Unsupervised Terminological Ontology Learning based on Hierarchical Topic Modeling

Xiaofeng Zhu, Diego Klabjan
Northwestern University, USA
xiaofengzhu2013@u.northwestern.edu, d-klabjan@northwestern.edu

Patrick N Bless
Intel Corporation, USA
patrick.n.bless@intel.co

Abstract

*In this paper, we present **h**ierarchical **r**elation-based **l**atent **D**irichlet **a**llocation (hrLDA), a data-driven hierarchical topic model for extracting terminological ontologies from a large number of heterogeneous documents. In contrast to traditional topic models, hrLDA relies on noun phrases instead of unigrams, considers syntax and document structures, and enriches topic hierarchies with topic relations. Through a series of experiments, we demonstrate the superiority of hrLDA over existing topic models, especially for building hierarchies. Furthermore, we illustrate the robustness of hrLDA in the settings of noisy data sets, which are likely to occur in many practical scenarios. Our ontology evaluation results show that ontologies extracted from hrLDA are very competitive with the ontologies created by domain experts.*

1 Introduction

Although researchers have made significant progress on knowledge acquisition and have proposed many ontologies, for instance, WordNet [22], DBpedia [3], YAGO [30], Freebase, [8] Nell [9], DeepDive [26], Domain Cartridge [24], Knowledge Vault [12], INS-ES [33], iDLER [10], and TransE-NMM [25], current ontology construction methods still rely heavily on manual parsing and existing knowledge bases. This raises challenges for learning ontologies in new domains. While a strong ontology parser is effective in small-scale corpora, an unsupervised model is beneficial for learning new entities and their relations from new data sources, and is likely to perform better on larger corpora.

In this paper, we focus on unsupervised terminological ontology learning and formalize a terminological ontology as a hierarchical structure of subject-verb-object triplets. We divide a terminological ontology into two components: **topic hierarchies** and **topic relations**. Topics are presented in a tree structure where each node is a topic label (noun phrase), the root node represents the most general topic, the leaf nodes represent the most specific topics, and every topic is composed of its topic label and its descendant topic labels. Topic hierarchies are preserved in topic paths, and a topic path connects a list of topics labels from the root to a leaf. Topic relations are semantic relationships between any two topics or properties used to describe one topic. Figure 1 depicts an example of a terminological ontology learned from a corpus about European cities. We extract terminological ontologies by applying unsupervised hierarchical topic modeling and relation extraction to plain text.

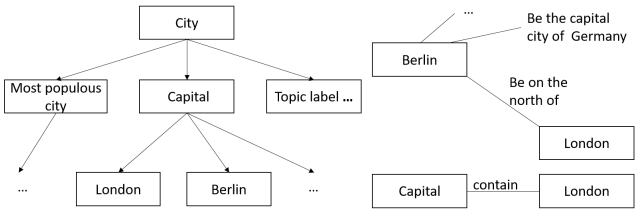


Figure 1: A representation of a terminological ontology. (Left: topic hierarchies) Topic $city$ is composed of $most\ populous\ city$, $capital$, $London$, $Berlin$, etc. $City \rightarrow capital \rightarrow London$ and $city \rightarrow capital \rightarrow Berlin$ are two topic paths. (Right: topic relations) Every topic label has relations to itself and/or with other labels. $Be\ the\ capital\ city\ of\ Germany$ is one relation/property of topic $Berlin$. $Be\ on\ the\ north\ of$ is one relation of topic $Berlin$ to $London$.

Topic modeling was originally used for topic extraction and document clustering. The classical topic model, latent Dirichlet allocation (LDA) [7], simplifies a document as a bag of its words and describes a topic as a distribution of words. Prior research [27, 32, 34, 17, 29, 11, 23, 16] has shown that LDA-based approaches are adequate for (terminological) ontology learning. However, these models are deficient in that they still need human supervision to decide the number of topics, and to pick meaningful topic labels usually from a list of unigrams. Among models not using

unigrams, LDA-based Global Similarity Hierarchy Learning (LDA+GSHL) [32] only extracts a subset of relations: "broader" and "related" relations. In addition, the topic hierarchies of KB-LDA [23] rely on hypernym-hyponym pairs capturing only a subset of hierarchies.

Considering the shortcomings of the existing methods, the main objectives of applying topic modeling to ontology learning are threefold.

1. In topic models, a topic is usually represented with a list of unigrams. In a terminological ontology, a topic/entity needs to be represented with a more descriptive identifier (i.e., noun phrase). Currently, the number of topics is usually a fixed parameter, which restricts the number of classes an ontology could have. For instance, it is difficult to add a new species to an animal ontology.

2. Both relations among different noun phrases and relations/properties (see the relations in Figure 1) for describing single noun phrases should be captured during the topic generation process.

3. Hierarchies need to be built on topical affiliations. If topic B is a sub-topic of topic A, B has a more specific meaning than A. The depth of each topic path should be determined by a data-driven method.

To achieve the first objective, we extract noun phrases and then propose a sampling method to estimate the number of topics. For the second objective, we use language parsing and relation extraction to learn relations for the noun phrases. Regarding the third objective, we adapt and improve the hierarchical latent Dirichlet allocation (hLDA) model [6, 5]. hLDA is not ideal for ontology learning because it builds topics from unigrams (which are not descriptive enough to serve as entities in ontologies) and the topics may contain words from multiple domains when input data have documents from many domains (see Section 2 and Figure 9). Our model, hrLDA, overcomes these deficiencies. In particular, hrLDA represents topics with noun phrases, uses syntax and document structures such as paragraph indentations and item lists, assigns multiple topic paths for every document, and allows topic trees to grow vertically and horizontally.

The primary contributions of this work can be specified as follows.

- We develop a hierarchical topic model, hrLDA, that does not require one to set the topic number at every level of a topic tree or to set the topic path lengths from the root to leaves.

- We integrate relation extraction into topic modeling leading to lower perplexity.

- We propose a multiple topic path drawing strategy, which is an improvement over the simple topic path drawing method proposed in hLDA.

- We present automatic extraction of terminological ontologies via hrLDA.

