CoRel: Seed-Guided Topical Taxonomy Construction by Concept Learning and Relation Transferring

Jiaxin Huang[1], Yiqing Xie[2], Yu Meng[1], Yunyi Zhang[1], Jiawei Han[1]
[1]University of Illinois at Urbana-Champaign, IL, USA
[2]The Hong Kong University of Science and Technology, Hong Kong, China
[1]{jiaxinh3, yumeng5, yzhan238, hanj}@illinois.edu [2]yxieal@ust.hk

ABSTRACT

Taxonomy is not only a fundamental form of knowledge representation, but also crucial to vast knowledge-rich applications, such as question answering and web search. Most existing taxonomy construction methods extract hypernym-hyponym entity pairs to organize a "universal" taxonomy. However, these generic taxonomies cannot satisfy user's specific interest in certain areas and relations. Moreover, the nature of instance taxonomy treats each node as a single word, which has low semantic coverage. In this paper, we propose a method for seed-guided topical taxonomy construction, which takes a corpus and a seed taxonomy described by concept names as input, and constructs a more complete taxonomy based on user's interest, wherein each node is represented by a cluster of coherent terms. Our framework, CoRel, has two modules to fulfill this goal. A relation transferring module learns and transfers the user's interested relation along multiple paths to expand the seed taxonomy structure in width and depth. A concept learning module enriches the semantics of each concept node by jointly embedding the taxonomy and text. Comprehensive experiments conducted on real-world datasets show that CoRel generates high-quality topical taxonomies and outperforms all the baselines significantly.

CCS CONCEPTS

• **Information systems → Data mining**; *Clustering and classification*; • **Computing methodologies → Information extraction**; **Ontology engineering**.

KEYWORDS

Taxonomy Construction; Semantic Computing; Topic Discovery; Relation Extraction

ACM Reference Format:
Jiaxin Huang[1], Yiqing Xie[2], Yu Meng[1], Yunyi Zhang[1], Jiawei Han[1]. 2020. CoRel: Seed-Guided Topical Taxonomy Construction by Concept Learning and Relation Transferring. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, CA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394486.3403244

KDD '20, August 23–27, 2020, Virtual Event, CA, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403244

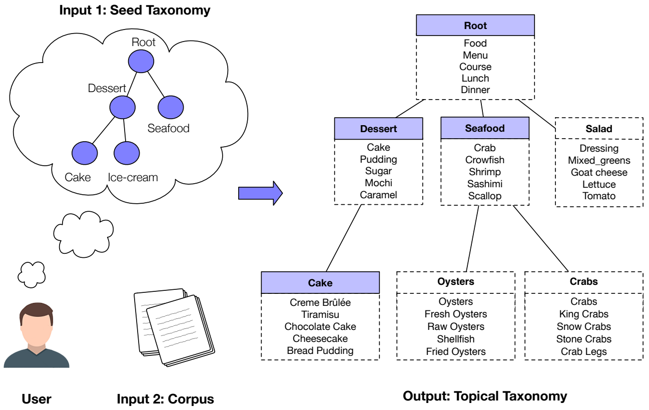


Figure 1: Seed-guided topical taxonomy construction. User inputs a partial taxonomy, and CoRel **extracts a more complete topical taxonomy based on user-interested aspect and relation, with each node represented by a cluster of words.**

1 INTRODUCTION

Taxonomy is an essential form of knowledge representation and plays an important role in a wide range of applications [19, 39, 41]. A taxonomy constructed from a large corpus organizes a set of concepts into a hierarchy, making it clear for people to understand relations between concepts.

Most existing taxonomy construction methods organize hypernym-hyponym entity pairs into a tree structure to form an instance taxonomy. However, a "universal" taxonomy so constructed cannot cater to user's specific needs. For example, a user might want to learn about concepts in a certain aspect (e.g., *food* or *research areas*) from a corpus. Generic taxonomy has two noteworthy limitations: (1) Countless irrelevant terms and fixed "is-a" relations dominate the instance taxonomy, failing to capture user's interested aspects and relations, and (2) each node is represented by a single word without considering term correlation, limiting people's understanding due to low semantic coverage, not to mention that synonyms could appear at multiple nodes.

We study the problem of seed-guided topical taxonomy construction, where user provides a seed taxonomy as guidance, and a more complete topical taxonomy is generated from text corpus, with each node represented by a cluster of terms (topics). As shown in Figure 1, a user provides a seed taxonomy and wants to generate a more complete food taxonomy from a given corpus. Such a more complete topical taxonomy can be hopefully constructed by expanding various types of food both in width and depth, with a cluster of descriptive terms for each concept node as a topic.

To fulfill this, we propose a framework CoRel, which approaches the problem with two modules: (i) A relation transferring module learns the specific relation preserved in seed taxonomy and attaches new concepts to existing nodes to complete the taxonomy structure.; and (ii) a concept learning module captures user-interested aspects and enriches the semantics of each concept node. Two challenges are met in the course: (1) Fine-grained concept names can be close in the embedding space, and enriching the concepts might result in relevant but not distinctive terms (e.g., "sugar" is relevant to both "cake" and "ice-cream"); and (2) with minimal user input, it is nontrivial to directly apply weakly supervised relation extraction methods to expand the taxonomy structure. Overall, noisy terms may harm the quality of new topics found.

To address these challenges, the relation transferring module first captures the relation preserved in seed parent-child pairs and transfers it upwards and downwards for finding the first-layer topics and subtopics, attached by a co-clustering technique to remove inconsistent subtopics. The concept learning module learns a discriminative embedding space by jointly embedding the taxonomy with text and separating close concepts in the embedding space.

We demonstrate the effectiveness of our framework through a series of experiments on two real-world datasets, and show that CoRel outperforms all the baseline methods in multiple metrics. We also provide qualitative analysis to demonstrate the high quality of our generated topical taxonomy and its advantages over other methods.

Our contribution can be summarized as follows: (1) A novel framework for seed-guided topical taxonomy construction. (2) A relation transferring module that passes the user-interested relation along multiple paths in different directions for taxonomy structure completion. (3) A concept learning module that enriches the semantics for a taxonomy of words by extracting distinctive terms. (4) Comprehensive experiments on real-world data with qualitative and quantitative studies that prove the effectiveness of CoRel.

2 RELATED WORK

Unsupervised Taxonomy Construction.

