NetTaxo: Automated Topic Taxonomy Construction from Text-Rich Network

Jingbo Shang*
University of California San Diego
jshang@ucsd.edu

Xinyang Zhang*
University of Illinois at Urbana-Champaign
xz43@illinois.edu

Liyuan Liu
University of Illinois at Urbana-Champaign
ll2@illinois.edu

Sha Li
University of Illinois at Urbana-Champaign
shal2@illinois.edu

Jiawei Han*
University of Illinois at Urbana-Champaign
hanj@illinois.edu

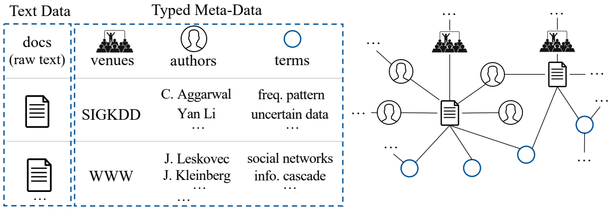
ABSTRACT

The automated construction of topic taxonomies can benefit numerous applications, including web search, recommendation, and knowledge discovery. One of the major advantages of automatic taxonomy construction is the ability to capture corpus-specific information and adapt to different scenarios. To better reflect the characteristics of a corpus, we take the meta-data of documents into consideration and view the corpus as a text-rich network. In this paper, we propose NetTaxo, a novel automatic topic taxonomy construction framework, which goes beyond the existing paradigm and allows text data to collaborate with network structure. Specifically, we learn term embeddings from both text and network as contexts. Network motifs are adopted to capture appropriate network contexts. We conduct an instance-level selection for motifs, which further refines term embedding according to the granularity and semantics of each taxonomy node. Clustering is then applied to obtain sub-topics under a taxonomy node. Extensive experiments on two real-world datasets demonstrate the superiority of our method over the state-of-the-art, and further verify the effectiveness and importance of instance-level motif selection.

ACM Reference Format:
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 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3366423.3380259

1 INTRODUCTION

Constructing high-quality topic taxonomies for document collections is an important task. A topic taxonomy is a tree-structured hierarchy, where each taxonomy node contains a set of semantically similar terms. A high-quality topic taxonomy benefits various downstream applications, such as search and indexing [43], personalized content recommendation [46], and question answering [42]. For example, organizing copious scientific papers into a well-structured taxonomy gives researchers a bird's-eye view of the field, and then they can quickly identify their interests, and easily acquire desired information [35]. A high-quality taxonomy for business reviews on Yelp[1] can facilitate more accurate recommendations and improve user's browsing experience.



(a) An example digital collection of massive scientific papers.

(b) An text-rich network view of the example digital collection.

Figure 1: Document collections with meta-data can be viewed as text-rich networks.

Different applications usually require different taxonomies, therefore, automatic taxonomy construction capturing corpus-specific information becomes beneficial. The last decade has witnessed an explosive growth of digital document collections. By linking documents with their meta-data, we can view any document collection as a text-rich network. As illustrated in Figure 1, a collection of scientific papers can be viewed as a text-rich network with interconnected venue, author, term and paper nodes, and raw texts are associated with the paper nodes. Similarly, reviews from online platforms like Yelp and TripAdvisor [2] can be seen as a part of a text-rich network with nodes of businesses, users, and reviews.

While most existing methods solely rely on text data [2, 11, 16, 44], incorporating network structures can bring additional, valuable information to text. Let's use the computer science paper collection to convey our intuition. The term "*frequent pattern*" appears along with "*transaction database*" frequently. Judging only from text data, one may put this term into the *database* community. However, information embedded in the network structure, such as its associated venues (e.g., "*SIGKDD*") and authors (e.g., "*Charu C. Aggarwal*"), indicates the strong relatedness between the term "*frequent pattern*" and the *data mining* community, enabling us to assign it to the right taxonomy node.

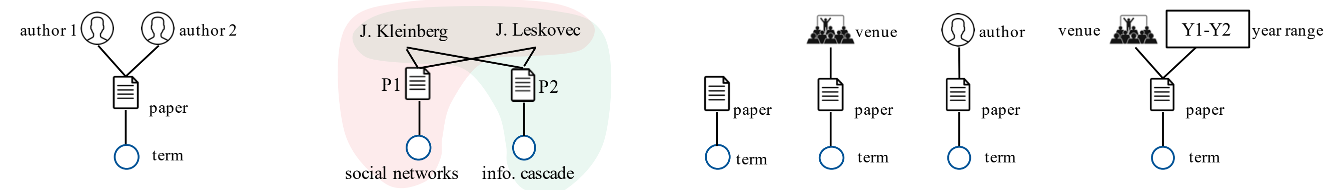
*These authors contributed equally to this work.

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

[2] https://www.tripadvisor.com/

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380259



(a) An example motif pattern. (b) Two terms connected by a motif instance. (c) Some other motif patterns. Meta-paths are special cases of motif patterns.

Figure 2: Example Motif Patterns and Motif Instances. (a) This motif pattern suggests that two terms are similar when they are from the papers published by the same author pairs. (b) The two terms are connected by a motif instance of the motif patterns in (a), which has two authors instantiated. The shades indicate two full instantiations of the motif pattern. (c) The other motif patterns that we used in the DBLP-5 dataset, including meta-path shaped patterns.

Acknowledging that network provides useful information for taxonomy construction, how to effectively integrate network and text remains a major challenge. We leverage *motif patterns* in our framework to extract useful features from the heterogeneous text-rich network. Meta-paths [31] and motif patterns [5, 18] have been widely adopted to extract useful structural information from networks. As illustrated in Figure 2, motifs are subgraph patterns that capture higher-order connectivity and the semantics represented by these connections. We observe two issues of applying motif patterns in our problem. First, motif patterns are not created equal. Some motif patterns are more useful in identifying top-level concepts, while other motif patterns are better at differentiating finer concepts. Second, even only looking at one motif pattern, its motif instances are by no means equally informative. Some of them could even interfere the taxonomy construction, leading to a worse result. For example, using the motif pattern in Figure 2(a) which captures co-authorship, some of its instances may be occasional and coincidental collaborations, thus will not help much when constructing the scientific taxonomy. To address these two issues, we propose a novel instance-level motif selection mechanism, which is specifically tailored to current node's granularity and semantics. We show in our experiments that such selection mechanism is crucial especially when the network is relatively noisy.

We propose NetTaxo, a hierarchical embedding and clustering framework for automatic topic taxonomy construction. The general workflow is sketched in Figure 3. To begin with, we ask the user to provide a set of motif patterns as guidance. This set is never assumed to be clean and equally effective. At each taxonomy node, we propose to learn term embedding from both text and network data, and then apply a soft clustering method to obtain term clusters. We first obtain initial term clusters based on term embedding learned on text data. An inter-cluster comparative analysis is then conducted to select the most representative terms as anchor terms from each cluster. We make an assumption that a helpful motif instance should have the ability to separate one cluster's anchor terms from others. Building upon this assumption, we further distill the motif instances to include those that are relevant to the clustering, thus avoiding to introduce noise from network data. After that, we combine textual context and selected motif instances to learn term embedding jointly. Final clusters are then decided based on such joint embedding.

Experimental results demonstrate the success of our instance-level motif selection. For example, we show that, for a collection of computer science papers, at the top level of the taxonomy construction, our method locates the venue of publication (e.g., "*SIGKDD*") as a strong indicator of research fields (e.g., "*data mining*"). Drilling down to lower levels of the taxonomy, our objective becomes to distinguish research sub-areas. Our proposed method identified specific author groups as more useful signals, such as "*Cheng-Wei Wu*" and "*Philip S. Yu*" — All their collaborations focus on the topic of *high-utility itemset discovery.*

To our best knowledge, this is the first work that bridges text and network data for automatic construction of topic taxonomy. Our contributions can be summarized as follows.

- We propose a novel topic taxonomy construction framework, NetTaxo, which integrates text data and network structures effectively and systematically.
- We design an instance-level motif selection method to choose the appropriate information from network data. Moreover, it's adaptive to the granularity and semantics of each taxonomy node.
- We conduct extensive experiments on real-world datasets to demonstrate the superiority of NetTaxo over many baselines and verify the importance and effectiveness of the instance-level motif selection.