2 Background

In this section, we introduce our main baseline model, hierarchical latent Dirichlet allocation (hLDA), and some of its extensions. We start from the components of hLDA - latent Dirichlet allocation (LDA) and the Chinese Restaurant Process (CRP)- and then explain why hLDA needs improvements in both building hierarchies and drawing topic paths.

LDA is a three-level Bayesian model in which each document is a composite of multiple topics, and every topic is a distribution over words. Due to the lack of determinative information, LDA is unable to distinguish different instances containing the same content words, (e.g. "I trimmed my polished nails" and "I have just hammered many rusty nails"). In addition, in LDA all words are probabilistically independent and equally important. This is problematic because different words and sentence elements should have different contributions to topic generation. For instance, articles contribute little compared to nouns, and sentence subjects normally contain the main topics of a document.

Introduced in hLDA, CRP partitions words into several topics by mimicking a process in which customers sit down in a Chinese restaurant with an infinite number of tables and an infinite number of seats per table. Customers enter one by one, with a new customer choosing to sit at an occupied table or a new table. The probability of a new customer sitting at the table with the largest number of customers is the highest. In reality, customers do not always join the largest table but prefer to dine with their acquaintances. The theory of distance-dependent CRP was formerly proposed by David Blei [4]. We provide later in Section 3.3 an explicit formula for topic partition given that adjacent words and sentences tend to deal with the same topics.

hLDA combines LDA with CRP by setting one topic path with fixed depth L for each document. The hierarchical relationships among nodes in the same path depend on an L dimensional Dirichlet distribution that actually arranges the probabilities of topics being on different topic levels. Despite the fact that the single path was changed to multiple paths in some extensions of hLDA - the nested Chinese restaurant franchise processes [1] and the nested hierarchical Dirichlet Processes [28], - this topic path drawing strategy puts words from different domains into one topic when input data are mixed with topics from multiple domains. This means that if a corpus contains documents

in four different domains, hLDA is likely to include words from the four domains in every topic (see Figure 9).

In light of the various inadequacies discussed above, we propose a relation-based model, hrLDA. hrLDA incorporates semantic topic modeling with relation extraction to integrate syntax and has the capacity to provide comprehensive hierarchies even in corpora containing mixed topics.

3 Hierarchical Relation-based Latent Dirichlet Allocation

The main problem we address in this section is generating terminological ontologies in an unsupervised fashion. The fundamental concept of hrLDA is as follows. When people construct a document, they start with selecting several topics. Then, they choose some noun phrases as subjects for each topic. Next, for each subject they come up with relation triplets to describe this subject or its relationships with other subjects. Finally, they connect the subject phrases and relation triplets to sentences via reasonable grammar. The main topic is normally described with the most important relation triplets. Sentences in one paragraph, especially adjacent sentences, are likely to express the same topic.

We begin by describing the process of reconstructing LDA. Subsequently, we explain relation extraction from heterogeneous documents. Next, we propose an improved topic partition method over CRP. Finally, we demonstrate how to build topic hierarchies that bind with extracted relation triplets.

3.1 Relation-based Latent Dirichlet Allocation

Documents are typically composed of chunks of texts, which may be referred to as sections in Word documents, paragraphs in PDF documents, slides in presentation documents, etc. Each chunk is composed of multiple sentences that are either atomic or complex in structure, which means a document is also a collection of atomic and/or complex sentences. An atomic sentence (see module T in Figure 2) is a sentence that contains only one subject (S), one object (O) and one verb (V) between the subject and the object. For every atomic sentence whose object is also a noun phrase, there are at least two relation triplets (e.g., "*The tiger that gave the excellent speech is handsome*" has relation triplets: (*tiger, give, speech*), (*speech, be given by, tiger*), and (*tiger, be, handsome*)). By contrast, a complex sentence can be subdivided into multiple atomic sentences. Given that the syntactic verb in a relation triplet is determined by the subject and the object, a document d in a corpus D can be ultimately reduced to N_d subject phrases (we convert objects to subjects using passive voice) associated with N_d relation triplets T_d. Number N_d is usually larger than the actual number of noun phrases in document d. By replacing the unigrams in LDA with relation triplets, we retain definitive information and assign salient noun phrases high weights.

We define $Dir(\alpha)$ as a Dirichlet distribution parameterized by hyperparameters α, $Multi(\theta)$ as a multinomial distribution parameterized by hyperparameters θ, $Dir(\eta)$ as a Dirichlet distribution parameterized by η, and $Multi(\beta)$ as a multinomial distribution parameterized by β. We assume the corpus has K topics. Assigning K topics to the N_d relation triplets of document d follows a multinomial distribution $Multi(\theta)$ with prior $Dir(\alpha)$. Selecting the N_d relation triplets for document d given the K topics follows a multinomial distribution $Multi(\beta)$ with prior $Dir(\eta)$. We denote $T = \{T_d\}_{d \in D}$ as the list of relation triplet lists extracted from all documents in the corpus, and Z as the list of topic assignments of T. We denote the relation triplet counts of documents in the corpus by $N = \{N_d\}_{d \in D}$. The graphical representation of the relation-based latent Dirichlet allocation (rLDA) model is illustrated in Figure 2.