Most existing taxonomy construction algorithms perform a two-step approach: Hypernym-hyponym pairs are first extracted from a corpus and then organized into a tree structure. The task of hypernym-hyponym extraction traditionally relies on pattern-based methods which utilize Hearst patterns [11] like "NP such as NP" to acquire parent-child pairs that satisfy the "is-a" relation. Later, researchers design more lexical patterns [22, 23] or extract such patterns automatically [31, 32] in a bootstrapping method. These pattern-based methods suffer from low recall due to the diversity of expressions. Distributional methods alleviate the problem of sparsity by representing each word as a low-dimensional vector to capture their semantic similarity. There exist approaches [2, 37] inspired by Distributional Inclusion Hypothesis [42] (the context of a hyponym should be the subset of that of a hypernym) to detect hypernym-hyponym pairs without supervision. On the end of term organization, graph-based methods [14, 24, 34] are used to remove conflicts that form loops in the taxonomy. The aforementioned algorithms are not suitable for constructing a topical taxonomy, since they treat each word as a single node instead of forming topics with relevant terms for people to comprehend.

As another line of work, clustering-based taxonomy construction is closer to our problem setting. Clustering-based taxonomy construction methods first learn a representation space for terms, then perform clustering to separate terms into different topics by different measures [3, 16, 36, 38]. A recent method TaxoGen [40] finds fine-grained topics by spherical clustering and local-corpus embedding. Hierarchical topic modeling algorithms [1, 21] are comparable to these methods, since they organize terms to form a taxonomy of topics, each represented by a word distribution.

The above methods do not require supervision in taxonomy construction. They suffer from two disadvantages: (1) Without user input seed terms, they cannot capture users' specific interest in certain aspects of the corpus (e.g., "food" or "research areas"), thus the final output may include a large number of irrelevant terms; and (2) these methods either capture generic "is-a" patterns or do not enforce specific relations in children finding (clustering-based methods), thus cannot cater to user-interested relations.

Seed-Guided Taxonomy Construction.

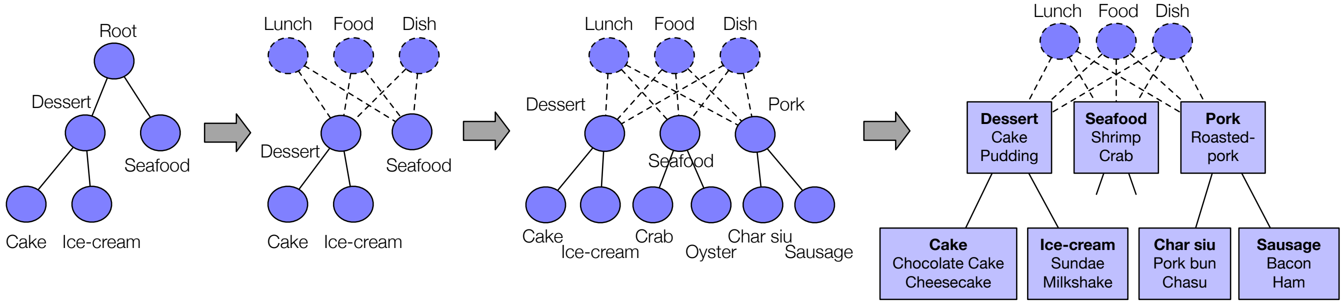
For seed-guided taxonomy construction, HiExpan [30] integrates the above two-step approach into a tree expansion process by width and depth expansion of the original seed taxonomy. Specifically, for width expansion that adds sibling nodes to those sharing the same parent, the method uses a set expansion algorithm [29] that leverages skip-gram features to calculate similarity between terms. For depth expansion that attaches children nodes to new node (e.g., attaching "oyster" to "seafood" in Figure 1), they use word analogy [20] to capture relations between parent-child pairs. However, in our setting of constructing topical taxonomy, HiExpan suffers from two drawbacks: (1) the set expansion algorithm is not good at expanding concepts; and (2) word analogy is only locally preserved in the Word2Vec space [8].

Weakly Supervised Relation Extraction.

To construct a taxonomy that fits in with a user-interested relation, we aim to preserve the same relation between all newly added parent-child topics. With only a few given seeds, it is impossible to train a highly accurate and complicated relation extraction model with a huge number of parameters. Traditional weakly supervised relation extraction methods [22, 32] find textual patterns from given instances, suffering from sparsity of relation expressions. Recent studies [26] combine textual patterns with distributional features for mutual enhancement in a co-training framework. Neural-based methods like prototypical network [9] which represents each relation as a vector have shown to be effective in few shot relation (FSL) extraction. Recent advances in contextualized text representation show that deep transformers (e.g., BERT [6]) learn task-agnostic representations achieving strong performance on various NLP tasks. Researchers show that by learning from large amounts of entity pairs co-occurring in Wikipedia corpus, BERT can achieve state-of-the-art [33] on FSL relation extraction on benchmark datasets [10].

Supervised Taxonomy Construction.

Most supervised taxonomy construction methods focus on extracting hypernym-hyponym pairs. Word analogy (v(man)−v(king) = v(woman) − v(queen)) is preserved in local clusters, and a piecewise linear projection from words to their hypernyms is trained



Step 1: Relation transferring upwards

Step 2: Relation transferring downwards

Step 3: Concept learning for generating topical clusters

Figure 2: Workflow of CoRel.

in [8]. For neural-based methods, order embedding [4, 35] is proposed to express partial orders between words. Later studies show that Poincaré space can be viewed as a continuous generalization of tree structures, so they embed large taxonomy structures extracted from WordNet [25] or Wikipedia [15]. However, in our setting, the user-given taxonomy is of very limited size, and thus cannot be adaptive to these frameworks.

3 PROBLEM DESCRIPTION

In this section we describe the task of seed-guided topical taxonomy construction. The inputs are a collection of documents $\mathcal{D} = \{d_1, d_2, ..., d_{|\mathcal{D}|}\}$ and a tree-structured seed taxonomy T^0 provided by user. Each node e in T^0 is represented by a single word from the corpus, and each edge $\langle p_0, c_0 \rangle$ implies user-interested relation between a parent-child pair, such as "is a subfield of" or "is a type of". The output is a more complete topical taxonomy T, with each node e as a conceptual topic, represented by a coherent cluster of words describing the topic. Figure 1 shows an example of our task.

The meanings of the notations used in this paper are presented in Table 1.

Table 1: Notations and Meanings.

Notation	Meaning
T	The tree structure of taxonomy.
R	Root node of taxonomy T.
e	A concept node on taxonomy T.
C_e	The topic cluster of concept e.
N_e	Nodes sharing common parent with e, including e.
B_e	The children nodes of e.
$\langle p, c \rangle$	A pair of direct parent and child nodes in T.

4 METHODOLOGY

In this section, we introduce our proposed method by first giving an overview in Section 4.1 and then describing the details of two modules in Sections 4.3 and 4.2.

