Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 159 (2019) 562–571

Procedia Computer Science

www.elsevier.com/locate/procedia

23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Modular Ontology Learning with Topic Modelling over Core Ontology

Ziwei Xu[a*], Mounira Harzallah[a], Fabrice Guillet[a], Ryutaro Ichise[b]

[a] *LS2N, Polytech Nantes - Ecole Polytechnique de l'Université de Nantes, Rue Christian Pauc, Nantes, 44300, France*
[b] *National Institute of Informatics, 2-1-2 Hitotsubashi, Tokyo, 101-8430, Japan*

Abstract

Nowadays, modular domain ontology, where each module represents a subdomain of the ontology domain, facilitates the reuse of information and provides users with domain-specific knowledge. In this paper, we focus on modular taxonomy learning from text, where each module collects terms with the same topic insights, and in parallel we manage to discover hypernym and 'related' relations among those collected terms. However, it is difficult to automatically fit terms into modules and discover relations. We propose to employ twice trained LDA to partition terms of each subdomain, and relate subdomains into modules of ontology. Meanwhile, we apply core concept replacement and subdomain knowledge supplementation as supportive information embedding technique over the corpus. This shows that the twice trained LDA strategy can effectively identify topic-relevant terms into subdomains, with nearly two-fold precision comparing to that of normal LDA training. The combination of core concept replacement and subdomain knowledge supplementation contributes to significant improvements in modular taxonomy learning.

Keywords: LDA; term partition; modular ontology; ontology learning; core ontology; knowledge supplementation

1. Introduction

Modular ontology has become an important issue to overcome the complexity of managing large domain knowledge [12]. To build a modular ontology, the domain core ontology based approaches are more prevalent. A core ontology of a domain is a basic and minimal ontology composed only of the minimal concepts (i.e core concepts) and the principal relations between them that allow defining the other concepts of the domain [16, 3]. A

[*] Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.
E-mail address: ziwei.xu@etu.univ-nantes.fr

10.1016/j.procs.2019.09.211

core concept, corresponding to the name of a subdomain of the ontology, is specialized to be extended with subconcepts, in order to define a taxonomy for each subdomain. Each taxonomy can be considered as a module of this ontology. However, modular ontology building is still complex, especially when the ontology domain is large. The increasing availability of massive data on the web makes ontology learning from texts become inevitable and meaningful. However, it shows that current works are remaining preliminary, so that further examination of this issue is still required.

We are interested in modular domain ontology learning from texts, derived from core ontology. More precisely, as the first step of modular ontology learning, we are focusing on grouping together prominent terms of each subdomain, using a clustering approach. In the literature, there are many terms clustering algorithm, however, they are faced with the difficulty of cluster label recognition with respect to subdomains of the corpus.

In this paper, we propose an approach to adapt the topic modeling method LDA (latent Dirichlet allocation)[2], by integrating domain core ontology in its model, in order to obtain term clusters, where each cluster is close to the insights of a subdomain. The approach is dedicated to modular ontology learning from a corpus of scientific papers, and a list of keywords is associated with each paper. Then hypernym relations and 'related' relations between terms of the same cluster can be extracted and added to build a modular ontology.

This paper is organized as follows. In section 2, we discuss the related work on term partition with LDA for ontology learning from texts and on modular ontology learning using core ontology. In section 3 we detail principles, hypothesis, and steps of our approach. In section 4, we present and discuss four experiments with different inputs and processing steps, applied on computer science paper corpus, and also the evaluation principles over partition results. Then it ends with the analysis of the results and the evaluation of our approach. Finally, we conclude and discuss future work.

2. Related Work

Topic modeling algorithms could participate in different phases of ontology learning. Kim et al. [11] relates a knowledge database to LDA, in order to capture the semantic relation between terms for the purpose of conceptualization. Yeh [24] connects terms with topics by their occurrence in documents, so as to enrich concept hierarchy based on topics. Furthermore, Wang [22] benefited from LDA to provide feature space of terms and use their similarity for relation discovery and concept taxonomy construction. Rani et al.[18] proposed to automatically learn taxonomy of ontology from text by using LSI (Latent Semantic Indexing), comparing to that by MapReduce-implemented LDA. On the other side, clustering approaches are widely learned in order to build ontology automatically. Louge et al.[13] implemented affinity propagation clustering algorithms[7] upon string similarity measurement for the construction and population of ontology. Moreover, Xu et al.[23] presented a term clustering framework for modular ontology construction, which examines various feature representations and their relative clustering algorithms, e.g. K-means and affinity propagation. Noticeably, little efforts have been made to apply LDA for term clustering aiming at concept identification, due to the existence of overlapping among topic clusters. However, once the dominant topic feature of terms could be selected with a certain purpose, LDA could be considered as a productive algorithm for term clustering as well.