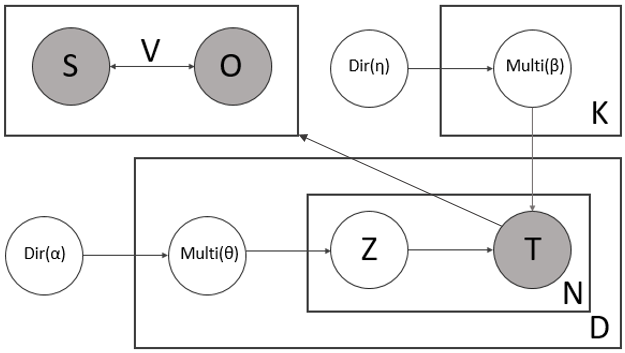


Figure 2: Plate notation of rLDA

The plate notation can be decomposed into two types of Dirichlet-multinomial conjugated structures: document-topic distribution $Dir(\alpha) \rightarrow Multi(\theta) \rightarrow Z$ and topic-relation distribution $Dir(\eta) \rightarrow Multi(\beta) \rightarrow T|Z$. Hence, the joint distribution of T and Z can be represented as

$$\begin{aligned} & P(T, Z|\alpha, \eta) \\ & = P(T|Z, \eta)\ P(Z|\alpha) \\ & = \prod_{k=1}^{K} \frac{Dir(C_k + \eta)}{Dir(\eta)} \prod_{d}^{D} \frac{Dir(B_d + \alpha)}{Dir(\alpha)} \\ & C_k = (C_k^1, C_k^2, ..., C_k^w, ..., C_k^W) \\ & B_d = (B_d^1, B_d^2, ..., B_d^k, ..., B_d^K), \end{aligned} \tag{1}$$

where W is the number of unique relations in all documents, C_k^w is the number of occurrences of the relation triplet w generated by topic k in all documents, and B_d^k is the number of relation triplets generated by topic k in document d. $Dir(\alpha)$ is a conjugate prior for $Multi(\theta)$ and

thus the posterior distribution is a new Dirichlet distribution parameterized by $(B_d + \alpha)$. The same rule applies to $Dir(C_k + \eta)$.

3.2 Relation Triplet Extraction

Extracting relation triplets is the essential step of hrLDA, and it is also the key process for converting a hierarchical topic tree to an ontology structure. The idea is to find all syntactically related noun phrases and their connections using a language parser such as the Stanford NLP parser [19] and Ollie [20]. Generally, there are two types of relation triplets:

- Subject-predicate-object-based relations,
  e.g., *New York is the largest city in the United States* ⇒ (*New York, be the largest city in, the United States*);
- Noun-based/hidden relations,
  e.g., *Queen Elizabeth* ⇒ (*Elizabeth, be, queen*).

A special type of relation triplets can be extracted from presentation documents such as those written in PowerPoint using document structures. Normally lines in a slide are not complete sentences, which means language parsing does not work. However, indentations and bullet types usually express inclusion relationships between adjacent lines. Starting with the first line in an itemized section, our algorithm scans the content in a slide line by line, and creates relations based on the current item and the item that is one level higher.

3.3 Acquaintance Chinese Restaurant Process

As mentioned in Section 2, CRP always assigns the highest probability to the largest table, which assumes customers are more likely to sit at the table that has the largest number of customers. This ignores the social reality that a person is more willing to choose the table where his/her closest friend is sitting even though the table also seats unknown people who are actually friends of friends. Similarly with human-written documents, adjacent sentences usually describe the same topics. We consider a restaurant table as a topic, and a person sitting at any of the tables as a noun phrase. In order to penalize the largest topic and assign high probabilities to adjacent noun phrases being in the same topics, we introduce an improved partition method, Acquaintance Chinese Restaurant Process (ACRP).

The ultimate purposes of ACRP are to estimate K, the number of topics for rLDA, and to set the initial topic distribution states for rLDA. Suppose a document is read from top to bottom and left to right. As each noun phrase belongs to one sentence and one text chunk (e.g., section, paragraph and slide), the locations of all noun phrases in a document can be mapped to a two-dimensional space where **sentence location** is the x axis and text **chunk location** is the y axis (the first noun phrase of a document holds value (0, 0)). More specifically, every noun phrase has four attributes: **content**, **location**, **one-to-many relation triplets**, and **document ID**. Noun phrases in the same text chunk are more likely to be "acquaintances;" they are even closer to each other if they are in the same sentence. In contrast to CRP, ACRP assigns probabilities based on closeness, which is specified in the following procedure.

1. Let z_n be the integer-valued random variable corresponding to the index of a topic assigned to the n^{th} phrase. Draw a probability $P(z_{n+1})$ from Equations 2 to 5 below for the $(n+1)^{th}$ noun phrase t^{n+1}, joining each of the existing k topics and the new $(k+1)^{th}$ topic given the topic assignments of previous n noun phrases, $Z_{1:n}$. If a noun phrase joins any of the existing k topics, we denote the corresponding topic index by $i \in [1, k]$.
   - The probability of choosing the $(k+1)^{th}$ topic:
   $$P(z_{n+1} = (k+1)|Z_{1:n}) = \frac{\gamma}{n+\gamma}. \quad (2)$$
   - The probability of selecting any of the k topics:
     - if the content of t^{n+1} is synonymous with or an acronym of a previously analyzed noun phrase t^m $(m < n+1)$,
     $$P(z_{n+1} = i|Z_{1:n}) = 1 - \gamma; \quad (3)$$
     - else if the document ID of t^{n+1} is different from all document IDs belonging to the i^{th} topic,
     $$P(z_{n+1} = i|Z_{1:n}) = \gamma; \quad (4)$$
     - otherwise,
     $$P(z_{n+1} = i|Z_{1:n}) = \frac{C_i - (1 - \frac{1}{min(Q_{1:i})})}{(1 + min(S_{1:i}))n + \gamma}, \quad (5)$$
     where C_i refers to the current number of noun phrases in the i^{th} topic, $Q_{1:i}$ represents the vector of **chunk location** differences of the $(n+1)^{th}$ noun phrase and all members in the i^{th} topic, $S_{1:i}$ stands for the vector of **sentence location** differences, and γ is a penalty factor.

   Normalize the $(k+1)$ probabilities to guarantee they are each in the range of [0, 1] and their sum is equal to 1.

2. Based on the probabilities 2 to 5, we sample a topic index z from $\{1, ..., (k+1)\}$ for every noun phrase, and we count the number of unique topics K in the end. We shuffle the order of documents and iterate ACRP until K is unchanged.