4.1 Method Overview

Figure 2 shows the workflow of CoRel. To expand the tree structure of a user-given seed taxonomy T^0, CoRel first leverages a relation transferring module to capture seed relations of edge $\langle p_0, c_0 \rangle$. In step 1, it attempts to discover potential root concepts by transferring the relation upwards, such as "Lunch", "Food" and "Dish" as more general concepts to cover the topics. In step 2, the relation is transferred downwards to attach new topics (internal nodes) as well as new subtopics (leaf nodes). Finally, a concept learning module is used to learn a discriminative embedding space to generate topical clusters for each concept node. Below we address the two modules in detail.

4.2 Taxonomy Completion by Relation Transferring

The relation transferring module is used to complete the taxonomy structure by finding new topics and subtopics. This module first captures the relation between user given $\langle p, c \rangle$ pairs by training a relation classifier on the given corpus. The relation classifier takes a relation statement (will be defined in section 4.2.1) of a pair of terms as input, and judges whether there exists user-interested relation and which direction it is between the pair. After training the relation classifier, we transfer the relation upwards for root node discovery, and then transfer the relation downwards to find new topics/subtopics as the child of root/topic node. In both the example and evaluation we construct two layers of topics, though this module can be applied to discover more fine-grained topics by further going down.

4.2.1 Self-supervised Relation Learning. Previous studies show that word analogy can capture relations between words to some extent [20, 30], but vector offset is only preserved in a local area in the embedding space [8]. To deal with more complicated relations, our choice of the model is inspired by the effectiveness of the pre-trained deep language model, BERT[6], on wide downstream applications. [33] also shows its power in few-shot relation learning by training the model in a distant supervised setting using large amounts of pairs of entities on Wikipedia corpus. In our setting, user gives minimal seeds which limits the potential to train such deep language model, thus we only employ the pre-trained BERT model and train a relation classifier as shown in Figure 3.

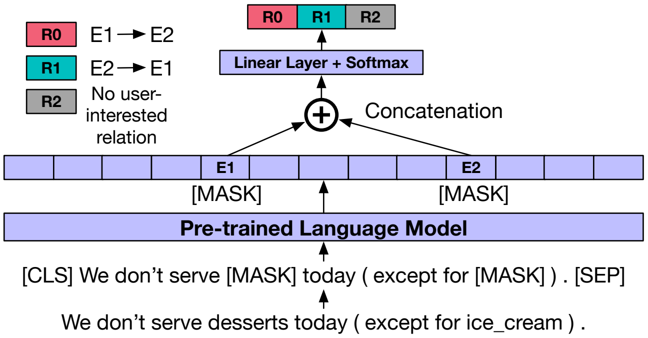

R0
E1 → E2
R1
E2 → E1
R2
No user-interested relation
R0
R1
R2
Linear Layer + Softmax
Concatenation
E1
E2
[MASK]
[MASK]
Pre-trained Language Model
[CLS] We don't serve [MASK] today (except for [MASK]) . [SEP]
We don't serve desserts today (except for ice_cream) .

Figure 3: Our Weakly Supervised Relation Classifier.

Relation Statement. We assume that if a pair of $\langle p,c \rangle$ co-occurs in a sentence in the corpus, then that sentence implies their relation. We refer to sentences containing $\langle p,c \rangle$ as their relation statements and leverage a pre-trained deep language model to understand the relation statements. To learn the user-interested specific relation, we extract all the relation statements of user-given $\langle p_0,c_0 \rangle$ as positive training samples. We collect negative training sentences in two ways: (1) relation statements of sibling nodes, thus avoiding the model to only find closely related terms; and (2) random sentences from the corpus, so the model can learn from irrelevant contexts to avoid overfitting.

Sequence Input Representation. Since user gives only a minimal number of seeds, which is not enough to train deep encoders, we cannot simply add explicit markers around pairs of terms to let the model pay attention to. Therefore, we take the original sentence and replace the two terms by "[MASK]" tokens with two justifications: (1) aligning with the pre-trained objective of masked language model; and (2) avoiding the classification layer to only remember relations from training pairs instead of looking into contexts.

Classification Layer. We take the output of two "[MASK]" tokens from the last layer of the pre-trained language model, and concatenate them to be the input of the classification layer, where we use a simple linear layer before the softmax layer. The output label chooses the relation from e_1 to e_2 among three classes in the relation set Q: $\langle e_1 \rightarrow e_2 \rangle$ (i.e., e_1 is a parent node of e_2), $\langle e_2 \rightarrow e_1 \rangle$, or non-user interested relation.

Data Augmentation. To fully utilize the asymmetric property along the taxonomy edges, we augment each input training sequence by reversing the order of concatenation of $e1$ and $e2$. Then the label would switch if user-interested relation exists between the pair, but will not change otherwise.

4.2.2 *First-layer Topic Finding by Root Node Discovery.*

After deriving a relation classifier, we can easily transfer the user-interested relation along the paths in the taxonomy. This is done by targeting an existing node and finding entities to be its potential parent node (transferring upwards) or child node (transferring downwards). To "expand" more first-layer topics based on user-given ones, previous work like set expansion is good at extending a list of instances like country names or company names [12, 27, 29], but is not perfect at expanding concept names. Another seemingly straightforward solution is to train a few-shot sibling relation classifier based on relation statements of sibling nodes. However, this would result in low recall of extracted new topics, since the co-occurrence of two relatively unrelated topics might be sparse in the corpus.

To resolve the above issue, we assume that if we can discover potential root nodes, such as "Food" for "Dessert" and "Seafood", then the root node would have more general contexts for us to find connections with potential new topics. Previous studies [17, 42] also found general concepts cover broader semantic meaning and have more varied contexts.

Specifically, we transfer the relation upwards by using the relation classifier learned to extract a list of parent nodes for each seed topic. The common parent nodes for all topics are treated as root nodes R.

Finding common root nodes. To find the parent or child of an existing node, we extract relation statements of a concept e and a candidate term w into the relation classifier to judge sentence-based relations. Corpus-based relation between w and e is then averaged over confident sentence-based results over the corpus, with the confidence threshold being δ.

$$\text{Score}(w \longrightarrow e) = \frac{\sum_{s_{w \to e}} \mathbb{1}\left(KL\left(\boldsymbol{l} \middle\| \boldsymbol{p}_w\right) > \delta\right)}{\sum_{q \in Q} \sum_{s_q} \mathbb{1}\left(KL\left(\boldsymbol{l} \middle\| \boldsymbol{p}_w\right) > \delta\right)} \tag{1}$$

where s_q denotes relation statements in which the relation of q exists, $\boldsymbol{p}_w$ denotes the output probability from the relation classifier, and l is the uniform distribution vector among three classes of relations. Thus if the KL divergence between the two distributions is larger than a threshold δ, we treat the prediction as a confident one. Eq. (1) calculates the portion of term w being the parent of concept e among all the confident predictions, and we confirm w as the parent node of e if the portion is larger than a threshold. For each user-given first-layer topic, we can generate a list of parent nodes, and their common parent nodes are treated as root nodes R.

