0.90) by directly generating the topic-conditional terms from documents. TopicExpan also shows significantly higher relation accuracy and subtopic integrity than the other expansion methods, with the help of its GNN-based topic encoder that captures a holistic topic structure beyond the first-order topic relation.

5.2.2 Topic-Conditional Phrase Generation

We investigate the topic phrase prediction performance of our framework and other keyphrase extraction/generation models. We leave out 10% of the positive triples (c_j, d_i, p_k) from the training set $\mathcal{X}$ and use them as the test set. We measure **perplexity (PPL)** and **accuracy (ACC)** by comparing

---

[2]The implementation details and hyperparameter selection for the baselines and TopicExpan are in Sections A.2 and A.3.

[3]The details of the evaluation protocol are in Section A.5.

[4]The evaluation results obtain the Kendall coefficient of 0.96/0.91/0.84 (Amazon) and 0.93/0.90/0.91 (DBPedia) respectively for each aspect, which indicates strong inter-rater agreement on ranks of the methods.

Table 2: Quantitative evaluation on output topic taxonomies. The average and standard deviation for the three aspects are reported. The relation accuracy and subtopic integrity are considered only for the expansion methods, whose identified new topic nodes can be clearly compared with the ground-truth ones at each valid position.

Part	Methods	Amazon			DBPedia		
		Term Coherence	Relation Accuracy	Subtopic Integrity	Term Coherence	Relation Accuracy	Subtopic Integrity
	hLDA	0.2417 (0.0398)	N/A	N/A	0.2688 (0.0320)	N/A	N/A
	TaxoGen	0.4333 (0.1062)	N/A	N/A	0.4906 (0.1523)	N/A	N/A
$\mathcal{T}_1$	CoRel	0.5167 (0.1512)	0.4833 (0.1501)	0.2708 (0.1263)	0.5083 (0.1377)	0.6583 (0.1762)	0.2813 (0.1727)
	TaxoCom	0.6667 (0.1411)	0.5167 (0.0992)	0.3177 (0.1006)	0.5250 (0.2151)	0.6833 (0.1808)	0.2917 (0.1282)
	TopicExpan	**0.9750** (0.0496)	**0.8833** (0.1113)	**0.4948** (0.1309)	**0.9667** (0.0713)	**0.9333** (0.0713)	**0.5781** (0.1389)
$\mathcal{T}_2$	CoRel	0.5583 (0.1967)	0.6333 (0.1594)	0.2569 (0.1215)	0.4417 (0.1815)	0.5583 (0.1231)	0.1458 (0.1488)
	TaxoCom	0.6083 (0.1466)	0.6167 (0.1369)	0.4514 (0.1464)	0.4833 (0.1944)	0.7083 (0.1467)	0.2708 (0.1282)
	TopicExpan	**0.8917** (0.0707)	**0.8583** (0.1650)	**0.6597** (0.2062)	**0.9583** (0.0707)	**0.9167** (0.1054)	**0.5729** (0.1035)
$\mathcal{T}_3$	CoRel	0.5667 (0.1638)	0.5833 (0.1222)	0.2344 (0.1527)	0.6250 (0.1669)	0.7167 (0.1321)	0.3177 (0.1195)
	TaxoCom	0.5917 (0.1571)	0.6083 (0.0972)	0.1563 (0.1179)	0.5667 (0.1852)	0.6917 (0.1179)	0.4167 (0.1069)
	TopicExpan	**0.9167** (0.0690)	**0.9083** (0.1004)	**0.4531** (0.1068)	**0.9833** (0.0309)	**0.9417** (0.0904)	**0.6719** (0.0916)

Table 3: Performance for topic phrase generation.

Methods	Amazon		DBPedia	
	PPL ↓	ACC ↑	PPL ↓	ACC ↑
TopicExpan	**5.2553**	**0.6958**	**3.1108**	**0.7768**
(Encoder) BERT→Bi-GRU	5.7844	0.6884	3.5322	0.7645
(Decoder) Transformer→GRU	6.6649	0.6754	5.3690	0.6798
w/o Topic-attentive context	7.0907	0.6643	7.1679	0.6553
w/o Hierarchical topic relation	6.5345	0.6772	3.9802	0.7423
w/o Sideward topic relation	5.8705	0.6807	3.6985	0.7506
TextRank (Mihalcea and Tarau, 2004)	-	0.3023	-	0.1628
TopicRank (Bougouin et al., 2013)	-	0.2099	-	0.1092
TopicKG (Wang et al., 2019)	13.1298	0.2770	11.5663	0.3238
BERT-KG (Liu et al., 2020)	11.0229	0.4165	9.4723	0.4734
BERT-TKG (Zhou et al., 2021)	10.9746	0.4308	8.3607	0.4894

Figure 4: Examples of topic-conditional phrase generation, given a document and its relevant/irrelevant topic.

each generated phrase with the target phrase at the token-level and phrase-level, respectively.