3.4 Nested Acquaintance Chinese Restaurant Process

The procedure for extending ACRP to hierarchies is essential to why hrLDA outperforms hLDA. Instead of a predefined tree depth L, the tree depth for hrLDA is optional and data-driven. More importantly, clustering decisions are made given a global distribution of all current non-partitioned phrases (leaves) in our algorithm. This means there can be multiple paths traversed down a topic tree for each document. With reference to the topic tree, every node has a noun phrase as its label and represents a topic that may have multiple sub-topics. The root node is visited by all phrases. In practice, we do not link any phrases to the root node, as it contains the entire vocabulary. An inner node of a topic tree contains a selected topic label. A leaf node contains an unprocessed noun phrase. We define a hashmap $leaves$ with a document ID as the key and the current leaf nodes of the document as the value. We denote the current tree level by l. We next outline the overall algorithm.

1. We start with the root node ($l = 0$) and apply rLDA to all the documents in a corpus.
   (a) Collect the current leaf nodes of every document. $leaves$ initially contains all noun phrases in the corpus. Assign a cluster partition to the leaf nodes in each document based on ACRP and sample the cluster partition until the number of topics of all noun phrases in $leaves$ is stable or the iteration reaches the predefined number of iteration times (whichever occurs first).
   (b) Mark the number of topics (child nodes) of parent node m at level l as K^{l_m}. Build a K^{l_m}-dimensional topic proportion vector θ based on $Dir(\alpha)$.
   (c) For every noun phrase $\{t_n\}_{n=1}^{N_d}$ in document d, form the topic assignments $Z_{\{1,...,K_{l_m}\}}$ based on $Multi(\theta)$.
   (d) Generate relation triplets from $Multi(\beta)$ given $Dir(\eta)$ and the associated topic vector $\{Z_k\}_{k=1}^{K^{l_m}}$.
   (e) Eliminate partitioned leaf nodes from $leaves$. Update the current level l by 1.
2. If phrases in $leaves$ are not yet completely partitioned to the next level and l is less than L, continue the following steps. For each leaf node, we set the top phrase (i.e., the phrase having the highest probability) as the topic label of this leaf node and the leaf node becomes an inner node. We next update $leaves$ and repeat procedures $1(a) - 1(e)$.

To summarize this process more succinctly: we build the topic hierarchies with rLDA in a divisive way (see Figure 3). We start with the collection of extracted noun phrases and split them using rLDA and ACRP. Then, we apply the procedure recursively until each noun phrase is selected as a topic label. After every rLDA assignment, each inner node only contains the topic label (top phrase), and the rest of the phrases are divided into nodes at the next level using ACRP and rLDA. Hence, we build a topic tree with each node as a topic label (noun phrase), and each topic is composed of its topic labels and the topic labels of the topic's descendants. In the end, we finalize our terminological ontology by linking the extracted relation triplets with the topic labels as subjects.

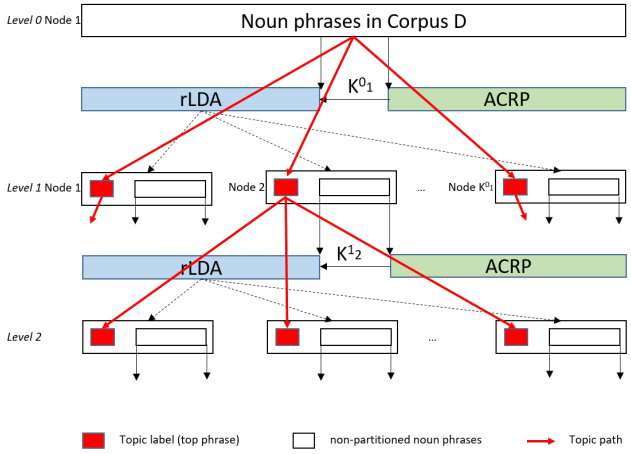


Figure 3: Graphical representation of hrLDA

We use collapsed Gibbs sampling [15] for inference from posterior distribution $P(Z|T, \alpha, \eta)$ based on Equation 1. Assume the n^{th} noun phrase $t_n = \hat{t}$ in parent node m comes from document d. We denote unassigned noun phrases from document d in parent node m by d^m, and unique noun phrases in parent node m by $\hat{T}_m$. We simplify the probability of assigning the n^{th} noun phrase in parent node m to topic k among K^{l_m} topics as

$$
\begin{aligned}
&P(z_n = k|Z_{\neg n}, \hat{T}_m, \alpha, \eta) \\
&\propto P(t_n = \hat{t}, z_n = k|Z_{\neg n}, \hat{T}_{m\neg n}, \alpha, \eta) \\
&= \int P(t_n = \hat{t}, z_n = k|Z_{\neg n}, \hat{T}_{m\neg n}, \theta_{d^m}, \beta_k) d\theta_{d^m}, d\beta_k \\
&= \frac{C_{k,\hat{t}\neg n} + \eta}{\sum_{\hat{t}}^{\hat{T}_m}(C_{k,\hat{t}\neg n} + \eta)} \frac{C_{d^m,k\neg n} + \alpha}{\sum_{k=1}^{K^{l_m}}(C_{d^m,k\neg n} + \alpha)}
\end{aligned}
\tag{6}
$$

where $Z_{\neg n}$ refers to all topic assignments other than z_n, θ_{d^m} is multinational document-topic distribution for unassigned noun phrases d^m, β_k is the multinational topic-relation distribution for topic k, $C_{k,\hat{t}\neg n}$ is the number of occurrences of noun phrase $\hat{t}$ in topic k except the n^{th} noun phrase in m, $C_{d^m,k\neg n}$ stands for the number of times that topic k occurs in d^m excluding the n^{th} noun phrase in m.

In order to build a hierarchical topic tree of a specific domain, we must generate a subset of the relation triplets using external constraints or semantic seeds via a pruning process [31]. As mentioned above, in a relation triplet, each relation connects one subject and one object. By assembling all subject and object pairs, we can build an undirected graph with the objects and the subjects constituting the nodes of the graph [18]. Given one or multiple semantic seeds as input, we first collect a set of nodes that are connected to the seed(s), and then take the relations from the set of nodes as input to retrieve associated subject and object pairs. This process constitutes one recursive step. The subject and object pairs become the input of the subsequent recursive step.