Finding new first-layer topics. We apply the relation classifier to extract child terms for each root node $r \in R$. This is done in a similar way as root node discovery, but we only reverse the direction of relation. Thus we need to replace $(w \longrightarrow e)$ in Eq. (1) with $(r \longrightarrow w)$. New topics are selected by their average score over all root nodes.

$$\text{Score}(R \longrightarrow w) = \frac{\sum_{r \in R} \text{Score}(r \longrightarrow w)}{|R|} \tag{2}$$

4.2.3 *Candidate term extraction for subtopics.*

After generating the first-layer topics, we transfer the relation downwards to discover subtopics of each first-layer topic. This can be done by applying Eq. (1) again and replacing $(w \longrightarrow e)$ with $(e \longrightarrow w)$. The candidate terms will later be clustered into subtopics.

4.3 Generating Topical Clusters by Concept Learning

Our concept learning module is used to learn a discriminative embedding space, so that each concept is surrounded by its representative terms. Within this embedding space, subtopic candidates are also clustered to form coherent subtopic nodes. This is motivated by the fact that relevant concept names can be close to each other in the embedding space, and directly using unsupervised word embedding such as Word2Vec [20] might result in relevant but not distinctive terms (e.g., "food‘‘ is relevant to both "seafood‘‘ and “dessert‘’). Thus we use the expanded taxonomy as input to guide the word embedding learning process.

4.3.1 Concept Learning based on Taxonomy and Corpus. We basically design three loss functions to embed the concepts, words and documents in a joint embedding space. Our starting point is the assumption that similar words share similar local contexts, as is used by the Skip-Gram model [20]. Two sets of embedding for each word are used: center word embedding denoted as u_w and context word embedding as v_w. The training objective is to maximize the log probability of observing words in a fixed local context window with the size of h.

$$L_l = -\sum_{d \in D} \sum_{1 \le i \le |d|} \sum_{0 < |j-i| \le h} \log P(w_j \mid w_i) \tag{3}$$

where $P(w_j \mid w_i) \propto \exp\left(u_{w_i} v_{w_j}\right)$.

Recent studies [17, 18] also observe the importance of modeling the documents where a word appears in, since words in similar documents share topical coherence. We denote document embedding as d and maximize the log probability of predicting the correct document that a word belongs to.

$$L_d = -\sum_{d \in D} \sum_{1 \le i \le |d|} \log P(d \mid w_i) \tag{4}$$

where $P(d \mid w_i) \propto \exp\left(u_{w_i} u_d\right)$.

Since we want to regularize the embedding space to be discriminative among the concepts in the expanded taxonomy, we wish to form topical clusters in the embedding space where each concept embedding is surrounded by its representative terms. We use an iterative approach to gradually grab distinctive words at each epoch. Specifically, we add one distinctive word to each concept cluster C_e at each epoch to avoid semantic drift. We then enforce the proximity between concept embedding and their clusters by

$$L_{prox} = \sum_{e \in \mathcal{T}} \sum_{w \in C_e} \log P(e \mid w) \tag{5}$$

where $P(e \mid w) \propto \exp\left(u_{w_i} u_e\right)$

The overall training objective is a weighted sum of the above terms.

$$L = L_l + \lambda_d L_d + \lambda_p L_{prox} \tag{6}$$

4.3.2 Topic and Relation aware Subtopic Finding. To find subtopics for existing concepts in the seed taxonomy, we apply two constraints for generating potential candidates for subtopics: (1) Topical constraints: candidates should belong to the topic cluster C_e of that concept e; and (2) relational constraints: candidates should bear user-interested relation with the concept. The two constraints can be applied by using the learned relation classifier and concept embedding. However, directly using the few-shot relation classifier can still include noisy and non-consistent terms, thus we carry out a co-clustering method to further filter out those noisy terms.

An example is shown in Table 2, where we use a Topic-Type table to organize the valid terms from the relation classifier. Valid terms are divided in columns by semantic meaning and in rows by semantic type (e.g., *food*, *cooking style* or *sauces*). It is easily observable that the fourth subtopic of “pieces, slices” is an outlier sharing little type similarity with other subtopics. Thus we apply co-clustering method to retain those subtopics sharing similar semantic type distribution.

Beef		Pork	Bread	
sliced beef	sirloin, rare beef	roasted pork		flat bread, wheat
stewed				toasted
			pieces, slices	
	black pepper	spicy sauce		buttery

Table 2: An example of a Topic-Type table.

Topic-Type Matrix Creation. We construct an indicative (0/1) Topic-Type matrix to represent the joint distribution of subtopics and types of candidates from the Topic-Type table as shown in Table 2, and the table is created by the following process: The topic-wise clustering is done by affinity propagation (AP) clustering [7] in the discriminative embedding space trained by concept learning, where the concepts are separated away from each other to avoid overlapping. The type-wise clustering is conducted by AP on the average BERT embedding space: We first retrieve the contextualized embedding of each candidate mention using the last layer output of BERT, and then average over the mentions to get the embedding for each candidate.

Co-clustering of the Matrix. Finally, to extract high quality subtopics, we apply co-clustering [13] on the indicative Topic-Type matrix M and define a consistency score for each cluster.

$$\text{Consistency}(\text{Cluster}_k) = \frac{\sum_{\text{row label[i]==k}} \sum_{\text{col. label[j]==k}} M_{i,j}}{\sum_{\text{row label[i]==k}} \sum_{\text{col. label[j]==k}} 1} \tag{7}$$

If the consistency score of a cluster is high, then the cluster consists of multiple subtopics that share similar semantic types. We retain high quality subtopics by setting a threshold for the consistency score.

4.4 Overall Algorithm

We summarize the overall algorithm of seed-guided topical taxonomy construction in Algorithm 1.

5 EXPERIMENTS AND RESULTS

5.1 Experiment Setup

Datasets. Our experiments are conducted on two large real-world datasets: (1) **DBLP** contains around 157 thousand abstracts from publications in the field of computer science. For preprocessing, we use AutoPhrase [28] to extract meaningful phrases to serve as our vocabulary, we further discard infrequent terms occuring less than 50 times, resulting in 16650 terms. (2) **Yelp** is collected from the recent released *Yelp Dataset Challenge*[1], containing around 1.08 million restaurant reviews. Similarly, we extract meaningful phrases and remove infrequent terms, resulting in 14619 terms[2].