In Table 3, TopicExpan achieves the best PPL and ACC scores. We observe that TopicExpan more accurately generates topic-related phrases from input documents, compared to the state-of-the-art keyphrase generation methods which are not able to consider a specific topic as the condition for generation. In addition, ablation analyses validate that each component of our framework contributes to accurate generation of topic phrases. Particularly, the hierarchical (i.e., upward and downward) and sideward relation modeling of the topic encoder improves the quality of generated phrases.

5.3 Qualitative Evaluation

5.3.1 Comparison of Topic Terms

We qualitatively compare the topic terms found by each method. In case of TopicExpan, we sort all confident topic terms by their cosine distances to the topic name (i.e., center term) using the global embedding features (Pennington et al., 2014).
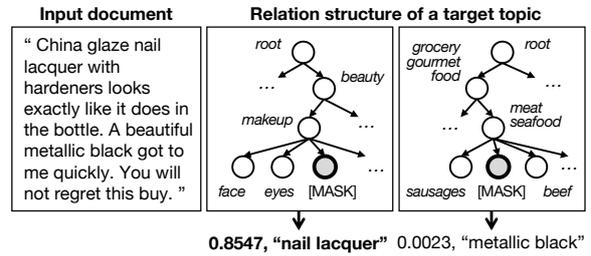
Table 4 shows that the topic terms of TopicEx-

pan are superior to those of the baseline methods, in terms of the expressiveness as well as the topic relevance. In detail, some of the terms retrieved by CoRel and TaxoCom are either off-topic or too general (marked with a strikethrough); this indicates that their topic relevance score for each term is not good at capturing the hierarchical topic knowledge of a text corpus. On the contrary, TopicExpan generates strongly topic-related terms by capturing the relation structure of each topic. Furthermore, TopicExpan is effective to find infrequently-appearing multi-word terms (underlined), which all the extraction-based methods fail to obtain.

5.3.2 Comparison of Novel Topics

Next, we examine novel topics inserted by each expansion method. To show the effectiveness of sideward relation modeling adopted by our topic encoder (Section 4.2.1), we additionally present the results of **TopicExpan^{+sr}** and **TopicExpan^{-sr}**, which computes topic representations with and without capturing the sideward topic relations.

In Table 5, TopicExpan^{+sr} successfully discovers new topics that should be placed in a target position. Notably, the new topics are clearly distin-

Table 4: Top-5 topic terms included in each topic node. The off-topic (or too general) terms are marked with a strikethrough, and the multi-word terms that are not obtainable by the extraction-based methods are underlined.

	Amazon	DBPedia
Topic	**Root → grocery gourmet food → beverages → tea**	**Root → work → periodical literature → magazine**
CoRel	blend, alvita, green tea, white tea, herbal tea	science fiction, comics, anthology, ~~monthly~~, newspaper
TaxoCom	earl grey, taylors of harrogate, black tea, herbal tea, teabags	~~books~~, cover, fashion, ~~specializing~~, publishers
TopicExpan	green tea, herbal tea, black tea, vanilla chai tea, jasmine tea	news mag.., monthly mag.., business mag.., comics mag.., fashion mag..
Topic	**Root → personal care → household supplies → dishwashing**	**Root → event → societal event → film festival**
CoRel	~~household~~, fresh, flushes, dirty, ~~biological~~	film festival, ~~music festival~~, annual event, ~~festival~~, cinema
TaxoCom	dishwasher, bubbles, dirty, smelly, nasty	dishwasher, ~~fest~~, ~~festival~~, independent film, annual festival
TopicExpan	dishwasher, dish soap, dish detergent, dishwasher gel, dishwasher soap	film festi.., short film festi.., annual film festi.., cannes film festi.., venice film festi..
Topic	**Root → baby products → diapering → cloth diapers**	**Root → sports season → sports team season → ncaa team season**
CoRel	cloth diaper, ~~diapers~~, ~~diaperbag~~, cloth diapering, baby wipes	naia, div.. ii, ncaa div.. ii, atlantic coast conference, mid american conference
TaxoCom	~~disposable diapers~~, ~~biodegradeable~~, cloth diaper, ~~diapering~~, bulky	mastodons, basketball tournament, college football, bulldogs, hoosiers
TopicExpan	cloth diaper, prefold diaper, diaper cover, diaper liner, reusable diaper	ncaa national team, ncaa tournament, ncaa div.. ii, ncaa div.. i, ncaa championship

Table 5: Novel topics identified at each target position. The center term (i.e., topic name) of each identified topic is presented. Correct topics (○), incorrect topics (⊗), and redundant topics (✄) are annotated.