Reproducibility: Data and code packages can be found on the GitHub: https://github.com/xinyangz/NetTaxo.

2 RELATED WORK

Hyponymy-based Methods. Taxonomies have been designed to group entities into hierarchies where each node is a concept term and each parent-child pair expresses a hyponymy (a.k.a. "is-a") relation (e.g., panda "is-a" mammal). In order to construct such taxonomies automatically, researchers have developed a number of pattern-based methods. Typically, these methods first acquire hyponymy relations from text data using lexical patterns (e.g., "A such as B"), and then organize the extracted pairs into a taxonomy by applying algorithms like maximum spanning tree. The lexical patterns are either manually designed [14, 21, 23, 25] or derived from the corpus using some supervision or seeds [1, 6, 15, 20, 28, 47]. Such patterns have demonstrated their effectiveness at finding hyponymy relations, however, they are not suitable for constructing a topic taxonomy as (1) each node in a topic taxonomy is a cluster of terms instead of a single concept term, and (2) pattern-based methods often suffer from low recall due to the large variation of expressions in natural language on hyponymy relations.

Recently, term embedding has been widely adopted in automatic topic taxonomy construction. A common practice is to first learn

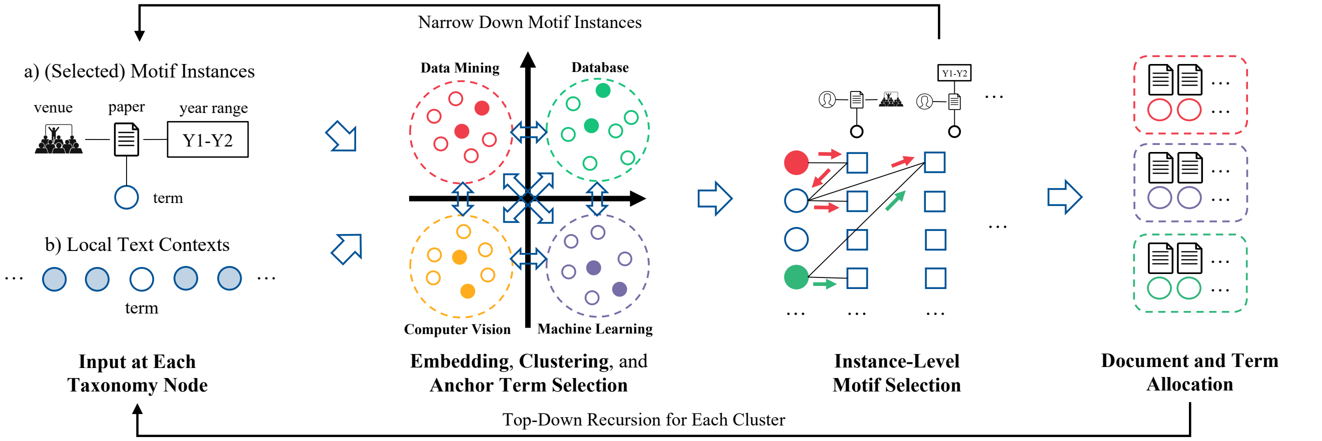

Narrow Down Motif Instances
a) (Selected) Motif Instances
venue
paper
year range
Y1-Y2
term
b) Local Text Contexts
term
Data Mining
Database
Computer Vision
Machine Learning
Y1-Y2
Input at Each Taxonomy Node
Embedding, Clustering, and Anchor Term Selection
Instance-Level Motif Selection
Document and Term Allocation
Top-Down Recursion for Each Cluster

Figure 3: An overview of NetTaxo**, which learns term embedding jointly from textual and motif contexts. We conduct a motif instance-level selection to pick the most informative network structures for better topic taxonomy construction.**

term embedding from text data and then organize them into a structure based on their representation similarity [4] and cluster separation measures [7]. Utilizing pairwise hyponymy relation labels, taxonomic relations between terms and clusters can be identified through supervised models, for example, semantic projection in the embedding space [11] and neural network classifier [2]. In our setting, there are no hyponymy labels.

Term Clustering-based Methods. A number of clustering methods have been proposed towards automatic topic taxonomy construction from text corpora. In pioneer studies, hierarchical topic modeling [10, 12, 19, 37, 38] and bottom-up agglomerative clustering-based [8] methods are arguably the most popular and effective frameworks, before word embedding techniques become mature.

Among unsupervised frameworks using term embedding, top-down hierarchical clustering methods [16, 44] achieve the state-of-the-art. For example, TaxoGen [44] learns local term embedding from the documents associated with a taxonomy node, and then clusters terms at a deeper level. Most of these methods, including TaxoGen, only utilize the information embedded in text data, but ignore the underlying network structures in the digital document collections. In our NetTaxo framework, we follow the top-down, local embedding approach but go beyond and leverage network structures to significantly improve the quality of clustering.

Network Clustering-based Methods. CATHYHIN [38] is arguably the state-of-the-art method solely based on network structures for automatic topic taxonomy construction. Specifically, with unigram words as a part of its node set, it attempts to mine terms (i.e., phrases) and clusters simultaneously. It ignores the context of the words, thus sacrificing the abundant information embedded in the text data, yielding unsatisfactory results in our experiments.

Another related thread is the clustering algorithms on heterogeneous information networks (i.e., networks of typed nodes and edges) [32, 33]. For example, NetClus [33] starts with user-provided seed nodes and applies authority ranking together with node clustering to cluster nodes. We adopt a similar authority ranking process as a part of our instance-level motif seleciton.

Network Motifs. Network motifs are higher-order subgraph structures that are critical in complex networks across various domains, such as neuroscience [30], bioinformatics [18], and information networks [5]. In the context of heterogeneous information networks, network motifs, sometimes also referred to as meta-graphs, can offer more flexibility and capture richer network semantics than the widely used meta-path [31] patterns. Recent studies have shown that incorporating motifs for node embedding leads to superior performance [24, 41, 45] compared to conventional path-based methods [9, 27]. In this work, the quality of term embedding is the key to the overall quality of the constructed taxonomy. While taking advantage of network motifs in our embedding learning, we further select a subset of motif instances according to the current taxonomy node. This novel approach enables us to refine the rich semantics captured by network motifs, generating embedding better suited for taxonomy construction.

3 PRELIMINARY

In this section, we first introduce the preliminary concepts and then formulate the problem by specifying the input and output.

3.1 Concept Definitions

A **topic taxonomy** is a tree-structured hierarchy $\mathcal{H}$, where each node $c \in \mathcal{H}$ contains a small set of terms $\mathcal{T}_c \subset \mathcal{T}$, which are semantically coherent and represent a conceptual topic. Moreover, the parent-child nodes in $\mathcal{H}$ should follow the topic-subtopic relation. That is, suppose a node c has a set of children $\mathcal{S}_c = c_1, c_2, \ldots, c_n$, then each $c_i (1 \leq i \leq n)$ should be a sub-topic of c and of the same granularity as its siblings in $\mathcal{S}_c$.

Note that, one term may belong to multiple conceptual topics and thus appear in multiple nodes. For example, "deep learning" could be a part of both "deep learning theory in machine learning" and "deep learning models in computer vision"; "data stream" could belong to "stream data indexing in database" and "stream data classification in data mining".

As mentioned before, a document collection with meta-data can be naturally viewed as a **text-rich network**, consisting of text data and network structure:

- **Text Data**: A corpus $\mathcal{D}$ and a set of terms $\mathcal{T}$. $\mathcal{T}$ includes terms in $\mathcal{D}$, which can be either specified by users or extracted from the corpus. In our experiments, we form the term set $\mathcal{T}$ by extracting high-quality phrases from the corpus $\mathcal{D}$ using AutoPhrase [26].

- **Network Structure**: A heterogeneous information network $G = (V, E, \phi, \psi)$, where V is the node set and E is the edge set. Type mapping ϕ and ψ map each node v to its type $\phi(v)$ and each edge e to a relation $\psi(e)$.