Relation discovery relies heavily on characteristics of terms. Generally, identifying hypernym relation has been addressed with two main approaches: pattern-based and morphology-based methods. Pattern-based method [8] handcrafts several lexical-syntactic patterns to harvest is-a relation, which provides higher precision at the price of lower coverage. Conversely, morphology-based method supposes that the central noun of noun phrase contributes the majority of semantic insights, which holds the hypernym relation between them [17, 15]. This method exploits the meronymy of noun phrases with high recall but low precision. Traditionally, hypernym relation discovery over noun phrases could be taken to constitute the baseline structure of ontology learning, which is widely explored for concept hierarchy engineering [22, 18, 20]. Specifically, Snow et al.[21] does not rely on hand-crafted regular expression pattern to learn hypernym relation, instead, [21] introduces a general and formal pattern by using 'dependency path' to rediscover the hand-designed patterns. Additionally, the hypernyms can be expressed as semantic compression, Dey [5] applied hypernym replacement to unify the diverse words appearing in corpus. However, this technique is only restricted for the single term rather than noun phrases, because they simply use the predefined hypernym relation on Wordnet [14]. It exists a gap of hypernym replacement from single terms to noun phrases, especially with domain-specific consideration.

Ontology could be learned toward different subdomains. Besbes [1] defined different subdomains by their core concepts and developed taxonomic relation and conceptual relation between terms within partition. The subdomains can be directed by prior knowledge of core concept, conversely, the topic feature from documents can also lead to subdomain representation in the ontology. Mustapha et al. [15] proposed the topic ontology where topic and relation definitions are specified in advance, then each topic will go deep to relate entity inside that partition. However, this proposal only exploits topics to find the core concepts, but does not make use of topics to identify all terms inside subdomains. It results in difficulties to tackle the problem of scalability in subdomains of ontology.

3. Adaptation over LDA Algorithm

In topic modelling approach, through using topics attributes of terms for modular ontology learning, we frequently encountered the difficulties to match learned topics with sudomains of modular ontology correctly. This issue could be derived from the lack of prior knowledge during training. Learning ontology automatically is attractive, but not all terms from the corpus are meaningful for subdomains. This problem shows the demand of term identification with subdomain interests. In our approach, we aim to identify subdomain related terms and match topics of these terms to their subdomains, so as to construct each module by aggregating terms of the same domain. Meanwhile, we attempt to discover "is-a" and "related" relations during the learning process, for the purpose of modular taxonomic hierarchy construction.

To explore the semantic meaning of modular ontology, normally we would match the separated modules of a modular ontology with the divided subdomains of this ontology respectively. Latent Dirichlet Allocation strategy(LDA) is adopted to allocate topic features to terms in an automatic way[2]. However, it is incapable to train LDA with prior knowledge. In our approach, for ontology learning, we propose to to implant these principles by two techniques using prior knowledge. The fig. 1 depicts the framework of modular taxonomic hierarchy construction with the support of supportive information embedding techniques. This workflow starts by acquiring the prior knowledge from the corpus and applying them with two different supportive information embedding techniques, namely core concept replacement and subdomain knowledge supplementation, in the stage 1-1 and the stage 1-2 of fig. 1. These techniques insert prior knowledge to the corpus, which reconstitutes the new corpus. Based on the new corpus, twice trained LDA is proposed to satisfy different objectives(stage 2 in fig. 1): first training is to acquire the normal topic representation of terms for term identification purpose(stage 2-1); second training is to obtain the advanced topic representation of residual terms for the interests of subdomain(stage 2-4). Among them, term identification is executed to find topic significant terms in stage 2-2, which leads to the reconstituted corpus in stage 2-3. From those procedures, in stage 3, the obtained “is-a” relation and “related” relation would support modular taxonomic hierarchy construction corresponding to each subdomain.

3.1. The Pre-Processing of Data

The corpus contains separated plain text documents, where each document is associated with a core concept and a list of supportive terms. A core concept represents a subdomain of the modular ontology and the supportive terms are a set of significant keywords extracted from documents that are closely connected to this subdomain. In our experiments, the keywords in documents are considered as the supportive terms to its corresponding subdomain. In context, nouns (also noun phrases) refer to individuals or classes (types) of entities sharing some essential defining properties, while verbs usually serve as functions in sentences, from the terminology of Reichenbach [19]. The appearance of NPs and verbs provides central constituents of a sentence, where the semantic relation between the NPs of the subject and of the object is labelled by a verb. During the pre-processing steps, owing to the integrated parser spaCy [9], we would like to designate our data to only include NPs with the syntactic role as subject and as object, or include their co-occurred verbs in a sentence. In fig. 1, after the pre-processing box, those extracted terms will be stored into new separative documents similar to their original documents, namely "restricted corpus". Through this way, we are able to focus on syntactically