	Amazon	DBPedia
Position	**Root → grocery gourmet food → beverages → ?**	**Root → agent → sports team → ?**
Sibling topics	**tea, coffee, hot cocoa, water, sports drinks**	**basketball team, cycling team, football team**
CoRel	apple cider (○), bottles (⊗), drinking (⊗), fruit juice (○), matcha (✄)	baseball (⊗), domestic competition (⊗), football club (✄), national ice hockey team (○), soccer (⊗)
TaxoCom	decaf chai (✄), espresso (✄), fizzy (○), juice (○), lipton (✄), mouthwash (⊗)	hockey (⊗), junior football team (✄), national team (○), regular season (⊗), rugby club (○)
TopicExpan^{-sr}	breakfast tea (✄), coconut water (○), fruit juice (○), natural cocoa (✄), vanilla coffee (✄)	american football team (✄), cricket team (○), cycling team (✄), professional basketball team (✄), rugby union team (○)
TopicExpan^{+sr}	coconut water (○), cream soda (○), decaf tea (✄), diet smoothie (○), redline energy drink (○)	beach handball team (○), cricket team (○), football club (✄), ice hockey team (○), rugby union team (○)

guishable from the sibling topics (i.e., known topics given in the initial topic hierarchy), which reduces the redundancy of the output topic taxonomy. On the other hand, CoRel and TaxoCom show limited performance for novel topic discovery; some new topics are redundant (✄) while some others do not preserve the hierarchical relation with the existing topics (⊗). Some of the new topics found by TopicExpan^{-sr} semantically overlap with the sibling topics, even though they are at the correct position in the hierarchy; this implies that our topic encoder with sideward relation modeling makes the representation of a virtual topic node *discriminative* with its sibling topic nodes, and it eventually helps to discover new conceptual topics of novel semantics.

5.3.3 Case Study of Topic Phrase Generation

To study how the generated phrases and their topic-document similarity scores (i.e., confidences) vary depending on a topic condition, we provide examples of topic-conditional phrase generation. The input document in Figure 4 contains a review about nail care products. In case that the relation structure of a target topic implies the nail product (Figure 4 Left), TopicExpan obtains the desired *topic-relevant* phrase "nail lacquer" along with the high topic-document similarity of 0.8547. On the other hand, given the relation structure of a target topic which is inferred as a kind of meat foods (Figure 4 Right), it generates a *topic-irrelevant* phrase "metallic black" from the document along with the low topic-document similarity of 0.0023. That is, TopicExpan fails to get a qualified topic phrase when the textual contents of an input document is obviously irrelevant to a target topic. In this sense, TopicExpan filters out non-confident phrases having a low topic-document similarity score to collect only the phrases relevant to each virtual topic.

5.4 Analysis of Topic-Document Similarity

Finally, we investigate the changes of generated phrases in two aspects, with respect to the topic-document similarity scores. The first aspect is the ratio of three categories for generated phrases, which have been focused on in the literature of keyphrase generation (Meng et al., 2017; Zhou et al., 2021): **(1) present phrases** appearing in the input document, **(2) absent phrases** not appear-
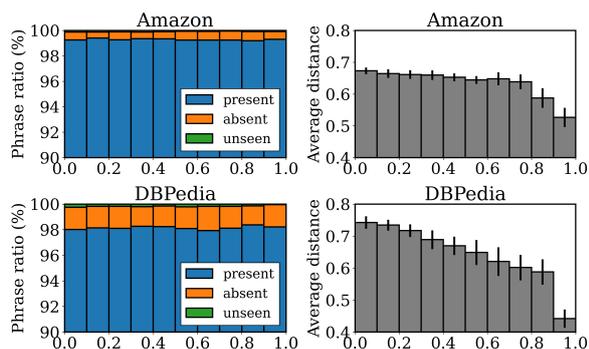
Figure 5: The ratio of three categories for generated phrases (Left) and the average semantic distance among generated phrases (Right). The horizontal axis shows 10 bins of normalized topic-document similarity scores.

ing in the input document but in the corpus at least once, and **(3) unseen (i.e., totally-new) phrases** that are not observed in the corpus at all. The second aspect is the average semantic distance among the phrases, measured by using the semantic features. For the plots in Figure 5, the horizontal axis represents 10 bins of normalized topic-document similarity scores over all generated phrases.