A **motif pattern** Ω refers to a subgraph pattern at the meta level (i.e., every node is abstracted by its type). In this paper, we study only the motif patterns having at least one node of *term* type. A **motif instance** m is an instantiation of a motif pattern by replacing the node types with concrete values. Figure 2 presents some examples. We define "open" nodes as those single-degree nodes except for the term node, playing a role of connecting two terms. We say that two terms are connected following a motif pattern, if and only if both terms appear in motif instances sharing the same values at those "open" nodes. Therefore, we represent motif instances only by the values of "open" nodes. As an example, in Figure 2(b), the motif instances linking to the terms "*social network*" and "*information cascade*" are the same. Both motif instances can be represented by the combination of two authors (i.e., "*Jure Leskovec*" and "*Jon Kleinberg*").

It is worth noting that meta-path [31] can be viewed as a special case of motif patterns when they degenerate to lines. For example, the meta-path describing the shared venue relation between two terms is equivalent to the 2nd motif pattern in Figure 2(c). The only "open" node in this motif pattern is the venue node.

3.2 Problem Formulation

In this paper, we aim to construct a topic taxonomy with a text-rich network as input. In addition, we ask user to provide a set of motif patterns as the guidance to incorporate information from network. However, the user-provided set can be noisy and we will conduct a motif instance-level selection later. Our goal is to construct a tree-structured taxonomy hierarchy $\mathcal{H}$, i.e., a topic taxonomy.

4 OUR FRAMEWORK

In this section, we describe our proposed NetTaxo framework.

4.1 Overview

NetTaxo is a top-down, recursive framework. Our main goal is to allocate terms into sub-topics at each taxonomy node. The allocation module relies on term embedding that is jointly learned from textual and motif contexts. We use *local embedding* and *motif instance selection* to refine the textual and motif contexts respectively.

To support our local embedding and motif instance selection module, we associate every taxonomy node with a set of weighted documents. Specifically, we maintain a weight $w_{c,d} \in [0, 1]$ for each document d at the taxonomy node c. The weights are initialized to 1 for all documents in the root node. Alongside with term allocation, we also allocate documents from a taxonomy node to its children nodes. During the allocation process, we update $w_{c,d}$ for documents in the children nodes $c_1, c_2, \ldots, c_n$.

Figure 3 gives an overview of NetTaxo. At each taxonomy node, the system needs to determine the sub-topics, and then distribute terms and documents into its children accordingly. The key contribution of NetTaxo is our designed effective way of leveraging both text data and network structures.

Based on our observations and previous work [44], using term embedding learned from textual contexts alone can cluster subtopics roughly, although not necessarily perfect. Therefore, we decide to leverage such clustering results as the initialization to our subsequent motif instance selection step. Specifically, we first follow previous work [44] to learn local term embedding and obtain initial term clusters. To be more accurate, we conduct a comparative analysis between clusters to select the most representative terms from each cluster to serve as anchor terms. Such anchor terms can be viewed as consolidated clustering information. Based on the anchor terms, we choose the appropriate motif instances. After that, we learn the term embedding jointly from the text data and the selected motif instances, which will in turn yield better clustering results. Before recursing to the next level, anchor terms chosen from the new clustering results are set as the final term set for this taxonomy node.

The details are presented in the remaining of this section. Sections 4.2 and 4.3 present how to learn term embedding from textual and motif contexts, respectively. Section 4.4 introduces anchor term selection method which is used multiple times across our framework. Section 4.6 discusses the joint term embedding after we introduce our motif selection technique in Section 4.5. Finally, Section 4.7 shows how to allocate terms and documents into child taxonomy nodes.

4.2 Local Embedding from Text Data

In NetTaxo framework, term embedding is the key to discover subtopic clusters at every taxonomy node.

Term embedding learning is typically conducted on the entire document collection [17, 22]. However, such learning paradigm faces a major drawback in topic taxonomy construction: the discriminative power of learned term embedding becomes limited at deep levels. For example, term embedding learned from all computer science papers shall be able to distinguish "*machine learning*"-related terms from terms in other research field. However, it may have difficulties in further discovering sub-topics under "*machine learning*", as those "*machine learning*"-related terms are already quite close to each other. This problem will only get worse as we drill down further. Therefore, it is a necessity to condition the term embeddings to the current taxonomy node.

To this end, we follow previous work [44] and adopt the idea of *local embedding* [13] to learn term embedding from text data. The basic idea of local embedding is to fine-tune term embedding at each node according to its own associated (weighted) documents. Its effectiveness has been verified in [44] through ablation tests.

We use skip-gram with negative sampling (SGNS) [17] as our base embedding model. At each taxonomy node, we use local documents $\mathcal{D}_c$ instead of $\mathcal{D}$ for training. Similar to the original SGNS model, the objective is to maximize the probability of the local context given a term in a document. The loss function to minimize is given by:

$$\mathcal{L}_{\text{text}} = \mathbb{E}_{d \sim P_D(\mathcal{D}_c)} \left[\sum_{t \in d} \sum_{t' \in C(t)} -P(t' \mid t) \right] \tag{1}$$

$C(t)$ stands for the set of terms within a context window of term t. We sample documents according to the multinomial distribution

$P_D(\mathcal{D}_c)$ parameterized by the document weights $\{w_{c,d}\}$ under the current taxonomy node. Therefore, our loss function slightly differs from the ones in the previous work [44] as well as the original local embedding work [13].

4.3 Motif Instances as Term Contexts

We generalize the distributional hypothesis, which is fundamental in word embedding, to network by using motif instances. In text data, every word within sliding windows of a term is regarded as a part of its contexts. Similarly, a term's *motif context* is characterized by the set of motif instances, which can be matched based on the network structures around the term and the provided motif patterns. The network version of distributional hypothesis therefore becomes: terms with similar motif contexts are similar.

Now we can generalize the SGNS embedding model to incorporate motif context. Specifically, we use each term to predict its motif context, generating the following loss term.

$$\mathcal{L}_{\text{motif}} = \mathbb{E}_{d\sim P_D(\mathcal{D}_c)}\left[\sum_{t\in d}\mathbb{E}_{m\sim\tilde{\mathcal{M}}_c(t)} - \log P(m \mid t)\right] \tag{2}$$

where $\tilde{\mathcal{M}}_c(t)$ is the associated motif instances of term t. We will describe how to select $\tilde{\mathcal{M}}_c$ in section 4.5.

The probabilities are approximated with negative sampling [17].

$$\log P(m \mid t) = \log\sigma(\boldsymbol{r}_m^T\boldsymbol{u}_t) - \mathbb{E}_{m\sim P_{\text{neg}}(m)}\left[\log\sigma(-\boldsymbol{r}_j^T\boldsymbol{u}_t)\right] \tag{3}$$

where $\boldsymbol{r}$ and $\boldsymbol{u}$ are embedding vectors of motif instances and terms and $P_{\text{neg}}(m)$ is the negative sampling distribution.

In this way, term embedding can be also derived from network structures given the user-provided motif patterns.

4.4 Anchor Term Selection

In order to provide more accurate initialization for the latter instance-level motif selection module, we first introduce our anchor term selection method.

The goal of the anchor term selection is to find a concise, discriminative subset of terms from each cluster. It is a critical step for us to obtain clean semantics of a cluster, given that our vocabulary is large and noisy. For this very reason, we use anchor terms (1) as the initialization for our instance-level motif selection module, in which they provide more accurate initial clustering information; (2) as input to the clustering algorithm, in order to find sub-topics under the current taxonomy node; and (3) as the final list of terms presented at each taxonomy node.

We formulate the anchor term selection as an unsupervised term ranking problem.

Ranking Principles. Given a specific taxonomy node, we define the anchor terms from the following criteria.

- **Popularity**: An anchor term should be popular enough at the given node. Very low frequency terms within a node do not contribute substantially to its semantics, thus are not considered representative.
- **Discriminativeness**: An anchor term should be able to distinguish a node from its parent node and its sibling nodes. Discriminativeness is particularly critical in taxonomy scenarios, so analysts won't be confused by two similar taxonomy nodes during the navigation to find subsets of interest. Non-discriminative terms will appear in documents associated with many nodes and offer redundant and confusing information. For example, "*extensive experiments*" might be popular at both nodes about "*data mining*" and "*database*", thus being non-discriminative.
- **Informativeness**: An anchor term should not be a stopword-like term. As the taxonomy construction goes deeper and deeper, some terms become less and less informative. For example, "*data mining*" is an informative term at the node representing the "*computer science*" field, but has much less information at the node focusing on "*frequent pattern mining*".

