



Automatic ontology construction from text: a review from shallow to deep learning trend

Fatima N. Al-Aswadi^{1,2} · Huah Yong Chan¹ · Keng Hoon Gan¹

Published online: 8 November 2019
© Springer Nature B.V. 2019

Abstract

The explosive growth of textual data on the web coupled with the increase on demand for ontologies to promote the semantic web, have made the automatic ontology construction from the text a very promising research area. Ontology learning (OL) from text is a process that aims to (semi-) automatically extract and represent the knowledge from text in machine-readable form. Ontology is considered one of the main cornerstones of representing the knowledge in a more meaningful way on the semantic web. Usage of ontologies has proven to be beneficial and efficient in different applications (e.g. information retrieval, information extraction, and question answering). Nevertheless, manually construction of ontologies is time-consuming as well extremely laborious and costly process. In recent years, many approaches and systems that try to automate the construction of ontologies have been developed. This paper reviews various approaches, systems, and challenges of automatic ontology construction from the text. In addition, it also discusses ways the ontology construction process could be enhanced in the future by presenting techniques from shallow learning to deep learning (DL).

Keywords Concept classification · Deep learning · Ontology construction · Ontology learning · Semantic relation

1 Introduction

Ontology construction from text is a procedure that involves analysing the collected text for a specific domain; identifying the relevant terms, concepts and their relationships; mapping and representing the ontology by representation language [e.g. OWL (Web Ontology

✉ Fatima N. Al-Aswadi
fatima7aswadi@gmail.com; fnsa15_com016@student.usm.my

Huah Yong Chan
hychan@usm.my

Keng Hoon Gan
khgan@usm.my

¹ School of Computer Sciences, Universiti Sains Malaysia, 11800 Gelugor, Pulau Pinang, Malaysia

² Faculty of Computer Sciences and Engineering, Hodeidah University, Hodeidah, Yemen

Language), RDF (Resource Description Framework), or RDFS (Resource Description Framework Schema)]; and finally evaluating the constructed ontology. The procedure of ontology construction, in general, can be done in one of three ways: manual construction; cooperative construction (need the human intervention during the ontology constructing process) and (semi-) automatic construction. Ontology Learning (OL) from text is a process of (semi-) automatic ontology construction from text.

In recent years, many OL approaches and systems that try to automate the construction of ontologies have emerged. OL systems change the way of processing textual data from text mining to knowledge mining. This knowledge has to be represented in the form of concepts and relationships between those concepts (ontologies) to be in machine-understandable form.

The purpose of this paper is to give a review in ontology construction approaches, challenges, systems, and explain the importance of proceeding towards DL rather than shallow learning. In addition, it discusses and illustrates how to enhance the process of ontology construction from the text by using DL. The rest of this paper is organized as follows: Sect. 2 defines ontology construction and its challenges. Section 3 presents the ontology construction layers and process while Sect. 4 presents the ontology construction approaches. Section 5 presents the ontology construction systems and a comparison among them as well as the evaluation metrics of the ontology construction systems. Section 6 explains and discusses the power and importance of DL for ontology construction. Section 7 presents and explains how to use the DL for the ontology construction process. Finally, we conclude in Sect. 8.

2 Ontology construction

2.1 Formal definition

2.1.1 Ontology

According to W3C (World Wide Web Consortium) “*Ontologies define the terms used to describe and represent an area of Knowledge*”. Ontology is a data model that represents a set of concepts and the relationships among those concepts within a domain (Mishra and Jain 2015). Zouaq (2011) defined the components of an ontology by the following tuple:

$$O = \langle C, H, R, A \rangle$$

where O represents ontology, C represents a set of classes (concepts), H represents a set of hierarchical links between the concepts (taxonomic relations), R represents the set of conceptual links (non-taxonomic relations), and A represents the set of rules and axioms.

2.1.2 Ontology construction

Ontology construction can be defined as an iterative process of creating an ontology from scratch or reusing an existing ontology for enriching or populating. The process of constructing the ontology includes six tasks as follows:

1. Specifying the domain to create well-defined terms and concepts.
2. Identifying the key terms, concepts, and their relations in the domain.

3. Establishing or inferring the rules and axioms that describe the structural properties of the domain.
4. Encoding (representing) the constructed ontologies by using representation languages which support the ontology such as RDF, RDFS or OWL.
5. Combining the constructed ontologies with existing ontologies (if available).
6. Evaluating the constructed ontologies by using generic and specific evaluation metrics.

The numbering of these tasks does not indicate the order of execution of them, and some of these different tasks may work iteratively to be accomplished. Ontology construction process may be done by one of the following three ways:

1. Manual construction: experts perform manual construction of ontology
2. Cooperative construction: most or all tasks of the ontology construction system are supervised by experts.
3. (Semi-) Automatic construction: the ontology construction process is performed automatically with limited intervention by users or experts. Automatic construction means that the level of human intervention is slightly less than semi-automatic construction but does not mean fully automatic construction. It is worth mentioning that full automatic construction for ontology by a system is still a significant challenge and it is not likely to be possible (Maimon and Browarnik 2015; Wong et al. 2012).

2.2 Challenges in ontology construction

Following the closer look into many ontology construction studies, it is clear that there is a consensus among several aspects of ontology construction challenges that require more efforts. The following list presents the common aspects that define the main challenges of automatic ontology construction:

1. Fully automatic construction for ontologies could be not possible, but there is an acute need for more effort to decrease human intervention in the ontology construction process to build (semi-) automatic construction rather than the existing cooperative systems of ontology construction (Buitelaar et al. 2005; Gómez-Pérez and Manzano-Macho 2003; Maimon and Browarnik 2015; Mishra and Jain 2015; Wong et al. 2012; Zhou 2007).
2. The need to avoid the noise terms (irrelevant or very general) that lead to unnecessary additional efforts required for pre-processing. This could be addressed by paying more attention to filter terms in the ontology construction process as early as possible, i.e. in the early stage of construction (Wong et al. 2012; Zhou 2007).
3. Discovering the relations between concepts still unsatisfactory in terms of its result and more efforts are needed in this aspect. (Albukhitan et al. 2017; Buitelaar et al. 2005; Maimon and Browarnik 2015; Wong et al. 2012; Zhou 2007).
4. Learning axiom is still in the initial stage on existing ontology construction systems and requires plenty of work (Maimon and Browarnik 2015; Mishra and Jain 2015; Wong et al. 2012; Zhou 2007).
5. The transformation of data whether it is from small static text collections, or from massive heterogeneous on the World Wide Web, should be taken into account when designing ontology construction system (Buitelaar et al. 2005; Mishra and Jain 2015; Wong et al. 2012; Zhou 2007).

6. Building a standard platform to evaluate ontology construction systems is still a hard task (Albukhitan et al. 2017; Buitelaar et al. 2005; Wong et al. 2012; Zhou 2007).

The validity of these aspects and challenges will be evident from further discussion on the approaches and prominent systems of ontology construction in the later sections.

3 Ontology construction via learning

OL is a process of automatically or semi-automatically creating of new ontology or reusing (for enriching or populating) an existing ontology; with the minimum exertion of a human (Gillani Andleeb 2015). OL from text is a process of acquiring knowledge from text by applying a set of methods and techniques from various fields such as natural language processing (NLP), data mining, and machine learning for extracting ontological elements (Maimon and Browarnik 2015) and then constructing ontologies.

Ontology Learning Layer Cake is proposed by Buitelaar et al. (2005). This approach is a dominant approach and it is considered the cornerstone in OL (Maimon and Browarnik 2015). According to Buitelaar et al. (2005), there are six layers in OL, they are: *Terms*, *Synonyms*, *Concepts*, *Concepts Hierarchies*, *Relations*, and *Rules*. Based on these layers, the process of OL can be divided into six sub-tasks as follows:

- *Term extraction* is a prerequisite for all aspects of OL from text. A term is multi-word or single-word token, which denotes a specific meaning in a given domain.
- *Synonym discovery* aims to find the terms that indicate the same concept and appear in the same set for a selected concept. A synonym set is the most common type lexical relations. It is possible to use a readily available set such as WordNet *synsets*, clustering techniques or any other similar methods (Maimon and Browarnik 2015).
- *Concept formation* is considered as an unclear task because there is no consensual definition of what a concept is. The general definition of a concept is that it should include the intention of the concept, the extension of the concepts, and the lexical signs (terms) which are used to refer to these concepts.
- *Concept Hierarchy* is the backbone of an OL that includes inducing, extending and refining the concepts. This task aims to build the hierarchical taxonomy of concepts.
- *Relation discovery* aims to extract a novel relationship between known concepts. This task still an open problem (Gillani Andleeb 2015; Maimon and Browarnik 2015; Mishra and Jain 2015; Wong et al. 2012). There are a few approaches that address the relation extraction issue for OL from text such as the association rule extraction algorithm based on a sentence, that proposed in Maedche and Staab (2000).
- *Rule or axiom extraction* is the final sub-task in the OL process. It aims to infer the rules based on extracted concepts and relations. This task is still in the initial stage and needs more efforts (Gillani Andleeb 2015; Maimon and Browarnik 2015; Mishra and Jain 2015; Wong et al. 2012; Zhou 2007). There are very little attempts to generate the rules and axioms in existing OL systems. So far, logic-based approaches can be used for this task such as in (Fleischhacker and Völker 2011; Oliveira et al. 2001).

4 Ontology learning approaches

Approaches of OL from text can be divided into two categories: linguistics-based approaches and machine learning (ML) approaches (statistic-based and logic-based approaches). The following sub-sections present the details of these approaches.