Interestingly, TopicExpan hardly generates absent phrases (about 0.7% for Amazon, 1.7% for DBPedia) and unseen phrases (about 0.1% for Amazon, 0.2% for DBPedia) regardless of the topic-document similarity; instead, it generates present phrases in most cases (Figure 5 Left). In other words, if the input document is not relevant to a target topic, it tends to generate an irrelevant-but-present phrase rather than a relevant-but-absent phrase, as shown in Section 5.3.3. One potential risk of TopicExpan is to generate unseen phrases that are nonsense or implausible, also known as *hallucinations* in neural text generation, and such unseen phrases can degrade the quality and credibility of output topic taxonomies. This result supports that we can easily exclude all unseen phrases, which account for less than 0.2% of generated phrases, to effectively address this issue.

Moreover, the negative correlation between the topic-document similarity score and the inter-phrase semantic distance (Figure 5 Right) provides empirical evidence that the similarity score can serve as the confidence of a generated topic phrase. There is a clear tendency toward decreasing the average semantic distance as the topic-document similarity score increases; this implies that the phrases generated from topic-relevant documents are semantically coherent to each other, and accordingly, they are likely to belong to the same topic.

6 Conclusion

In this paper, we study the problem of topic taxonomy expansion, pointing out that the existing approach has shown limited term coverage and inconsistent topic relation. Our TopicExpan framework introduces hierarchy-aware topic term generation, which generates a topic-related term by using both the textual content of an input document and the relation structure of a topic as the condition for generation. The quantitative and qualitative evaluation demonstrates that our framework successfully obtains much higher-quality topic taxonomy in various aspects, compared to other baseline methods.

For future work, it would be promising to incorporate an effective measure for the topic relevance of multi-word terms (i.e., phrases) into our framework. Since learning and utilizing the representation of multi-word terms remains challenging and worth exploring, it can be widely applied to many other text mining tasks.

7 Limitations

Despite the remarkable performance of TopicExpan on our tested corpus, there is still room to improve regarding how to better handle topics, documents, and phrases, for effective mining of topic knowledge. First, TopicExpan uses only the topic names (i.e., center terms) as the base node features in the topic relation graph, which makes our topic encoder difficult to capture the collective meaning of each topic from its set of topic-related phrases. Second, the confidence of each generated phrase considers only the topic relevance of its source document, instead of all the documents in which this phrase appears. Finally, the clustering process does not leverage the contextualized textual features computed by our BERT-based document encoder, which makes it hard to consolidate the context of the phrase within its source document.

References

Ramakrishna Bairi, Rishabh Iyer, Ganesh Ramakrishnan, and Jeff Bilmes. 2015. Summarization of multi-document topic hierarchies using submodular mixtures. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 553–563, Beijing, China. Association for Computational Linguistics.

Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. TopicRank: Graph-based topic ranking for keyphrase extraction. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 543–551, Nagoya, Japan. Asian Federation of Natural Language Processing.

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

Thomas Griffiths, Michael Jordan, Joshua Tenenbaum, and David Blei. 2003. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, volume 16.

Xiaotao Gu, Zihan Wang, Zhenyu Bi, Yu Meng, Liyuan Liu, Jiawei Han, and Jingbo Shang. 2021. UCPhrase: Unsupervised context-aware quality phrase tagging. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 478–486, New York, NY, USA. Association for Computing Machinery.

Jiaxin Huang, Yiqing Xie, Yu Meng, Yunyi Zhang, and Jiawei Han. 2020. CoRel: Seed-guided topical taxonomy construction by concept learning and relation transferring. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 1928–1936, New York, NY, USA. Association for Computing Machinery.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Dongha Lee, Jiaming Shen, Seongku Kang, Susik Yoon, Jiawei Han, and Hwanjo Yu. 2022. TaxoCom: Topic taxonomy completion with hierarchical discovery of novel topic clusters. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 2819–2829, New York, NY, USA. Association for Computing Machinery.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. 2015. Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Jialu Liu, Jingbo Shang, Chi Wang, Xiang Ren, and Jiawei Han. 2015. Mining quality phrases from massive text corpora. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, SIGMOD '15, page 1729–1744, New York, NY, USA. Association for Computing Machinery.

Rui Liu, Zheng Lin, and Weiping Wang. 2020. Keyphrase prediction with pre-trained language model. *CoRR*, abs/2004.10462.