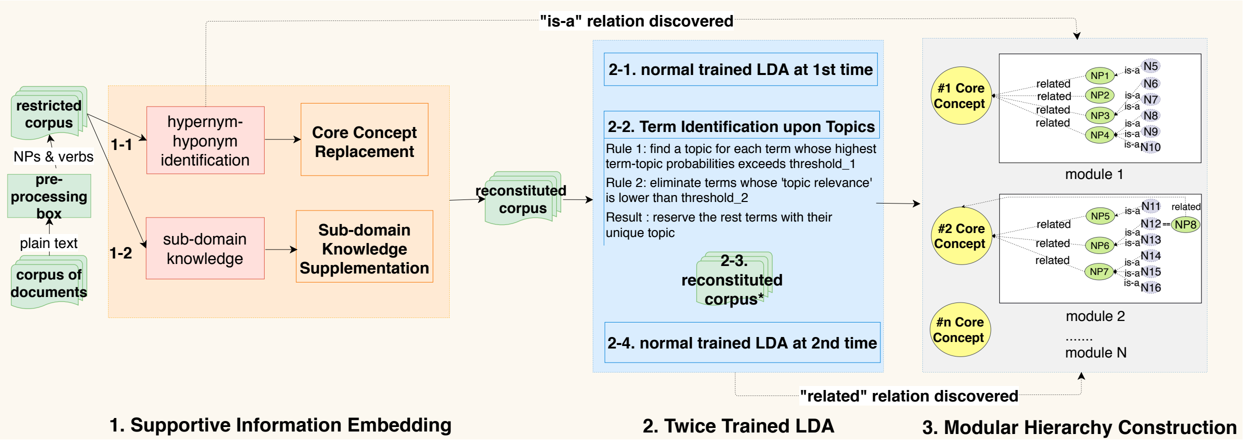
"is-a" relation discovered
restricted corpus
NPs & verbs
pre-processing box
plain text
corpus of documents
1-1
1-2
hypernym-hyponym identification
Core Concept Replacement
sub-domain knowledge
Sub-domain Knowledge Supplementation
reconstituted corpus
2-1. normal trained LDA at 1st time
2-2. Term Identification upon Topics
Rule 1: find a topic for each term whose highest term-topic probabilities exceeds threshold_1
Rule 2: eliminate terms whose 'topic relevance' is lower than threshold_2
Result : reserve the rest terms with their unique topic
2-3. reconstituted corpus*
2-4. normal trained LDA at 2nd time
#1 Core Concept
related
NP1
NP2
NP3
NP4
is-a
N5
N6
N7
N8
N9
N10
module 1
#2 Core Concept
NP5
NP6
NP7
NP8
N11
N12
N13
N14
N15
N16
module 2
....
module N
#n Core Concept
"related" relation discovered
1. Supportive Information Embedding
2. Twice Trained LDA
3. Modular Hierarchy Construction

Fig. 1. The framework of ontology Learning with prior knowledge.

significant NPs with the limited size of vocabulary. Since then the restricted corpus would play a major role as input for all techniques and training procedures.

3.2. Core Concept Replacement

The hypernyms possess the containment relation to their hyponyms, presenting much more general meaning than their descendants. Generally, the slight difference between hypernym and hyponym brings little impact to the topics or the subdomains that they belong to. Accordingly, the terms replacement by its hypernyms tends to be independent to the topic representation of terms [4].

Intuitively, with the assistance of hypernym replacement, the meaning of topics obtained from LDA tend to be close to hypernyms. This technique could be considered as prior knowledge embedding, among which the hypernym relation discovery is dominant but also tough. The terms substituting others play an important role on topic modeling, to serve the purpose of matching topics to subdomains (core concepts), we prefer those hypernyms who have the same names as subdomains(core concepts).

For instance, in a music domain, we can develop two subdomains from it: the ’music genre subdomain’ and the ’instrument subdomain’. In this way, the two associated core concepts of the music core ontology are ’music genre’ and ’instrument’. After the terms are replaced by the core concepts, the topics would have a higher tendency to correspond to the subdomains of corpus. Hence, the step of core concept replacement would support LDA training to provide a closer connection between topics and subdomains. With respect to the bag-of-word assumption, this technique brings many benefits in topic modeling:

- The size of vocabulary is dramatically reduced, which leads to the reduction of computational complexity.
- The frequency of hypernym terms has a great increase, such that their statistic importance is highlighted during training.
- The neighbor terms of original hyponyms (terms being replaced) could be captured easily because they become to co-occur with the more frequent hypernyms after replacement, so as to implicitly reinforce the connection between neighbor terms and the substitute hypernyms.

Therefore, it seems to be profitable to apply the core concept replacement technique upon corpus and feed the new corpus for topic modeling. Notably, not every term is acceptable to be replaced by core concepts, it will induce a semantic drift if the replaced terms are far from the original meaning. Therefore, the recognition of hypernym-hyponym pairs turns to be the key parts. Since the terms appeared in “keywords” are feasible to express the key meaning of the subdomain where this document belongs to, the terms appeared in “keywords” section are suitable to be selected as hyponyms. Accordingly, it is convincing to set the core concepts (the name of subdomains) as their hypernyms. Thereby the hypernym-hyponym pairs can be identified in the format as pairs of $< CoreConcept, Keyword >$, denoting as 'Approach2' in section 4. Concisely, the hypernym relation recognition is restricted to those terms occurred in keyword list, because of their significant presence in documents.