4 Empirical Results

4.1 Implementation

We utilized the Apache poi library to parse texts from pdfs, word documents and presentation files; the MALLET toolbox [21] for the implementations of LDA, optimized_LDA [2] and hLDA; the Apache Jena library to add relations, properties and members to hierarchical topic trees; and Stanford Protege[1] for illustrating extracted ontologies. We make our code and data available [2]. We used the same hyper-parameter setting (i.e., $\alpha = 1.0$, $\eta = 0.1$, and $\gamma = 0.01$) across all our experiments.

4.2 Evaluation and Examples

In this section, we present the evaluation results of hrLDA tested against optimized_LDA, hLDA, and phrase_hLDA (i.e., hLDA based on noun phrases) as well as ontology examples that hrLDA extracted from real-world text data. The entire corpus we generated contains 349,362 tokens (after removing stop words and cleaning) and is built from articles on *semiconductor packaging*. It includes 84 presentation files, articles from 1,782 Wikipedia pages and 3,000 research papers that were published in IEEE manufacturing conference proceedings within the last decade. In order to see the performance in data sets of different scales, we also used a smaller corpus Wiki that holds the articles collected from the Wikipedia pages only.

[1] http://protege.stanford.edu/

[2] https://github.com/UnsupervisedOntologyLearning/hrLDA

We extract a single level topic tree using each of the four models; hrLDA becomes rLDA, and phrase_hLDA becomes phrase-based LDA. We have tested the average perplexity and running time performance of ten independent runs on each of the four models [14, 13]. Equation 7 defines the perplexity, which we employed as an empirical measure.

$$ln(perplexity) = -\frac{\sum_d^D \log(\sum_{k=1}^K P(T_d|Z_k)P(Z_k|d))}{\sum_d^D N_d}, \quad (7)$$

where T_d is a vector containing the N_d relation triplets in document d.

The comparison results on our Wiki corpus are shown in Figure 4. hrLDA yields the lowest perplexity and reasonable running time. As the running time spent on parameter optimization is extremely long (the optimized_LDA requires 19.90 hours to complete one run), for efficiency, we adhere to the fixed parameter settings for hrLDA. We then demonstrate the evaluation results from two aspects: topic hierarchy and ontology rule.

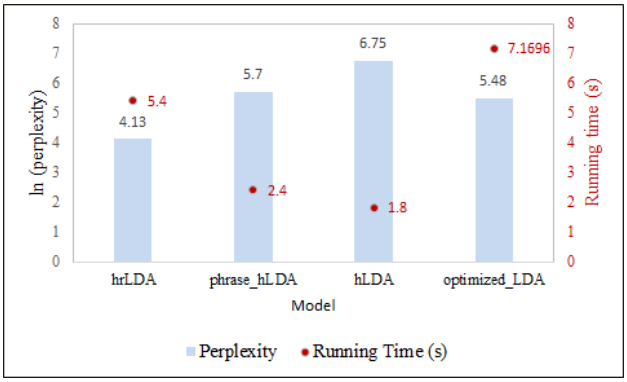
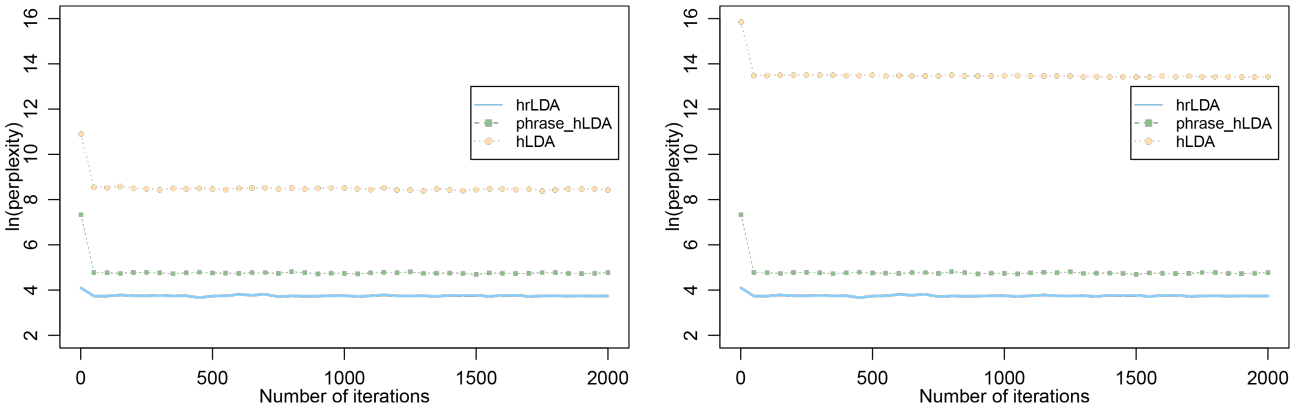


Figure 4: Comparison results of hrLDA, phrase_hLDA, hLDA and optimized_LDA on perplexity and running time

