Ontology

In other words, every knowledge base has to be committed to a conceptualization, whether implicitly or explicitly. This conceptualization is what we refer to as ontologies [Gruber 1993]. With this in mind, knowledge bases can be created by extracting the relevant instances from information to populate the corresponding ontologies, a process known as *ontology population* or *knowledge markup*. Ontology learning from text is then essentially the process of deriving high-level concepts and relations as well as the occasional axioms from information to form an ontology.

ONTOLOGY LEARNING FROM TEXT

Ontology learning from text is the process of identifying terms, concepts, relations, and optionally, axioms from textual information and using them to construct and maintain an ontology. Techniques from established fields, such as information retrieval, data mining, and natural language processing, have been fundamental in the development of ontology learning systems.

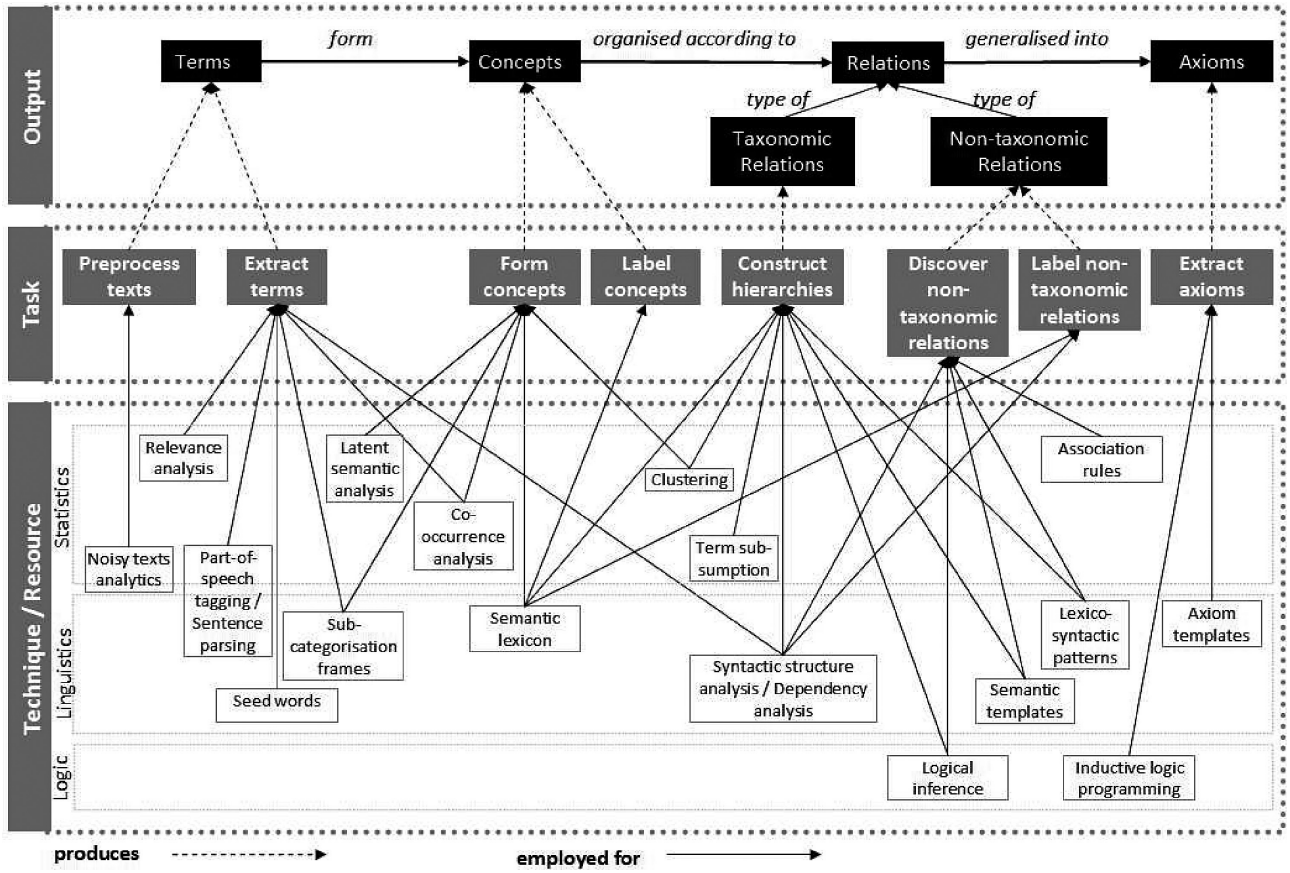
3.1. Lightweight versus Formal Ontologies

Ontologies can be thought of as directed graphs consisting of *concepts* as nodes and *relations* as the edges between the nodes. A concept is essentially a mental symbol often realized by a corresponding lexical representation (i.e., natural language name). For instance, the concept *"food"* denotes the set of all substances that can be consumed for nutrition or pleasure. In Information Science, an ontology is a *"formal, explicit specification of a shared conceptualisation"* [Gruber 1993]. This definition imposes the requirement that the names of concepts and how the concepts are related to one another have to be explicitly expressed and represented using formal languages, such as Web Ontology Language (OWL).1 An important benefit of a formal representation is the ability to specify *axioms* for reasoning in order to determine validity and to define constraints in ontologies. Moreover, a formal ontology is natural language independent or, in other words, does not contain lexical knowledge

3.2. Outputs and Tasks in Ontology Learning

There are five types of output in ontology learning, namely, *terms*, *concepts*, *taxonomic relations*, *non-taxonomic relations*, and *axioms*.

