ONTOLOGY LEARNING: REVISTED

AHAMAD ABDOLLAHZADEH BARFORUSH   ALI RAHNAMA

Computer Engineering and IT department, Amirkabir University of Technology

424 Hafez Ave., Tehran, Iran

{ahmad, arahnama}@aut.ac.ir

The term "ontology" comes from the field of philosophy that is concerned with the study of being or existence. In general computer science defines ontology as an "explicit specification of a conceptualization," which is, "the objects, concepts, and other entities that are presumed to exist in some area of interest and the relationships that hold among them". Ontologies hold a great importance to modern knowledge based systems. They enable shared knowledge and reuse where information resources can be communicated between human or software agentsand should be machine readable. Manual construction of ontologies is an expensive and time consuming task. An answer to this problem is to provide an automatic or semi- automatic tool for ontology construction. Over the past years, this field of research has not yet reached the goal of fully automating the ontology development process. In this paper we will review the ontology creation process with the help of ontology learning (OL) and extend our previous OL framework. We will examine OL applications with respect to the extensions of our framework. And last we will define a roadmap for future work.

Communicated by: B. White & Y. Deshpande

1 Introduction

The term "ontology" comes from the field of philosophy that is concerned with the study of being or existence. In general computer science defines ontology as an "explicit specification of a conceptualization," which is, "the objects, concepts, and other entities that are presumed to exist in some area of interest and the relationships that hold among them". They hold a great importance to modern knowledge based systems. They can be used to support a great variety of tasks in different domains such as knowledge representation, natural language processing, information retrieval, information exchange, collaborative systems, databases, knowledge management, database integration, digital libraries, information retrieval, or multi agent systems.

Ontologies also enable shared knowledge and reuse where information resources can be communicated between human or software agents. Semantic relationships in ontologies should be machine readable. It should be noted here that there are differences between ontologies and other conceptual structures such as taxonomies, thesauruses, folksonomies… . We have presented the differences of such structures from ontologies in [1].

Manual construction of ontologies is an expensive and time consuming task, this is due to the difficulty in capturing knowledge also known as the "knowledge acquisition bottleneck". An answer to this problem is to provide an automatic or semi- automatic tool for ontology construction, which is the subject of this paper. Although progress has been made over the last years, this field of research has not yet reached the goal of fully automating the ontology development process. Work on automatic learning of non-taxonomic relationships has already been conducted but has not reached an advanced development stage yet.

This paper is an extended and revised version of our previous framework published in [2] which is based on over 8 years of ontology engineering research. In this paper we will first in section 2 discuss the ontology creation process with the help of ontology learning (OL). We will also present the newest approaches presented in recent work. In section 3 we present the state of the art in ontology learning. In examining each OL application, we state the types of their inputs, type of ontology creation, and their types of outputs. And finally in section 4 we conclude and define future work areas.

2   Ontology Creation

Two main methods exist in ontology construction. The first aids manual ontology construction by providing tools, including editors, consistency checkers, mediators to support shared decisions, and ontology import tools. The second relies on machine learning and automated language processing techniques to extract concepts and ontological relations from structured and unstructured data such as databases and text or more precisely it relies on ontology learning. Manual ontology creation is very time consuming and difficult and usually requires an expert ontology engineer for the domain. But the manual ontology created is almost always at a higher level of semantic richness. For this reason when semantic richness is the goal, manual approaches are preferred to automatic approaches. To help speed up the manual ontology creation process, semi-automatic tools and methods have been presented. These tools have automated the manual process with the assistance of a human expert at certain points of the process. The main objective of ontology learning systems which have attempted to fully automate the ontology creation process is to eliminate the need of any human interaction in the process. The ultimate achievement of automated ontology learning systems is to produce an ontology with the exact or better richness than the one created by the manual process. In this paper we will cover only semi and automatic methods of ontology creation with the help of ontology learning. In other words, ontology learning is the acquisition of knowledge for the semi or automatic creation of ontologies. Mostly ontology learning is from text, but it can also be from other sources. In order to achieve our ontology learning goal we first need to establish the subtasks that together constitute the complex task of ontology development.

2.1 Ontology learning process

There are two viewpoints in ontology learning (OL). The first one is the availability of prior knowledge. The other, is the type of input used by the learning process. Here, we will emphasize on the type of input and assume that we are creating ontology from scratch. In this section after investigating the type of input, we will discuss the ontology creation techniques.

2.1.1 Types of input

Ontologies can be learnt from various sources, they can be databases, structured and unstructured documents or even dictionaries, taxonomies and directories. There are three different kinds of input: structured data, semi-structured data, and unstructured data.

2.1.1.1 OL from Structured Data

These ontology learning procedures extract parts of the ontology using the available structural information. Examples of structured information sources are database schemas, existing ontologies and knowledge bases. The central problem in learning from structured data is to determine which pieces of structural information can provide relevant knowledge [3].

2.1.1.2 OL from semi-structured data

Usually the quality of the results of ontology learning procedures using structural information is better than the ones using completely unstructured input data [4]. It is obvious that more structured input data yields richer results. So, semi-structured data will help create richer results in comparison with unstructured data. The reason for this is that more semantics is provided in the data, so in response better inference and deduction can be performed. Unfortunately, most of the available knowledge is in the form of unstructured text. Examples of semi-structured data are WordNet [5], HTML and XML documents.

2.1.1.3 OL from unstructured data

Unstructured data extraction methods are those that do not rely on any structural information for improving the quality of its results. They are important because unstructured data is the most available format for ontology learning input [4]. Unstructured documents consist of natural language texts such as Word, or text documents.

2.2 Ontology creation methods

Different methods for ontology creation and construction have been presented in the past few years. In this section, we intend to first present the traditional (main) approach and then present the newest approaches presented in recent work. The main steps and knowledge acquisition techniques employed for building ontologies can be seen in figure 1 which we will elaborate in the following subsections.

2.2.1 Concept learning and taxonomy construction

In order to perform the ontology construction process from scratch, the first step is to retrieve an initial base of knowledge for the desired domain. So, our first objective is to retrieve terms that are related to a domain (Term extraction) and build a hierarchy (Hierarchy construction).

Three different learning approaches can be used. First, some approaches rely on the document-based notion of term subsumption (Concept formation) [6]. Secondly, some researchers claim that words or terms are semantically similar to the extent to which they share similar syntactic contexts (Synonym extraction) [7][8]. And finally, other researchers have attempted to find taxonomic relations expressed in texts by matching certain patterns associated to the language in which documents are presented [9]. Pattern-based approaches are heuristic methods using regular expressions that have been successfully

applied in information extraction. The text is scanned for instances of distinguished lexical-syntactic patterns that indicate a relation of interest. For example Hearst patterns [10] such as "X, Ys *and other* Zs" or "Ws *such as* X, Y and Z" can be used to achieve this goal. Also a detailed state of the art survey on pattern based semantic relation extraction can be found in [11]. Also "doubly anchored" hyponym patterns such as "X *such as* Xs *and* Zs" have been presented in [12] which provide a means of achieving better precision. In [13] performance issues of different patterns have been investigated. Also, work has been done on automating the lexico-syntactic pattern generation process such as "*moved to (location)X*" [14]. Automatic cluster labeling has been proposed as a means of creating better labels for the relations extracted [15].

Concept learning
- Term extraction
- Synonym extraction
- Concept formation

Taxonomic relation learning
- Hierarchy construction
- *is-a* relation extraction

Non-taxonomic relation learning
- *part-of* relation extraction
- *related-to* relation extraction
- *similar-to* relation extraction
- *Qualia* relation extraction
- *Causation* relation extraction
- ...