4.1 Linguistics-based approaches

- *Pattern-based extraction* (Morin 1999): It is used to recognize the relations by matching a pattern from a sequence of words in the text. Lexico-syntactic patterns and semantic templates are techniques under this approach. Lexico-syntactic patterns technique uses any defined patterns such as “NP is type of NP” to extract hypernym and meronym relations. Semantic templates are similar to lexico-syntactic patterns technique but with more detailed rules and conditions and also has been used to extract non-taxonomic relations. It is well known that these approaches have reasonable precision, but they have very low recall (Maimon and Browarnik 2015; Wong et al. 2012).
- *POS tagging and sentence parsing* (Abney 1997): It is considered a rule-based approach. Part-Of-Speech (POS) tagging is used to assign parts of speech to each word in the text, such as noun, verb, adjective, etc., while the sentence parser is used to recover the complete and exact parses for each sentence in the text. However, many words are ambiguous (e.g. in English, the word “plant” may be a noun or a verb), so certain parsers are built on statistical-based parsing such as the Stanford Parser (Klein and Manning 2003). Statistical Parsing based on the probability of certain tag occurring, given various possibilities. This approach is used for term extraction.
- *Syntactic structure analysis and dependency structure analysis* (Gamallo et al. 2002; Nivre 2004): It is used to uncover syntactic and dependency of terms and relations at the sentence level. Syntactic structure analyses the words and modifiers in syntactic structures such as noun phrases and/or verb phrases to discover potential terms and relations while ignoring other phrases. As for dependency structure analysis, it uses grammatical relations (e.g. subject, object, and complement) to determine more complex relations. However, these approaches may be inadequate, they need to cooperate with other algorithms and/or rules for better performance (Mudhsh et al. 2015). This approach is useful for term and concept extraction, and also for relation discovery. For example, concepts can be extracted based on terms dependency within a noun phrase while relations can be extracted based on terms dependency within a verb phrase.

4.2 Machine learning approaches

ML approaches can be divided into two kinds: statistical-based approaches and logic-based approaches.

4.2.1 Statistic-based approaches

- *Co-occurrence analysis* (Budanitsky 1999): It is used to identify lexical units that tend to occur together for purposes ranging from extracting related terms to discovering implicit relations between concepts.
- *Association rules* (Maedche and Staab 2000): It is used to extract the non-taxonomic relations between concepts by using a small seed knowledge as background (e.g. using concept hierarchy as background).
- *Heuristic and conceptual clustering* (Faure and Nédellec 1998; Faure and Poibeau 2000a): It is used to group the concepts based on the semantic distance between them to make up hierarchies. Formal Concept Analysis (FCA) is one method under this approach that uses conceptual clustering technique to provide intentional descriptions for the abstract concepts or data units (Cimiano et al. 2005; Drymonas et al. 2010).
- *Ontology pruning* (Kietz and Maedche 2000): It is used to build a domain relevant ontology by using heterogeneous sources (e.g. comparing domain sources with the generic sources to determine which concepts are more relevant to the specific domain and which concepts are general).

4.2.2 Logic-based approaches

- *Inductive Logic Programming* (Zelle and Mooney 1993): It is used to derive the rules from positive and negative examples of the existing collection of concepts and relations. For example, firstly, the following positive examples: “cats have fur”, “dogs have fur”, and “tigers have fur”, then “mammals have fur” are generated. After that, from negative example, “humans do not have fur”, then the generalization of “mammals have fur” will be dropped and deduces that only “canines and felines have fur”. However, this approach depends on the good predefined rules templates by the expert. For instance, if there are no good negation examples, then an invalid rule may be generated. The considerable disadvantage of this approach is that in the search process sometimes can prune searched hypothesis (Boycheva 2002).
- *Logical inference* (Shamsfard and Barforoush 2004): It is used to infer implicit relations from existing ones. For example, “Steven is a man” and “all men are mortal”, then the following relation “Steven is mortal” is inferred. However, in this approach, there is a high possibility of introducing conflicting and/or invalid relations and rules (Wong et al. 2012). For example, “human eats fish” and “fish eats the worms” potentially generate invalid new relation. In addition, it can generate only very basic relations most of the time.

To conclude this section, Table 1 shows the summary of the discussed OL approaches and OL tasks corresponding with these approaches.

Table 1 Ontology learning approaches and their corresponding tasks

	OL Approaches	OL Tasks	OL Approaches Advantages	OL Approaches Disadvantages
Linguistics-Based Approaches	<i>Pattern-based extraction:</i> – <i>Lexico-syntactic patterns</i> – <i>Semantic templates</i>	Concept hierarchy & Relation discovery	Has reasonable precision	Have very low recall
	<i>POS tagging & sentence parsing</i>	Term extraction	Good for characterizing the context	Ambiguity (one word may have more than one tag)
	<i>Syntactic structure analysis & dependency structure analysis</i>	Term extraction, Concept hierarchy & Relation discovery	Good for uncovering the syntactic and grammatical relations in context	Need to incorporate with other algorithms or rules to has better performance
Statistical-Based Approaches	<i>Co-occurrence analysis</i>	Term extraction & Concept formation	Has good result for extracting related terms that occur together	Not appropriated for relation discovery task
	<i>Association rules</i>	Relation discovery	Has good result for well-defined problems	Need large support factor specified in advance
	<i>Heuristic/ conceptual clustering</i>	Synonym discovery, Concept formation & Concept hierarchy	Good for grouping the concepts based on the semantic distance	Not applicable for non-taxonomic relation discovery
	<i>Ontology pruning</i>	Term extraction	Good for reducing the noise data by determining the relevant concepts	Not applicable for relation discovery task
Logic-Based Approaches	<i>Inductive Logic Programming</i>	Axiom extraction	Has good result for the good predefined rules problems	May introduce invalid rules & May prune the searched hypothesis
	<i>Logical inference</i>	Concept hierarchy, & Relation discovery	Can generate new basic rule from existing ones	May infer invalid or conflicting relations & May only generate very basic relations

5 Ontology learning and construction systems

5.1 Tools or systems

Since last 2 decades, many semi-automatic or automatic OL systems (or tools) have been developed. These systems tried to enhance the OL process to make it more effective and efficient. Park et al. (2010) divided OL into three types as follow: ontology editing tools that help ontology engineer for acquiring, visualizing, and organizing domain knowledge; ontology merging tools that combine two or more existing ontologies to construct one coherent ontology; and ontology extraction tools (it also can be called automatic ontology construction tools) that try to extract concepts and/or relations by using some NLP and/or machine learning techniques.

As ontology extraction tools play a more promising role in automating ontology construction, therefore, in this section, our discussion focus on ontology extraction tools. Table 2 shows the comparison among ten prominent automatic ontology construction systems that take into account the discovering for any type of relations (taxonomic or non-taxonomic).

5.1.1 Asium

“*Acquisition of Semantic knowledge Using Machine learning methods*” (ASIMUM) (Faure and Nédellec 1998; Faure and Poibeau 2000b) is a semi-automated OL system that aims to learn the taxonomic relations. ASIMUM can also be considered as an ontology editing tool because it is designed to help experts in the acquisition of semantic knowledge from technical domains. The learning method in ASIMUM is based on heuristic and conceptual clustering. The basic clusters are formed by words that occur with the same verb after the same preposition (e.g. “ballpoint pen” and “pencil” are an adjunct of the verb “to write” and it may occur after the preposition “by” or “with”).

Table 2 A Comparison of the Ten Prominent Ontology Learning Systems

System Name	Input Type & Language	Learned Elements				Used Approach	User Intervention	Output Format	P & R, or F %
		Terms	Concepts	Taxonomic Relations	Non-taxonomic Relations				
ASIMUM	Unstructured text French	✓	✓			Syntactic structure analysis	Whole process	Frame based	Not provided
						Conceptual clustering			
CRCTOL	Unstructured (plain text documents only) English	✓	✓			Syntactic structure analysis, POS tagging, & relevance measures	Validation & Evaluation	RDFS or OWL	Not provided
				✓		Lexico-syntactic patterns, syntactic structure analysis			
					✓				
DODDL EII	Unstructured & structured data English			✓		co-occurrence (4-grams) & association rules algorithm	Whole process	Information not provided	F= 23, R= 56
		✓	✓			Lexico-syntactic patterns & semantic templates			
HASTI	Unstructured Persian		✓			Semantic templates, heuristic clustering & logical inference	Validation & Evaluation	Subset of KIF	Not provided
				✓					
					✓	Inductive logic programming			
OntoCmaps	Unstructured English	✓				Dependency structure analysis & POS tagging	Validation & Evaluation	OWL	Not provided
				✓		Dependency structure analysis, hierarchical clustering & filtering matrices			
					✓				
SYNDIKATE	Unstructured text German	✓	✓			Syntactic structure analysis & Dependency structure analysis	Evaluation	Special format	1 (P= 97, R= 57) (P= 94, R= 317)
				✓		Dependency structure analysis & Semantic templates			
					✓				
TEXT2ONTO	Unstructured text Spanish & German	✓				POS tagging, Syntactic structure analysis & relevance metrics	Validation & Evaluation	F-Logic, OWL or RDFS	F=22, R= 17, Re= 30
			✓			formal concept analysis			
				✓		hierarchical clustering & lexico-syntactic patterns			
					✓	association rules			
TextStorm and Clouds	Unstructured English	✓		✓		POS tagging, & Syntactic structure analysis	Whole process	Part of Dr. Diogo project	F=52
					✓	Inductive logic programming			
		✓				POS tagging, & Syntactic structure analysis			
TEXT-TO-ONTO	Structured, or sim-structured German		✓			Formal concept analysis & pruning	Validation & Evaluation	F-Logic, RDFS / DAML+OIL, & Part of EKON	Not provided
				✓		Hierarchical clustering & lexico-syntactic patterns			
					✓	Association rules			
PROMINE	Sim-structured English	✓	✓			POS tagging & relevance measures	Validation & Evaluation	Subset of PROKEX project	P= 89, R= 86
				✓		Heuristic clustering & filtering measures			

*Medical domain, **IT domain

5.1.2 Text-to-Onto

Text-to-Onto (Maedche and Volz 2001) is a semi-automated system that builds a domain ontology from an initial core ontology by using data mining and NLP. Text-to-Onto uses statistical-based approaches to learn ontology. The concepts are formed by using formal concept analysis, such as co-occurrence analysis, with no additional information required. Lexico-syntactic patterns are used for the hypernym (taxonomic relations) extraction, while association rules approach is used for non-taxonomic relation extraction. POS tagging and syntactic structure analysis are used for term extraction. This system uses the pruning approach by comparing domain sources with the generic sources to determine the domain-relevant concepts. Text-to-Onto takes the German web data (e.g. HTML free text, dictionaries) as input.