Yuning Mao, Tong Zhao, Andrey Kan, Chenwei Zhang, Xin Luna Dong, Christos Faloutsos, and Jiawei Han. 2020. Octet: Online catalog taxonomy enrichment with self-supervision. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 2247–2257, New York, NY, USA. Association for Computing Machinery.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, page 165–172, New York, NY, USA. Association for Computing Machinery.

Rui Meng, Sanqiang Zhao, Shuguang Han, Daqing He, Peter Brusilovsky, and Yu Chi. 2017. Deep keyphrase generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 582–592, Vancouver, Canada. Association for Computational Linguistics.

Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2019. Weakly-supervised hierarchical text classification. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI'19, pages 6826–6833. AAAI Press.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Yu Zhang, Chao Zhang, and Jiawei Han. 2020. Hierarchical topic mining via joint spherical tree and text embedding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 1908–1917, New York, NY, USA. Association for Computing Machinery.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 3111–3119, Red Hook, NY, USA. Curran Associates Inc.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748.

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

Yves Petinot, Kathleen McKeown, and Kapil Thadani. 2011. A hierarchical model of web summaries. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 670–675, Portland, Oregon, USA. Association for Computational Linguistics.

Jingbo Shang, Jialu Liu, Meng Jiang, Xiang Ren, Clare R. Voss, and Jiawei Han. 2018. Automated phrase mining from massive text corpora. *IEEE Transactions on Knowledge and Data Engineering*, 30(10):1825–1837.

Jingbo Shang, Xinyang Zhang, Liyuan Liu, Sha Li, and Jiawei Han. 2020. NetTaxo: Automated topic taxonomy construction from text-rich network. In *Proceedings of The Web Conference 2020*, page 1908–1919, New York, NY, USA. Association for Computing Machinery.

Jiaming Shen, Wenda Qiu, Yu Meng, Jingbo Shang, Xiang Ren, and Jiawei Han. 2021. TaxoClass: Hierarchical multi-label text classification using only class names. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4239–4249, Online. Association for Computational Linguistics.

Jiaming Shen, Zhihong Shen, Chenyan Xiong, Chi Wang, Kuansan Wang, and Jiawei Han. 2020. TaxoExpan: Self-supervised taxonomy expansion with position-enhanced graph neural network. In *Proceedings of The Web Conference 2020*, page 486–497, New York, NY, USA. Association for Computing Machinery.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Yue Wang, Jing Li, Hou Pong Chan, Irwin King, Michael R. Lyu, and Shuming Shi. 2019. Topic-aware neural keyphrase generation for social media language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2516–2526, Florence, Italy. Association for Computational Linguistics.

Qingkai Zeng, Jinfeng Lin, Wenhao Yu, Jane Cleland-Huang, and Meng Jiang. 2021. Enhancing taxonomy completion with concept generation via fusing relational representations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 2104–2113, New York, NY, USA. Association for Computing Machinery.

Qingkai Zeng, Wenhao Yu, Mengxia Yu, Tianwen Jiang, Tim Weninger, and Meng Jiang. 2020. Tri-train: Automatic pre-fine tuning between pre-training and fine-tuning for SciNER. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4778–4787, Online. Association for Computational Linguistics.

Chao Zhang, Fangbo Tao, Xiusi Chen, Jiaming Shen, Meng Jiang, Brian Sadler, Michelle Vanni, and Jiawei Han. 2018. TaxoGen: Unsupervised topic taxonomy construction by adaptive term embedding and clustering. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 2701–2709, New York, NY, USA. Association for Computing Machinery.

Cangqi Zhou, Jinling Shang, Jing Zhang, Qianmu Li, and Dianming Hu. 2021. Topic-attentive encoder-decoder with pre-trained language model for keyphrase generation. In *IEEE International Conference on Data Mining, Auckland, New Zealand, December 7-10, 2021*, pages 1529–1534. IEEE.

A Supplementary Material

A.1 Pseudo-code of TopicExpan

Algorithm 1 describes the detailed process of our framework, including the training step (Lines 1–9) and the expansion step (Lines 10–23). The final output is the expanded topic taxonomy (Line 24).

Algorithm 1: The process of TopicExpan.