Axiom and rule learning
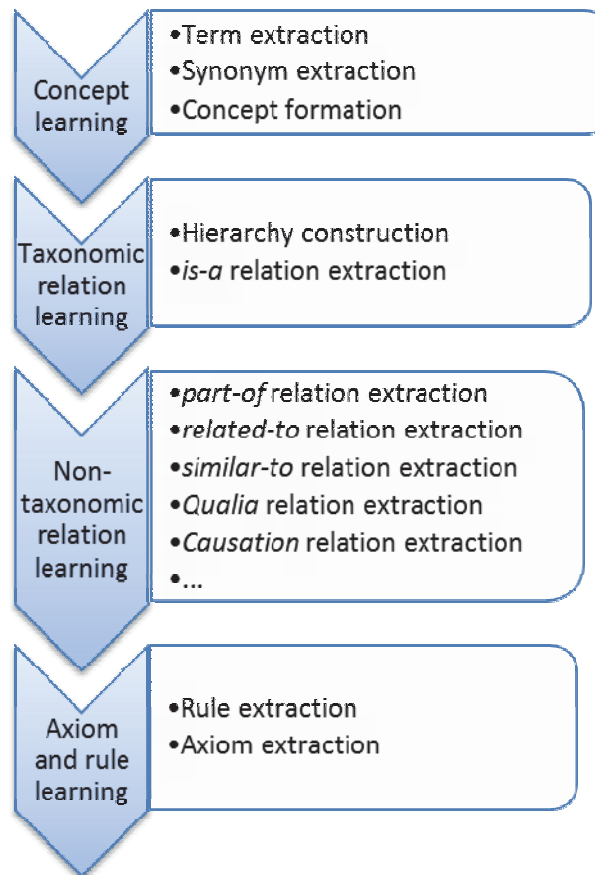- Rule extraction
- Axiom extraction

Figure 1 An illustration of the main steps in ontology construction

Next the construction of an initial taxonomy of concepts using *is-a* relations is done (Hierarchy construction). Some typical approaches to build a hierarchy of concepts use the following techniques: clustering[7][16], WordNet-based approaches[5], document-subsumption[17], lexico-syntactic patterns[18], [19], shallow linguistic parsing[20][21][22], taxonomy extension and refinement[23][24], and statistical analysis[25][26][27].

2.2.2 Identifying non-taxonomic relations

The discovery of non–taxonomic relations is considered as one of the least tackled problems within ontology learning [28]. In general, two tasks have to be performed. First, we have to detect which concepts are related. Second, we have to figure out how these concepts are related. And a name for the relation has to be found. This is typically specified by a verb. Verbs specify the interaction between the participants of some action or event by expressing relations between them.

There are several trends in learning relationships from text depending on the degree of generality of the extracted relations (Relation extraction). Some authors have developed approaches for learning specific relationships such as *subtype-supertype* relations like *part-of* [9], Qualia relations which consist of four types (*Formal* relations such as "NP_{Qualia} (*and|or*) *other* NP_{Formal}" for example "cars *or other* automobiles**"** , *Constitutive* relations such as "NP_{Qualia} *consists (of)* $NP_{Constructive}$" for example "a car *consists of* engine, chasis, gear box**"**, *Agentive* relations such as "*to* $Verb_{Telic}$ *a* (*new|complete*) NP_{Qualia}" for example "*to* develop *a new* car", and *Telic* relations such as "*purpose of* NP_{Qualia} *is (to)* Verb NP_{Telic}" for example "*purpose of* a car *is to* transport") [29] or Causation relations such as "NP *verb* NP" for example "earthquakes *generate* tsunamis" [30], by using specific language related linguistic. Co-occurrence analysis between terms is used to infer relations with little attention to sentence structure[31][32]. Other approaches aim to learn more general relations by exploiting the linguistic structure of text. Relation extraction is therefore related to the problem of acquiring selection restrictions for verb arguments. Reinberger and Spyns in [33] employ statistical methods based on frequency information over linguistic dependencies in order to establish relations between entities from a corpus of the biomedical domain. Schutz and Buitelaar [34] developed a system (RelExt) that is capable of automatically identifying highly relevant triples (pairs of concepts connected by a relation) over concepts from an existing ontology. RelExt works by extracting relevant verbs and their grammatical arguments (i.e. terms) from a domain-specific text collection and computing corresponding relations through combination of linguistic and statistical processing.

Ruenes in [35] worked by studying the sentence structure (subject, verb, object). They used verbs as the central point for discovering non-taxonomic relationships. They started from domain-related verbs that were learned automatically in a previous stage. They considered specific verb phrases as domain dependent semantic patterns that express non-taxonomic relations for a domain.

For the identification and labeling of non-taxonomic relations (such as *part-of*, *related-to*, *similar-to*, *cause/effect*, and other domain dependent relations), the following techniques are typically considered: anonymous relation extraction mainly done with association rules [36][37], and named relation extraction usually by linguistic parsing[38], [39][40].

To our knowledge work on axiom learning has not been performed much. Most of the work done has been by Volker et al.[41]. They have presented methods to automatically create complex class descriptions [42], and a semi-automatic approach to taxonomy refinement [43]. Other than Volker's works, in [44][45][46] automatic approaches to the axiomatization on WordNet have been presented.

Ontology population commonly refers to the extraction of instances of ontological concepts from text. In general, the recognition of named entities and their associated categories within unstructured text traditionally relies on semantic lexicons and gazetteers. Many named entity recognizers traditionally rely on lists of names [47]. It is also possible to build recognizers that identify names

automatically in text [48]. Such approaches usually attempt to learn general categories such as organizations or persons rather than refined categories.

Other authors such as Lamparter et al. in [49] are using WordNet to perform this detection: if the word is not found in the dictionary, it is assumed to be a named entity. However, sometimes, a named entity can be composed by common words, so the use of a dictionary is not enough. Other approaches take into consideration the way in which named entities are presented in the specific language. Languages such as English distinguish proper names from other nouns through capitalization. This method, combined with linguistic pattern analysis, has been applied by several authors [50][51][52][39], obtaining good results.

An important problem in information retrieval is semantic disambiguation: a word may have multiple meanings (polysemy), yet several words can have the same meaning (synonymy). Treatment of semantic ambiguity (mainly polysemy and synonymy) is done mainly in order to improve the quality of the results. In general, solving polysemy increases the quality of the returned results (precision) by eliminating results of the wrong word-sense; treating synonymy increases the proportion of correct results (recall) by including terms that have the same meaning [53].

The resolution of the lexical ambiguity that appears when a given word in a context has several different meanings is commonly called Word Sense Disambiguation (WSD). There are two approaches to this problem: supervised [54] and unsupervised approaches [55]. In unsupervised methods usually one of two general approaches are taken. The first, there are global, context-independent approaches, which assign meanings retrieved from an external dictionary by applying special heuristics. And the second, there are context-sensitive approaches. This kind of methods uses the context of a word to disambiguate it. For example Rosso et al. in [56] have been using the Web to disambiguate, analyzing text contexts in comparison to WordNet definitions or hyponym sets. Attempting a general solution for complete disambiguation is a very hard task which can be seen in [54].

2.3 New ontology creation approaches

In the following part of this section we will present the newest approaches presented in recent work. It should be stated here that these are not the only work done in ontology learning field, but they are a few of the good novel methods presented.

2.3.1 An iterative view

Brewster et al. in their paper [14] present a novel approach to ontology learning which takes an iterative view of knowledge acquisition for ontologies. Their incremental, weakly-supervised approach views OL as a process involving three resources: the corpus of texts, the extraction patterns set (a set of lexico-syntactic textual patterns), and the ontology (a set of RDF triples). And the system seeks to maintain these in equilibrium. As events occur which disturb this equilibrium, actions are triggered to re-establish a balance between the resources.