5.1.3 Text2Onto

Text2Onto (Cimiano and Völker 2005) is a redesign of Text-To-Onto system. Text2Onto applies different measures such as tf-idf and c-value/nc-value to find the relevance of a term with respect to the corpus. There is no significant difference between Text2Onto

and Text-To-Onto for the OL methods. User interface is friendlier in Text2Onto, e.g. the translation of the extracted ontologies to ontology languages such as OWL and RDFS is easier and the ontology experts (or users) have more control.

5.1.4 HASTI

HASTI (Shamsfard and Barforoush 2003, 2004) is considered an automatic OL system that tries to build dynamic ontologies from scratch (it uses a small kernel of primitive concepts as initial input). *HASTI* uses logic-based, linguistic-based and statistics-based approaches for OL process. Lexico-syntactic patterns and semantic templates are used for concept extraction. Also, semantic templates, heuristic clustering analysis and logical inference are used for taxonomic and non-taxonomic relation extraction. The limitation of the conceptual hierarchy method in this system is that each intermediate node has at most two children (Wong et al. 2007). In addition, *HASTI* is one of the little few systems that tries to learn axioms, it uses the inductive logic programming for axioms learning. However, these learned axioms at most are very general.

5.1.5 SYNDIKATE

SYnthesis of DIstributed Knowledge Acquired from TExtS (SYNDIKATE) (Hahn and Marko 2002; Hahn and Romacker 2001) is an automatic system for acquiring knowledge from real-word text. *SYNDIKATE* automatically bootstraps its domain knowledge as text analysis proceeds by learning module. *SYNDIKATE* uses syntactic structure, dependency analysis, and semantic templates for OL. Syntactic structure and dependency analysis are used for terms and concept extraction. Semantic templates as well as dependency analysis are used for taxonomic and non-taxonomic relation extraction. *SYNDIKATE* is more directed towards evolution and maintenance of ontologies, and other knowledge sources rather than to construct an ontology from scratch. The shortcomings are that this approach could generate too many hypotheses hence it becomes unmanageable and the calculations involved are also resource demanding. It is also somehow unclear what the actual output of the system is and how that output can be used. The result of this system may be viewed more as assistance for constructing an initial ontology rather than a finished ontology.

5.1.6 DODDLE-II

DODDLE-II (Nakaya et al. 2002) is a development of a *Domain Ontology rapiD DeveLopment Environment (DODDLE)* (Sekiuchi et al. 1998). *DODDLE* is OL system that uses an existing machine-readable dictionary (MRD) for OL, while *DODDLE-II* uses domain-specific English text corpus as well as an existing machine-readable dictionary (MRD) for acquiring taxonomic and non-taxonomic relationships of concepts. *DODDLE-II* uses co-occurrence (4-g) and association rules algorithm for extracting taxonomic and non-taxonomic relations.

5.1.7 TextStorm and clouds

TextStorm and Clouds (Oliveira et al. 2001; Pereira et al. 2000) is semi-automated OL system that aims to build a taxonomy. It consists of two main modules: *TextStorm* is for

extracting relations between concepts and Clouds is for completing these relations and inferring rules. TextStorm and Clouds system uses logic-based and linguistics-based approaches for performing OL. In this system, logical inference approach (using binary predicates) is used for taxonomic and non-taxonomic relation extraction, while inductive logic programming approach is used for axioms learning. POS tagging and syntactic structure analysis are used for term extraction.

5.1.8 CRCTOL

Concept-Relation-Concept Tuple based Ontology Learning (CRCTOL) (Jiang and Tan 2005; Jiang and Tan 2010) is OL system that aims to extract key concepts and finds the semantic relations of these concepts. CRCTOL uses linguistics-based and statistics-based approaches for performing OL. Multi-word terms in the form of nouns and noun phrases, predefined POS tagging, and syntactic structure analysis are used for extracting the terms and concepts. Lexico-syntactic patterns and syntactic structure analysis are used for extracting the taxonomic relations. For extracting non-taxonomic relation, tuples $\langle \textit{noun1} \gg \textit{verb} \gg \textit{noun2} \rangle$ is adopted. One of the shortcomings of this system is that the lexicon of specific terms of the domain is built and maintained manually. Another shortcoming is that this system observes only general concepts, and it ignores the whole-part relations which are likewise important in ontology constructing (Gillani Andleeb 2015). Furthermore, the resulting ontology in this system is based on domain specific documents, that makes these ontologies are not the accurate and comprehensive representation of the given domain. These ontologies may not be useful for applications with domain different than that knowledge base (Gillani Andleeb 2015).

5.1.9 OntoCmaps

OntoCmaps (Zouaq et al. 2011a, b) is OL system that tries to extract deep semantic representations in the form of concept maps. According to the authors, OntoCmaps generates rich conceptual representations in the form of concept maps and filters the important concepts and relationships. This system uses linguistics-based and statistics-based approaches for performing OL. The dependency structure analysis as well as POS tagging to represent the sentences, then the patterns are divided into conceptual patterns and hierarchical patterns. After that, the filtering metrics are used to filter the concept maps. The good thing about this system is that it does not rely on any predefined template for its semantic representation. However, this system does not specify formally the OL requirements when designing the system (Herrera 2014), which is considered one of the shortcomings of this system. In addition to this shortcoming, this system includes little documents details of OL process supporting and there is also lack of ontology evaluation details (Herrera 2014).

5.1.10 ProMine

ProMine (Gillani Andleeb 2015) is a semi-automated OL system for business processes/ organizations on domain ontology. The main aim of this system is to extract the semantic concepts that are most relevant for a domain. ProMine system includes three steps: first is extracting the terms and concepts, then these extracted concepts are filtered to find most relevant terms of a domain, and the last step is to build semantic concept categorization. This system uses linguistics-based and statistics-based approaches for performing OL.

Linguistics techniques such as POS tagging and frequency count are used to extract concepts, then filtering measures are applied. After that, the shortest path length and depth of concept methods are used to build the semantic concepts categories. Lastly, semantic lexical databases are used to enrich these semantic concepts categories. According to the authors, the limitation of this system is that it just extracts two words (compound words) concepts, but three words can also possibly represent a concept. This limitation will be followed up in the next iteration of ProMine system. The other limitation of this system is that it does not take into account the non-taxonomic relation extraction.

5.2 Evaluation metrics

This section presents five important evaluation measurements for an OL system. Precision, Recall and *F measure* are the main performance measures that are used for evaluating an OL system. Most of the existing OL systems use these measures. Precision (*P*) means the number of selected items that are relevant. While recall (*R*) means the number of relevant items that are selected. F-measure (harmonic mean of precision and recall) is an aggregated performance score for evaluating the algorithms and systems.

$$P = \frac{TP}{TP + FP} \quad (1)$$

$$R = \frac{TP}{TP + FN} \quad (2)$$

$$F = 2 \times \frac{P \times R}{P + R} \quad (3)$$

where *TP* is the true positive, *FP* is the false positive and *FN* is the false negative. True positive means the number of extracted (selected) items that are relevant. False positive means the number of extracted items that are irrelevant. While false negative means the total number of the relevant items.

Furthermore, there are two additional measures which missed in the evaluation process for most existing OL systems. Ontological Improvement (*OImp*) is a measure that accounts the newly discovered concepts which are absent in the benchmark, and Ontological Loss (*OLoss*) is a measure that accounts the concepts which exist in the benchmark, but they were not discovered. These two additional measurements are suggested in (Sabou et al. 2005) and they are defined as the following:

$$OImp = \frac{|C_d \setminus C_m|}{|C_m|} \quad (4)$$

$$OLoss = \frac{|C_m \setminus C_d|}{|C_m|} \quad (5)$$

where C_d is the discovered concepts and C_m is the recommended concepts.

5.3 Comparison of learning elements and approaches

Table 2 presents a comparison between ten prominent OL tools that consider relation extraction. This comparison includes the input type, the learned elements, and the used approaches for each learned element. It also includes the evaluation metrics that have been used to evaluate the system, R refers to recall measure, P refers to precision measure, and F refers to harmonic mean measure. As shown in Table 2, ASIUM and TEXT-TO-ONTO do not offer an overall figure of recall and precision for the whole OL system. HASTI, CRCTOL and OntoCmaps offer recall and precision for some sub-tasks but not for the whole OL systems. TextStorm and Clouds report the average figure of correctness. It has been assumed that this result represents the F measure (harmonic mean). In addition, this comparison also presents the output format of the ontology, as well as the level of human intervention in the systems.