4.2.1 Hierarchy Evaluation

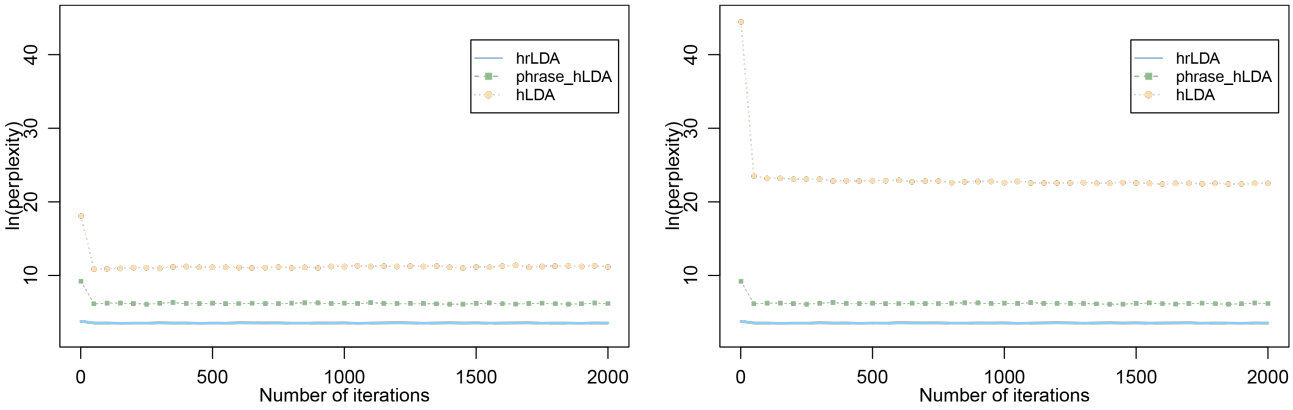
Superiority

Figures 5 to 7 illustrates the perplexity trends of the three hierarchical topic models (i.e., hrLDA, phrase_hLDA and hLDA) applied to both the Wiki corpus and the entire corpus with *seed* "*chip*" given different level settings. From left to right, hrLDA retains the lowest perplexities compared with other models as the corpus size grows. Furthermore, from top to bottom, hrLDA remains stable as the topic level increases, whereas the perplexity of phrase_hLDA and especially the perplexity of hLDA become rapidly high. Figure 8 highlights the perplexity values of the three models with confidence intervals in the final state. As shown in the two types of experiments, hrLDA has the lowest average perplexities and smallest confidence intervals, followed by phrase_hLDA, and then hLDA.



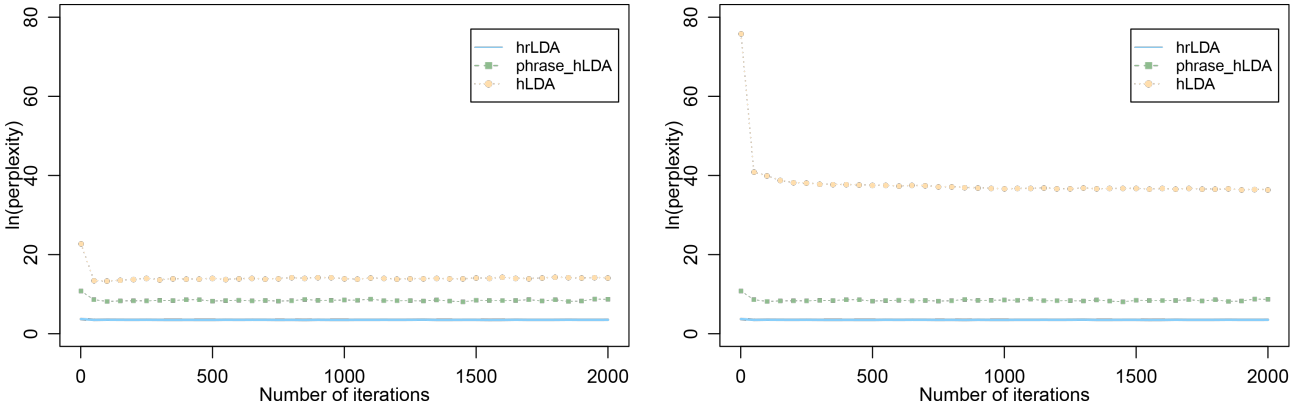
(a) The Wiki corpus (b) The entire corpus

Figure 5: Perplexity trends within 2000 iterations with level = 2



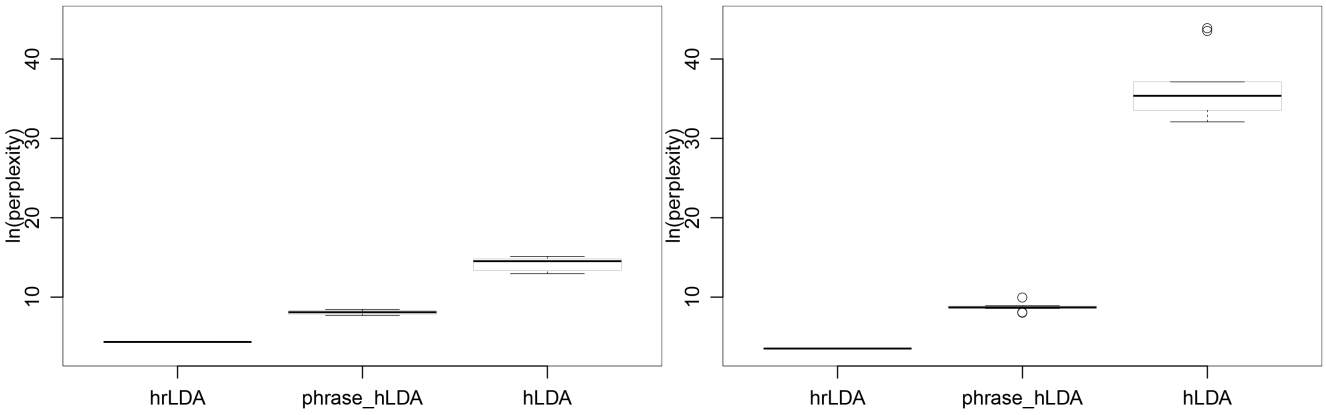
(a) The Wiki corpus (b) The entire corpus

Figure 6: Perplexity trends within 2000 iterations with level = 6



(a) The Wiki corpus (b) The entire corpus

Figure 7: Perplexity trends within 2000 iterations with level = 10



(a) The Wiki corpus

(b) The entire corpus

Figure 8: Average perplexities with confidence intervals of the three models in the final 2000th iteration with level = 10

Robustness

Figure 9 shows exhaustive hierarchical topic trees extracted from a small text sample with topics from four domains: *semiconductor*, *integrated circuit*, *Berlin*, and *London*. hLDA tends to mix words from different domains into one topic. For instance, words on the first level of the topic tree come from all four domains. This is because the topic path drawing method in existing hLDA-based models takes words in the most important topic of every document and labels them as the main topic of the corpus. In contrast, hrLDA is able to create four big branches for the four domains from the root. Hence, it generates clean topic hierarchies from the corpus.

4.2.2 Gold Standard-based Ontology Evaluation

The visualization of one concrete ontology on the *semiconductor packaging* domain is presented in Figure 10. For instance, Topic *packaging* contains topic *integrated circuit packaging*, and topic label *jedec* is associated with relation triplet *(jedec, be short for, joint electron device engineering council)*.