They provide a measure which classifies the suitability of ontological concepts and relations to generate patterns from the documents, which is formalized by assigning the relationship of any resource to the domain a confidence value, which they name "resource confidence" (RC). They also present a gold standard based evaluation of the final output of the system, the intermediate output

showing the iterative process and a comparison of performance using different seed input. Therefore, it receives corpus text as input. It processes them and creates a domain ontology. It can be considered as an automatic ontology creation tool. The ontology created goes under the subclass of ontology in our framework.

2.3.2 A multi-lingual view

Hans Hjelm in his PhD thesis [57] presents the development of a new evaluation measure, which remedies some of the problems regarding robustness and predictability. Also he provides a comparison of distributional similarity models and a statistical word alignment system on the task of bilingual dictionary extraction, as well as the introduction of an ensemble method for combining the two approaches. His main contribution was how to merge information across different languages. So, he covered three main topics within the field of ontology learning. First, he examined which sources of information are useful within an ontology learning system and how the information sources can be combined effectively. Secondly, he did this with a special focus on cross-language text collections, to see if he could learn more from studying several languages at once, than he could from a single-language text collection. And finally, he investigated new approaches to formal and automatic evaluation of the quality of a learned ontology. All in all he didn't provide a novel ontology learning methodology but he has experimented and evaluated quite a number of different approaches and presented their results.

2.3.3 A web based knowledge acquisition view

Ruenes in his PhD thesis [35] presents a novel approach for ontology learning, introducing new methods for knowledge acquisition from the Web. He presented an automatic, unsupervised and domain independent approach to domain ontology learning. In the ontology building process, the following methods have been developed: 1) extraction and selection of domain related terms, organizing them in a taxonomical way; 2) discovery and label of non-taxonomical relationships between concepts; 3) additional methods for improving the final structure, including the detection of named entities, class features, multiple inheritance and also a certain degree of semantic disambiguation. He has evaluated his approach and achieved good quality results. In short, it receives text as input. It processes them and creates a domain ontology. It can be considered as an automatic ontology creation tool. The ontology created goes under the subclass of ontology in our framework.

2.3.4 A process engineering view

Simperl et al. in their paper [58] aim at contributing to the operationalization of the ontology learning process by introducing a methodology describing the major coordinates of these processes in terms of activities, actors, inputs, outputs and support tools. Their methodology distinguishes among the following eight process stages: Feasibility study, requirements specification, selection of information sources, selection of ontology learning, methods and tools, learning preparation, learning execution, ontology evaluation, and ontology integration. They state that depending on the outcomes of a particular process stage the learning workflow can be executed in a linear or iterative manner. Overview of their learning-driven ontology engineering process can be seen in [58].

2.3.5 A design pattern view

Ontology Patterns can be used for ontology design. The notion of ontology design patterns (OP) was first presented by Gangemi [59]. Presutti et al. in [60] present the basic building blocks that are used in ontology design which are named Content Ontology Design Patterns (CP). CPs are small ontologies that mediate between use cases (problem types) and design solutions. Presutti et al. in [61] have identified several types of OPs, and have grouped them into six families: Structural OPs, Correspondence OPs, Content OPs (CPs), Reasoning OPs, Presentation OPs, and Lexico- Syntactic OPs. OPs help in creating better designed ontologies. Work on automating the ontology creation process with the help of OPs has been investigated by Maynard et al. [18] and Blomqvist [62]. Also work on defining a pattern language for ontology design patterns has been presented by Noppens et al. [63].

In this section we have reviewed the ontology creation processes and the new approaches (Brewster's iterative view, Hjelm's multi-lingual view, Ruenes's web based knowledge acquisition view, Simperl's process engineering view, and Gangemi's design pattern view) that have started to take shape. It is too early to precisely state whether these new approaches are good enough, but their results are somewhat assuring.

3   ONTOLOGY LEARNING TOOLS: STATE OF THE ART

It needs to be noted here that the presented list of tools is not intended to be an exhaustive and complete review of all work done so far but we have to the best of our knowledge tried to make it as complete as possible. We divide ontology applications into two different categories. First, those which mainly focus on ontology learning from plain text and second, those that mainly focus on ontology learning from semi-structured text. In each of the following sub sections we will introduce OL applications presented between the years of 2004 and 2012 with the intention of familiarizing the reader of what has happened in ontology learning during this 8 year period. The reason that we will not present ontology learning applications which emphasize on structured input is that little work has been done in this field. This is due to the unavailability of both structured input and standard structure for the input provided. Each application has its own standard and input, and this makes comparison and evaluation of those applications very difficult.

In our previous paper [2] we presented a framework for classifying and comparing OL systems as presented in the following figure 2. In this paper we intend to take a closer look at the *result* section (outlined in figure 2), especially the *ontology* section. In our extended framework we investigate the ontology created in three different sub dimensions, as seen in figure 3.

In the ontology *definition type* section of the framework the following are considered:

- By *terms and key phrases* we mean those applications that create a set of terms and key phrases as output and have considered it as an ontology.

- By *taxonomy* we mean those applications that create a taxonomy (a set of concepts related by only taxonomic relations) and have considered it as an ontology.

- By *ontology* we mean those applications that create a set of concepts, taxonomic and non-taxonomic relations, axioms and a specific domain of interest and call it an ontology.

- By *axioms* we mean those applications that create a set of rules (assertions) as an output and have considered it as ontology. They can also be considered as rules.

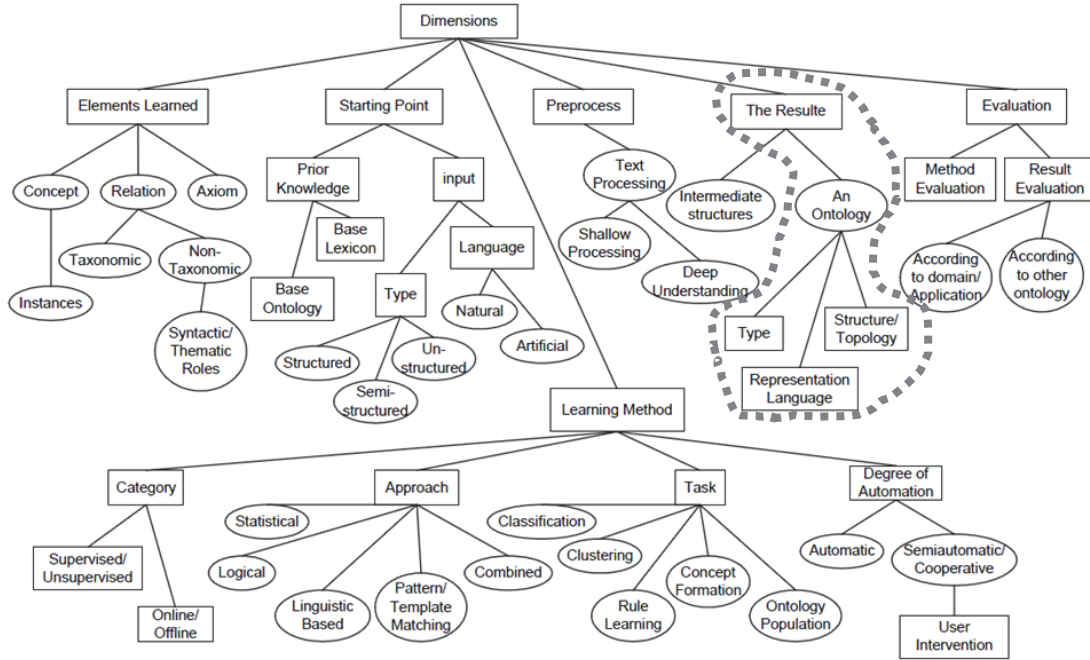Figure 2 Framework dimensions, sub-dimensions and values
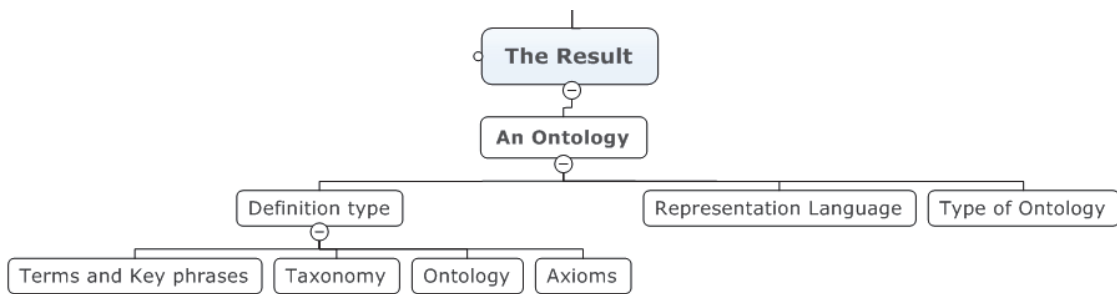