Bearing these principles in minds, we design the following scoring functions accordingly.

Popularity Score. We denote the number of occurrences of the term t in the document d as $\text{tf}(t,d)$. As the documents are weighted, term frequency is weighted by the importance of the document. Given the document weights $w_{c,d}$, we define the popularity of the term t at the node c as

$$\text{pop}(c,t) = \frac{\sum_{d\in\mathcal{D}} w_{c,d}\cdot \text{tf}(t,d)}{\sum_{d\in\mathcal{D}} w_{c,d}\cdot |d|} \tag{4}$$

where $|d|$ represents the total number of terms in the document d. This formula captures the relative weighted term frequency of the term t at the node c.

Discriminativeness Score. A discriminative term t at the taxonomy node c should have a significantly larger relative weighted term frequency at the node c than that at its parent node p_c or other sibling nodes $c'_1, c'_2, \ldots, c'_m$. Therefore, we define the following ratio to capture this intuition.

$$\text{discriminative}(c,t) = \frac{\text{pop}_{c,t}}{\max\{\text{pop}_{p_c,t}, \max_{i=1}^{m}\text{pop}_{c'_i,t}\}} \tag{5}$$

The larger $\text{discriminative}(c,t)$ should imply a better anchor term candidate. When $\text{discriminative}(c,t)$ is smaller than 1, it is unlikely that t is a good choice of an anchor term at taxonomy node c.

Informativeness Score. Inverse document frequency (IDF) has been widely adopted in information retrieval to measure the informativeness of a term within a given corpus [29]. At each taxonomy node c, we calculate the weighted inverse document frequency as follows.

$$\text{idf}(c,t) = \log\frac{\sum_{d\in\mathcal{D}} w_{c,d}}{\sum_{d\in\mathcal{D}}\mathbb{I}(t\in d)\cdot w_{c,d}} \tag{6}$$

where $\mathbb{I}(t\in d)$ is a boolean indicator function about whether the term t appears in the document d.

Combined Anchor Score. As an unsupervised ranking problem, we follow the previous comparative analysis work [36] and use a geometric mean to combine these three signals.

$$\text{anchor\_score}(c,t) = \Big(\text{pop}(c,t)\cdot\text{discriminative}(c,t)\cdot\text{idf}(c,t)\Big)^{1/3} \tag{7}$$

In summary, at each taxonomy node c, we will rank the terms based on the anchor scores and pick the top K_t terms as anchor terms. We expect these anchor terms can express clear semantics of the topic at each node.

4.5 Instance-Level Motif Selection

So far we have already shown how to learn term embedding separately from text and motif using local corpus. Trivially putting them together, however, gives sub-optimal performance based on our observation. As discussed before, motif instances should be weighed accordingly at each taxonomy node during the construction process. Specifically, based on anchor terms selected from initial clusters, we further narrow down a set of useful motif instances. This instance-level motif selection step is designed to make the collaboration between text and network more effective.

We identify two principles for instance-level motif selection:

- **Importance**: The motif instance should be associated with a set of important terms, providing useful information for term embedding learning.
- **Concentration**: The motif instance should be concentrated on one or a small number of sub-topics under the current taxonomy node, thereby including it will help us better separate sub-topics.

We realize these two principles by applying *authority ranking* [32] upon the *motif context graph*.

The motif context graph at a taxonomy node c is a bipartite graph $G_c^M = (\mathcal{T}_c, \mathcal{M}_c, \boldsymbol{W})$, where $\mathcal{T}_c$ is the terms under the current taxonomy node and $\mathcal{M}_c$ is the set of motif instances. We use the notation G_c^M to avoid the ambiguity of mixing this graph with the network structure G. Note that we exclude motif instances which do not include any term or document under the current taxonomy node. The bipartite graph connects each term to the motif instances it occurs in. The weight matrix $\boldsymbol{W} \in \mathbb{R}^{|\mathcal{T}_c| \times |\mathcal{M}_c|}$ describes the number of occurrences of term t in each motif instance m (i.e., $\boldsymbol{W}_{t,m}$).

We apply authority ranking to obtain importance scores between each motif instance and each cluster. In the ranking process, we maintain two matrices $\boldsymbol{I}_T \in \mathbb{R}^{|\mathcal{T}_c| \times n}$ and $\boldsymbol{I}_M \in \mathbb{R}^{|\mathcal{M}_c| \times n}$ to store the importance scores of terms and motif instances. Each row of the matrix denotes the importance scores of a specific term (or a motif instance) under all n clusters. As initialization, we set $\boldsymbol{I}_T^{(0)}(t, k) = 1/K_t$ for all anchor terms in all clusters and zero for all other terms. This is based on the assumption that all anchor terms are important in the first place. The authority ranking is an iterative importance propagation process. Specifically, in each iteration,

$$\boldsymbol{I}_M^{(t)} \leftarrow \tilde{\boldsymbol{W}}^T \boldsymbol{I}_T^{(t-1)}, \quad \boldsymbol{I}_T^{(t)} \leftarrow \tilde{\boldsymbol{W}} \boldsymbol{I}_M^{(t)}$$

$\tilde{\boldsymbol{W}} = \boldsymbol{D}_r^{1/2} \boldsymbol{W} \boldsymbol{D}_c^{1/2}$ is the normalized weight matrix with row degree $\boldsymbol{D}_r$ and column degree $\boldsymbol{D}_c$ matrices. The iterative process can be repeated to a max iteration number or until convergence. In practice, we found that 5 iterations are enough to achieve good results.

For each motif instance m, we take the mean of its importance score across different clusters as the overall importance.

$$\text{importance}(m) = \text{mean}\left(\boldsymbol{I}_M(m, \cdot)\right)$$

Moreover, with the importance scores on different clusters, we can measure the concentration of a motif instance m based on entropy.

$$\text{concentration}(m) = 1 - \frac{1}{\log n} \sum_{i=1}^{n} \tilde{\boldsymbol{I}}_M(m, i) \log \tilde{\boldsymbol{I}}_M(m, i)$$

We use normalized entropy here to keep its range in 0 to 1. $\tilde{\boldsymbol{I}}_M$ denotes $\boldsymbol{I}_M$ after row normalization.

Finally, we define the final score of a motif instance m as

$$\text{motif\_score}(m) = \left(\text{importance}(m) \cdot \text{concentration}(m)\right)^{1/2}$$

We rank all motif instances based on their final scores, and select a subset $\tilde{\mathcal{M}}_c$ of the instances ranked in the top K_m percent. Note that, the motif instance ranking is across all motif patterns. Therefore, we are implicitly selecting motif patterns by pruning most of instances from uninformative motif patterns.

4.6 Joint Embedding from Textual and Motif Contexts

At each taxonomy node c, given the local corpus $\mathcal{D}_c$ and locally selected motif instances $\tilde{\mathcal{M}}_c$, we refine term embedding by joint embedding training of text and motif instances. Specifically, putting text and motif together, we minimize the joint loss function:

$$\mathcal{L} = \lambda \mathcal{L}_{\text{text}} + (1 - \lambda) \mathcal{L}_{\text{motif}} \tag{8}$$

We use λ to balance text and motif losses. In our implementation, we optimize the loss function with stochastic gradient descent, and approximate the expectations in previous equations using sampling.

4.7 Term and Document Allocation

With the joint embedding trained on text and motif instances, we are ready to allocate terms and documents into children nodes. In principle, our method is flexible in the choice of clustering method. Consider that cosine similarity between term embedding has demonstrated its effectiveness in term similarity search [17], we apply vMF mixture clustering [3] in NetTaxo. It is a classical, effective soft clustering method on the unit hyper-sphere. Since the constructed topic taxonomy rarely changes, we leave the choice of k, the number of topics, to human experts.

It is worth noting that we fit the vMF distributions only on anchor terms of the current taxonomy node. The rationale is that the automatically extracted term vocabulary is often noisy, while anchor terms selected from comparative analysis are much cleaner, which makes the clustering more accurate. After fitting the vMF mixture model, each cluster is represented by a vMF distribution in the embedding space. We then use these distributions to estimate the clustering probability of each term in $\mathcal{T}_c$. Finally, we allocate terms to children clusters.