Input: Initial topic taxonomy $\mathcal{T} = (\mathcal{C}, \mathcal{R})$ and Text corpus $\mathcal{D}$
Output: Expanded taxonomy $\mathcal{T}'$

  // Step 1: Learning the topic taxonomy
1 $\mathcal{X} \leftarrow \text{COLLECTTRIPLES}(\mathcal{T}, \mathcal{D})$
2 $\mathcal{G} \leftarrow \text{CONSTRUCTGRAPH}(\mathcal{T})$
3 **while** *not converged* **do**
4   **for** $(c_j, d_i, p_k) \in \mathcal{X}$ **do**
5   Obtain the model outputs for the inputs $(\mathcal{G}, c_j, d_i)$
6   Compute $\mathcal{L}_{sim}$ by Equation (3)
7   Compute $\mathcal{L}_{gen}$ by Equation (6)
8      $\mathcal{L} \leftarrow \mathcal{L}_{sim} + \mathcal{L}_{gen}$
9      $\Theta \leftarrow \Theta - \eta \cdot \partial\mathcal{L}/\partial\Theta$

  // Step 2: Expanding the topic taxonomy
10 $\mathcal{T}' \leftarrow \mathcal{T}$
  // For each valid position
  (the child position of each topic)
11 **for** $c_j \in \mathcal{C}$ **do**
12    $\mathcal{T}^*, \mathcal{P}^* \leftarrow \mathcal{T}, \emptyset$
13    $c_j^* \leftarrow \text{MAKEVIRTUALNODE}(c_j)$
14    $\mathcal{T}^*.\text{INSERTNODE}(c_j, \{c_j^*\})$
15    $\mathcal{G}^* \leftarrow \text{CONSTRUCTGRAPH}(\mathcal{T}^*)$
16   **for** $d_i \in \mathcal{D}$ **do**
17   Obtain the model outputs for the inputs $(\mathcal{G}^*, c_j^*, d_i)$
18      $\hat{s} \leftarrow \exp(c_j^{*\top} M d_i)$
19      $\hat{p} \leftarrow [\hat{v}_1, \ldots, \hat{v}_T], \hat{v}_t \sim P(v_t|\hat{v}_{<t}, d_i, c_j^*)$
20      $\mathcal{P}^*.\text{APPEND}((\hat{s}, \hat{p}))$
21    $\mathcal{P}^* \leftarrow \text{FILTERBYNORMALIZEDSCORE}(\mathcal{P}^*, \tau)$
22    $c_{j1}^*, \ldots, c_{jK}^* \leftarrow \text{CLUSTERPHRASES}(\mathcal{P}^*)$
23    $\mathcal{T}'.\text{INSERTNODE}(c_j, \{c_{j1}^*, \ldots, c_{jK}^*\})$
24 **return** $\mathcal{T}'$

Training Step (Lines 1–9). TopicExpan first collects all positive triples (c_j, d_i, p_k) from an initial topic taxonomy $\mathcal{T}$ and a text corpus $\mathcal{D}$ (Line 1; Section 4.1), and constructs a topic relation graph $\mathcal{G}$ from the topic hierarchy (Line 2; Section 4.2.1). Then, it updates all the trainable parameters based on the gradient back-propagation (Lines 5–9) to minimize the losses for the topic-document similarity prediction task (Line 6; Section 4.3.1) and the topic-conditional phrase generation task (Line 7; Section 4.3.2).

Expansion Step (Lines 10–23). Using the trained model, TopicExpan discovers new topics that need to be inserted into each valid position in the topic
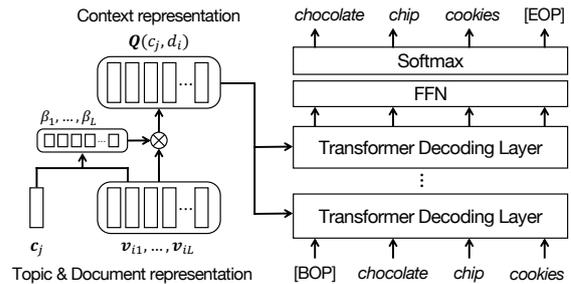

Figure 6: The phrase generator architecture. It generates the token sequence given a topic and a document, by using topic-attentive token representations as the context.

hierarchy (Line 11). For a virtual topic node c_j^* as a newly-introduced child of each topic node c_j (Line 13), it constructs a topic relation graph $\mathcal{G}^*$ from the topic hierarchy augmented with the virtual topic node (Lines 14–15). Then, it collects all pairs of a topic-document similarity score and a generated topic phrase $(\hat{s}, \hat{p})$, which are obtained by using the trained model on the augmented topic relation graph and all the documents (Lines 16–20; Section 4.4.1). Next, it filters out non-confident (i.e., irrelevant) phrases according to the normalized score (Line 21), then it performs clustering to find out multiple phrase clusters, each of which is considered as a new topic node having a novel topic semantics (Line 22; Section 4.4.2). In the end, it inserts the identified new topic nodes into the target position (i.e., the child of a topic node c_j) to expand the current topic taxonomy (Line 23).