Figure 3 Extension to our previous framework

As it can be seen we have extended our previous framework focusing on the output (*result*) section for which an ontology is created. We have renamed the *Type* and *Structure/Topology* sections to *Type of ontology* and *Definition type* for more clarity respectively. In [1] we have presented and discussed the reasons of why the output of an ontology based system should be clearly specified whether it is a

classification of terms, or just a hierarchy of concepts, or really an ontology with rich semantic relations, or just a set of rules and axioms. The reason that the above situations occur is due to the fact that each ontology created may be at a different level of semantic richness. This is the reason we believe that when evaluating OL systems it should be specified which level of ontology is really created. Here the question arises that if you create an ontology which is just a hierarchy of concepts, is it correct to call it ontology or should we name it taxonomy? In the mentioned paper we have discussed this problem and presented a new representation model to support ontology development which we named Cognibase. The answer to the question above is not the concept of this paper, so we will not discuss it any further. In the scope of this paper and our framework we will specify the ontology created into four types (terms, taxonomy, ontology, and axioms). In the following subsections we will present OL applications (tools) which have created ontologies from unstructured and semi-structured texts.

3.1 OL from unstructured text

In this section we review different ontology learning applications that mainly use text as their input from years 2004 to 2012.

3.1.1 Application from 2004 to 2006

3.1.1.1 Bole

The BOLE platform [64] takes advantage of the automatic acquisition methods. It enables creating the core taxonomy of a bio-ontology sub domain in the bottom-up manner, from ontologies with a very simple structure to more complex ones, in a continual iterative process. It is also able to extend, refine and update bio-ontologies with respect to new data. Virtually any method of automated knowledge acquisition can be employed because of its modularity and flexibility. In short, BOLE is primarily intended for autonomous creation and management of domain specific bio-ontologies. The bottom- up approach to the ontology acquisition is emphasized. It can be classified as an automatic method of ontology learning from text. The structure created goes under the subclass of ontology in our framework.

3.1.1.2 OLE

It is very similar to BOLE. OLE stands for Ontology LEarning [65]. The purpose of the project is to develop a system for bottom-up automatic generation and merging of domain specific ontologies, representing particular domains of human scientific knowledge. In order to gain ontology concepts from textual resources, various methods can be used. Automatic acquisition of semantic relations is examined. It can be classified as an automatic method of ontology learning from text. The structure created goes under the subclass of ontology in our framework.

3.1.1.3 Text2Onto

Text2Onto presented in [66] is a tool and framework for ontology learning from textual resources. Text2Onto is a complete re-design and re-engineering of their previous system TextToOnto[67]. It represents the learned knowledge at a meta level in the form of instantiated modeling primitives via a so called Probabilistic Ontology Model (POM). User interaction is added to the framework. They implement strategies for data-driven change discovery, and avoid re-processing the whole corpus from

scratch each time it changes. It can be classified as an automatic method of ontology learning from unstructured text. The structure created goes under the subclass of ontology in our framework.

3.1.2 Applications from 2006 to 2008

3.1.2.1 Englmeier et al.'s tool

Englmeier and his colleagues in their paper [68] present a method that automatically builds a domain concept hierarchy from a textual corpus. Their approach is based on n-gram detection to extract term candidates. And the selected terms are organized in order to derive a concept hierarchy. The longest terms (longest n-grams in terms of words) are considered as leaves of the hierarchy and their hyperonyms are either automatically built from the n-gram components or selected from the other extracted n-grams. Then the upper level of the hierarchy is automatically extracted from a general resource (WordNet). They have also evaluated the different steps of the method using a Wikipedia sub-collection. In short, it receives corpus text as input. It processes them and creates a concept hierarchy. It can be considered as an automatic concept hierarchy creation tool. The structure created goes under the subclass of taxonomy in our framework.

3.1.2.2 Isolde

The ISOLDE (Information System for Ontology Learning and Domain Exploration) system [69] generates a domain ontology by extracting class candidates from the linguistic context of a given set of ontology instances and by deriving further knowledge on these class candidates from available web resources. It is based on techniques for unsupervised named-entity recognition. It finds additional information on the extracted classes in order to organize them into a taxonomy or full ontology. Therefore, it is a system for web based ontology learning that uses web resources such as Wikipedia and Wiktionary in combination with a domain corpus, a general purpose named-entity tagger and a seed or 'base' ontology to derive a domain ontology. Their experiments show that best results may be obtained from semi-structured data resources (e.g. web dictionaries). It can be classified as an automatic method of ontology learning from both semi-structured and unstructured text. The structure created goes under the subclass of ontology in our framework.

3.1.2.3 Sole

SOLE [70] is almost the same as OLE and BOLE but adds web interface support. It can be classified as an automatic method of ontology learning from text. The structure created goes under the subclass of ontology in our framework.

3.1.2.4 Dynamo

Dynamo [71] which is short for DYNAMic Ontologies is a tool based on an adaptive multi-agent system to construct and maintain an ontology from a domain-specific set of texts. Their adaptive multi-agent system is used both to represent the ontology itself and to produce the ontology. It is a semi-automatic tool since it needs the assistance of an ontologist in its creation process. As input, Dynamo uses the results of terminological and syntactic analysis of texts carried out by the term extractor Syntex[72]. And as output, Dynamo gives the ontologist a hierarchical organization of concepts (which can be seen as a *is-a* hierarchy). So, Dynamo only builds domain-specific taxonomies. Therefore, it receives as input preprocessed text from the term extractor. It processes them and creates

a domain-specific taxonomy. It can be considered as a semi-automatic ontology creation tool. The structure created goes under the subclass of taxonomy in our framework.

.3.1.3 Application from 2008 to 2012

3.1.3.1 LexO

LExO (Learning EXpressive Ontologies) is a tool which basically relies on KAON2 [73], an ontology management infrastructure for OWL DL, and a Minipar dependency parser[74], [75]. Given a natural language definition of a class, LExO starts by analyzing the syntactic structure of the input sentence. A set of manually engineered transformation rules which are then applied to the XML-based parse tree, make use of XPath expressions for transforming the dependency structure into one or more OWL DL axioms. In short, it receives corpus text as input. It processes them and creates an expressive ontology. It can be considered as a semi-automatic ontology creation tool. The structure created goes under the subclass of axioms in our framework.

3.1.3.2 OntoLancs

OntoLancs [76] is a flexible framework for the ontology learning process. This framework is designed as a cyclical process to experiment with different techniques and combinations of techniques. An ontology engineer can decide techniques or combinations which will be used to extract concepts and turn them into an ontology. The framework provides support to evaluate the usefulness and accuracy of different techniques and possible combinations of techniques into specific processes. It can be classified as an automatic method of ontology learning from text. The structure created goes under the subclass of ontology in our framework.