For documents in $\mathcal{D}_c$, we estimate their clustering probability by aggregating clustering probability from their connected terms. This process is the same as that in [44]. The aggregated probabilities of a document, multiplied by its current weight, will be the weights of the document on the next level.

5 EXPERIMENTS

In this section, we first introduce the experimental settings, including datasets, compared methods, and evaluation metrics. We then present quantitative evaluation results. In the end, we showcase parts of the constructed topic taxonomies as well as several interesting findings.

5.1 Datasets

We conduct our experiments on two real-world document collections: computer science papers in DBLP and business reviews in

Table 1: Dataset Statistics. Motifs patterns in DBLP-5 and Yelp-5 datasets are visualized in Figures 2 and 4, respectively.

	#doc	#term	#node	#edge	#motif
DBLP-5	79,896	26,684	182,290	1,897,226	5
Yelp-5	1,308,371	74,951	1,760,025	6,809,152	4

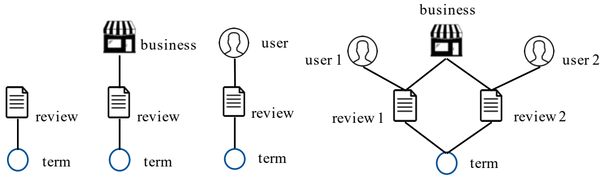


Figure 4: All motif patterns used in Yelp-5 dataset. The most complex pattern indicates a term mentioned by two users under the same business.

Yelp. The statistics about the two datasets are shown in Table 1. Details about the two datasets are as below.

- **DBLP-5.** The first document collection is from the AMiner dataset about computer science papers[3]. We select five closely-related research areas: (1) *data mining*, (2) *database*, (3) *machine learning*, (4) *computer vision*, and (5) *natural language processing*. From these five areas, 79,896 papers are selected, containing 26,684 distinct terms. The network contains node types of author, venue, year, paper, and term (as available in this DBLP dataset). We augment the network by adding "year range" nodes, each representing a five consecutive years (e.g., 2010-2014). The text data, i.e., title and abstract, is associated with each paper node. The edges describe author–paper, venue–paper, year–paper, year range–paper, and term–paper relations. Note that, previous methods [38, 44] choose five areas from this dataset too, for example in [44], *information retrieval*, *computer vision*, *robotics*, *security & network*, and *machine learning*. In contrast, our chosen five areas are more closely related to each other, thus being more challenging.
- **Yelp-5.** The second document collection is from the Yelp Dataset Challenge[4]. Since some baselines are too slow if we use the full dataset, we have to choose a subset of these reviews. Particularly, we choose the most popular state (i.e., *Arizona*) and the top-5 popular business categories (i.e., (1) *automotive*, (2) *beauty & spas*, (3) *hotels & travel*, (4) *restaurants*, and (5) *shopping*). We also remove rare businesses with less than 50 reviews. As a result, we obtain 1,308,371 reviews in total and extract 74,951 terms from them. We build the network using nodes of business, user, review, and term and edges of business–review, user–review, and term–review, as they are available in the meta-data. The text data, i.e., review comments, is associated with each review node.

We present all motif patterns used in DBLP-5 and Yelp-5 datasets in Figures 2 and 4, respectively.

5.2 Compared Methods

We compare our proposed methods with different types of topic taxonomy construction methods: (1) using text data, (2) using network data, and (3) using both text and network data. The details are listed below.

- **HPAM++** is a method enhanced by us from the original Hierarchical Pachinko Allocation Model (HPAM) [19]. HPAM is a state-of-the-art hierarchical topic model built upon the Pachinko Allocation Model *using text data*. Although it was designed to work on all unigrams, to make the comparison more fair, we improve the HPAM by focusing on only very high-quality phrases. Also, we set the topic numbers at different levels as the same as the numbers of clusters in NetTaxo. We have tested our enhanced Hierarchical Latent Dirichlet Allocation (HLDA) model [12] and its performance is quite similar to HPAM++. Therefore, we only present the results of HPAM++ here.
- **TaxoGen** [44] is the state-of-the-art topic taxonomy construction method *using text data*. As demonstrated in its paper, it beats many strong baselines, such as hierarchical topic models [10, 12, 19, 37, 38]. It utilizes the same local embedding idea as our model, but ignores network structures.
- **CATHYHIN++** is a method enhanced by us from the original CATHYHIN [38] method. CATHYHIN [38] is a topic taxonomy construction method *using network data*. It treats unigrams as nodes and attempted to mine terms (i.e., phrases) and clusters simultaneously. Its performance is limited due to (1) the poor phrase quality compared to the state-of-the-art method [26] and (2) the poor term clustering results compared to methods that use the term embedding technique. To make the comparison more fair, we improve the CATHYHIN by adding only very high-quality phrases.
- **HClusEmbed** is a baseline method that we propose *using both text and network data*. It is a straightforward solution to combine the term embedding technique with the network structure. Specifically, we first learn term embedding vectors from text using word2vec [17] and network using LINE [34] separately, where every embedding vector has a dimension of 300. And then, we concatenate the two vectors for each term, and then apply hierarchical spherical k-Means algorithm. We name this method as hierarchical topic clustering based on term and node embedding, and therefore denote it as HClusEmbed.

We denote our proposed method as **NetTaxo**. To demonstrate the necessity and effectiveness of our proposed motif instance selection, we introduce an ablated version of NetTaxo without this step, denoted as **NetTaxo w/o Selection**.

Note that, in order to conduct a fair comparison, the same set of terms are used across different methods. They are the extracted from raw texts by the state-of-the-art distantly supervised phrase mining method [26].

5.3 Parameter Setting

The number of mixtures k for vMF mixture clustering is manually selected by incrementally increasing k by 1 in the range of $[3, 6]$ until coherent clusters are observed. We set $k = 5$ for the top level and $k = 4$ for the second level of the taxonomy in both the DBLP and Yelp dataset. In TaxoGen [44], this number is set to 5 for all levels, which is not far from our observation. Note that this parameter will only need to set once for a given dataset, so this process will not put a large burden on humans. For anchor term selection, we use $K_t = 50$ for each cluster. For motif selection, we keep top $K_m = 10\%$ of motif instances.

[3] https://aminer.org/citation
[4] https://www.yelp.com/dataset/challenge

Table 2: Quantitative Evaluations. Scores are averaged over 10 annotators.

	DBLP-5					Yelp-5				
	Coherence	Sibling	Parent-Child Relations			Coherence	Sibling	Parent-Child Relations		
	Measure	Exclusiveness	Precision	Recall	F_1	Measure	Exclusiveness	Precision	Recall	F_1
HPAM++	0.796	0.680	0.348	0.451	0.393	0.832	0.740	0.171	0.247	0.202
TaxoGen	0.840	0.740	0.780	0.713	0.745	0.920	0.800	0.650	0.618	0.633
CATHYHIN++	0.880	0.533	0.850	0.744	0.793	0.742	0.420	0.705	0.638	0.670
HClusEmbed	0.624	0.420	0.525	0.409	0.460	0.744	0.560	0.655	0.610	0.632
NetTaxo w/o Selection	0.908	0.680	0.895	0.808	0.849	0.816	0.540	0.668	0.681	0.674
NetTaxo	**0.912**	**0.880**	**0.898**	**0.810**	**0.852**	**0.928**	**0.854**	**0.790**	**0.825**	**0.807**

5.4 Evaluation Tasks & Metrics

Systematic evaluation of the constructed topic taxonomy has long been a very challenging task. Inspired by the state-of-the-art work on topic taxonomy construction [38, 44] and recent work on topic modeling [39, 40], we design a set of tasks for human evaluation. For each dataset, we recruited 10 in-domain human experts. In their annotation process, they were encouraged to use search engines (e.g., Google) to better understand unfamiliar terms.

We identify the following aspects for judging the taxonomy quality, and then design three evaluation tasks accordingly.