A.2 Baseline Methods

For the baselines, we employ the official author codes while following the parameter settings provided by (Lee et al., 2022). For all the methods that optimize the Euclidean or spherical embedding space (i.e., TaxoGen, CoRel, and TaxoCom), we fix the number of negative terms (for each positive term pair) to 2 during the optimization.

- **hLDA**[5] (Griffiths et al., 2003) performs hierarchical latent Dirichlet allocation. It models a document generation process as sampling its words along the path selected from the root to a leaf. We set the smoothing parameters $\alpha = 0.1$ and $\eta = 1.0$, respectively for document-topic distributions and topic-word distributions, and the concentration parameter in the Chinese restaurant process $\gamma = 1.0$.

---

[5]https://github.com/joewandy/hlda

- **TaxoGen**[6] (Zhang et al., 2018) is the unsupervised framework for topic taxonomy construction. To identify hierarchical term clusters, it optimizes the term embedding space with SkipGram (Mikolov et al., 2013). We set the maximum taxonomy depth to 3 and the number of child nodes to 5, as done in (Zhang et al., 2018; Shang et al., 2020).

- **CoRel**[7] (Huang et al., 2020) is the first topic taxonomy expansion method. It trains a topic relation classifier by using the initial taxonomy, then recursively transfers the relation to find out candidate terms for novel subtopics. Finally, it identifies novel topic nodes based on term embeddings induced by SkipGram (Mikolov et al., 2013).

- **TaxoCom**[8] (Lee et al., 2022) is the state-of-the-art method for topic taxonomy expansion. For each node from the root to the leaf, it recursively optimizes term embedding and performs term clustering to identify both known and novel subtopics. we set $\beta = 1.5, 2.5, 3.0$ (for each level) in the novelty threshold τ_{nov}, and fix the signficance threshold $\tau_{sig} = 0.3$.

A.3 Implementation Details

Model Architecture. For the topic encoder, we use two GCN layers to avoid the over-smoothing problem, and fix the dimensionality of all node representations to 300. For the document encoder, we employ the `bert-base-uncased` provided by huggingface (Devlin et al., 2019), as the initial checkpoint of a pretrained model. It contains 12 layers of transformer blocks with 12 attention heads, thereby obtaining 768-dimensional contextualized token representations $[\boldsymbol{v}_{i1}, \ldots, \boldsymbol{v}_{iL}]$ (and a final document representation $\boldsymbol{d}_i =$ mean-pooling$(\boldsymbol{v}_{i1}, \ldots, \boldsymbol{v}_{iL})$) for an input document d_i. Consequently, the size of the interaction matrix $\boldsymbol{M}$ in our topic-document similarity predictor (Equation (3)) becomes 300×768. For the phrase generator, we adopt a single layer of the transformer decoder with 16 attention heads[9] and train its parameters from scratch without using the checkpoint of a pretrained text decoder. We limit

the maximum length of a generated phrase to 10. Figure 6 shows the phrase generator architecture. In total, our neural model contains 540K (for the topic encoder), 110M (for the document encoder), 230K (for the similarity predictor), and 30M (for the phrase generator) parameters.

Training Step. For the optimization of model parameters, we use the Adam optimizer (Kingma and Ba, 2015) with the initial learning rate 5e-5 and the weight decay 5e-6. The batch size is set to 64, and the temperature parameter γ in Equation (3) is set to 0.1. The best model is chosen using the best perplexity of generated topic phrases on the validation set of positive triples (c_j, d_i, p_k), which is evaluated every epoch.

Expansion Step. To filter out non-confident phrases (Section 4.4.1), we set the threshold value τ to 0.8 after applying min-max normalization on all topic-document similarity scores computed for each virtual topic node. To perform k-means clustering on the collected topic phrases (Section 4.4.2), we set the initial number of clusters k to 10, then select top-5 clusters by their cluster size (i.e., the number of phrases assigned to each cluster). The center phrase of each cluster is used as the final topic name of the new topic node.

A.4 Computing Platform

All the experiments are carried out on a Linux server machine with Intel Xeon Gold 6130 CPU @2.10GHz and 128GB RAM by using a single RTX3090 GPU. In this environment, the model training of TopicExpan takes around 2 hours and 6 hours for Amazon and DBPedia, respectively.

A.5 Quantitative Evaluation Protocol

For exhaustive evaluation on a large-scale topic taxonomy with hundreds of topic nodes, the output taxonomy of topic taxonomy expansion methods (i.e., CoRel, TaxoCom, and TopicExpan) is divided into three parts $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$ so that each part covers some of the first-level topics (and their subtrees) listed in Table 6.

Table 6: Three disjoint parts of the topic taxonomy.