We use KB-LDA, phrase_hLDA, and LDA+GSHL as our baseline methods, and compare ontologies extracted from hrLDA, KB-LDA, phrase_hLDA, and LDA+GSHL with DBpedia ontologies. We use precision, recall and F-measure for this ontology evaluation. A true positive case is an ontology rule that can be found in an extracted ontology and the associated ontology of DBpedia. A false positive case is an incorrectly identified ontology rule. A false negative case is a missed ontology rule. Table 1 shows the evaluation results of ontologies extracted from Wikipedia articles pertaining to *European Capital Cities* (Corpus E), *Office Buildings in Chicago* (Corpus O) and *Birds of the United States* (Corpus B) using hrLDA, KB-LDA, phrase_hLDA (tree depth $L = 3$), and LDA+GSHL in contrast to these

Semiconductor is a material characterized by its intermediate electrical property. A semiconductor material has an electrical conductivity value between a conductor, such as copper, and an insulator, such as glass. Semiconductors are the foundation of modern electronics.

An integrated circuit is a set of electronic circuits on one small plate chip of semiconductor material, normally silicon. Integrated circuits are used in virtually all electronic equipment today and have revolutionized the world of electronics.

Berlin is the capital city of Germany and one of the 16 states of Germany. With a population of 3.4 million people, Berlin is Germany's largest city, the second most populous city proper, and the seventh most populous urban area in the European Union.

London is the capital city of England and the United Kingdom. It is the most populous region, urban zone and metropolitan area in the United Kingdom. Standing on the River Thames, London has been a major settlement for two millennia.

(a) A toy corpus in domains: semiconductor, integrated circuit, Berlin, and London

Level 1, topic: material electrical london integrated urban

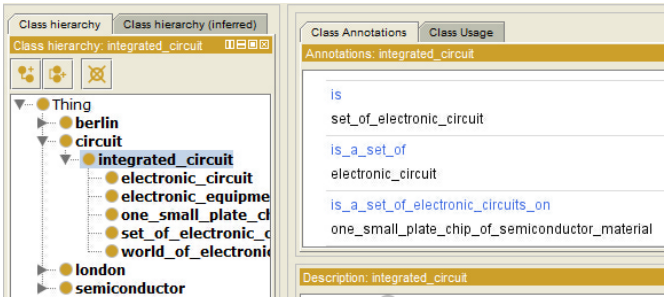
Level 2, topic: semiconductor berlin settlement thames river

Level 3, topic: populous germany millennia metropolitan union

Level 3, topic: electronic world today virtually silicon

Level 3, topic: modern foundation semiconductor insulator copper

(b) The topic tree obtained from hLDA; each node contains the top five words ordered by their probabilities of being in the corresponding topics



(c) The topic tree (left panel *class hierarchy*) with relations (right panel *class annotations*) obtained from hrLDA

Figure 9: Performance of hLDA and hrLDA on a toy corpus of diversified topics

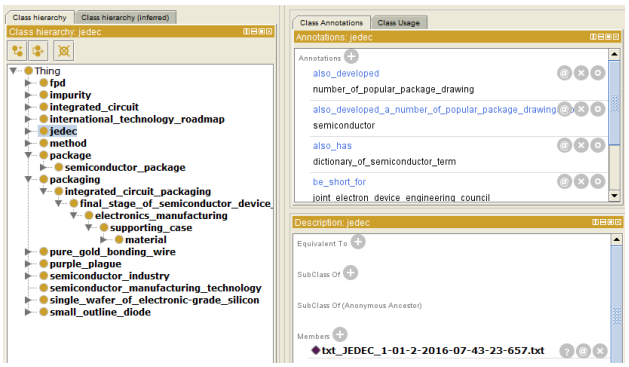


Figure 10: A 10-level semiconductor ontology that contains 2063 topics and 6084 relation triplets

Table 1: Precision, recall and F-measure (%)

Domain		Corpus E	Corpus O	Corpus B
Precision	hrLDA	**96.0**	**92.4**	**84.0**
	KB-LDA	90.7	89.9	79.4
	phrase_hLDA	27.6	27.4	24.5
	LDA+GSHL	52.4	19.8	28.6
Recall	hrLDA	**86.9**	74.7	**81.9**
	KB-LDA	83.8	**75.4**	63.3
	phrase_hLDA	50.6	57.5	36.5
	LDA+GSHL	20.0	73.1	11.8
F-measure	hrLDA	**91.2**	**82.6**	**82.9**
	KB-LDA	87.1	82.0	70.4
	phrase_hLDA	35.7	26.8	29.3
	LDA+GSHL	29.0	31.2	16.7

gold ontologies belonging to DBpedia. The three corpora used in this evaluation were collected from Wikipedia abstracts, the same text source of DBpedia. The seeds of hrLDA and the root concepts of LDA+GSHL are "*capital*," "*building*," and "*bird*." For both KB-LDA and phrase_hLDA we kept the top five tokens in each topic as each node of their topic trees is a distribution/list of phrases. hrLDA achieves the highest precision and F-measure scores in the three experiments compared to the other models. KB-LDA performs better than phrase_hLDA and LDA+GSHL, and phrase_hLDA performs similarly to LDA+GSHL. In general, hrLDA works well especially when the pre-knowledge already exists inside the corpora. Consider the following two statements taken from the corpus on *Birds of the United States* as an example. In order to use two short documents "*The Acadian Flycatcher is a small insect-eating bird.*" and "*The Pacific Loon is a medium-sized member of the loon.*" to infer that the Acadian Flycatcher and the Pacific Loon both belong to topic "bird," the pre-knowledge that "the loon is a species of bird" is required. This example explains why the accuracy of extracting ontologies from this kind of corpus is low.

5 Concluding Remarks

In this paper, we have proposed a completely unsupervised model, hrLDA, for terminological ontology learning. hrLDA is a domain-independent and self-learning model, which means it is very promising for learning ontologies in new domains and thus can save significant time and effort in ontology acquisition.