3.1.3.3 Wong et al.'s tool

Wong et al. in [77] present a hybrid approach which combines techniques in lexical simplification, word disambiguation and association inference for acquiring coarse-grained relations between potentially ambiguous and composite terms using only dynamic Web resources. The output created is a lightweight ontology. Giunchiglia et al. [78] define a lightweight ontology in its simplest version, as an ontology consisting of backbone taxonomies only (taxonomy with mainly is-a relations). Therefore lightweight ontologies from our point of view can be called taxonomies. They have used lexical simplification, word disambiguation and association inference techniques to learn relations in between terms. This is done by an iterative process of term matching and term resolution. The main contribution of their paper is it can easily handle complex and ambiguous terms which were not stated in their background knowledge from web sources (e.g. Wikipedia). So, it receives as input a set of domain terms. It processes them and creates a lightweight ontology. It can be considered as an automatic ontology creation tool. The structure created goes under the subclass of taxonomy in our framework.

3.1.3.4 OntoCase

OntoCase [79] is a pattern-based ontology construction approach and tool. They based their semi-automatic approach on the principles of case-based reasoning (CBR). In the OntoCase approach the case base corresponds to a pattern catalogue (pattern base), containing both ontology design patterns and architecture patterns. The first OntoCase phase performs case retrieval and analyses the input (text corpus) and matches it to the pattern base, to select appropriate patterns. The second phase, case reuse,

constitutes the process of reusing retrieved patterns and constructing a first version of the ontology. The third phase concerns revision of the ontology to improve the fit and the ontology quality. The final phase includes the discovery of new patterns as well as storing pattern feedback. We refer the reader to [80] for a more detailed specification of the tool and approach. It can be classified as a semi-automatic method of ontology creation from unstructured text. The structure created goes under the subclass of ontology in our framework.

3.1.3.5 ORE

ORE (Ontology Repair and Enrichment) [81] tool provides guidance for knowledge engineers to detect problems in their knowledge base and repair them. ORE also provides suggestions for extending a knowledge base by using supervised machine learning on the instance data in the knowledge base. It integrates state of the art methods from ontology debugging and supervised learning in OWL. They have used Inductive Logic Programming [82] as a means of solving the learning class problem. It can be considered as a semi-automatic ontology enrichment tool. The structure created goes under the subclass of axioms in our framework.

3.1.3.6 LexOnt

LexOnt [83] is a semi-automatic ontology generator that helps in the ontology creation of a high-level service ontology. It uses the Programmable Web directory of services[84], Wikipedia[85], WordNet and the current state of the generated ontology to suggest relevant terms that may be incorporated within the ontology. LexOnt builds the ontology iteratively, by interacting with the user, taking in terms that the user has chosen, adding these to the ontology and ranking terms according to the external knowledge base. It is a plugin tab for the Protégé [86] ontology editor which interacts with the user to facilitate the ontology creation process. It accepts unstructured text as input. It can be considered as a semi-automatic ontology construction tool. The structure created goes under the subclass of ontology in our framework.

3.1.3.7 OntoCmaps

OntoCmaps[87] is a domain-independent and ontology learning tool that extracts deep semantic representations from corpora. OntoCmaps generates rich conceptual representations in the form of concept maps and proposes an innovative filtering mechanism based on Degree (number of edges from and to a given term), Betweenness (number of shortest paths that pass through a term), PageRank (fraction of time spent visiting a term) and Hits (ranks terms according to the importance of hubs and authorities[88]) metrics from graph theory. It relies on the inner structure of graphs to identify the important elements without using any other knowledge source. OntoCmaps resulting ontology was compared with the ontology created by Text2Onto and has achieved better results. It accepts both unstructured corpus text and other concept maps as input. It can be considered as a semi-automatic ontology construction tool. The structure created goes under the subclass of ontology in our framework.

3.2 OL from semi-structured text

In this section we review different ontology learning applications that mainly use semi-structured text as their input from years 2004 to 2012.

3.2.1 Applications from 2004 to 2006

3.2.1.1 AEON

AEON [89] is a tool which automatically tags concepts with appropriate OntoClean meta properties. AEON makes use of the World Wide Web as the biggest existing source of common sense knowledge. It is a tool which matches lexico-syntactic patterns on the Web to obtain positive and negative evidence for rigidity, unity, dependence and identity of concepts in an RDFS or OWL ontology. It can be considered as an automatic ontology learning tool. The structure created goes under the subclass of ontology in our framework.

3.2.1.2 RelExt

RelExt [34] is capable of automatically identifying relevant triples (pairs of concepts connected by a relation) over concepts from an existing ontology. RelExt works by extracting relevant verbs and their grammatical arguments (i.e. terms) from a domain-specific text collection and computes corresponding relations through a combination of linguistic and statistical processing. It is concerned with the extraction of domain specific verbal relations other than *is-a*. It relies on an already existing ontology for the domain and extends it. It has been implemented and used in the football domain in World Cup 2006 in Germany. It can be classified as an automatic method of ontology learning from semi-structured text. The structure created goes under the subclass of rules in our framework.

3.2.2 Applictions from 2006 to 2008

3.2.2.1 ISOLDE

ISOLDE [69] has been explained once previously. It has been presented again here because it uses both unstructured and semi-structured text as inputs to its ontology learning process. The structure created goes under the subclass of ontology in our framework.

3.2.2.2 OntoGen

OntoGen [90], a tool which helps the user by suggesting the possible new topics and visualizing the topic ontology created in real time. It aims at assisting the user in a fast semi-automatic construction of the topic ontology from a large document collection and helps by automatically assigning documents to the topics, by suggesting names for the topics, etc. OntoGen uses statistical analysis and clustering methods and also uses text collections as its input sources. OntoGen can be considered as a semi-automatic ontology creation tool. The structure created goes under the subclass of ontology in our framework.

3.2.3 Applications from 2008 to 2012

3.2.3.1 GALEON

The General Architecture for Learning Ontologies (GALEON) [91] is an open, extensible and domain independent architecture that automates the process of building or extending domain ontologies. It learns the new ontology or parts of it, using the given heterogeneous sources with minimum user intervention. Its architecture is composed of six main phases: processing, acquisition, action, consolidation, evaluation and knowledge augmentation. The user provides a set of sources that describe the domain of the ontology to be created and GALEON extracts and selects the core terminology which it calls candidate elements (CEs) to use in the ontology construction. It then uses a

set of hypotheses to relate CEs to actions which then include those CEs into the ontology. GALEON also accepts as an optional input an existing domain ontology that will be enriched as a result of the learning process. The provided knowledge acts as background knowledge of the domain and is used by the operators to improve the accuracy of the learning process. They have tested GALEON in the Universities and Economics domains achieving good results. As it can be seen it accepts heterogeneous sources as input. It has the capability to receive an ontology as input which is a structured input. As they have stated the ontology is an optional means of input to provide better accuracy therefore we will classify it as both unstructured and semi-structured data inputs. All of the process except the data sources input is done automatically. It only extracts hierarchy (taxonomic) relations so the structure created goes under the subclass of taxonomy in our framework.

After reviewing the presented tools and applications, for more clarity we have summarized them into table 1.