3.3. Subdomain Knowledge Supplementation

The technique of core concept replacement modifies the reconstituted corpus with supportive information, alternatively, there is an another manner to prepare corpus with less deliberate intervention, namely subdomain knowledge supplementation. In this approach, the supportive information will be appended at the tail of each document, which could extend the content with a higher occurrence of these appended words. Comparatively, the core concept replacement strategy tends to delicately adjust corpus to help topics get close to subdomains, while the subdomain knowledge supplementation strategy preserves original terms but add supportive information in order to increase the subdomain prominence of terms for topic modeling. In this context, the supportive information ought to convey the information of subdomains. For one practice of the supportive information, the appending knowledge should at least present with the most distinguishing and expressive terms to accentuate the dependence between documents and their subdomains(i.e. keywords in each document). As for another practice, to be straightforward, the appending terms could be specified by using core concepts. In overall, the supportive information can be the keywords list of documents or the reiterated core concepts with the length as the keywords list, corresponding to 'Approach3' and 'Approach4' of section 4 respectively.

3.4. Twice Trained LDA

Heretofore, corpus has been furnished with prior knowledge after the supportive information embedding techniques. From the re-constituted corpus, topic modelling is able to represent terms by topic features. We plan to discover the relation between the interesting terms and the subdomains of corpus, based on terms feature representations. It is noteworthy that not all terms are interesting to be engaged in the partitions of domains. For instance, if some terms are not significantly related to a certain topic, the involvement of these terms will bring some fuzzy meanings into topic partitions, and even gives rise to semantic drift of topics.

To address the problem, we propose to employ twice trained topic modeling of LDA, as an approach to sorely concentrate on the topic-significant terms. From the first training of LDA, we can obtain the term probabilities of topics and term significance of topics from the corpus. These information can be used as indicators to identify topic-significant terms. Then the residual terms will be kept for the second training of LDA. To present the details, we provide an example of term identification upon topic(see fig. 2). To be simple, 4 terms are represented with 5 topics as the result of first normal trained LDA. After the normal trained LDA, we can obtain the term probabilities of topics, denoted as $P(w|t)$, and the term significance of topics, denoted as $\frac{P(w|t)}{P(w)}$. In step 2-2, specifically, two thresholds (shown in two diamonds) are applied to reject the topic-insignificant terms in each partition:

- In the first diamond of step 2-2, in order to find the dominant topics, the highest value of term probabilities of topics $P(w|t)$ are highlighted for each term. If this value exceeds threshold1, terms will be assigned with corresponding topics and be preserved for the next step. Otherwise, this terms will be rejected.
- In the second diamond of step 2-2, for all the remaining terms after threshold 1, if term significance of topics $\frac{P(w|t)}{P(w)}$, is higher than threshold 2, the assigned topics for terms will be kept, unless the terms will be eliminated.
- In the bottom box of step 2-2, the rest of the terms are stored to prepare for a newly reconstituted corpus only with the remaining terms.

Before the second time of topic modelling, it is desirable to clean out those topic-insignificant terms in the corpus. To simplify the training procedures in this stage, we determine to reconstitute corpus with only the identified topic-relevant terms. In the stage 2-3 of fig. 2, the topic-insignificant terms are deleted from previous reconstituted corpus, which turns to be a new reconstituted corpus. Starting from the new one, topic modelling is carried out to provide the ultimate topic features for terms. In brief, the twice trained topic modelling strategy gets rid of the influence of topic-irrelevant terms and pays attention to those topic-meaningful terms. We