Hyperparameter setting. For our relation classifier, the hyperparameter is set to be: batch size = 16, training epochs = 5, model: Bert-Base (12 layers, 768 hidden size, 12 heads). When training

[1] https://www.yelp.com/dataset/challenge

[2] Our code and data are available at https://github.com/teapot123/CoRel

Algorithm 1: Seed-guided Topical Taxonomy Construction.

```
Input: A text corpus $\mathcal{D}$; a given taxonomy $\mathcal{T}^0$ consisting
       of nodes $\{e_i\}|_{i=1}^{n}$ and edges $\langle p_i, c_i \rangle|_{i=1}^{m}$.
Output: A more complete taxonomy $\mathcal{T}$ with each node $e$
        represented by a cluster of terms.
Initialize relation training sample list $\mathcal{S}$;
for $i \leftarrow 1$ to $m$ do
    Extract sentences $S_{p_i,c_i}$ where $p_i$ and $c_i$ co-occur;
    $\mathcal{S} \leftarrow S_{p_i,c_i}$;
Train the relation classifier $F$ according to Section 4.2.1;
$R \leftarrow$ root nodes discovered by Section 4.2.2;
Initialize new first-layer topic candidates $e_{\text{new}}$ by
  co-occurred terms of $r \in R$;
Score($R \rightarrow e_{\text{new}}$) $\leftarrow$ Equation 2 ;
$\mathcal{B}_R \leftarrow \mathcal{B}_R \cup \{e_{\text{new}}$ if Score($R \rightarrow e_{\text{new}}$) $> \gamma\}$ ;
  ▹ New first-layer topics found are attached to root node.;
for internal nodes $e$ in $\mathcal{T}$ do
    $\mathbf{B} \leftarrow$ Candidate terms extraction by Section 4.2.3 ;
    $\mathcal{B}_e \leftarrow \mathcal{B}_e \cup \mathbf{B}$;
Train a joint embedding space of words and concepts;
Extract topical words $C_e$ for internal nodes $e \in \mathcal{T}$ ;
Subtopic Finding by Section 4.3.2;
Return topical taxonomy $\mathcal{T}$;
```

the relation classifier, we make a 90/10 training/validation split on training samples. In our relation transferring process, we set the threshold for relation score in Eqs. (1) and (2) to 0.7, and the threshold for KL divergence δ to 0.5. For our concept learning module, we set the following hyperparameters for embedding training process: embedding dimension = 100, local context window size = 5, λ_d = 1.5, and λ_p = 1.0. The threshold for Cluster Consistency is 0.5. We use the same hyperparameters for both datasets.

Compared Methods. We compare CoRel to several previous corpus-based taxonomy construction algorithms. To the best of our knowledge, there is no seed-guided topical taxonomy construction methods where each node is represented by a cluster of words, so we also compare CoRel with some unsupervised methods.

- Hi-Expan [30] + Concept Learning: Hi-Expan is a seed-guided instance-based taxonomy construction algorithm, which has the same input as our setting and constructs the taxonomy structure by set expansion and word analogy. Since its output node is represented by single word, we apply our concept learning module to enrich each node with a cluster of words.
- TaxoGen [40]: An unsupervised topical taxonomy construction method. It uses spherical clustering and local-corpus embedding to discover fine-grained topics represented by clusters of words.
- HLDA [1]: A non-parametric hierarchical topic model. It models the generation of documents in a corpus as sampling words from the paths when moving from the root node to a leaf node. Thus we can take each node as a topic.
- HPAM [21]: A state-of-the-art hierarchical topic model which requires a pre-defined number of topics and outputs topics at different levels.

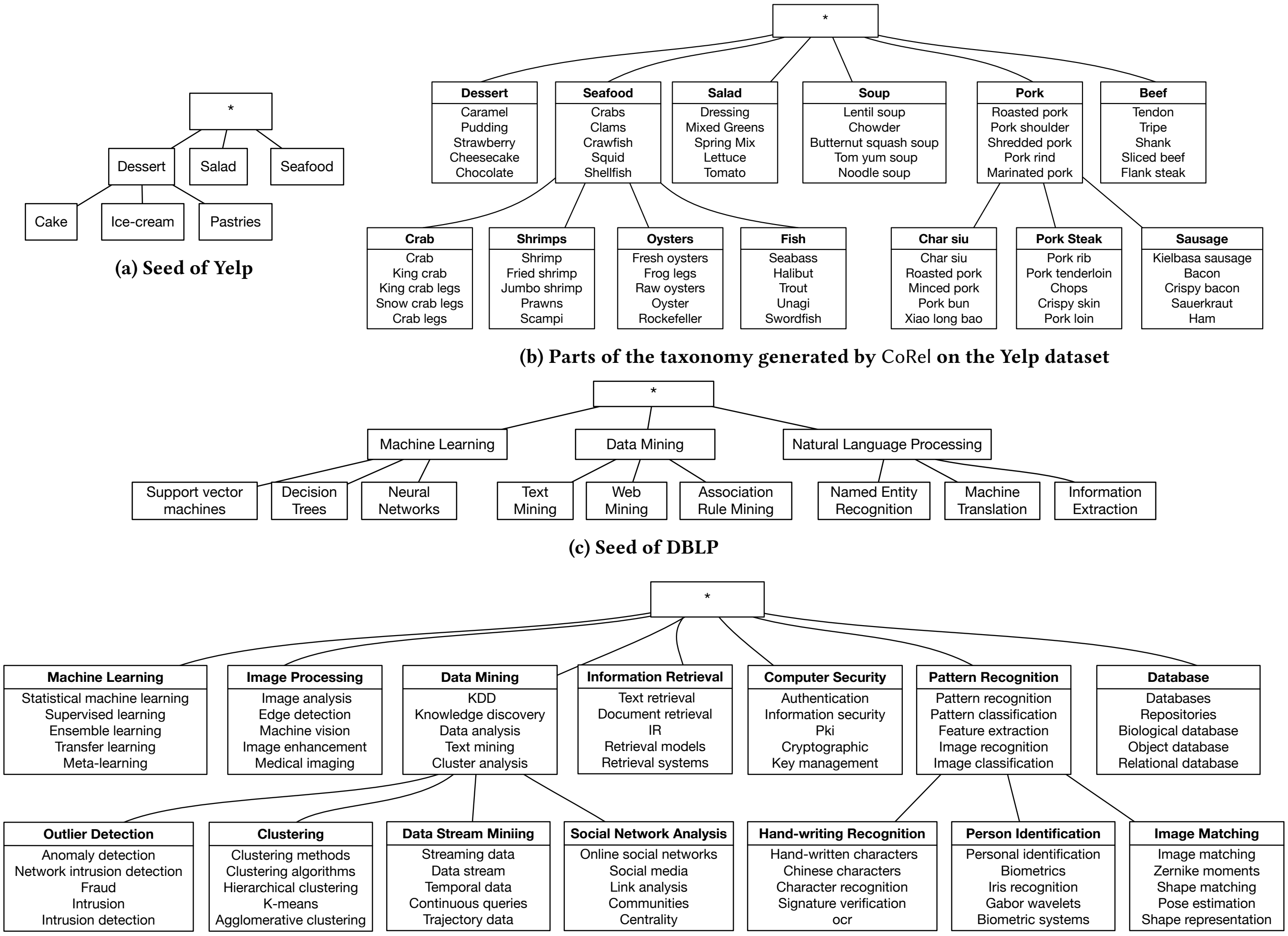
5.2 Qualitative Results

In this section we show the topical taxonomy generated by CoRel on both datasets. We further exhibit the effectiveness of our concept and relation learning modules by comparing with baseline methods.