Table 1: The comparison of ontology tools

Application name	Year	Input type	Degree of automation	Output structure
BOLE	2005	Unstructured Text	Automatic	Ontology
OLE	2005	Unstructured Text	Automatic	Ontology
Text2Onto	2005	Unstructured Text	Automatic	Ontology
AEON	2005	Semi-structured Text	Automatic	Ontology
RelExt	2005	Semi-structured Text	Automatic	Axioms
Englmeier et al.	2006	Unstructured Text	Automatic	Taxonomy
ISOLDE	2006	Semi/Unstructured Text	Automatic	Ontology
SOLE	2006	Unstructured Text	Automatic	Ontology
OntoGen	2006	Semi-structured Text	Semi-automatic	Ontology
Brewster et al.	2007	Unstructured Text	Automatic	Ontology
Dynamo	2007	Unstructured Text	Semi-automatic	Taxonomy
Ruenes et al.	2007	Unstructured Text	Automatic	Ontology
LExO	2008	Unstructured Text	Semi-automatic	Axioms
OntoLancs	2008	Unstructured Text	Automatic	Ontology
GALEON	2008	Semi/Unstructured Text	Automatic	Taxonomy
Wong et al.	2009	Unstructured Text	Automatic	Taxonomy
OntoCase	2009	Unstructured Text	Semi-automatic	Ontology
ORE	2010	Unstructured Text	Semi-automatic	Axioms
OntoCmaps	2011	Semi/Unstructured Text	Semi-automatic	Ontology
LexOnt	2012	Unstructured Text	Semi-automatic	Ontology

4   Conclusion and Future Work

In this paper we updated our previous framework published in [2] by extending the ontology output section and examined OL applications from that aspect. In recent years plain text methods have added web based support and are changing their input sources from local resources to global web text resources such as Ruenes's approach or Luong's framework [92]. Also, it can be said that recent

applications have started to use both plain text and (semi) structured text as their inputs to improve their ontology learning results as seen in ISOLDE, GALEON, and OntoCmaps. Also text based applications are working on combining different ontology learning methods improve their results [92]. To be able to better evaluate these results a few of them have chosen to create frameworks to provide an environment to test different methods and compare their results such as in [76][92][93][94]. As for semi-structured methods, it can be seen that recent work has been done providing better preprocessed input such as better tagging in [89]. Recently work on creating different language WordNets have been seen such as Romanian[95], Japanese[96], Chinese[97], Persian[98][99] and Arabic[100] to name just a few. The growth of such WordNets will help ontology learning systems which use the Hjelm's multi-lingual view to create better ontologies.

So we believe that in the future more work can be done in the following fields: First, to produce tools to better transform plain text which is the most available source of input to create semi-structured and if possible structured text. By this a great amount of (semi) structured text input could be created and this would help to better feed and evaluate the semi-structured and structured text based ontology learning methods and applications. One approach that we recommend to achieve this is that more web based ontology learning applications need to be created to help, the creation and growth of ontologies. Second, although progress has been made over the last years, research has not yet reached the goal of fully automating the ontology development process. So there still needs to be work done in this section. Third, the development and test of more structured text approaches to help the process of extracting more and richer non-taxonomic relations are needed. Work on automatic learning of non-taxonomic relationships have been conducted but have not reached yet an advanced and mature development stage. Fourth, the establishment of formal, standard methods to evaluate ontology learning systems by proving their learning methods, accuracy, efficiency and completeness of the built ontology are needed [101]. Fifth, of the ontology learning subtasks, the one that is currently still in the earlier stage of development is the axiom learning and needs to be investigated more in the future. And last, pattern-based approaches to ontology construction have been presented in section 2.3.5, we believe that in the future more work in this field is needed. Because of the complexity of ontology development and the importance of the ontologies for the web and knowledge systems, we believe that ontology learning will remain an active and central field of research for quite some time.

Acknoledgments

References

[1]   A. Rahnama and A. Abdollahzadeh Barforoush, "Cognibase: a new representation model to support ontology development," in IADIS International Conference Information Systems (IS 2011), Avila, Spain, 2011.

[2]   M. Shamsfard and A. Abdollahzadeh Barforoush, "The state of the art in ontology learning: a framework for comparison," The Knowledge Engineering Review, vol. 18, no. 04, pp. 293–316, 2004.

[3]   L. Drumond and R. Girardi, "A Survey of Ontology Learning Procedures," presented at the 3rd Workshop On Ontologies And Their Applications (WONTO'2008), Salvador, Bahia, Brazil, 2008.

[4]   K. Dellschaft and S. Staab, "Measuring the similarity of concept hierarchies and its influence on the evaluation of learning procedures," Master's Thesis (Diplomarbeit), University of Koblenz-Landau, 2005.

[5]   G. A. Miller, "WordNet: a lexical database for English," Communications of the ACM, vol. 38, no. 11, pp. 39–41, 1995.

[6]   M. Sanderson, "Deriving concept hierarchies from text," in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 1999, pp. 206–213.

[7]   G. Bisson, C. Nedellec, and D. Canamero, "Designing clustering methods for ontology building: The Mo'K workbench," in Proceedings of the First Workshop on Ontology Learning OL, 2000, pp. 13–18.

[8]   S. A. Caraballo and E. Charniak, "Determining the specificity of nouns from text," in Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, 1999, pp. 63–70.

[9]   M. Berland and E. Charniak, "Finding parts in very large corpora," in Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, 1999, pp. 57–64.

[10]   M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in Proceedings of the 14th conference on Computational linguistics-Volume 2, 1992, pp. 539–545.

[11]   A. Auger and C. Barrière, "Pattern-based approaches to semantic relation extraction: A state-of-the-art," Terminology: international journal of theoretical and applied issues in specialized communication, vol. 14, no. 1, pp. 1–19, 2008.

[12]   Z. Kozareva, E. Riloff, and E. Hovy, "Semantic class learning from the web with hyponym pattern linkage graphs," Proceedings of ACL-08: HLT, pp. 1048–1056, 2008.

[13]   N. Aussenac-Gilles and M.-P. Jacques, "Designing and evaluating patterns for relation acquisition from texts with Cameleon," Terminology, vol. 14, no. 1, pp. 45–73, 2008.

[14]   C. A. Brewster, J. Iria, Z. Zhang, F. Ciravegna, L. Guthrie, and Y. Wilks, "Dynamic iterative ontology learning," in Proceedings of Recent Advances in Natural Language Processing, 2007.

[15]   V. Kashyap, C. Ramakrishnan, C. Thomas, and A. Sheth, "TaxaMiner: an experimentation framework for automated taxonomy bootstrapping," International Journal of Web and Grid Services, vol. 1, no. 2, p. 240, 2005.

[16]   C. Papatheodorou, A. Vassiliou, and B. Simon, "Discovery of Ontologies for Learning Resources Using Word-Based Clustering.," 2002.

[17]   P. Pantel and D. Lin, "Document clustering with committees," in Proceedings of the 25th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, New York, 2002, pp. 199–206.

[18]   D. Maynard, A. Funk, and W. Peters, "Using lexico-syntactic ontology design patterns for ontology creation and population," in Proc. of the Workshop on Ontology Patterns, 2009.

[19]   E. Morin and E. Martienne, "Using a Symbolic Machine Learning Tool to Refine Lexico-syntactic Patterns," in Machine Learning: ECML 2000: 11th European Conference on Machine Learning, Barcelona, Catalonia, Spain, May/June 2000. Proceedings, 2000, pp. 515–524.

[20]   C. Giuliano, A. Lavelli, and L. Romano, "Exploiting shallow linguistic information for relation extraction from biomedical literature," in Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics (EACL-2006), 2006, pp. 5–7.

[21]   A. M. Giuglea and A. Moschitti, "Shallow semantic parsing based on FrameNet, VerbNet and PropBank," in Proceeding of the 2006 conference on ECAI 2006: 17th European Conference on Artificial Intelligence August 29–September 1, 2006, Riva del Garda, Italy, 2006, pp. 563–567.

[22]   K. Erk and S. Pado, "Shalmaneser–a toolchain for shallow semantic parsing," in Proceedings of LREC, 2006, vol. 6.