6 Shallow versus deep learning for ontology construction

In this paper, the term of shallow learning system for OL refers to the systems of OL that use traditional ML and/or traditional (artificial) neural networks (ANNs/NNs). While shallow learning term refers to ANNs and DL term refers to deep neural networks.

6.1 Shallow learning

Based on the after-mentioned literature in existing shallow OL approaches and systems, it can be concluded that the existing approaches and systems of OL have many shortcomings and drawbacks. For example, most of these systems depend on human intervention in the whole or most of their tasks (e.g. concept formation, relation discovery etc.) such as Text-to-Onto, Text2Onto, and TextStorm and Clouds systems. In addition, most of existing shallow learning systems do not infer new relations or axioms, and some of them (e.g. Text2Onto and CRCTOL) use predefined templates for taxonomic and non-taxonomic relation extraction that lead to very low recall results.

In addition, most of these shallow learning systems of OL work with a small dataset and/or only the domain's dataset. For instance, Text-to-Onto used only 21 web articles as the input dataset, while CRCTOL based on selected documents of the domain, so the constructed ontology was not comprehensive and accurate representation of the given domain (Gillani Andleeb 2015).

The main challenges faced by OL systems are the relation discovery and axiom learning. There are many works in information extraction (IE) field that try to extract the semantic relations between pairs of concepts or named entities. The following sub-sections explore some of the important recent advancements of shallow systems in the relation discovery and axiom learning tasks.

6.1.1 Relation discovery

Sombatsrisomboon et al. (2003) suggested a simple method for discovering taxonomic relations between pairs of terms by using search engines. This study used only the pattern "NP is a/an NP" for searching. However, this proposed method often fails in acquiring hypernyms of a general noun as the authors stated (Sombatsrisomboon et al. 2003). Specia

and Motta (2006) introduced a hybrid of existing tools that used data mining and linguistic techniques for extracting the semantic relationship between pairs of named entities.

Sánchez and Moreno (2008) proposed unsupervised methods to discover non-taxonomic relations by using domain-relevant verb phrases to learn domain patterns, and then using statistical and linguistic analysis together with the learned domain patterns to extract non-taxonomic relations. Liu et al. (2008) developed a technique named Catriple. Catriple is a system for extracting triples automatically by using Wikipedia's categorical system. In this approach, syntactic rules and sentence parsers were used to extract the explicit values and attributes of the category names.

Suchanek et al. (2007) developed a new ontology named YAGO (Yet Another Great Ontology). YAGO is an ontology that is built on top of both WordNet and Wikipedia. In this study (Suchanek et al. 2007), the authors used the fact that Wikipedia had category pages (lists of articles that belong to a specific category) rather than using information extraction methods to leverage the knowledge of Wikipedia. In 2007, YAGO contained more than 1 million entities and 5 million facts. The semantic relations included the Is-A hierarchy and non-taxonomic relations between entities. However, it only used structured data for building the ontology. The main disadvantage of this approach is that the space can be wasted if not all arguments of n-array facts are known. YAGO was succeeded by YAGO2 at 2012. YAGO2 was an ontology that based on WordNet, Wikipedia, as well as GeoNames (a geographical database that contained over eight million place names and it covers all countries). YAGO2 contained more than 10 million entities and more than 120 million facts.

Etzioni et al. (2008) and Zhang et al. (2016) used the Conditional Random Fields (CRFs) for extracting attributes or information. In (Etzioni et al. 2008), the authors developed TextRunner tool for extracting information across different domains from the Web based on CFRs model. Zhang et al. (2016) proposed Simultaneously Entity and Relationship Extraction (SERE) model based on CFRs to extract binary relationship from unstructured text. The authors combined between IOB2 (named entity tags) and some defined tags to build the training file. The authors stated that this study focused only on the relationship between the two named entities in one sentence without considering the relationship between the named entities across all sentences.

El-Kilany et al. (2017) proposed an unsupervised clustering-based relation extraction method to construct a dataset of relations and then used these constructed extraction templates in order to generate and extract the readable relations from collections of news data. This study used Stanford Named Entity Tagger to extract the named entities under three types of entities (person, organization, and location). Then HITS algorithm was applied to get the importance of actors' entities and sentences. The authors concluded that if a score of high importance was at least 75% then two actors' entities in a sentence had a relation. In this case, the Stanford Parser was used to parse each sentence with a relation for producing the relevant dependency graph. Then the shortest path nodes between every two entities was used to find if the sentence matched one of the extracted templates. This proposed method gave better results in recall measure (R) but it gave less results in precision measure (P) compared to traditional methods.

Another useful study which related to OL was Herrera (2014). This study aimed to improve the knowledge management (KM) process based on OL. The suggested method in this study used the previous developed ontologies, database, and documents to recover the demanded knowledge through the OL process. This study was more aligned with ontology enrichment and populating rather than ontology constructing.

Despite all research works and efforts of shallow analysis in relation discovery, the results are still less than satisfactory and it needs more efforts to improve it (Albukhitan et al. 2017; Chen et al. 2010; Maimon and Browarnik 2015; Wong et al. 2012; Zhang et al. 2016; Zhou 2007).

6.1.2 Axiom learning

From all studies of the OL that have been referenced in this study, which include more than 40 OL approaches and systems, there are only few methods for axiom learning. Lin and Pantel (2001) showed some of the extracted similarities corresponded to inverse relations such as *author_of* and *written_by* to find the similar dependency tree paths, which could be used to axiomatize the meaning of some relation. Shamsfard and Barforoush (2004) suggested deriving axioms from conditional or quantified sentences such as 'All babies need milk'. Thus, can simply be used as a basis to define general rules. In (Oliveira et al. 2001; Pereira et al. 2000), the authors proposed a method to produce the rules by Is-A relation (such as $Is - A(X, vegetarian) : -eat(X, vegetables)$) and/or property (such as $property(X, friendly) : -property(X, small), have(X, fur)$). However, as mentioned above, producing the rules based on the property often are invalid (Wong et al. 2012).

In the (Fleischhacker and Völker 2011) study, the authors proposed inductive methods of enriching engineered ontologies with generating disjointness axioms. These methods used semantic similarity (statistical correlation analysis to rate the strength of linear relationships between two value sequences) and the association rule mining techniques for learning disjointness axioms. It also outlined the ideas underlying two alternative methods supporting the discovery of negative association rules. The experiments of this study showed that it was possible to induce disjointness axioms from an existing knowledge base with some accuracy, but there were two sources of errors. The first one was that the disjointness axioms determined by the automatic learning process can be incorrect, the other one was the incorrect, explicit or implicit *rdf* : *type* assertions.

Völker et al. (2007) proposed a semi-automatic ontology engineering method for automatically generating formal class descriptions from natural language definitions extracted (from Wikipedia definitions and from a fishery glossary which was provided by the Food and Agriculture Organization of the United Nations). The implementation of this method was based on a syntactic transformation of natural language definitions into OWL DL axioms in line with lexico-syntactic patterns. This proposed method had many limitations and shortcomings according to the authors themselves. For example, the parser which was used in this method failed to deliver a parse particularly for the structurally complex or ill-formed sentences, as well as there were several problems apart from the quality or efficiency of the syntactic analysis, which concerned semantic ambiguity related to quantifier or homonymy.

Mathews and Kumar (2017) proposed a new Controlled Natural Language (CNL) called TEDEI (TEXTual DEscription Identifier) to generate corresponding axioms. This method based on grammar-based syntactic transformation. However, as the authors themselves stated, this proposed approach could not handle any sentence that was not guided by a grammar. In addition, only one formalization per sentence was generated without taking into account the impact of ambiguity in the formalization.

As it is observed from mentioned literature, the extracting rules or axioms from unstructured or even semi-structured data is really hard task (Gillani Andleeb 2015; Mathews and

Kumar 2017) and it depends on the level of precision and recall of the concept and relation extraction tasks.

6.2 Differences between shallow learning and deep learning

DL is considered the second generation of ANNs (Mo 2012). Traditional ANN models have shallow-structure architectures, these architectures typically contain a single layer of nonlinear feature transformations and they lack multiple layers of adaptive non-linear features (Deng 2012; Deng and Yu 2014). These shallow architectures can effectively solve many simple or well-constrained problems, but their modelling and representational power are limited (Deng 2012). Hence, for more complicated real-world applications such as human speech, natural language, and natural image; these shallow architectures face many difficulties in dealing with these problems (Deng 2012; Deng and Yu 2014; Zouaq 2011). Whereas deep architectures produce better results when dealing with these complicated problems.

DL architecture can learn representations and features directly from the input with a little or no prior knowledge. This learning about features is a promise to get rid of features engineering which use other shallow learning architecture in the future (LeCun et al. 2015). In other words, the main distinguishing feature of DL from shallow learning is that DL derives its own features directly from data (feature learning), while shallow learning relies on handcrafted features based upon heuristics of the target problem. That means DL is more able to take advantage of increases in the amount of available computation and data (LeCun et al. 2015).

6.3 Deep learning techniques

There are three learning types of DL networks: (Deng 2012; Deng and Yu 2014) (1) unsupervised or generative: when no target labels data are available; (2) supervised or discriminative: when target label data are always available in direct or indirect forms; and (3) hybrid: when discriminative criteria for supervised learning are used to estimate the parameters in any of the deep generative or unsupervised deep networks. These DL networks can be feed-forward or recurrent networks. In the feed-forward networks, there is no connection between the neurons in the same layer. While in the recurrent networks, there may be connections between the neurons in the same layer. Figure 1(a) shows an example of a feed-forward network structure while (b) shows an example of a recurrent network structure.