Our Topical Taxonomy. Figure 4 shows the input seed taxonomy and parts of the topical taxonomy generated by CoRel on both datasets. For the **Yelp** dataset, we use minimal user input by only giving child nodes for one input topic to test the robustness of our method. The output in Figure 4b shows that we can find new food types such as "soup", "pork" and "beef". For subtopic finding, we can also distinguish between various western and eastern cooking style of pork. The word clusters for each topic/subtopic are obtained by the concept learning module trained on the corpus and the whole taxonomy. For **DBLP**, we use the same input seed as Hi-Expan [30]. We show that CoRel successfully finds various computer science fields in Figure 4d other than user's input, such as "information retrieval" and "pattern recognition". CoRel is also capable of finding separate fields for seed and new research areas found.

Subtopic Quality. We exhibit the effectiveness of our relation learning module by comparing the subtopics we found with those of HiExpan (seed-guided tree expansion baseline). We randomly choose the common topics shared by both methods, and show all subtopics found by each of them in Table 3. The bold ones are seeds from the input taxonomy, while the wrong subtopics are marked by (×). Since generating synonyms or included concepts at the same level harms the overall quality of a taxonomy, we mark these terms as redundant (✂). For example, under the topic of "**DBLP**-Machine Learning", HiExpan generates lots of synonyms and subconcepts for "neural networks" at the same level, such as "artificial neural networks" and "multilayer perceptron". These terms should be formed in the topic of "neural networks" instead of its sibling nodes. Our design of concept learning puts these terms into the distinctive term clusters of corresponding topics, thus avoiding such pitfalls. Under the topic of "**Yelp**-Seafood", we show that simply using word analogy is not robust in capturing parallel relations in the global embedding space, and it is essential to utilize more advance text representations and operations upon them as in our relation learning module.

Concept clusters. We wish to evaluate how well our concept learning module forms meaningful clusters. We compare with TaxoGen, HLDA and HPAM that output hierarchical clusters of terms. Since they do not need user-given seeds as supervision, and generate many irrelevant topics, we set the number of topics for TaxoGen and HPAM to 10 at each layer, and use the default setting for HLDA. Then we manually pick out common topics/subtopics found by different algorithms. We list the centermost 5 terms for CoRel topics/subtopics, and the 5 terms with top probability of TaxoGen, HLDA and HPAM in Table 4. We observe that without user-given seeds, topics from TaxoGen, HLDA and HPAM include mixtures of cross-concept terms or irrelevant terms, while our concept learning module is able to find coherent and distinctive terms for each concept.



(a) Seed of Yelp

(b) Parts of the taxonomy generated by CoRel **on the Yelp dataset**

(c) Seed of DBLP

(d) Parts of the taxonomy generated by CoRel **on the DBLP dataset**

Figure 4: Input and part of output taxonomy generated by CoRel **on DBLP and Yelp.**

Table 3: Comparison of subtopics found under the same topics.

Topics	Method	All Subtopics found
DBLP-Machine Learning	Hi-Expan	**Support vector machine**, **Decision trees**, **Neural networks**, Regression models, Genetic algorithm, Naive Bayes, Classification, Random forest, Markov random field, Nearest Neighbor, Unsupervised learning, Artificial neural networks (✂), Multilayer perceptron (✂), Support vector regression (✂), Conditional random fields (✂), hidden markov models (✂), Radial basis function(✂), Self-organizing map (✂), Recurrent neural networks (✂), Extreme learning machine(✂), Particle swarm optimization (×),
	CoRel	**Support vector machine**, **Decision trees**, **Neural networks**, Regression, Genetic algorithm, Bayesian networks, Classification, Random forest, Inductive logic programming, Reinforcement learning, Active learning, Boosting algorithms, Transfer learning, object recognition (×), Text classification (×)
DBLP-Data Mining	Hi-Expan	**association rule mining**, **text mining**, **web mining**, outlier detection, anomaly detection, spectral clustering, social network analysis, density estimation,(✂) association rules (✂)
	CoRel	**association rule mining**, **text mining**, **web mining**, outlier detection, anomaly detection, clustering algorithms, social network analysis, data stream mining, data visualization, online analytical processing, rule discovery, predictive modeling, sequence analysis (✂)
Yelp-Dessert	Hi-Expan	**cake**, **ice cream**, **pastries**, latte, cream, gelato, sauce (×), cheesecake (✂), taste (×), pancake (✂)
	CoRel	**cake**, **ice cream**, **pastries**, milk tea, cream, fruit, juice, cooked (×), sugar (×)
Yelp-Seafood	Hi-Expan	fish, shrimps, salmon (✂), meat (×), chicken (×), beef (×), steak (×), pork (×), rice (×)
	CoRel	fish, shrimps, crab, scallop, oysters, mussel, camarones (✂), soy sauce (×), pho (×), low mein (×)

Table 4: Topic clusters generated by different methods.

Met.	DBLP Recommender System	DBLP Image Matching	Yelp Beef
HLDA	recommended items	illumination variation	reuben
	recommendation framework	relevant document (×)	Don Juan (×)
	sentimental analysis (×)	affine distortion	cutter (×)
	source entropy (×)	BNMTF	corned beef
	customer (×)	phase diagram (×)	turnover beef
HPAM	ranking	face recognition	BBQ
	web page (×)	image (×)	brisket
	propose (×)	video (×)	ribs
	different (×)	detection (×)	meat (×)
	recommendation	segmentation (×)	good
TG	linked data (×)	fisher criterion	wellington
	social network analysis (×)	face verification	wagyu beef
	recommendation systems	decision fusion (×)	dry aged
	user interests	classifier design (×)	walleye (×)
	user feedback	discriminative power (×)	red meat (×)
CoRel	recommender systems	image matching	tendon
	collaborative filtering	zernike moments	tripe
	recommendation	shape matching	beef ball
	user preferences	pose estimation	flank
	user rating	feature point	rare beef

5.3 Quantitative Results

In this section, we quantitatively evaluate the quality of the taxonomies constructed by different methods.