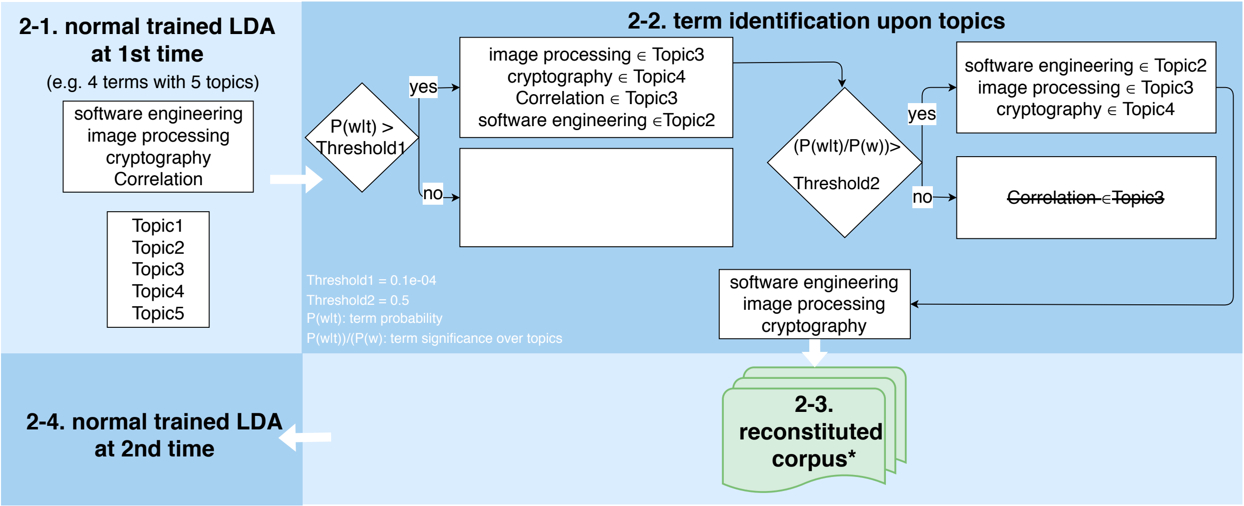
2-1. normal trained LDA at 1st time
(e.g. 4 terms with 5 topics)
software engineering
image processing
cryptography
Correlation
Topic1
Topic2
Topic3
Topic4
Topic5
2-2. term identification upon topics
P(w|t) > Threshold1
yes
no
image processing ∈ Topic3
cryptography ∈ Topic4
Correlation ∈ Topic3
software engineering ∈Topic2
(P(w|t)/P(w))> Threshold2
yes
no
software engineering ∈ Topic2
image processing ∈ Topic3
cryptography ∈ Topic4
~~Correlation ∈ Topic3~~
Threshold1 = 0.1e-04
Threshold2 = 0.5
P(w|t): term probability
P(w|t))/(P(w): term significance over topics
software engineering
image processing
cryptography
2-3. reconstituted corpus*
2-4. normal trained LDA at 2nd time

Fig. 2. An example of term identification upon topic.

expect that the resulting terms would possess more significant topic features, which implies a stronger “related” relation to the core concept of each subdomains for modular taxonomic hierarchy construction.

4. Experiments

4.1. Experiment Preparation

The original plain text corpus contains 6,514 documents, in which each document records the abstract of an academic paper in computer science domain. In this corpus, 11 subdomains are considered and the corresponding core concepts are : "computer graphics", "machine learning", "network security", "cryptography", "operating systems", "software engineering", "distributed computing", "algorithm design", "computer programming", "data structures" and "bioinformatics". Each document also provides its related prior knowledge, which contains the keywords list (including around 11 sets of keywords) and the name of their subdomains (corresponding to 11 core concepts). The keywords list will support the identification of hypernym-hyponym pairs $< CoreConcept, Keyword >$ in the following sections.

For the convenience of evaluation, it is required to build the Gold Standard for the selected corpus, we manually identify terms which are similar or hyponyms of Keywords. We have 791 Keywords as prior knowledge, and this step provides 1,861 pairs of $\langle CoreConcept, NPs \rangle$ as Gold Standard. To serve as a baseline, in table 1, we build the reconstituted corpus ‘Data1’ only containing NPs that exists in Gold Standard, denotes as NP*, and another reconstituted corpus ‘Data2’ containing the same NP* and their co-occurred verbs as well, denotes as NP*+verbs*. To serve for terms identification, we provide the reconstituted corpus ‘Data3’ with complete NPs from documents and ‘Data4’ with complete NPs and their co-occurred verbs as well. To ensure that there are enough elements in each document, the filtering of document is required, in which each reconstituted document contains at least 2 NPs in Data1 and Data2 and contains at least 10 NPs in Data3 and Data4.

In section 3.2, the hypernym-hyponym pairs can be expressed as pairs of $\langle CoreConcept, Keyword \rangle$. Those relations would be employed in the core concept replacement strategy. This procedure is represented as ‘Approach2’ in table 1. For the strategy of subdomain knowledge supplementation, as we talked in section 3.3, two choices are provided to be supplemented as supportive information for each document. Concisely, in table 1, ‘Approach3’ appends each document with their corresponding keywords and ‘Approach4’ appends each document with its core concept (name of their subdomains). After the preparation with supportive information embedding techniques, the modified corpus will be fed into topic modelling learning.

		Data1(NPs*)	Data2(NPs*+verbs*)	Data3(NPs)	Data4(NPs+verbs)
corpus	# files	1,709 files		6,514 files	
	# occurrence	2,345	4,690	118,251	236,502
	# unique terms	1,850	1,906	53,677	55,929
Approach1: file selection with sufficient amount of NPs	# files	39 files		5,801 files	
	# occurrence	139	278	113,244	226,488
	# unique terms	112	182	51,756	53,971
Approach2: core concept replacement upon Approach1	# occurrence after core concept replacement	139	278	113,244	226,488
	# unique terms after core concept replacement	109	179	51,394	53,609
Approach3: keywords supplementation upon Approach1	# occurrence after keywords supplementation	223	362	118,977	232,221
	# unique terms after keywords supplementation	173	243	52,077	54,290
Approach4: core concept replacement and supplementation upon Approach1	# occurrence after both techniques	223	362	118,977	232,221
	# unique terms after both techniques	120	189	51,394	53,541

Table 1. The statistic of the reconstituted corpus applied with 4 different approaches. NPs appeared in Gold Standard denote by NPs*, and their co-occurred verbs denote by verbs*. Data3 and Data4 contain all NPs as subject or object of sentences, while Data1 and Data2 extract only NPs* from Data3 and Data4.