We have compared hrLDA with popular topic models to interpret how our algorithm learns meaningful hierarchies. By taking syntax and document structures into consideration, hrLDA is able to extract more descriptive topics. In addition, hrLDA eliminates the restrictions on the fixed topic tree depth and the limited number of topic paths. Furthermore, replacing random topic assignments with acquaintance topic assignments in ACRP allows hrLDA to create more reasonable topics and to converge faster in Gibbs sampling.

We have also compared hrLDA to several unsupervised ontology learning models and shown that hrLDA can learn applicable terminological ontologies from real world data. Although hrLDA cannot be applied directly in formal reasoning, it is efficient for building knowledge bases for information retrieval and simple question answering. Finally, one issue we have not addressed in our current study is capturing pre-knowledge. Although a direct solution would be adding the missing information to the data set, a more advanced approach would be to generate topic embeddings to extract hidden information.

Acknowledgments

This work was supported in part by Intel Corporation, Semiconductor Research Corporation (SRC). We are obliged to Professor Goce Trajcevski from Northwestern University for his insightful suggestions and discussions. This work was partly conducted using the Protege resource, which is supported by grant GM10331601 from the National Institute of General Medical Sciences of the United States National Institutes of Health.

References

[1] A. Ahmed, L. Hong, and A. J. Smola. Nested chinese restaurant franchise process: Applications to user tracking and document modeling. In *ICML (3)*, pages 1426–1434, 2013.

[2] A. Asuncion, M. Welling, P. Smyth, and Y. W. Teh. On smoothing and inference for topic models. *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 27–34, 2009.

[3] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Conference*, pages 722–735. Springer, 2007.

[4] D. M. Blei and P. I. Frazier. Distance dependent chinese restaurant processes. *The Journal of Machine Learning Research*, 12:2461–2488, 2011.

[5] D. M. Blei, T. L. Griffiths, and M. I. Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM)*, 57(2):7, 2010.

[6] D. M. Blei, T. L. Griffiths, M. I. Jordan, and J. B. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems 16*, page 17. MIT Press, 2004.

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JOURNAL of Machine Learning Research*, 3:993–1022, 2003.

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

[9] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka, and T. M. Mitchell. Toward an architecture for never-ending language learning. In *AAAI*, 2010.

[10] S. P. Chatzis. Inducing space dirichlet process mixture large-margin entity relationshipinference in knowledge bases. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1311–1320. ACM, 2015.

[11] F. Colace, M. De Santo, L. Greco, F. Amato, V. Moscato, and A. Picariello. Terminological ontology learning and population using latent dirichlet allocation. *Journal of Visual Languages & Computing*, 25(6):818–826, 2014.

[12] X. L. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. ACM, 2014.

[13] D. Downey, C. S. Bhagavatula, and A. Yates. Using natural language to integrate, evaluate, and optimize extracted knowledge bases. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 61–66. ACM, 2013.

[14] A. Gangopadhyay, M. Molek, Y. Yesha, M. Brady, and Y. Yesha. A methodology for ontology evaluation using topic models. In *Intelligent Networking and Collaborative Systems (INCoS), 2012 4th International Conference on*, pages 390–395. IEEE, 2012.

[15] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235, 2004.

[16] Z. Hu, G. Luo, M. Sachan, E. Xing, and Z. Nie. Grounding topic models with knowledge bases. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, 2016.

[17] Y. Jing, W. Junli, and Z. Xiaodong. An ontology term extracting method based on latent dirichlet allocation. In *Multimedia Information Networking and Security (MINES), 2012 Fourth International Conference on*, pages 366–369. IEEE, 2012.

[18] S. Krause, H. Li, H. Uszkoreit, and F. Xu. Large-scale learning of relation-extraction rules with distant supervision from the web. *The Semantic WebISWC 2012*, pages 263–278, 2012.

[19] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard., and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, 2014.

[20] Mausam, M. Schmitz, R. Bart, S. Soderland, and O. Etzioni. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 524–534, 2012.

[21] A. K. McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

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

[23] D. Movshovitz-Attias and W. W. Cohen. Kb-lda: Jointly learning a knowledge base of hierarchy, relations, and facts. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1449–1459, 2015.

[24] S. Mukherjee, J. Ajmera, and S. Joshi. Domain cartridge: Unsupervised framework for shallow domain ontology construction from corpus. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 929–938. ACM, 2014.

[25] D. Q. Nguyen, K. Sirts, L. Qu, and M. Johnson. Neighborhood mixture model for knowledge base completion. *arXiv preprint arXiv:1606.06461*, 2016.

[26] F. Niu, C. Zhang, C. Ré, and J. W. Shavlik. Deepdive: Web-scale knowledge-base construction using statistical learning and inference. *VLDS*, 12:25–28, 2012.

[27] I. Ocampo-Guzman, I. Lopez-Arevalo, and V. Sosa-Sosa. Data-driven approach for ontology learning. In *Electrical Engineering, Computing Science and Automatic Control, CCE, 2009 6th International Conference on*, pages 1–6. IEEE, 2009.

[28] J. Paisley, C. Wang, D. M. Blei, and M. I. Jordan. Nested hierarchical dirichlet processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):256–270, 2015.

[29] A. Slutsky, X. Hu, and Y. An. Tree labeled lda: A hierarchical model for web summaries. In *Big Data, 2013 IEEE International Conference on*, pages 134–140. IEEE, 2013.

[30] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706, 2007.

[31] M. Thelen and E. Riloff. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 214–221. Association for Computational Linguistics, 2002.

[32] W. Wei, P. Barnaghi, and A. Bargiela. Probabilistic topic models for learning terminological ontologies. *IEEE Transactions on Knowledge and Data Engineering*, 22(7):1028–1040, 2010.

[33] Z. Wei, J. Zhao, K. Liu, Z. Qi, Z. Sun, and G. Tian. Large-scale knowledge base completion: Inferring via grounding network sampling over selected instances. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, pages 1331–1340. ACM, 2015.

[34] A. Weichselbraun, G. Wohlgenannt, and A. Scharl. *Ontology learning and knowledge discovery using the web: challenges and recent advances*. IGI Global, 2011.