Evaluation Metrics. The evaluation of the quality of a topical taxonomy is a challenging task since there are different aspects to be considered, and it is hard to construct a gold standard taxonomy that contains all the correct child nodes under each parent node. Following [30, 40], we propose three evaluation metrics: *Term Coherency*, *Relation F1* and *Sibling Distinctiveness* in this study.

- **Term Coherency (TC)** measures the semantic coherence of words in a topic.
- **Relation F1** measures the portions of correct parent-children pairs in a taxonomy that preserve user-interested relation.
- **Sibling Distinctiveness (SD)** measures how well the topics are distinctive from their siblings.

We calculate the three metrics as follows. For *TC*, we recruited 5 Computer Science students to judge the results. Specifically, we extract the top 10 representative words under each topic, ask the evaluators to divide these words into different clusters by concepts and compute the size of the cluster with the most words, thus a mixture of terms from different concepts scores lower. Then we take the mean of the results given by all the evaluators as the *TC* of this topic, and average the *TC* of all the topics in a taxonomy.

For *Relation F1*, we show the evaluators all the parent-children pairs of topics in a taxonomy and ask them to judge independently whether each pair truly holds user-interested relation. Then we use majority votes to label the pairs and use all the true parent-children pairs from different methods to construct a gold standard taxonomy. Since each topic is represented by a cluster of words, for simplicity, we consider two clusters as the same if they share the same concept. The *Relation F1* is computed as follow:

$$\begin{aligned} P_r &= \frac{|is\_ancestor_{pred}| \bigcap |is\_ancestor_{gold}|}{|is\_ancestor_{pred}|}, \\ R_r &= \frac{|is\_ancestor_{pred}| \bigcap |is\_ancestor_{gold}|}{|is\_ancestor_{gold}|}, \\ F1_r &= \frac{2P_r * R_r}{P_r + R_r} \end{aligned} \tag{8}$$

where P_r, R_r and $F1_r$ denote the *Relation Precision*, *Relation Recall* and *Relation F1*, respectively.

Finally, we calculate *Sibling Distinctiveness (SD)* as follows: we compute the similarity between a topic cluster C_i and each of its sibling topics C_j by Jaccard index [5]. Then we calculate *SD* of C_i as 1 minus the largest similarity score among all C_j. A larger *SD* means the sibling topics sharing a common parent are truly separate from each other.

Evaluation Results. Table 5 shows the *Term Coherency (TC)*, *Relation F1*, and *Sibling Distinctiveness (SD)* of different methods. For unsupervised baselines, we only take relevant topics (topics with more than half terms belonging to *food* or *research areas*) into account. Overall, weakly-supervised methods (Hi-Expan and CoRel) outperform unsupervised methods by a large margin, which shows the constructed taxonomies are well guided by the user given seeds. We can see that CoRel achieves the best performance under all evaluation metrics, especially in terms of the *Relation F1*, showing that CoRel is able to find related terms for each concept and retain ones holding certain relations with current topics in relation transferring module. For *TC*, CoRel also significantly outperforms HLDA, HPAM and TaxoGen, which model the intrinsic distribution of terms in documents or corpus, and might generate topics as mixtures of terms relevant but not distinctive of user-interested topics. They also have inferior performance in *SD* since they do not enforce distinctiveness when forming topics. On the other hand, though HiExpan + Concept Learning only achieves slightly worse or equal results on *TC* and *SD* compared with ours, HiExpan itself only outputs an instance taxonomy and cannot generate topics for each concept node. We enhance it by our own concept learning module to extract distinctive terms for each node. This further demonstrates the effectiveness of both of our modules.

6 CONCLUSIONS AND FUTURE WORK

In this paper we explore the problem of seed-guided topical taxonomy construction. Our proposed framework CoRel completes the taxonomy structure by a relation transferring module and enriches the semantics of concept nodes by a concept learning module. The relation transferring module learns the user-interested relation preserved in seed parent-child pairs, then transfers it along multiple paths to expand the taxonomy in width and depth. The concept learning module finds discriminative topical clusters for each concept in the process of jointly embedding concepts and words. Extensive experiments show that both modules work effectively in generating a high-quality topical taxonomy based on user-given seeds.

For future work, it is interesting to study how we can generate multi-faceted taxonomy automatically, so that each concept node is described by terms from different aspects (e.g., *ingredients* and *cooking style* for foods). Though these terms can be captured by our

Table 5: Quantitative evaluation on topical taxonomies.

Methods	DBLP					Yelp				
	TC	SD	Precision$_r$	Recall$_r$	F1-score$_r$	TC	SD	Precision$_r$	Recall$_r$	F1-score$_r$
HLDA	0.582	0.981	0.188	0.577	0.283	0.517	0.991	0.135	0.387	0.200
HPAM	0.557	0.905	0.362	0.538	0.433	0.687	0.898	0.173	0.615	0.271
TaxoGen	0.720	0.979	0.450	0.429	0.439	0.563	0.965	0.267	0.381	0.314
Hi-Expan + CoL.	0.819	0.996	0.676	0.532	0.595	0.815	**1.000**	0.429	0.677	0.525
CoRel	**0.855**	**1.000**	**0.730**	**0.607**	**0.663**	**0.825**	**1.000**	**0.564**	**0.710**	**0.629**

concept learning module, how to recognize them and organize them into meaningful clusters remains challenging and worth exploring.

ACKNOWLEDGMENTS

Research was sponsored in part by US DARPA KAIROS Program No. FA8750-19-2-1004 and SocialSim Program No. W911NF-17-C-0099, National Science Foundation IIS 16-18481, IIS 17-04532, and IIS 17-41317, and DTRA HDTRA11810026. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and should not be interpreted as necessarily representing the views, either expressed or implied, of DARPA or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright annotation hereon. We thank anonymous reviewers for valuable and insightful feedback.

REFERENCES

[1] David M. Blei, Thomas L. Griffiths, Michael I. Jordan, and Joshua B. Tenenbaum. 2003. Hierarchical Topic Models and the Nested Chinese Restaurant Process. In *NIPS*.

[2] Haw-Shiuan Chang, ZiYun Wang, Luke Vilnis, and Andrew McCallum. 2017. Distributional Inclusion Vector Embedding for Unsupervised Hypernymy Detection. In *NAACL-HLT*.

[3] Philipp Cimiano, Andreas Hotho, and Steffen Staab. 2004. Comparing Conceptual, Divise and Agglomerative Clustering for Learning Taxonomies from Text. In *ECAI*.

[4] Sarthak Dash, Md. Faisal Mahbub Chowdhury, Alfio Massimiliano Gliozzo, Nandana Mihindukulasooriya, and Nicolas R. Fauceglia. 2019. Hypernym Detection Using Strict Partial Order Networks.