[23]   V. Novacek, "Imprecise Empirical Ontology Refinement: Application to Taxonomy Acquisition," 2007.

[24]   A. Weichselbraun, G. Wohlgenannt, and A. Scharl, "Refining non-taxonomic relation labels with external structured data to support ontology learning," Data & Knowledge Engineering, vol. 69, no. 8, pp. 763–778, 2010.

[25]   F. M. Suchanek, G. Ifrim, and G. Weikum, "Combining linguistic and statistical analysis to extract relations from web documents," in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, 2006, pp. 712–717.

[26]   B. Zhang, D. Schmoyer, S. Kirov, and J. Snoddy, "GOTree Machine (GOTM): a web-based platform for interpreting sets of interesting genes using Gene Ontology hierarchies," BMC bioinformatics, vol. 5, no. 1, p. 16, 2004.

[27]   P. Khatri and S. Drăghici, "Ontological analysis of gene expression data: current tools, limitations, and open problems," Bioinformatics, vol. 21, no. 18, pp. 3587–3595, 2005.

[28]   M. Kavalec, A. Maedche, and V. Svatek, "Discovery of lexical entries for non-taxonomic relations in ontology learning," Lecture notes in computer science, pp. 249–256, 2004.

[29]   P. Cimiano and J. Wenderoth, "Automatically learning qualia structures from the web," Ann Arbor, vol. 100, 2005.

[30]   R. Girju and D. I. Moldovan, "Text mining for causal relations," in Proceedings of the FLAIRS Conference, 2002, pp. 360–364.

[31]   C. Biemann, S. Bordag, and U. Quasthoff, "Automatic acquisition of paradigmatic relations using iterated co-occurrences," in Proceedings of LREC, 2004, vol. 4.

[32]   D. Faure and C. Nedellec, "Asium: Learning subcategorization frames and restrictions of selection," in ECML98, Workshop on Text Mining, 1998, vol. 409, p. 410.

[33]   M. L. Reinberger and P. Spyns, "Discovering knowledge in texts for the learning of dogma-inspired ontologies," in ECAI Workshop on Ontology Learning and Population, Valencia, Spain, 2004.

[34]   A. Schutz and P. Buitelaar, "Relext: A tool for relation extraction from text in ontology extension," Lecture notes in computer science, vol. 3729, p. 593, 2005.

[35]   D. Sanchez Ruenes, "Domain Ontology learning from the Web," PhD thesis, Universidad Politecnica de Catalufia, Departamento de Lenguajes y Sistemas Informaticos, 2007.

[36]   C. Antunes, "Mining rules in the Onto4AR framework for updating domain ontologies," in Proceedings of the IADIS European Conference on Data Mining (ECDM 2007), Lisbon, 2007, pp. 95–101.

[37]   A. Maedche and S. Staab, "Discovering Conceptual Relations from Text," Cognitive Science, no. iv, pp. 1–17, 2000.

[38]   O. Etzioni, M. Cafarella, D. Downey, A. M. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates, "Unsupervised named-entity extraction from the web: An experimental study," Artificial Intelligence, vol. 165, no. 1, pp. 91–134, 2005.

[39]   D. Downey, M. Broadhead, and O. Etzioni, "Locating complex named entities in web text," in Proc. of IJCAI, 2007, vol. 2007.

[40]   C. Giuliano, A. Lavelli, and L. Romano, "Relation extraction and the influence of automatic named-entity recognition," ACM Transactions on Speech and Language Processing (TSLP), vol. 5, no. 1, p. 2, 2007.

[41]   J. Volker, P. Haase, and P. Hitzler, "Learning Expressive Ontologies," in Ontology learning and population: bridging the gap between text and knowledge, Ios Pr Inc, 2008, pp. 45–69.

[42]		J. Völker, P. Hitzler, and P. Cimiano, "Acquisition of OWL DL axioms from lexical resources," The Semantic Web: Research and Applications, pp. 670–685, 2007.

[43]		J. Völker and S. Rudolph, "Lexico-logical acquisition of OWL DL axioms," Formal Concept Analysis, pp. 62–77, 2008.

[44]		R. Navigli and P. Velardi, "Ontology enrichment through automatic semantic annotation of on-line glossaries," Managing Knowledge in a World of Networks, pp. 126–140, 2006.

[45]		R. Navigli, P. Velardi, P. Buitelaar, and P. Cimiano, "From Glossaries to Ontologies: Extracting Semantic Structure from Textual Definitions," in Ontology Learning and Population: Bridging the Gap between Text and Knowledge, Ios Pr Inc, 2008, pp. 71–87.

[46]		D. Moldovan, C. Clark, S. Harabagiu, and D. Hodges, "Cogex: A semantically and contextually enriched logic prover for question answering," Journal of Applied Logic, vol. 5, no. 1, pp. 49–69, 2007.

[47]		A. Mikheev, M. Moens, and C. Grover, "Named entity recognition without gazetteers," in Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics, 1999, pp. 1–8.

[48]		M. Collins and Y. Singer, "Unsupervised models for named entity classification," in Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, 1999, pp. 189–196.

[49]		S. Lamparter, M. Ehrig, and C. Tempich, "Knowledge extraction from classification schemas," Lecture Notes in Computer Science, pp. 618–636, 2004.

[50]		P. Cimiano and S. Staab, "Learning by googling," ACM SIGKDD explorations newsletter, vol. 6, no. 2, pp. 24–33, 2004.

[51]		K. Schnattinger and U. Hahn, "Quality-based learning," in Proc. of the ECAI'98, 1998, pp. 160–164.

[52]		M. Pasca, "Acquisition of categorized named entities for web search," in Proceedings of the thirteenth ACM international conference on Information and knowledge management, 2004, pp. 137–145.

[53]		A. Burton-Jones, V. C. Storey, V. Sugumaran, and S. Purao, "A heuristic-based methodology for semantic augmentation of user queries on the Web," Lecture notes in computer science, pp. 476–489, 2003.

[54]		R. Mihalcea and P. Edmonds, "Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text," Association for Computational Linguistics, Barcelona, Spain, 2004.

[55]		N. Ide and J. Véronis, "Introduction to the special issue on word sense disambiguation: the state of the art," Computational linguistics, vol. 24, no. 1, pp. 2–40, 1998.

[56]		P. Rosso, M. Montes, D. Buscaldi, A. Pancardo, and A. Villase, "Two Web-based Approaches for Noun Sense Disambiguation," in Int. Conf. on Comput. Linguistics and Intelligent Text Processing, CICLing-2005, Springer Verlag, LNCS (3406), Mexico DF, Mexico, 2005, pp. 261–273.

[57]		H. Hjelm, "Cross-language ontology learning: incorporating and exploiting cross-language data in the ontology learning process," Ph.D., Stockholm University, Sweden, 2009.

[58]		E. Simperl, C. Tempich, and D. Vrandecic, "A methodology for ontology learning," in Ontology Learning and Population: Bridging the Gap between Text and Knowledge, 2008, pp. 225–249.

[59]		A. Gangemi, "Ontology design patterns for semantic web content," The Semantic Web–ISWC 2005, pp. 262–276, 2005.

[60]		V. Presutti and A. Gangemi, "Content ontology design patterns as practical building blocks for web ontologies," Conceptual Modeling-ER 2008, pp. 128–141, 2008.

[61]		V. Presutti, A. Gangemi, S. David, G. A. de Cea, M. Suarez-Figueroa, E. Montiel-Ponsoda, and M. Poveda, "Library of design patterns for collaborative development of networked ontologies," NeOn project, Deliverable D2.5.1, 2008.

[62]   E. Blomqvist, "Semi-automatic Ontology Engineering Using Patterns," The Semantic Web, vol. 4825, no. 3, pp. 911–915, 2007.