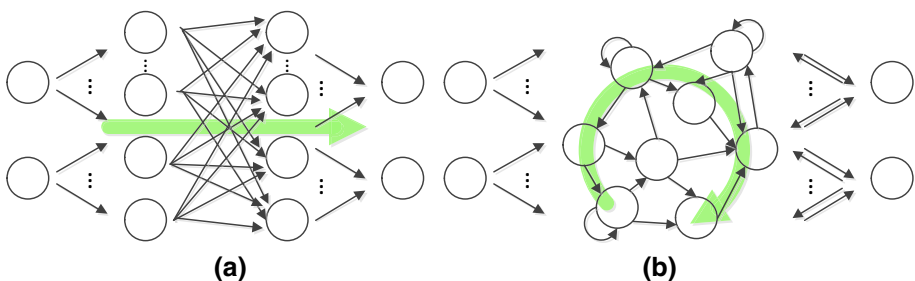


Fig. 1 Examples of feed-forward and recurrent networks structure

DL models use a deep graph with multiple processing layers, composed of multiple linear and non-linear transformations to model high-level abstractions in data (Deng and Yu 2014). There are four main DL models: Recurrent neural networks (RNNs), Convolutional Neural Networks (CNNs), Deep Belief Networks (DBNs) and Autoencoder Networks.

RNNs are unsupervised or supervised learning that are used to predict the future data sequence by using the previous data samples (Basesgmez 2014; Deng and Yu 2014). For language modelling tasks that involve sequential inputs, RNNs have a good performance, but training RNNs have proved to be problematic (so hard to train) (Basesgmez 2014; LeCun et al. 2015). If the goal is to predict the next word from the words previous given, then RNNs usually work better (Le 2015).

Convolutional Neural Networks (CNNs) and Deep Belief Networks (DBNs) (with their respective variations) are the two milestones in the field of DL (Arel et al. 2010; Chen and Lin 2014; Mo 2012). CNNs are multi-layer, supervised, and feed-forward networks. While DBNs are unsupervised or supervised, and feed-forward networks. CNNs have been used in wide applications in image and video recognition such as in (Basesgmez 2014; Kuang et al. 2018) and a little application of NLP such as in (Kim 2014). Also, DBNs have been used in many applications in image and video recognition such as in (Fischer and Igel 2012; Hinton et al. 2006; Huang et al. 2007) and in wide application of NLP such as in (Chen et al. 2010; Hinton and Salakhutdinov 2006; Salakhutdinov and Hinton 2007; Sarikaya et al. 2014; Zhong et al. 2016). In current literature of DL, it is observed that CNNs have performed better than DBN on the benchmark computer vision datasets such as MNIST, but if the dataset is not a computer vision, then DBNs can most definitely perform better (Deng 2012; Deng and Yu 2014; Hu et al. 2016).

Autoencoder network is an unsupervised or supervised feed-forward network that it is a closely related approach for DBNs (Hinton 2009). It is a network that its output target is the data input itself for compression objective as in (Le 2015) or denoising objective as in (Baldi 2012). Table 3 shows the summary of the RNNs, CNNs, DBNs, and Autoencoder Networks.

There are many other attempts to build DL models and techniques, for example, as in (Cohen 2005) and in (Deng and Yu 2011). In (Cohen 2005), the *author built stacks* of CRFs to develop a sequential classification when the segmentation information was unknown in the training data. While in (Deng and Yu 2011), the authors built a Deep Stacking Network (DSN) that also named Deep Convex Network (DCN). This model was developed for large-vocabulary speech recognition. Other example is as in (Albukhitan et al. 2017) and in (Arguello Casteleiro et al. 2017). The authors of these studies joined the Continuous Bag of Words (CBOW) model with a Skip-gram model to extract concepts and relations. CBOW and Skip-gram were machine learning models that each one of them had only one hidden layer. The authors called this a deep learning, because they used two models together and also the training of CBOW was little like the way of training DL models even if it had a single hidden layer.

6.4 From shallow learning to deep learning trend

The major problem in existing OL systems is the problem of language understanding by machine using shallow processing for text (Zouaq 2011). This section discusses the reasons for most of the recent techniques for NLP moved from shallow techniques toward the deep techniques.

Table 3 Summary of the four main DL models

DL model	Supervised/unsupervised	Using	Scope	Weaknesses
RNNs	Supervised and unsupervised	Predicting analysis and classifications	NLP and voice recognition	Training is very hard
CNNs	Supervised	Clustering and classifications	NLP and voice/image/video recognition	High computational cost and Need a lot of training datasets
DBNs	Supervised and unsupervised	Clustering and classifications with learning new features	NLP and voice/image recognition	Have less performance than other models for image recognition
Autoencoder Networks	Supervised and unsupervised	Compression and denoising	NLP and voice/image processing	Hard to form the generative model and treat the noisy or not noisy input equally

As it is observed from most recent research, the DL gives better handling for several NLP tasks versus shallow learning systems (Chen et al. 2010; Petrucci et al. 2016; Zhong et al. 2016; Zouaq 2011) as well as DL can handle large amount of data in an effective and efficient way. This would be more useful either in the most foundational NLP tasks such as POS tagging, or semantic role-labelling to the most complex tasks such as sentiment analysis, question-answering, OL, and machine translation (Grefenstette et al. 2014; Petrucci et al. 2016; Sarikaya et al. 2014).

Moreover, based on the state-of-the-art research results of NLP with DL, such as in (Albukhitan et al. 2017; Chen et al. 2010; Petrucci et al. 2016; Wang et al. 2018; Zhong et al. 2016), it can be concluded that using DL models for several NLP tasks can give better results rather than using traditional ANNs.

Furthermore, there are many advantages for using a DL rather than ANNs. One of DL advantages is that it can process unstructured data efficiently (Najafabadi et al. 2015). The other one of DL advantages is that it can learn prior knowledge from the input data and support different learning paradigms (Mo 2012). Also, it can extract the features from labelled or unlabelled data without the need to engineer these features; and it is more efficient in extract relationships and pattern in the data (Bengio and LeCun 2007; Najafabadi et al. 2015). Refer to Sect. 7 for more explanation and clarification.

6.5 Deep learning and ontology construction

It is noteworthy that in the last few years, there was a significant move towards deep analysis. The deeper techniques give better handling for text understanding and knowledge engineering versus shallow techniques. For example, Chen et al. (2010), and Zhong et al. (2016) used the DL model for extracting the attributes of named entity. In (Chen et al. 2010), the authors proposed an information extraction model to find and pair one of five types of relationships (Role, Part, At, Near, and Social) between each two named entities (Chinese entities) based on DBN, they used ACE 2004 Chinese dataset which have data of named entities under five type of entities (person, organization, GPE, location, and facility). While in (Zhong et al. 2016), the authors proposed unsupervised method based on DBN that extracted the Chinese named entity attributes for only persons, locations and organizations entities; this method called Entity Attribute Extraction Based on Deep Belief Network (EAEDB). This proposed method used the CRFs to extract the named entities and then used the proposed DBN to extract their attributes. The training file was built by combining between IOB (named entity tags) and POS tags. Based on the experiments in (Chen et al. 2010) study, using DBN gave better results rather than using Support Vector Machine (SVM) and traditional Back Propagation Neural Network (BP-NN).

In additions, the authors in (Albukhitan et al. 2017) designed a framework for OL from Arabic text. They used CBOW model with Skip-gram model to construct the words representation in vectors space and then extract the taxonomic relation extraction. The results of this study showed that this method gave better performance rather than traditional methods. Another study which uses the same technique is in (Arguello Casteleiro et al. 2017). It used CBOW and Skip-gram to extract biomedical terms, concepts, and relations from PubMed data regarding to sepsis domain.

Moreover, in the (Petrucci et al. 2016) study, the authors supposed that the OL could be a transductive reasoning process. The intention of the transductive process was to convert knowledge from natural language (source language) into a logic-based specification (target

language). The source language in this study was English and the target language was OWL. This supposed transduction process for sentences was divided to two phases: sentence transduction phase and sentence tagging phase. The RNN model was used for this process. The main limitation of this study was that the model had been trained and evaluated on a limited amount of data, that means it was just modelling a limited portion of natural language in a sentence-by-sentence. In addition, converting one sentence to be one axiom maybe not compatible with large data or with a different sample of collected data. Despite that, the results of this study gave a good evidence of the potential of DL towards long-term OL challenges.

Furthermore, in the (Wang et al. 2018) study, the authors used the CNN model to classify the text to set and then using this classified text and the TF-IDF matrix to construct the ontology for the shipping industry domain. This approach showed a high classification accuracy result that led to improve the ontology construction. However, the ontology construction framework in this study was built by experts.

In the (Wang 2015) report, the author summarized the recent advances in both DL and semantic data mining. In addition, he gave some explanation about how the DL can be capable for constructing better data representation for the machine. One of the essential discussed subjects in this report is knowledge bases and ontology building and how could DL techniques be applied to bridge semantic gaps (knowledge gaps) between the data, applications, data mining algorithms, and data mining results. At the end of this report, the author showed his intentions and thoughts on addressing the deep data representation with ontology. The author called his approach Deep Learning Ontology (DLO) which based on DBN. This DLO approach was considered an ontology-based DL framework and it sought to formally encode the concepts and three types of relations between these concepts (subclass, disjoint, and coexists) in the domain of the data label.