- **Coherence.** Within each node in the taxonomy, the terms should be able to form a semantically coherent topic. Similar to previous topic model evaluations [39, 40], we present the top-5 terms to human annotators from the same taxonomy node. Annotators are asked to first judge whether these terms form an interpretable topic. If not, all five terms at this node are automatically labeled as irrelevant. Otherwise, annotators are then asked to identify specific terms that are relevant to this topic. We define the **coherence measure** as the ratio of the number of relevant terms over the total number of presented terms.
- **Exclusive Siblings.** Besides the coherence, each taxonomy node should be distinguishable from its sibling nodes. Following previous taxonomy construction methods [38, 44], we perform the term intrusion test. Specifically, for each node, we collect its top-5 terms, and then randomly mix in an intruder term from the top-5 terms of its sibling nodes. We present the 6 terms in a random order and ask human annotators to identify the only intruder term. The more coherent and distinctive the topics are, the easier it is for human to spot intruder terms. We define the **sibling exclusiveness** as the successful identification ratio in this test.
- **Quality Parent-Child Relations.** Each taxonomy node should be an appropriate sub-topic of its parent node. Considering the huge vocabulary size, it is difficult to enumerate all children terms of a given topic, and further evaluate the relation quality. We instead use a sampling-based method for evaluation. Specifically, between two adjacent levels in a taxonomy, we first sample a child term t from lower-level nodes, and present t together with all upper-level (i.e., parent-level) nodes. Each upper-level node is visualized using its top-10 terms. We ask human annotators to mark all reasonable parent nodes of the child term t, which is denoted as $\hat{\mathcal{P}}(t)$. We merge the parent nodes of term t that identified by the model into a set $\mathcal{P}^*(t)$. Precision, recall, and F1 are employed to evaluate $\hat{\mathcal{P}}(t)$ against $\mathcal{P}^*(t)$ by treating all sampled together. Formally, we have

$$\text{Precision} = \frac{\sum_t |\hat{\mathcal{P}}(t) \cap \mathcal{P}^*(t)|}{\sum_t |\mathcal{P}^*(t)|} \quad \text{Recall} = \frac{\sum_t |\hat{\mathcal{P}}(t) \cap \mathcal{P}^*(t)|}{\sum_t |\hat{\mathcal{P}}(t)|}$$

  and F_1 is defined as their harmonic mean.

A quality topic taxonomy should have high scores in all the three evaluation tasks.

Annotation Details. First of all, it is worth mentioning that we mix the results from different methods together and shuffle them randomly before sending them to annotators. The annotators will not be aware of the method from which the results are produced.

Second, in order to avoid bias during the annotation, we first ask the annotators to do the *Exclusive Siblings* task, then the *Parent-Child Relations* task, and finally the*Coherence* task. So the annotator will not have any prior knowledge about which terms are in the same taxonomy node in the first two tasks.

In all tasks, we observe that annotators have inter-annotator agreements of more than 90%. The scores presented in the experiments are therefore all averaged across different annotators.

5.5 Quantitative Evaluation

In this section, we discuss the quantitative evaluation results of different methods on the two datasets. The results are summarized in Table 2. Overall, the topic taxonomy constructed by NetTaxo has demonstrated its significant advantage over taxonomies constructed by other methods, in all three evaluated aspects.

The two datasets have slightly different properties as text in the DBLP dataset is written in a more formal style and thus consists of cleaner terms while Yelp reviews often contain colloquial language. In terms of the underlying taxonomy, the Yelp taxonomy spans from very high-level distinctions (e.g., *auto repairs* vs. *restaurants*) to subtle distinctions (e.g., fusion dishes should belong to multiple nodes while common products do not fit into any node). The DBLP taxonomy has much fewer cases of ambiguity. This leads to closer but generally lower scores on the Yelp dataset for most metrics.

Within two methods using text data only, TaxoGen outperforms HPAM++ in all evaluated aspects. Compared with TaxoGen, NetTaxo improves most on the identification of parent-child relations. The network information provides a better overview of the hierarchical topic structure, whereas parent term and child terms often share the

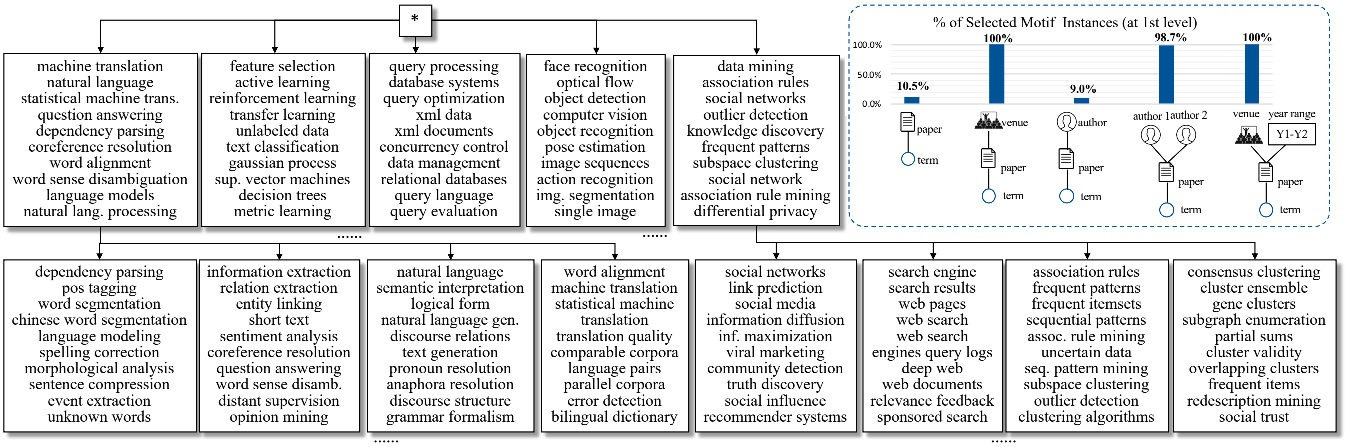


Figure 5: The Topic Taxonomy Constructed by NetTaxo on the DBLP Dataset. Due to the space limit, it is partially expanded. Each node is visualized as a rectangle block and its top-10 anchor terms. Arrows go from parent nodes to child nodes. In addition, at the first level, for each motif pattern, we show the percentage of selected instances over all instances of the motif.

same context in documents. This shows that the network structure truly provides complementary information to text.

Compared with CATHYHIN++, NetTaxo shows significant improvements in sibling exclusiveness and coherence. By initializing clusters using term embedding, we are able to better capture semantically similar terms for creating coherence clusters. The increase in sibling exclusiveness can be credited to the comparative analysis component, which puts sibling nodes under contrast to discover anchor nodes. Both of these components are only available with text data, which CATHYHIN++ does not leverage.

In short, NetTaxo outperforms TaxoGen and CATHYHIN++ in all metrics, demonstrating that text and network information are able to enhance each other.

HClusEmbed takes the same input as NetTaxo, but performs very poor among the baselines. This shows that applying pre-existing embedding models on text and network separately and then putting them together trivially is not enough to generate a high quality taxonomy. In HClusEmbed, term embedding aim to preserve semantic similarity while network embedding aim to preserve node proximity, both may not directly contribute to a better taxonomy. Moreover, checking the results of NetTaxo w/o Selection, one can observe that only a careful selection of the information from network structures can lead to performance gains. This is more significant on the Yelp-5 dataset, as the network information is much more noisy in this dataset. NetTaxo is carefully designed to select the most relevant motif contexts from network structures, and then incorporate them into a joint term embedding learning to further improve the quality of constructed taxonomy.

These comparisons further confirm the importance and effectiveness of our proposed motif selection process.

5.6 Constructed Topic Taxonomies

After the quantitative comparison, we present some case studies on both datasets for a closer look at the topic taxonomy constructed by our proposed NetTaxo.

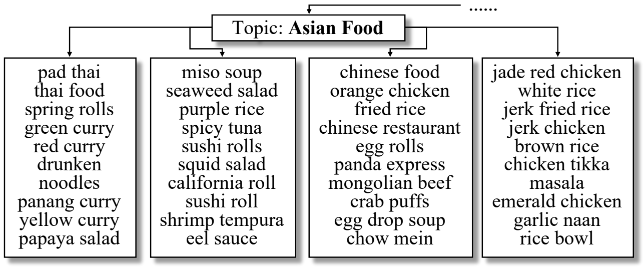


Figure 6: The Topic Taxonomy Constructed by NetTaxo on the Yelp Dataset. All sub-topics under the taxonomy node about "*Asian food*" are visualized.