[5] Barry de Ville. 2001. Introduction to Data Mining.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

[7] Brendan J. Frey and Delbert Dueck. 2007. Clustering by passing messages between data points. *Science* 315 5814 (2007), 972–6.

[8] Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Learning Semantic Hierarchies via Word Embeddings. In *ACL*.

[9] Tianyu Gao, Xu Han, Zhiyuan Liu, and Maosong Sun. 2019. Hybrid Attention-Based Prototypical Networks for Noisy Few-Shot Relation Classification. In *AAAI*.

[10] Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. FewRel: A Large-Scale Supervised Few-shot Relation Classification Dataset with State-of-the-Art Evaluation. In *EMNLP*.

[11] Marti A. Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. In *COLING*.

[12] Jiaxin Huang, Yiqing Xie, Yu Meng, Jiaming Shen, Yunyi Zhang, and Jiawei Han. 2020. Guiding Corpus-based Set Expansion by Auxiliary Sets Generation and Co-Expansion. *Proceedings of The Web Conference 2020* (2020).

[13] Yuval Kluger, Ronen Basri, Joseph T. Chang, and Mark B Gerstein. 2003. Spectral biclustering of microarray data: coclustering genes and conditions. *Genome research* 13 4 (2003), 703–16.

[14] Zornitsa Kozareva and Eduard H. Hovy. 2010. A Semi-Supervised Method to Learn and Construct Taxonomies Using the Web. In *EMNLP*.

[15] Matt Le, Stephen Roller, Laetitia Papaxanthos, Douwe Kiela, and Maximilian Nickel. 2019. Inferring Concept Hierarchies from Text Corpora via Hyperbolic Embeddings. In *ACL*.

[16] Xueqing Liu, Yangqiu Song, Shixia Liu, and Haixun Wang. 2012. Automatic taxonomy construction from keywords. In *KDD*.

[17] Yu Meng, Jiaxin Huang, Guangyuan Wang, Zihan Wang, Chao Zhang, Yu Zhang, and Jiawei Han. 2020. Discriminative Topic Mining via Category-Name Guided Text Embedding. In *WWW*.

[18] Yu Meng, Jiaxin Huang, Guangyuan Wang, Chao Zhang, Honglei Zhuang, Lance M. Kaplan, and Jiawei Han. 2019. Spherical Text Embedding. In *NeurIPS*.

[19] 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 *KDD*.

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS*.

[21] David M. Mimno, Wei Li, and Andrew McCallum. 2007. Mixtures of hierarchical topics with Pachinko allocation. In *ICML '07*.

[22] Ndapandula Nakashole, Gerhard Weikum, and Fabian M. Suchanek. 2012. PATTY: A Taxonomy of Relational Patterns with Semantic Types. In *EMNLP-CoNLL*.

[23] Roberto Navigli and Paola Velardi. 2010. Learning Word-Class Lattices for Definition and Hypernym Extraction. In *ACL*.

[24] Roberto Navigli, Paola Velardi, and Stefano Faralli. 2011. A Graph-Based Algorithm for Inducing Lexical Taxonomies from Scratch. In *IJCAI*.

[25] Maximilian Nickel and Douwe Kiela. 2017. Poincaré Embeddings for Learning Hierarchical Representations. In *NIPS*.

[26] Meng Qu, Xiang Ren, Yu Lin Zhang, and Jiawei Han. 2017. Weakly-supervised Relation Extraction by Pattern-enhanced Embedding Learning. In *WWW*.

[27] Xin Rong, Zhe Chen, Qiaozhu Mei, and Eytan Adar. 2016. EgoSet: Exploiting Word Ego-networks and User-generated Ontology for Multifaceted Set Expansion. In *WSDM*.

[28] Jingbo Shang, Jialu Liu, Meng Jiang, Xiang Ren, Clare R. Voss, and Jiawei Han. 2017. Automated Phrase Mining from Massive Text Corpora. *IEEE Transactions on Knowledge and Data Engineering* 30 (2017), 1825–1837.

[29] Jiaming Shen, Zeqiu Wu, Dongming Lei, Jingbo Shang, Xiang Ren, and Jiawei Han. 2017. SetExpan: Corpus-Based Set Expansion via Context Feature Selection and Rank Ensemble. In *ECML/PKDD*.

[30] Jiaming Shen, Zeqiu Wu, Dongming Lei, Chao Zhang, Xiang Ren, Michelle T. Vanni, Brian M. Sadler, and Jiawei Han. 2018. HiExpan: Task-Guided Taxonomy Construction by Hierarchical Tree Expansion. *ArXiv* abs/1910.08194 (2018).

[31] Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving Hypernymy Detection with an Integrated Path-based and Distributional Method. *ArXiv* abs/1603.06076 (2016).

[32] Rion Snow, Dan Jurafsky, and Andrew Y. Ng. 2004. Learning Syntactic Patterns for Automatic Hypernym Discovery. In *NIPS*.

[33] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the Blanks: Distributional Similarity for Relation Learning. In *ACL*.

[34] Paola Velardi, Stefano Faralli, and Roberto Navigli. 2013. OntoLearn Reloaded: A Graph-Based Algorithm for Taxonomy Induction. *Computational Linguistics* 39 (2013), 665–707.

[35] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2015. Order-Embeddings of Images and Language. *CoRR* abs/1511.06361 (2015).

[36] Chi Wang, Marina Danilevsky, Nihit Desai, Yinan Zhang, Phuong Nguyen, Thrivikrama Taula, and Jiawei Han. 2013. A phrase mining framework for recursive construction of a topical hierarchy. In *KDD '13*.

[37] Julie Weeds, David J. Weir, and Diana McCarthy. 2004. Characterising Measures of Lexical Distributional Similarity. In *COLING*.

[38] Grace Hui Yang and James P. Callan. 2009. A Metric-based Framework for Automatic Taxonomy Induction. In *ACL/IJCNLP*.

[39] Shuo Yang, Lei Zou, Zhongyuan Wang, Jun Yan, and Ji-Rong Wen. 2017. Efficiently Answering Technical Questions - A Knowledge Graph Approach. In *AAAI*.

[40] Chao Zhang, Fangbo Tao, Xiusi Chen, Jiaming Shen, Meng Jiang, Brian M. Sadler, Michelle Vanni, and Jiawei Han. 2018. TaxoGen: Unsupervised Topic Taxonomy Construction by Adaptive Term Embedding and Clustering. In *KDD '18*.

[41] Yuchen Zhang, Amr Ahmed, Vanja Josifovski, and Alexander J. Smola. 2014. Taxonomy discovery for personalized recommendation. In *WSDM '14*.

[42] Maayan Zhitomirsky-Geffet and Ido Dagan. 2005. The Distributional Inclusion Hypotheses and Lexical Entailment. In *ACL*.