There are much other research that showed how DL can improve the text analysis and knowledge representation such as in (Collobert and Weston 2008; Neelakantan 2017) studies. In (Neelakantan 2017), the authors developed a new DL model called Neural Programmer (based on RNN) for knowledge representation and reasoning. While in (Collobert and Weston 2008), the authors proposed a general DL architecture for NLP tasks (e.g. POS tagging, named-entity recognition, language modelling and semantic role-labelling). As well as, in the (Hassan and Mahmood 2018) study, the authors designed a joint CNN and RNN framework for sentences classification, they used the recurrent layer as a substitute for the pooling layer. This framework takes advantage of the encoded local features extracted from the CNN model and the long-term dependencies captured by the RNN model. In addition, in the (Chicco et al. 2014) study, the authors used deep autoencoder model to predict the novel gene functions and annotations for creating and enriching gene database. The result of this study showed that using autoencoder performed better than traditional systems of gene function prediction.

Based on state-of-the-art studies on DL and NLP, we can conclude that addressing the DL for OL and other NLP tasks have promising results.

The ontology construction process is combined between NLP and data mining techniques. Table 4 shows the summarization of the most relevant studies, that are mentioned in this research, which using DL for any main task of ontology construction. The tasks of the ontology construction process are divided into three main tasks for this summarization, they are: *Term Extraction* which includes the term, concept extraction and/or their similarities; *Relation Discovery* which includes attribute or relation extraction and/or classification; and *Axiom Learning* which includes axiom and rule extraction and/or prediction.

Table 4 The summary of deep learning studies for ontology construction tasks

Task	Study	DL model	Language	Input	Domain	Target	Other Details
Term extraction and relation discovery	Albukhitan et al. (2017)	CBOW and Skip-gram	Arabic	5 thousand words	Not determined	5022 concepts and 830 relations	The system extracted correctly 3861 concepts and 587 relations
Axiom learning	Arguello Castelleiro et al. (2017)	CBOW and Skip-gram	Not mentioned	301,202 PubMed publications (title and abstract)	Biomedical (sepsis)	Get the candidate terms related to sepsis	
Relation discovery	Chen et al. (2010)	DBN	Chinese	221 documents	5 entity types (Person, Organization, GPE, location, and facility)	5 types of relations (Role, Part, At, Near, and Social)	Dataset is ACE 2004
Term extraction and relation discovery	Chicco et al. (2014)	Autoencoder	Not mentioned	Bos taurus (cattle) and Gallus gallus, (red junglefowl), gene sets from the Genomic and proteomic data warehouse (GPDW 2009 and 2013)	Biomedical (Gene)	Create and enrich gene database with massive gene function annotation and prediction	
Term extraction and relation discovery	Hassan and Mahmood (2018)	CNN and RNN	Not mentioned	Stanford Large Movie Review dataset (IMDB) and the Stanford Sentiment Treebank dataset (SSTb)	Sentiment analysis	Sentence classification	8544 sentences for training, 2210 for testing, and 1101 for validation

Table 4 (continued)

Task	Study	DL model	Language	Input	Domain	Target	Other Details
Axiom learning	Neelakantan (2017)	RNN	English	Freebase and Google's entity linking in ClueWeb, with 10 k entity pairs and 2 million paths per relation type	Several domains (not determined)	to produce latent programs involving arithmetic and logic operations	Use WikiTableQuestions dataset as test set
Axiom learning	Petrucci et al. (2016)	RNN	English	123 millions sentences and formulas	Several domains (not determined)	learning expressive ALCQ axioms	Dataset collected from encyclopedias and previous studies
Relation discovery	Wang (2015)	DBN	Not mentioned	Not determined	Different domains (not determined)	3 types of relations (subclass, disjoint, and coexists)	Use top-down DBN
Relation discovery	Wang et al. (2018)	CNN	Chinese	68,000 texts	Shipping industry	classify the terminology of domain category	47,600 texts for training set, 13,600 for verification set and 6800 for test set
Relation discovery	Zhong et al. (2016)	CRF and DBN	Chinese	24 MB	Crawler in the shipping news and travel websites	4 categories of entities attributes (Port, ship, routes, and view)	more than 10,000 sentences as training set 26 extracted attributes

7 Automatic ontology construction by deep learning

According to the ability of DL to represent the data and learn the features directly from input data because of the hierarchical methods of DL, the features can be distributed in deferent layers and the level of the abstraction is increased while the input data is processed at each layer. For example, in the object recognition the lowest layer is pixels, the higher layers are edges, patterns, parts and finally the highest layer is recognized object.

Based on the referred ontology construction challenges (Sect. 2.2), decreasing the human intervention in the ontology construction process is one of the major challenges of the ontology construction systems. The most of these systems are cooperative (see Table 2), one of the advantages of DL is that it can learn features directly from input data (no more need to manually features engineering). This naturally reduces the human intervention. In addition, the results of concept and relation classification in the most exist ontology construction systems are still less than satisfactory. For more explanation, the concepts and relations are different from domain to other. There are no certain concepts and relations that can be expected for the domain, especially for a new domain area. The shallow learning systems for this type of problem is less effective and efficient in contrast to DL which promises to provide better performance for this type of problems.

7.1 Deep learning to ontology learning process

The ontology construction techniques are considered a compound of NLP techniques and data mining techniques. Based on our knowledge and related literature we can give a general vision of where the DL can be used to enhance the OL process. The problem of existing OL systems can be considered the problem of language understanding by machine. The text is an unstructured data and sometimes the sentences are complicated, the current shallow learning systems cannot handle these complicated sentences efficiently while the hierarchical technique of DL promise to efficiently process for unstructured data by learning new features and representing the data for more language understanding by machine.

For example, in the pre-processing and concept extraction phases of text, the appropriate DL model can be used for POS tagging or for semantic role-labelling and for semantic-syntactic parsing. Where most of the OL approaches are founded on the utilization of the syntactic analysis to further extract relevant structures (e.g. concept extraction, depends on noun phrase). So using DL to build deeper analysis for sentences structures will enhance the sentences understanding and relevant structure extraction. Then the relevant concepts and relations, which depend on the syntactic analysis, will be better in precision and recall.

In addition, as it is mentioned previously, there are no certain concepts and semantic relations because they are different from domain to other, and axiom learning depends on the concepts and relations. For the relation discovery and axiom learning phases, the appropriate DL model, which can learn new features of concepts and relations for classifying and representing them, can be used for building an appropriate trained model by using the pre-processed corpus as input. After that, this trained model can be used to classify the concepts and to extract the semantic relations; then to infer the rules based on classified concepts and extracted relations. Where DL can predicate and learn prior knowledge as well as extract the features without the need, for example, to static templates to engineer these features. For more explanation, the following example and figures (Figs. 2 and 3) give a general vision of DL network and how to classify the concepts by DL.

Fig. 2 An example of DL network

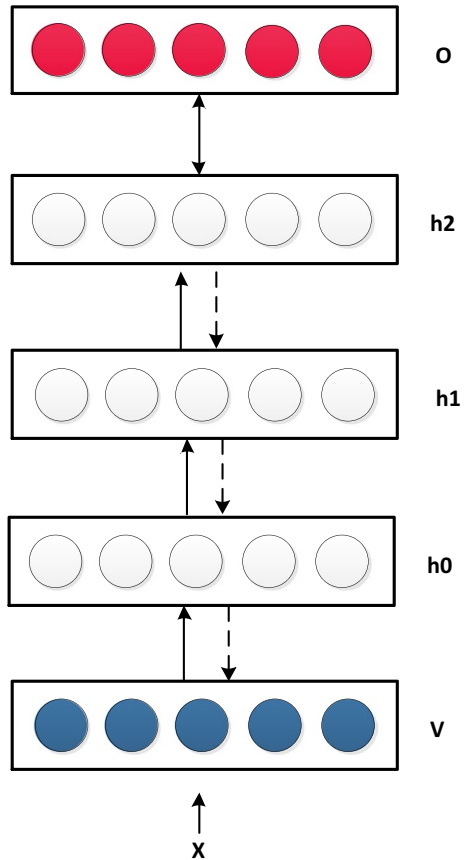


Figure 2 presents an example of DL network that has an input layer (V), three hidden layers (h_0, h_1, h_2), and output layer (O). X refers to input data. The dotted arrows represent the parameters for inferring samples from the posterior distribution at each hidden layer of the network. Figure 3 presents an example of how to build the concept classifier (trained model) through DL. This example gives a general vision of how DL classifier does regardless the details of DL model and its algorithms.

Now let us suppose that we have the following concepts of fishes' domain: hagfish, lampreys, sharks, sawfishes, and pelagic fish correspond the labels; while fishes, jawless fishes, cartilaginous fishes, and bony fishes are the other concepts that are defined through the domain knowledge. Note that the example in Fig. 3 uses a top-down method to build the concept classifier (trained model), then this trained model can be used to classify the concepts in the bottom-up method (jawless fishes, cartilaginous fishes, and bony fishes are considered as the labels).

Approximating any random function with a single hidden layer is still possible. However, learning through multiple layers (DL) is easier and faster (Bassez 2014; LeCun et al. 2015).