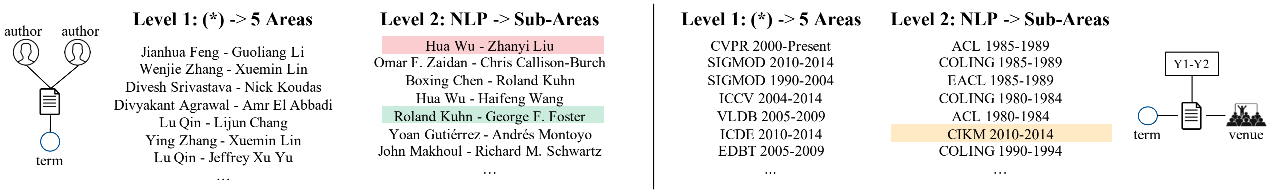
DBLP Taxonomy. We plot the final topic taxonomy in Figure 5. Due to the space limit, we only present the five nodes at the first level and expand two of them into the second level.

Looking at the top level topics, one can easily recognize the topics of the five nodes from the left to right as: (1) *natural language processing*, (2) *machine learning*, (3) *database*, (4) *computer vision*, and (5) *data mining*. These are exactly the five areas used in preparing the DBLP dataset.

In addition, we present the selected motif instance percentage for each motif pattern at this top level in Figure 5. The results are intuitive by putting more emphasis in venue-related motif patterns as well as the author-pair motif pattern. While we are conducting an instance-level motif selection, it actually implicitly selects motifs in pattern-level as well.

We then inspect the second-level results. Under the node about natural language processing, we can see four clear sub-topics: (1) *parsing*, (2) *information extraction*, (3) *language & grammar*, and (4) *machine translation*. Under the node about data mining, we can find (1) *social network analysis*, (2) *web mining and search*, (3) *frequent pattern/association rule mining*, and (4) *clustering*.

Yelp Taxonomy. In Figure 6, we present all four taxonomy nodes under the taxonomy node of "*Asian Food*" topic in our constructed



Level 1: (*) -> 5 Areas	Level 2: NLP -> Sub-Areas
Jianhua Feng - Guoliang Li	Hua Wu - Zhanyi Liu
Wenjie Zhang - Xuemin Lin	Omar F. Zaidan - Chris Callison-Burch
Divesh Srivastava - Nick Koudas	Boxing Chen - Roland Kuhn
Divyakant Agrawal - Amr El Abbadi	Hua Wu - Haifeng Wang
Lu Qin - Lijun Chang	Roland Kuhn - George F. Foster
Ying Zhang - Xuemin Lin	Yoan Gutiérrez - Andrés Montoyo
Lu Qin - Jeffrey Xu Yu	John Makhoul - Richard M. Schwartz
…	…

Level 1: (*) -> 5 Areas	Level 2: NLP -> Sub-Areas
CVPR 2000-Present	ACL 1985-1989
SIGMOD 2010-2014	COLING 1985-1989
SIGMOD 1990-2004	EACL 1985-1989
ICCV 2004-2014	COLING 1980-1984
VLDB 2005-2009	ACL 1980-1984
ICDE 2010-2014	CIKM 2010-2014
EDBT 2005-2009	COLING 1990-1994
…	…

Figure 7: Top motif instances selected by NetTaxo **at different taxonomy nodes. For venue + year range motif pattern, we merge consecutive instances for readability purpose if they are from the same venue and cover contiguous years. Three highlighted motif instances will be further elaborated with their most frequent terms in Figure 8.**

Hua Wu - Zhanyi Liu	Roland Kuhn - George F. Foster	CIKM 2010-2014
sentiment analysis	source language	question answering
semantic features	bilingual corpora	information extraction
semantic relations	bilingual word	language models
textual similarity	machine translation	sentiment analysis
sentiment words	statistical machine translation	sentiment classification
sentiment classification	bleu score	knowledge base
…	…	…

Figure 8: Most frequent terms linked to the three highlighted motif instances in Figure 7. Note that the frequency is calculated based on the weighted documents associated with the taxonomy node about NLP. Best viewed in color.

topic taxonomy on the Yelp dataset. Top-10 terms are presented at each node. While "*Asian Food*" is already a relatively fine-grained topic, NetTaxo successfully recovers its sub-topics: *Thai* cuisine, *Japanese* cuisine, *Chinese* cuisine, and *Other Asian* (e.g., *Indian*, *Mexican-Chinese Fusion*, . . .) cuisines. The first three sub-topics are quite clear, while the fourth one is a little vague. Remember that we set $k = 4$ here. So it makes sense to have an "other" sub-topic. At the first glance, "*Jade Red Chicken*" and "*Emrald Chicken*" look like Chinese dishes, and "*Jerk Fried Rice*" sounds like something from the Caribbean area. However, if one searches "*Jade Red Chicken*" in Google, a popular restaurant in *Arizona* named "*Chino Bandido*" pops up at the first place. It offers *Mexican-Chinese Fusion* dishes and these three dishes are strongly recommended by Yelp reviewers[5].

5.7 Effects of Instance-Level Motif Selection

Besides the final taxonomy quality, we're also interested in how the instance-level motif selection mechanism works at different taxonomy nodes. We visualize top motif instances selected by our method on the DBLP dataset. Figure 7 shows two motif patterns and their top instances at two taxonomy nodes, one from the first level and the other from the second level. Taking a closer look at three specific motif instances, we show most frequent terms linked to these motif instances in Figure 8.

On the first level, our goal is to identify major research fields, i.e., separating the 5 research areas in this dataset. From *co-authorship* motif pattern, we observe pairs of database researchers who share lots of research papers. The top-2 instances are all professors working in the same research group at the same university. From *venue-and-year-range* motif pattern, one can find many computer vision and database conferences. The reason for these motif instances to rank high is because *database* and *computer vision* are two relatively concentrated research areas, compared to *machine learning*, *data mining*, and *natural language processing* (NLP) which have more interconnections. Besides, professors and venues involved in the top-ranked instances are all highly reputed.

On the second level, the goal becomes more challenging — distinguishing research sub-areas. We use the NLP taxonomy node as an example. The *co-authorship* motif instances give us some less-known researchers. Therefore, we picked two of co-author pairs, sampled and visualized their associated terms from the motif context graph, shown in Figure 8. One can easily observe that these author groups work on relatively concentrated sub-topics under NLP, i.e., *sentiment analysis* and *machine translation*, respectively. From *venue-and-year-range* motif instances, we can see major NLP conferences in their early years, and data mining conferences in recent years. This is also quite interesting but explainable, as NLP conferences have a narrower scope in their early years, while the data mining community, as it evolves, has more overlaps with the NLP community recently. Specifically, we show most frequent terms linked to the motif instance "CIKM 2010-2014" in Figure 8, where we observe many NLP sub-topics such as "*question answering*" and "*information extraction*". These topics are also studied by information retrieval and data mining researchers recently.

Overall, from the empirical observations, we can verify that our instance-level motif selection is effective.

6 CONCLUSIONS & FUTURE WORK

In this paper, towards automatic topic taxonomy construction, we propose a novel hierarchical term embedding and clustering framework NetTaxo, which consumes a text-rich network as the input. Through a careful selection of motif contexts, NetTaxo learns term embedding jointly from the text data together with the most helpful network structures. To consolidate the foundation of such selection, we further design a method to choose anchor terms from the initial clusters based on text data only. Extensive experiments on two datasets demonstrate the superiority of our framework compared with baselines. Ablation experiments confirm the necessity and effectiveness of our proposed instance-level motif selection. Case studies illustrate the quality of our constructed taxonomy.

In future work, we would like to further improve NetTaxo in the following aspects. First, we would like to develop a more principled solution to determine the number of sub-topics at each taxonomy node. Second, incorporating user-provided seed examples of the desired taxonomy in construction process could be a promising and practically useful direction to pursue. Last but not least, we are interested in integrating our constructed taxonomy into downstream applications, such as recommender systems and question answering tasks.