4.2. Experiment Setting

We have discussed the criteria to find a dominant topic for each term with twice trained LDA in section 3.4. However, the task of controlling the size of topic partitions is a challenging and critical work. For a large topic partition, it leads to many difficulties to match a big partition into several small subdomains if the two sides are not in the same size. On the contrary, if the topic partitions could be learned into small size, it is easier to aggregate several related topic partitions into one subdomain, where each partition could represent as a sub-part of that subdomain. In fact, it is profitable to train LDA with a large number of topics because this result will provide more fine-grained features of terms. For this reason, we set the number of topics to 50 which is much larger than the number of subdomains. In table 2, we present the hyper-parameters of LDA and also the two thresholds to identify terms into different topic partitions in step 2-2. The setting of these thresholds indicates that the term probabilities of one topic should exceed 0.0001 (Threshold1) and the term significance of this topic should more than half (Threshold2). The previous setting regards to our practice of multiple experiments. Actually, empty partitions will exist if there are not enough features of terms for LDA training. In table 3, it shows that smaller data sets (i.e. Data1 and Data2) have less number of topics than others. From the comparison between Data3 and Data3* or between Data4 and Data4*, we can notice a dramatic decrease of the size of partitions, due to usage of the terms identification strategy in twice trained LDA. According to our setting, there are more topic partitions than subdomains. To tackle this unbalanced problem, we consider it as an n-to-1 labeling issue, in which the multiple topic partitions would match to one label. Here the label is regarded as the name of subdomain or core concepts.

First of all, we need to manage how to choose the label for topic partitions. Each partition consists of many labeled and unlabeled terms, where the labeled terms can be recognized with Gold Standard. Based on those labeled terms, we can vote for one dominant label from the multiple existing labels in one partition. The dominant label is supposed to be the most frequent label in a partition, this procedure also names as majority voting [23]. At this point, each topic partition could be assigned with a dominant label. However, for those topic partitions who do not possess any labeled terms, their insights seem to be far from any label-related terms. In this situation, it is beneficial to directly reject those topic partitions, ensuring that only core concepts related partitions are under consideration in the following procedures.

Then, we are faced with the label prediction of the unknown terms in partitions. After the label assignment of topic partitions, the entire terms in a partition are allocated with the same dominant label, including the label-unknown terms. This approach helps for label prediction of the unknown terms. To ensure a high-quality label prediction, we halve the corpus into two subsets randomly. The first half set offers the predicted labels for unknown terms, while the latter half set offers another prediction. Generally,the experiment will be repeated 10 times. If the labeled terms are always the same, it implies a good label prediction. Once they are different, the prediction will be rejected. Noticeably, this prediction could not be evaluated by Gold Standard, only human evaluation can contribute to this situation.

After the majority voting for labels, according to the assigned label, each partition can be exactly re-aggregated into the pre-defined subdomains of the corpus, where each label (core concept) represents a subdomain. Therefore the topic partitions could be merged into subdomains successfully.

Table 2. The parameter of learning.

LDA	k	50
	alpha	0.02
	beta	$\frac{k}{\#uniqueterms}$
	iteration	50
	filtering rate	0
term identification threshold	Threshold1	0.1e-04
	Threshold2	0.5

Table 3. The size of partitions for different data. * denotes the output from twice trained LDA

	# topics	avg terms	max terms	min terms
Data1	28	4	7	1
Data2	27	6	13	3
Data3	50	1,041	20,603	194
Data4	44	1,232	23,427	359
Data3*	50	104	963	59
Data4*	50	73	1.011	31

4.3. Experiment Results

Heretofore, we proposed several supportive information embedding techniques and different corpus reconstitution approaches. In order to measure their effectiveness of modular ontology learning, we designed many comparative experiments and adopted two main metrics to evaluate their outcomes, including precision and adjusted rand index. The precision presents intuitively the proportion of true labeled terms among all terms in the partition, where the higher the value, the partitions are more similar to Gold Standard. In parallel, the adjusted rand index [10] measures the agreement between two partitions. Specifically, it examines the performance by matching the term partitions to the classification of Gold Standard. When the resulted partitions agree perfectly with the partition of Gold Standard, the value will reach 1, otherwise it is supposed to be 0. Comparing to precision, the adjusted rand index is capable to measure the agreement of partitions even when they have the different number of partitions. To be convinced, each experiment is supposed to be trained for 10 times to provide the average score of metrics.