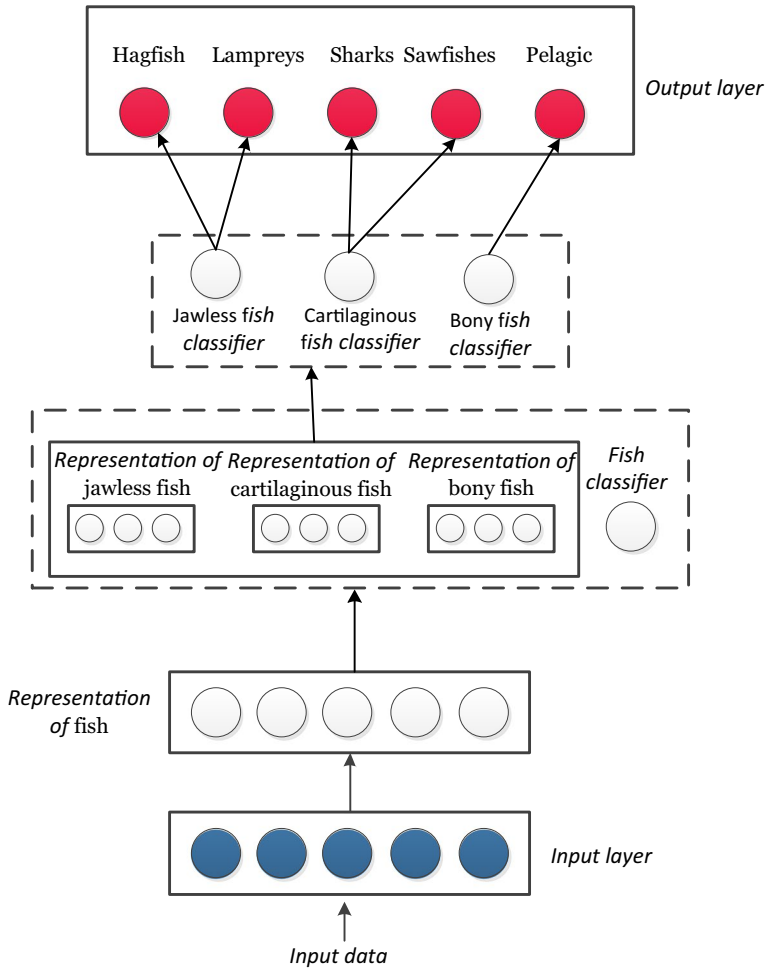


Fig. 3 An example of how to use DL for building the concept classifier

7.2 Deep automatic ontology construction system

Based on all of the above, we can be certain that building a deep automatic ontology construction system has promising results in terms of effectiveness, efficiency and reducing time and effort. Different DL models can be used in different stages of the ontology construction process. In other words, it is possible to build a different DL model for each different task in the ontology construction process. However, the main challenge of using the DL for ontology construction is to build an appropriate deep network and pick up the right method according to the particular task.

It is worth mentioning that despite all the promising results of using DL for ontology construction in the mentioned researches at this paper, but there is no research that uses the different DL models for the same task of the ontology construction process to compare and show which the DL model that is more appropriate for the selected task. That is what should be regarded in the future researches of deep automatic ontology construction.

8 Conclusion

This paper reviews, discusses, compares and criticizes different approaches, and systems of ontology construction from text. From the outcomes of reviewing existing ontology construction systems and approaches, this paper reveals that there is a consensus of the mentioned aspects of automatic ontology construction challenges that require more efforts. In brief, relation discovery, axioms learning, human intervention reduction, small/large scale input data transformation and standard platform building for ontology construction systems evaluation are considered the major challenges for an ontology construction system. Following that, this paper presents the rationale to move towards DL rather than traditional methods for OL. Then some research regarding to OL and DL are presented. Finally, this paper gives a vision of how and where the DL can be applied in ontology construction process.

We can summarise the key issues that will likely define the research directions in the near future of using DL for OL, as follows: using DL to build deeper sentences structures analysis (e.g. deep semantic-syntactic parser); using DL to classify concepts and relations; using DL to resolve the issues of learning and inferring relations and axioms.

References

- Abney S (1997) Part-of-speech tagging and partial parsing. In: Young S, Bloothoof G (eds) *Corpus-based methods in language and speech processing*, vol. 2. Springer, Netherlands, pp 118–136
- Albukhitan S, Helmy T, Alnazer A (2017) Arabic ontology learning using deep learning. Paper presented at the Proceedings of the international conference on web intelligence, Leipzig, Germany
- Arel I, Rose DC, Karnowski TP (2010) Deep machine learning—a new frontier in artificial intelligence research [research frontier]. *IEEE Comput Intell Mag* 5:13–18. <https://doi.org/10.1109/mci.2010.938364>
- Arguello Casteleiro M, Maseda Fernandez D, Demetriou G, Warren R, Fernandez-Prieto MJ, Des Diz J, Nenadic G, Keane J, Robert S (2017) A case study on sepsis using PubMed and deep learning for ontology learning. In: *Informatics for health: connected citizen-led wellness and population health*, vol 235. pp 516–520. <https://doi.org/10.3233/978-1-61499-753-5-516>
- Baldi P (2012) Autoencoders, unsupervised learning, and deep architectures. Paper presented at the proceedings of ICML workshop on unsupervised and transfer learning, Bellevue, Washington, USA
- Basegmez E (2014) The next generation neural networks: deep learning and spiking neural networks. In: *Advanced seminar in technical university of Munich, Munchen*, 2014, pp 1–40
- Bengio Y, LeCun Y (2007) Scaling learning algorithms towards AI. In: *Large-scale kernel machines*, p 34
- Boycheva S (2002) Overview of inductive logic programming (ILP) systems. *Cybern Inf Technol* 1:27–36
- Budanitsky A (1999) *Lexical semantic relatedness and its application in natural language processing*. University of Toronto
- Buitelaar P, Cimiano P, Magnini B (2005) Ontology learning from text: an overview. In: *Ontology learning from text: methods, evaluation and applications*, 123:3–12
- Chen X-W, Lin X (2014) Big data deep learning: challenges and perspectives. *IEEE Access* 2:514–525
- Chen Y, Li W, Liu Y, Zheng D, Zhao T (2010) Exploring deep belief network for chinese relation extraction. In: *Proceedings of the joint conference on Chinese language processing (CLP'10)*, pp 28–29
- Chicco D, Sadowski P, Baldi P (2014) Deep autoencoder neural networks for gene ontology annotation predictions. Paper presented at the Proceedings of the 5th ACM conference on bioinformatics, computational biology, and health informatics, Newport Beach, California
- Cimiano P, Völker J (2005) Text2Onto. In: Montoyo A, Muñoz R, Métails E (eds) *Natural language processing and information systems. Proceedings of 10th international conference on applications of natural language to information systems, NLDB 2005*, Alicante, Spain, June 15–17, 2005. Springer, Berlin, pp 227–238. https://doi.org/10.1007/11428817_21

- Cimiano P, Hotho A, Staab S (2005) Learning concept hierarchies from text corpora using formal concept analysis. *J Artif Intell Res* 24:305–339
- Cohen WW (2005) Stacked sequential learning. Carnegie-Mellon Univ Pittsburgh PA School of Computer Science
- Collobert R, Weston J (2008) A unified architecture for natural language processing: deep neural networks with multitask learning. In: Proceedings of the 25th international conference on machine learning. ACM, pp 160–167
- Deng L (2012) Three classes of deep learning architectures and their applications: a tutorial survey. *APSIPA transactions on signal and information processing*
- Deng L (2014) A tutorial survey of architectures, algorithms, and applications for deep learning. *APSIPA Transactions on signal and information processing* 3:1–29. <https://doi.org/10.1017/atsip.2013.9>
- Deng L, Yu D (2011) Deep convex net: A scalable architecture for speech pattern classification. In: Annual conference of the international speech communication association, 2011. Interspeech, pp 2285–2288
- Deng L, Yu D (2014) Deep learning. *Signal Process* 7:3–4
- Drymonas E, Zervanou K, Petrakis EGM (2010) Unsupervised ontology acquisition from plain texts: the OntoGain system. In: Natural language processing and information systems. Springer, Berlin, pp 277–287
- El-Kilany A, Tazi NE, Ezzat E (2017) Building relation extraction templates via unsupervised learning. Paper presented at the Proceedings of the 21st international database engineering and applications symposium, Bristol, United Kingdom
- Etzioni O, Banko M, Soderland S, Weld DS (2008) Open information extraction from the web. *Commun ACM* 51:68–74. <https://doi.org/10.1145/1409360.1409378>
- Faure D, Nédellec C (1998) Asium: Learning subcategorization frames and restrictions of selection. In: Kodratoff Y (ed) Text mining workshop, 10th European conference on machine learning (ECML 98), Chemnitz, Germany.
- Faure D, Poibeau T (2000a) First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX. In: Ontology learning ECAI-2000 workshop, pp 7–12
- Faure D, Poibeau T (2000b) First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX. In: Proceedings of the ECAI workshop on ontology learning
- Fischer A, Igel C (2012) An introduction to restricted Boltzmann machines. In: Iberoamerican congress on pattern recognition. Springer, pp 14–36
- Fleischhacker D, Völker J (2011) Inductive learning of disjointness axioms. In: On the move to meaningful internet systems: OTM 2011. Springer, Berlin, pp 680–697
- Gamallo P, Gonzalez M, Agustini A, Lopes G, De Lima VS (2002) Mapping syntactic dependencies onto semantic relations. In: Proceedings of the ECAI workshop on machine learning and natural language processing for ontology engineering, pp 15–22
- Gillani Andleeb S (2015) From text mining to knowledge mining: An integrated framework of concept extraction and categorization for domain ontology. Doctoral dissertation, Budapesti Corvinus Egyetem
- Gómez-Pérez A, Manzano-Macho D (2003) A survey of ontology learning methods and techniques. Deliverable 1.5, Onto Web consortium
- Grefenstette E, Blunsom P, de Freitas N, Hermann KM (2014) A deep architecture for semantic parsing. arXiv preprint [arXiv:14047296](https://arxiv.org/abs/14047296)
- Hahn U, Marko KG (2002) Ontology and lexicon evolution by text understanding. In: Proceedings of the ECAI 2002 workshop on machine learning and natural language processing for ontology engineering (OLT 2002), Lyon, France
- Hahn U, Romacker M (2001) The SYNDIKATE text knowledge base generator. Paper presented at the Proceedings of the first international conference on Human language technology research, San Diego
- Hassan A, Mahmood A (2018) Convolutional recurrent deep learning model for sentence classification. *IEEE Access* 6:13949–13957. <https://doi.org/10.1109/ACCESS.2018.2814818>
- Herrera RG (2014) Knowledge management systems based on ontology learning. Universidad de Granada, Granada
- Hinton GE (2009) Deep belief networks. *Scholarpedia* 4:5947
- Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. *Science* 313:504–507
- Hinton GE, Osindero S, Teh Y-W (2006) A fast learning algorithm for deep belief nets. *Neural Comput* 18:1527–1554
- Hu S, Zuo Y, Wang L, Liu P (2016) A review about building hidden layer methods of deep learning. *J Adv Inf Technol* 7:13–22. <https://doi.org/10.12720/jait.7.1.13-22>