Corpus	Part	First-level topics
Amazon	$\mathcal{T}_1$	grocery gourmet food, toys games
	$\mathcal{T}_2$	beauty, personal care
	$\mathcal{T}_3$	baby products, pet supplies
DBPedia	$\mathcal{T}_1$	agent, work, place
	$\mathcal{T}_2$	species, unit of work, event
	$\mathcal{T}_3$	sports season, device, topical concept

---

[6] https://github.com/franticnerd/taxogen
[7] https://github.com/teapot123/CoRel
[8] https://github.com/donalee/taxocom
[9] We empirically found that the number of decoding layers hardly affects the performance (i.e., accuracy) of topic-conditional phrase generation.

In case of hLDA and TaxoGen, the first-level topics in their output taxonomies are not matched with the ground-truth topics (in Table 6), because they build a topic taxonomy from scratch. For this reason, in Table 2, their output taxonomies are evaluated whole without partitioning. In addition, the two metrics for novel topic discovery (i.e., relation accuracy and subtopic integrity) are designed to evaluate the topic taxonomy expansion methods, so it is infeasible to measure the aspects on the output taxonomies of hLDA and TaxoGen. Thus, we only report the metric for topic identification (i.e., term coherence) in Table 2.

Term Coherence. It indicates how strongly terms in a topic node are relevant to each other. Evaluators count the number of terms that are relevant to the common topic (or topic name) among the top-5 terms found for each topic node.

Relation Accuracy. It computes how accurately a topic node is inserted into a given topic hierarchy (i.e., *precision* for novel topic discovery). For each valid position, evaluators count the number of newly-inserted topics that are in the correct relationship with the surrounding topics.

Subtopic Integrity. It measures the completeness of subtopics for each topic node (i.e., *recall* for novel topic discovery). Evaluators investigate how many ground-truth novel topics, which were deleted from the original taxonomy, match with one of the newly-inserted topics.

A.6 Examples of Topic Phrase Generation

We provide additional examples of topic-conditional phrase generation, obtained by TopicExpan. Figure 7 illustrates a confident phrase (Left) and a non-confident phrase (Right), generated from each input document and the given relation structure of a target topic, for both datasets. As discussed in Section 5.3.3, in case that a target topic is relevant to the document (i.e., high topic-document similarity score), TopicExpan successfully generates a phrase relevant to the target topic. On the other hand, in case that a target topic is irrelevant to the document (i.e., low topic-document similarity score), TopicExpan obtains a phrase irrelevant to the target topic.

(a) Dataset: Amazon
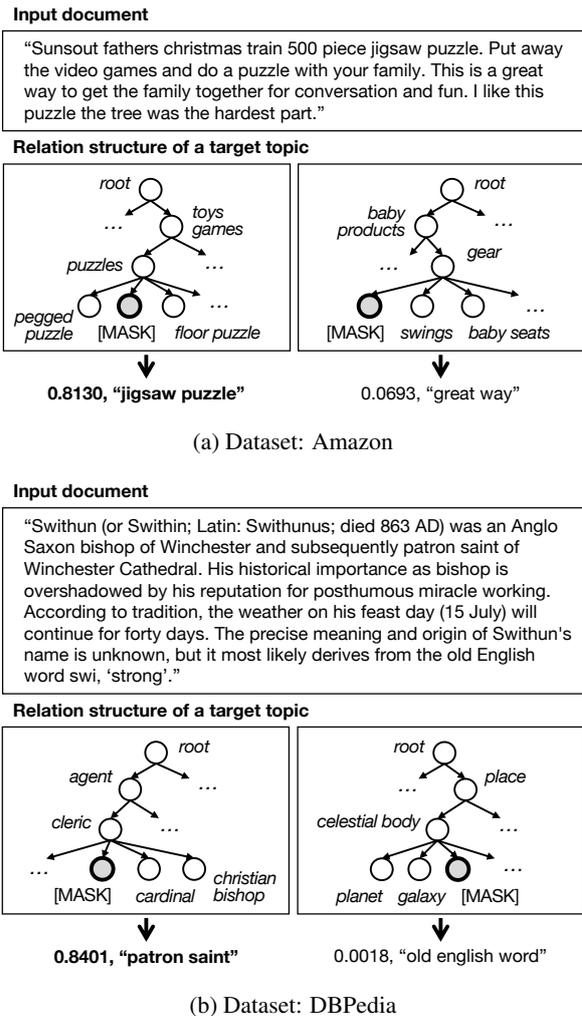

(b) Dataset: DBPedia

Figure 7: Examples of topic-conditional phrase generation, given a document and its relevant/irrelevant topic.