The evaluation results of all proposed experiments are shown in table 4. From the right side of table 4, it is obvious that the value arises dramatically from ‘normal LDA training’ to ‘twice trained LDA’ strategy. Due to the elimination of irrelevant terms in the right-hand strategy, terms are partitioned in a more accurate manner than that of normal LDA training. Please note that, for the subsequent comparisons, the two columns of twice trained LDA would be considered as indicators. In terms of types of the corpus, Data1 and Data3 only contain NPs where Data2 and Data4 contain NPs and verbs as well. Noticeably, it exists a great increase from Data3 to Data4, but not for Data1 and Data2, which indicates that the twice trained LDA strategy prefer the varied corpus containing NPs and verbs, rather than the pure corpus containing only NPs.

In the vertical direction of table 4, we notice a general decrease from Data1 to Data3 but a general increase from Data2 to Data4. The results are mixed because Data3 and Data4 contain a rather larger amount of terms than that in Data1 and Data2. We cannot conclude that it exists the relation between the size of corpus and term partition performance. However, comparing the metric scores between Data1 and Data4 in the row of Approach4, we observed the prominent scores in Data4. It is acceptable to indicate that our proposal can reach a rather higher term partition performance, even including many label-unknown terms.

In the horizontal direction of table 4, the difference from Approach1 to Approach2 is the usage of core concept replacement technique, and the variance from Approach1 to Approach3 and Approach4 is owing to the subdomain knowledge supplementation technique. However, only the second comparison could be observed with the

	Metrics of Evaluation	Data1	Data2	normal trained LDA		twice trained LDA	
				Data3	Data4	Data3	Data4
Approach1	Adjusted Rand Index	48.50%	45.97%	02.25%	01.31%	25.86%	66.12%
	Precision	77.54%	76.47%	31.78%	31.25%	69.06%	92.05%
Approach2	Adjusted Rand Index	47.08%	**50.16%**	01.83%	01.37%	32.96%	65.19%
	Precision	77.50%	**79.11%**	32.12%	32.17%	72.41%	92.20%
Approach3	Adjusted Rand Index	47.17%	46.65%	**02.28%**	**01.60%**	25.59%	56.42%
	Precision	76.49%	77.07%	32.83%	**33.78%**	68.99%	86.96%
Approach4	Adjusted Rand Index	**51.13%**	49.76%	01.95%	01.42%	**33.69%**	**75.50%**
	Precision	**78.91%**	78.84%	**33.00%**	32.14%	**75.64%**	**94.31%**

Table 4. The evaluation results of 4 different data and 4 approaches.

	Our proposal	**Wang et. al.** [22]	**Besbes et al.** [1]	**Mustapha et al.** [15]
Size of Corpus	6,514 documents	4,660 web pages	10 documents	10 web pages
Extraction of Corpus	NPs as subj./obj.;verbs	Nouns;Verbs;Adjectives	-	-
Domain	Computer Science	Semantic Web	multi-domains	multi-domains
Learning methods	twice trained LDA	Similarity Hierarchy Learning	Case-based reasoning	Attributed Typed Graph Model [6]
Number of Topics	50	30-90	-	-
Types of Ontology	modular ontology	terminological ontology	modular ontology	modular ontology
Highest precision	**94.3%**	87.5%	53.6%	70%

Table 5. The comparison of other ontology learning models.

significant increase, which implies that the subdomain knowledge supplementation technique is dominant for a better performance than that of core concept replacement. During the comparison between the two different approaches of subdomain knowledge supplementation, we notice a more dramatic increase from Approach1 to Approach4, than that of Approach1 to Approach3. It proves that it is better to directly supplement subdomain knowledge with core concepts(name of subdomains), rather than using keywords. In general, it is noteworthy that even though there is no significant increase from Approach1 to Approach2 with core concept replacement technique, but it has a substantial increase from Approach2 to Approach4 with the same technique. It shows that the combination of subdomain knowledge supplementation and core concept replacement gives the best performance of term partitions than any single technique.

Through comparison to other ontology learning models (see table 5), our proposal outperforms other learning algorithms with a larger corpus, according to the highest precision. As for the extraction procedure over corpus, Wang [22] has the similar pre-processing steps with us. In his work, the corpus is re-constituted with only specific terms, i.e. nouns, verbs and adjectives. The result presents a better precision than that of other models without specific emphasis over the corpus. It shows that the re-constituted corpus brings little influence to learning approaches, but even assist to increase precision somehow. This advantage owes to the decreased size of vocabulary, which gets rid of the negative impacts of redundant terms. Intuitively, it is easier to work on multi-domains rather than uni-domains for modular ontology learning. However, the research of Besbes [1] and Mustapha [15] indicates that the precision of modular ontology learning is comparatively lower than the learning based on uni-domains.