- Huang FJ, Boureau Y-L, LeCun Y (2007) Unsupervised learning of invariant feature hierarchies with applications to object recognition. In: 2007 IEEE conference on computer vision and pattern recognition. IEEE, pp 1–8
- Jiang X, Tan A-H (2005) Mining ontological knowledge from domain-specific text documents. In: Fifth IEEE international conference on data mining. IEEE
- Jiang X, Tan AH (2010) CRCTOL: a semantic-based domain ontology learning system. *J Am Soc Inf Sci Technol* 61:150–168
- Kietz J-U, Maedche A, Volz R (2000) A method for semi-automatic ontology acquisition from a corporate intranet. In: EKAW-2000 workshop “Ontologies and Text”, Juan-Les-Pins, France
- Kim Y (2014) Convolutional neural networks for sentence classification. arXiv preprint [arXiv:14085882](https://arxiv.org/abs/1408.5882)
- Klein D, Manning CD (2003) Accurate unlexicalized parsing. Paper presented at the Proceedings of the 41st Annual meeting on association for computational linguistics—volume 1, Sapporo, Japan
- Kuang Z, Yu J, Li Z, Zhang B, Fan J (2018) Integrating multi-level deep learning and concept ontology for large-scale visual recognition. *Pattern Recogn* 78:198–214. <https://doi.org/10.1016/j.patco.2018.01.027>
- Le QV (2015) A tutorial on deep learning part 2: autoencoders, convolutional neural networks and recurrent neural networks. *Google Brain* 1–20
- LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436
- Lin D, Pantel P (2001) DIRT- discovery of inference rules from text. Paper presented at the proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, California
- Liu Q, Xu K, Zhang L, Wang H, Yu Y, Pan Y (2008) Catriple: extracting triples from wikipedia categories. In: Asian semantic web conference. Springer, pp 330–344
- Maedche A, Staab S (2000) Discovering conceptual relations from text. In: Horn W (ed) Proceedings of the 14th European conference on artificial intelligence, Berlin, Germany. ECAI'00. IOS Press, pp 321–325
- Maedche A, Volz R (2001) The text-to-onto ontology extraction and maintenance environment. In: Proceedings of the ICDM-workshop on integrating data mining and knowledge management, San Jose, California
- Maimon O, Browarnik A (2015) Ontology learning from text: why the ontology learning layer cake is not viable. *Int J Signs Semiot Syst* 4:1–14. <https://doi.org/10.4018/ijsss.2015070101>
- Mathews KA, Kumar PS (2017) Extracting ontological knowledge from textual descriptions through grammar-based transformation. Paper presented at the proceedings of the knowledge capture conference, Austin, TX
- Mishra S, Jain S (2015) A study of various approaches and tools on ontology. In: 2015 IEEE international conference on computational intelligence and communication technology (CICIT), pp 57–61. <https://doi.org/10.1109/cicit.2015.43>
- Mo D (2012) A survey on deep learning: one small step toward AI Dept Computer Science, Univ of New Mexico, USA
- Morin E (1999) Automatic acquisition of semantic relations between terms from technical corpora. In: Proceedings of the fifth international congress on terminology and knowledge engineering-TKE'99
- Mudhsh BAD, Al-Takhaynih AH, Al-Dala'ien OA (2015) Immediate constituent analysis (ICA). *Int J Sci Res Publ* 5(6)
- Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E (2015) Deep learning applications and challenges in big data analytics. *J Big Data* 2:1
- Nakaya N, Kurematsu M, Yamaguchi T (2002) A domain ontology development environment using a MRD and text corpus. In: Proceedings of the fifth joint conference on knowledge-based software engineering frontiers in artificial intelligence and applications, pp 242–251
- Neelakantan AR (2017) Knowledge representation and reasoning with deep neural networks. Doctoral Dissertation, University of Massachusetts, Amherst
- Nivre J (2004) Incrementality in deterministic dependency parsing. Paper presented at the proceedings of the workshop on incremental parsing: bringing engineering and cognition together, Barcelona, Spain
- Oliveira A, Pereira FC, Cardoso A (2001) Automatic reading and learning from text. In: Proceedings of the international symposium on artificial intelligence (ISAI)
- Park J, Cho W, Rho S (2010) Evaluating ontology extraction tools using a comprehensive evaluation framework. *Data Knowl Eng* 69:1043–1061. <https://doi.org/10.1016/j.datak.2010.07.002>
- Pereira FC, Oliveira A, Cardoso A (2000) Extracting concept maps with clouds. In: Proceedings of the Argentine symposium of artificial intelligence (ASAI)
- Petrucci G, Ghidini C, Rospoche M (2016) Ontology learning in the deep. In: Knowledge engineering and knowledge management. Springer, Cham, pp 480–495

- Sabou M, Wroe C, Goble C, Mishne G (2005) Learning domain ontologies for Web service descriptions: an experiment in bioinformatics. Paper presented at the Proceedings of the 14th international conference on World Wide Web, Chiba, Japan
- Salakhutdinov R, Hinton G (2007) Semantic hashing. *RBM* 500:500
- Sánchez D, Moreno A (2008) Learning non-taxonomic relationships from web documents for domain ontology construction. *Data Knowl Eng* 64:600–623
- Sarikaya R, Hinton GE, Deoras A (2014) Application of deep belief networks for natural language understanding. *IEEE/ACM Trans Audio Speech Lang Process* 22:778–784
- Sekiuchi R, Aoki C, Kurematsu M, Yamaguchi T (1998) DODDLE: a domain ontology rapid development environment. In: Lee H-Y, Motoda H (eds) *PRICAI'98: topics in artificial intelligence*. Proceedings of 5th Pacific Rim international conference on artificial intelligence Singapore. Springer, Berlin, pp 194–204. <https://doi.org/10.1007/bfb0095269>
- Shamsfard M, Barforoush AA (2003) The state of the art in ontology learning: a framework for comparison. *Knowl Eng Rev* 18:293–316
- Shamsfard M, Barforoush AA (2004) Learning ontologies from natural language texts. *Int J Hum Comput Stud* 60:17–63
- Sombatsrisomboon R, Matsuo Y, Ishizuka M (2003) Acquisition of hypernyms and hyponyms from the WWW. In: Proceedings of the 2nd international workshop on active mining
- Specia L, Motta E (2006) A hybrid approach for relation extraction aimed at the semantic web. In: International conference on flexible query answering systems. Springer, pp 564–576
- Suchanek FM, Kasneci G, Weikum G (2007) Yago: a core of semantic knowledge. Paper presented at the proceedings of the 16th international conference on World Wide Web, Banff, Alberta, Canada
- Völker J, Hitzler P, Cimiano P (2007) Acquisition of OWL DL axioms from lexical resources. In: The semantic web: research and applications. Springer, Berlin, pp 670–685
- Wang H (2015) Semantic deep learning. University of Oregon, Oregon
- Wang J, Liu J, Kong L (2018) Ontology construction based on deep learning. In: *Advances in computer science and ubiquitous computing*. Springer, Singapore, pp 505–510
- Wong W, Liu W, Bennamoun M (2007) Tree-Traversing Ant Algorithm for term clustering based on featureless similarities. *Data Min Knowl Disc* 15:349–381. <https://doi.org/10.1007/s10618-007-0073-y>
- Wong W, Liu W, Bennamoun M (2012) Ontology learning from text: a look back and into the future. *ACM Comput Surv (CSUR)* 44:20
- Zelle JM, Mooney RJ (1993) Learning semantic grammars with constructive inductive logic programming. In: *AAAI*, pp 817–822
- Zhang J, Liu J, Wang X (2016) Simultaneous entities and relationship extraction from unstructured text. *Int J Database Theory Appl* 9:151–160
- Zhong B, Liu J, Du Y, Liao Zheng Y, Pu J (2016) Extracting attributes of named entity from unstructured text with deep belief network. *Int J Database Theory Appl* 9:187–196
- Zhou L (2007) Ontology learning: state of the art and open issues. *Inf Technol Manag* 8:241–252
- Zouaq A (2011) An overview of shallow and deep natural language processing for ontology learning. In: Wong W, Liu W, Bennamoun M (eds) *Ontology learning and knowledge discovery using the web: challenges and recent advances*, vol 2. Information Science Reference (IGI Global), USA, pp 16–37
- Zouaq A, Gasevic D, Hatala M (2011b) Towards open ontology learning and filtering. *Inf Syst* 36:1064–1081. <https://doi.org/10.1016/j.is.2011.03.005>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.