[63]   O. Noppens and T. Liebig, "Ontology Patterns and Beyond," in Workshop on Ontology Patterns, 2009, p. 179.

[64]   V. Novacek and P. Smrz, "Bole-a new bioontology learning platform," in Proceedings of the Workshop on Biomedical Ontologies and Text Processing, 2005.

[65]   V. Novacek and P. Smrz, "OLE—a new ontology learning platform," in Proceedings of the International Workshop on Text Mining, 2005.

[66]   P. Cimiano and J. Völker, "Text2Onto-A Framework for Ontology Learning and Data-driven Change Discovery," in Proceedings of the 10th International Conference on Applications of Natural Language to Information Systems-NLDB, 2005, vol. 5, pp. 15–17.

[67]   A. Maedche and S. Staab, "The text-to-onto ontology learning environment," in Software Demonstration at ICCS-2000-Eight International Conference on Conceptual Structures, 2000.

[68]   K. Englmeier, F. Murtagh, and J. Mothe, "Domain Ontology: Automatically Extracting and Structuring Community Language from Texts," IADIS Applied Computing, Spain, Espagne, 2007.

[69]   N. Weber and P. Buitelaar, "Web-based ontology learning with isolde," in Proc. of the ISWC Workshop on Web Content Mining with Human Language Technologies, 2006.

[70]   V. Novacek, "Using SOLE—Simple OLE Web Interface," Brno: Faculty of Informatics, Masaryk University, User Manual, 2006.

[71]   K. Ottens, M. P. Gleizes, and P. Glize, "A multi-agent system for building dynamic ontologies," in Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems, 2007, p. 227.

[72]   N. Aussenac-Gilles and D. Sörgel, "Text analysis for ontology and terminology engineering," Applied Ontology, vol. 1, no. 1, pp. 35–46, 2005.

[73]   B. Motik and R. Studer, "KAON2–A Scalable Reasoning Tool for the Semantic Web," in Proceedings of the 2nd European Semantic Web Conference (ESWC'05), Heraklion, Greece, 2005.

[74]   L. Dekang, "Dependency-based evaluation of MINIPAR," in Proceedings of the Workshop on the Evaluation of Parsing Systems, Granada, Spain, 1998, pp. 298–312.

[75]   D. Lin, "Dependency-based evaluation of MINIPAR," Treebanks, pp. 317–329, 2003.

[76]   R. Gacitua, P. Sawyer, and P. Rayson, "A flexible framework to experiment with ontology learning techniques," Knowledge-Based Systems, vol. 21, no. 3, pp. 192–199, 2008.

[77]   W. Wong, W. Liu, and M. Bennamoun, "Acquiring Semantic Relations using the Web for Constructing Lightweight Ontologies," in Proceedings of the 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), Bangkok, Thailand, 2009.

[78]   F. Giunchiglia and I. Zaihrayeu, "Lightweight ontologies," University of Trento, Department Of Information And Communication Technology, Technical Report DIT-07-071, 2007.

[79]   E. Blomqvist, "OntoCase-Automatic Ontology Enrichment Based on Ontology Design Patterns," Engineering, no. 2003, pp. 65–80, 2009.

[80]   E. Blomqvist, "Semi-automatic ontology construction based on patterns," Ph.D. thesis, Linkopings university, Sweden, 2009.

[81]   J. Lehmann and L. Bühmann, ORE - A tool for repairing and enriching knowledge bases, vol. 6497 LNCS. 2010.

[82]   S. H. Nienhuys-Cheng and R. De Wolf, Foundations of inductive logic programming, vol. 1228. Springer Verlag, 1997.

[83]   K. Arabshian, P. ₍Danielsen₎, and S. Afroz, "LexOnt: A Semi-Automatic Ontology Creation Tool for Programmable Web," in 2012 AAAI Spring Symposium Series, 2012.

[84]   "Programmable Web." [Online]. Available: http://www.programmableweb.com. [Accessed: 24-Jan-2012].

[85]   R. S. S. In, Wikipedia [updated 2007 January 31; cited 2007 April 4]. .

[86]   "The Protégé Ontology Editor and Knowledge Acquisition System." [Online]. Available: http://protege.stanford.edu/. [Accessed: 20-May-2012].

[87]   A. Zouaq, D. Gasevic, and M. Hatala, "Towards open ontology learning and filtering," Information Systems, vol. 36, no. 7, pp. 1064–1081, Nov. 2011.

[88]   J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," Journal of the ACM (JACM), vol. 46, no. 5, pp. 604–632, 1999.

[89]   J. Volker, D. Vrandecic, and Y. Sure, "Automatic evaluation of ontologies (AEON)," Lecture notes in computer science, vol. 3729, p. 716, 2005.

[90]   B. Fortuna, D. Mladenic, and M. Grobelnik, "Semi-automatic construction of topic ontologies," Lecture Notes in Computer Science, vol. 4289, p. 121, 2006.

[91]   D. Manzano-Macho, A. Gómez-Pérez, and D. Borrajo Millán, "Unsupervised and domain independent ontology learning: combining heterogeneous sources of evidence," 2008.

[92]   H. P. Luong, S. Gauch, Q. Wang, and A. Maglia, "An Ontology Learning Framework Using Focused Crawler and Text Mining," International Journal On Advances in Life Sciences, vol. 1, no. 2 and 3, pp. 99–109, 2009.

[93]   P. Haase, H. Lewen, R. Studer, D. T. Tran, M. Erdmann, M. d' Aquin, and E. Motta, "The neon ontology engineering toolkit," WWW, 2008.

[94]   A. Wróblewska, T. Podsiadły-Marczykowska, R. Bembenik, G. Protaziuk, and H. Rybiński, "Methods and Tools for Ontology Building, Learning and Integration – Application in the SYNAT Project," in Intelligent Tools for Building a Scientific Information Platform, vol. 390, R. Bembenik, L. Skonieczny, H. Rybinski, and M. Niezgodka, Eds. Springer Berlin / Heidelberg, 2012, pp. 121–151.

[95]   D. Tufiş, R. Ion, L. Bozianu, A. Ceauşu, and D. Ştefănescu, "Romanian WordNet: Current State, New Applications and Prospects," in Proceedings of 4th Global WordNet Conference, GWC-2008 (Attila Tanacs, Dora Csendes, Veronika Vincze, Christiane Fellbaum, Piek Vossen, eds.), Szeged, 2008, pp. 441–452.

[96]   H. Isahara, F. Bond, K. Uchimoto, M. Utiyama, and K. Kanzaki, "Development of the Japanese WordNet.," 2010.

[97]   C. R. Huang, S. K. Hsieh, J. F. Hong, Y. Z. Chen, I. L. Su, Y. X. Chen, and S. W. Huang, "Chinese Wordnet: Design, Implementation, and Application of an Infrastructure for Cross-lingual Knowledge Processing," Zhong Guo YuWen, vol. 24, no. 2, 2010.

[98]   M. Montazery and H. Faili, "Automatic Persian WordNet Construction," in Proceedings of the 23rd International Conference on Computational Linguistics: Posters, 2010, pp. 846–850.

[99]   M. Montazery and H. Faili, "Unsupervised Learning for Persian WordNet Construction," Proc. of RANLP'11, 2011.

[100]   H. Rodríguez, D. Farwell, J. Farreres, M. Bertran, M. Alkhalifa, M. A. Martí, W. Black, S. Elkateb, J. Kirk, A. Pease, and others, "Arabic wordnet: Current state and future extensions," in Proceedings of The Fourth Global WordNet Conference, Szeged, Hungary, 2008.

[101]   M. Shamsfard and A. Barforoush, "Learning ontologies from natural language texts," International Journal of Human-Computer Studies, vol. 60, no. 1, pp. 17–63, 2003.