5. Conclusion

We propose to employ twice trained LDA to identify terms into subdomains, within which terms are interconnected with "related" relations. In parallel, we apply supportive information embedding techniques to LDA training for hypernym relation discovery. Subsequently, the partitioned subdomains are associated into modules of ontology, inside each module, the discovered relations will be used to construct a modular taxonomic hierarchy. These procedures contribute to modular ontology building jointly. It proves that the twice trained LDA strategy can effectively identify terms into partitions, with around twice increase in precision than that of normal

LDA training. The combination of core concept replacement technique and subdomain knowledge supplementation contributes the significant improvement in modular taxonomic hierarchy construction of subdomains.

In the future, we would like to identify the ad-hoc relation inside each module and find some links between subdomains to organize a more sophisticated and interrelated ontology. To enrich the relations inside a module, we can make use of the syntactic features between nouns and verbs or the word embedding features from context, to discover the inner relations between those extracted terms. To tidy the relation between modules, we will only concentrate to discover the hypernym relation among terms that are belonging to different modules. We can benefit from the existing hierarchy of wordNet [14] or apply syntactic pattern recognition to find the hypernym relation among terms in different subdomains.

References

[1] Besbes, G., Baazaoui-Zghal, H., 2015. Modular ontologies and cbr-based hybrid system for web information retrieval. Multimedia Tools and Applications 74, 8053–8077.

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

[3] Burita, L., Gardavsky, P., Vejlupek, T., 2012. K-gate ontology driven knowledge based system for decision support. Journal of Systems Integration 3, 19–31.

[4] d'Aquin, M., Schlicht, A., Stuckenschmidt, H., Sabou, M., 2009. Criteria and evaluation for ontology modularization techniques. Modular ontologies 5445, 67–89.

[5] Dey, K., Shrivastava, R., Kaushik, S., 2016. A paraphrase and semantic similarity detection system for user generated short-text content on microblogs, pp. 2880–2890.

[6] Ehrig, H., Ehrig, K., Prange, U., Taentzer, G., 2006. Fundamental theory for typed attributed graphs and graph transformation based on adhesive hlr categories. Fundamenta Informaticae 74, 31–61.

[7] Frey, B.J., Dueck, D., 2007. Clustering by passing messages between data points. science 315, 972–976.

[8] Hearst, M.A., 1992. Automatic acquisition of hyponyms from large text corpora, Association for Computational Linguistics. pp. 539–545.

[9] Honnibal, M., Johnson, M., 2015. An improved non-monotonic transition system for dependency parsing. Proceedings of the Conference on Empirical Methods in Natural Language Processing , 1373–1378.

[10] Hubert, L., Arabie, P., 1985. Comparing partitions. Journal of classification 2, 193–218.

[11] Kim, D., Wang, H., Oh, A., 2013. Context-dependent conceptualization, AAAI Press. pp. 2654–2661.

[12] Kutz, O., Hois, J., 2012. Modularity in ontologies. Applied Ontology 7, 109–112.

[13] Louge, T., Karray, M.H., Archimède, B., 2018. Investigating a method for automatic construction and population of ontologies for services: Performances and limitations, in: 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA), IEEE. pp. 1–6.

[14] Miller, G.A., 1995. Wordnet: a lexical database for english. Communications of the ACM 38, 39–41.

[15] Mustapha, N.B., Aufaure, M.A., Zghal, H.B., Ghezala, H.B., 2012. Modular ontological warehouse for adaptative information search, Springer. pp. 79–90.

[16] Oberle, D., Lamparter, S., Grimm, S., Vrandečić, D., Staab, S., Gangemi, A., 2006. Towards ontologies for formalizing modularization and communication in large software systems. Applied Ontology 1, 163–202.

[17] Payne, J.R., 2006. Noun phrases. Encyclopedia of Language and Linguistics , 712–720.

[18] Rani, M., Dhar, A.K., Vyas, O., 2017. Semi-automatic terminology ontology learning based on topic modeling. Engineering Applications of Artificial Intelligence 63, 108–125.

[19] Reichenbach, H., 1947. Elements of Symbolic Logic. London: Dover Publications.

[20] Rios-Alvarado, A.B., Lopez-Arevalo, I., Sosa-Sosa, V.J., 2013. Learning concept hierarchies from textual resources for ontologies construction. Expert Systems with Applications 40, 5907–5915.

[21] Snow, R., Jurafsky, D., Ng, A.Y., 2005. Learning syntactic patterns for automatic hypernym discovery, in: Advances in neural information processing systems, pp. 1297–1304.

[22] Wang, W., Barnaghi, P.M., Bargiela, A., 2011. Learning skos relations for terminological ontologies from text, IGI Global. pp. 129–152.

[23] XU, Z., Harzallah, M., Guillet, F., 2018. Comparing of term clustering frameworks for modular ontology learning, SCITEPRESS - Science and Technology Publications, Seville, Spain. pp. 128–135.

[24] Yeh, J.h., Yang, N., 2008. Ontology construction based on latent topic extraction in a digital library, Springer. pp. 93–103.