[5] https://www.yelp.com/menu/chino-bandido-chandler/item/jade-red-chicken

ACKNOWLEDGMENTS

This work was sponsored in part by DARPA under Agreements No. W911NF-17-C-0099 and FA8750-19-2-1004, National Science Foundation IIS 16-18481, IIS 17-04532, and IIS-17-41317, and DTRA HDTRA11810026. Any opinions, findings, and conclusions or recommendations expressed in this document are those of the author(s) and should not be interpreted as the views of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes not withstanding any copyright notation hereon.

REFERENCES

[1] Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries*. ACM, 85–94.

[2] Tuan Luu Anh, Yi Tay, Siu Cheung Hui, and See Kiong Ng. 2016. Learning term embeddings for taxonomic relation identification using dynamic weighting neural network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 403–413.

[3] Arindam Banerjee, Inderjit S Dhillon, Joydeep Ghosh, and Suvrit Sra. 2005. Clustering on the unit hypersphere using von Mises-Fisher distributions. *Journal of Machine Learning Research* 6, Sep (2005), 1345–1382.

[4] Mohit Bansal, David Burkett, Gerard De Melo, and Dan Klein. 2014. Structured learning for taxonomy induction with belief propagation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 1041–1051.

[5] Austin R Benson, David F Gleich, and Jure Leskovec. 2016. Higher-order organization of complex networks. *Science* 353, 6295 (2016), 163–166.

[6] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R Hruschka Jr, and Tom M Mitchell. 2010. Toward an architecture for never-ending language learning.. In *AAAI*, Vol. 5. Atlanta, 3.

[7] David L Davies and Donald W Bouldin. 1979. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence* (1979), 224–227.

[8] R Lopez de Mantaras and L Saitia. 2004. Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. In *16th European Conference on Artificial Intelligence Conference Proceedings*, Vol. 110. 435.

[9] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 135–144.

[10] Doug Downey, Chandra Bhagavatula, and Yi Yang. 2015. Efficient methods for inferring large sparse topic hierarchies. 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)*, Vol. 1. 774–784.

[11] Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 1199–1209.

[12] Thomas L Griffiths, Michael I Jordan, Joshua B Tenenbaum, and David M Blei. 2004. Hierarchical topic models and the nested chinese restaurant process. In *Advances in neural information processing systems*. 17–24.

[13] Huan Gui, Qi Zhu, Liyuan Liu, Aston Zhang, and Jiawei Han. 2018. Expert Finding in Heterogeneous Bibliographic Networks with Locally-trained Embeddings. *arXiv preprint arXiv:1803.03370* (2018).

[14] Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics, 539–545.

[15] Meng Jiang, Jingbo Shang, Taylor Cassidy, Xiang Ren, Lance M Kaplan, Timothy P Hanratty, and Jiawei Han. 2017. Metapad: Meta pattern discovery from massive text corpora. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 877–886.

[16] Jialu Liu, Xiang Ren, Jingbo Shang, Taylor Cassidy, Clare R Voss, and Jiawei Han. 2016. Representing documents via latent keyphrase inference. In *Proceedings of the 25th international conference on World wide web*. International World Wide Web Conferences Steering Committee, 1057–1067.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[18] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. 2002. Network motifs: simple building blocks of complex networks. *Science* 298, 5594 (2002), 824–827.

[19] David Mimno, Wei Li, and Andrew McCallum. 2007. Mixtures of hierarchical topics with pachinko allocation. In *Proceedings of the 24th international conference on Machine learning*. ACM, 633–640.

[20] Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. PATTY: a taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 1135–1145.

[21] Alexander Panchenko, Stefano Faralli, Eugen Ruppert, Steffen Remus, Hubert Naets, Cédrick Fairon, Simone Paolo Ponzetto, and Chris Biemann. 2016. TAXI at SemEval-2016 Task 13: a taxonomy induction method based on lexico-syntactic patterns, substrings and focused crawling. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. 1320–1327.

[22] 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)*. 1532–1543.

[23] Simone Paolo Ponzetto and Michael Strube. 2007. Deriving a large scale taxonomy from Wikipedia. In *AAAI*, Vol. 7. 1440–1445.

[24] Aravind Sankar, Xinyang Zhang, and Kevin Chen-Chuan Chang. 2019. Meta-GNN: Metagraph Neural Network for Semi-supervised learning in Attributed Heterogeneous Information Networks. In *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE.

[25] Julian Seitner, Christian Bizer, Kai Eckert, Stefano Faralli, Robert Meusel, Heiko Paulheim, and Simone Paolo Ponzetto. 2016. A Large DataBase of Hypernymy Relations Extracted from the Web.. In *LREC*.

[26] 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 (2018), 1825–1837.

[27] Jingbo Shang, Meng Qu, Jialu Liu, Lance M Kaplan, Jiawei Han, and Jian Peng. 2016. Meta-path guided embedding for similarity search in large-scale heterogeneous information networks. *arXiv preprint arXiv:1610.09769* (2016).

[28] 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. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2180–2189.

[29] Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation* 28, 1 (1972), 11–21.

[30] Olaf Sporns and Rolf Kötter. 2004. Motifs in brain networks. *PLoS biology* 2, 11 (2004), e369.

[31] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment* 4, 11 (2011), 992–1003.

[32] Yizhou Sun, Jiawei Han, Peixiang Zhao, Zhijun Yin, Hong Cheng, and Tianyi Wu. 2009. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*. ACM, 565–576.

[33] Yizhou Sun, Yintao Yu, and Jiawei Han. 2009. Ranking-based clustering of heterogeneous information networks with star network schema. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 797–806.

[34] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1067–1077.

[35] Fangbo Tao, Chao Zhang, Xiusi Chen, Meng Jiang, Tim Hanratty, Lance Kaplan, and Jiawei Han. 2018. Doc2Cube: Allocating Documents to Text Cube without Labeled Data. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1260–1265.

[36] Fangbo Tao, Honglei Zhuang, Chi Wang Yu, Qi Wang, Taylor Cassidy, Lance R Kaplan, Clare R Voss, and Jiawei Han. 2016. Multi-Dimensional, Phrase-Based Summarization in Text Cubes. *IEEE Data Eng. Bull.* 39, 3 (2016), 74–84.

[37] 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 *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 437–445.

[38] Chi Wang, Marina Danilevsky, Jialu Liu, Nihit Desai, Heng Ji, and Jiawei Han. 2013. Constructing Topical Hierarchies in Heterogeneous Information Networks. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*. IEEE, 767–776.

[39] Pengtao Xie and Eric P Xing. 2013. Integrating document clustering and topic modeling. *arXiv preprint arXiv:1309.6874* (2013).

[40] Pengtao Xie, Diyi Yang, and Eric Xing. 2015. Incorporating word correlation knowledge into topic modeling. In *Proceedings of the 2015 conference of the north American chapter of the association for computational linguistics: human language technologies*. 725–734.

[41] Carl Yang, Yichen Feng, Pan Li, Yu Shi, and Jiawei Han. 2018. Meta-graph based hin spectral embedding: Methods, analyses, and insights. In *2018 IEEE*

International Conference on Data Mining (ICDM). IEEE, 657–666.

[42] Shuo Yang, Lei Zou, Zhongyuan Wang, Jun Yan, and Ji-Rong Wen. 2017. Efficiently Answering Technical Questions-A Knowledge Graph Approach.. In *AAAI*. 3111–3118.

[43] Xiaoxin Yin and Sarthak Shah. 2010. Building taxonomy of web search intents for name entity queries. In *Proceedings of the 19th international conference on World wide web*. 1001–1010.

[44] 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*. ACM, 2701–2709.

[45] Daokun Zhang, Jie Yin, Xingquan Zhu, and Chengqi Zhang. 2018. Meta-Graph2Vec: complex semantic path augmented heterogeneous network embedding. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 196–208.

[46] Yuchen Zhang, Amr Ahmed, Vanja Josifovski, and Alexander Smola. 2014. Taxonomy discovery for personalized recommendation. In *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 243–252.

[47] Jun Zhu, Zaiqing Nie, Xiaojiang Liu, Bo Zhang, and Ji-Rong Wen. 2009. Stat-Snowball: a statistical approach to extracting entity relationships. In *Proceedings of the 18th international conference on World wide web*. ACM, 101–110.