Terms are used to form concepts which in turn are organized according to relations. Relations can be further generalized to produce axioms. The solid arrows are used to relate techniques to tasks (i.e., technique X employed for task Y), while the dotted arrows indicate the connections between tasks and outputs (i.e., task X produces output Y).



Terms

Terms are the most basic building blocks in ontology learning. Terms can be simple (i.e., single word) or complex (i.e., multi word), and are considered as lexical realizations of everything important and relevant to a domain. The main tasks associated with terms are to *preprocess texts* and *extract terms*. The preprocessing task ensures that the input texts are in a format supported by the ontology learning system. Some of the techniques relevant to preprocessing include noisy text analytics and the extrac- tion of relevant contents from webpages (i.e., boilerplate removal). The extraction of terms, known as *term extraction* or *keyphrase extraction* [Medelyan and Witten 2005], typically begins with tokenization or part-of-speech tagging to break texts into smaller constituents. Statistical or probabilistic measures are then used to determine the col- locational stability of a noun sequence to form a term, also known as *unithood*, and the relevance or specificity of a term with respect to a domain, also known as *termhood*.

Concepts

Concepts can be abstract or concrete, real or fictitious. Broadly speaking, a concept can be anything about which something is said. Concepts are formed by grouping similar terms. The main tasks are therefore to *form concepts* and *label concepts*. The task of forming concepts involves discovering the variants of a term and grouping them together. Term variants can be determined using predefined background knowledge, syntactic structure analysis, or through clustering based on some similarity measures. Syntactic structure analysis, for instance, uses the common head such as *“tart”* of complex terms to form a unifying concept to encompass the corresponding longer strings *“egg tart”*, *“French apple tart”*, and *“chocolate tart”*. If labels are required for the concepts, existing background knowledge, such as WordNet, may be used to find the name of the nearest common ancestor.

Relations

Relations are used to model the interactions between the concepts in an ontology. There are two types of relations, namely, *taxonomic relations* and *non-taxonomic relations*. The main task that involves taxonomic relations is to *construct hierarchies*. Organizing concepts into a hierarchy requires the discovery of is-a relations (i.e., hy- pernym/ hyponym) [Cimiano et al. 2004], and hence, some researchers may also refer to this task as *extracting taxonomic relations*. Hierarchy construction can be performed in various ways, such as using predefined relations from existing background knowledge, using statistical subsumption models, relying on semantic similarity between con- cepts, and utilizing linguistic and logical rules or patterns. Non-taxonomic relations are the interactions between concepts (e.g., meronymy, thematic roles, attributes, pos- session, and causality) other than hypernymy. The less explicit and more complex use of words for specifying relations other than hypernymy causes the tasks of *discovering non-taxonomic relations* and *labeling non-taxonomic relations* to be more challenging. Discovering and labeling non-taxonomic relations are mainly reliant on the analysis of syntactic structures and dependencies. In this aspect, verbs are taken as good indicators for non-taxonomic relations, and help from domain experts may be required to label such relations.

Axioms

Lastly, axioms are propositions or sentences that are always taken as true. Axioms act as a starting point for deducing other truth, verifying correctness of existing ontological elements, and defining constraints. The task involved here is of *discovering axioms*. The task of learning axioms involves the generalization or deduction of a large number of known relations that satisfy certain criteria.

3.3. Techniques for Ontology Learning

Bootstrapping is a popular approach used to kickstart the construction of ontologies based on some user-provided resources, also known as *seeds*.

3.3.1. Statistics-Based Techniques

The lack of consideration for the underlying semantics and relations between the components of a text makes statistics-based techniques more prevalent in the early stages of ontology learning, such as term extraction and hierarchy construction.
Some of the common techniques include *clustering* [Wong et al. 2007], *latent semantic analysis* [Turney 2001], *cooccurrence analysis* [Budanitsky 1999], *term subsumption* [Fotzo and Gallinari 2004], *contrastive analysis* [Velardi et al. 2005], and *association rule mining* [Srikant and Agrawal 1997]. The main idea behind these techniques is that the (co-)occurrence of lexical units2 in samples often provides a reliable estimate about their semantic identity to enable the creation of higher-level entities.

3.3.2. Linguistics-Based Techniques and Resources. Linguistics-based techniques are applicable to almost all tasks in ontology learning and are mainly dependent on natural language processing tools. Some of the techniques include *part-of-speech tagging*, *sentence parsing*, *syntactic structure analysis*, and *dependency analysis*. Other techniques rely on the use of *semantic lexicon*, *lexico-syntactic patterns*, *semantic templates*, *subcategorization frames*, and *seed words*.
The Stanford Parser [Klein and Manning 2003] is a lexicalized probabilistic parser.

3.3.3. Logic-Based Techniques and Resources. Logic-based techniques are the least com- mon in ontology learning and are mainly adopted for more complex tasks involving relations and axioms.
The two main techniques em- ployed are *inductive logic programming* [Lavrac and Dzeroski 1994; Zelle and Mooney 1993] and *logical inference* [Shamsfard and Barforoush 2004].

3.4. Evaluation of Ontology Learning Techniques

task-based evaluation
corpus-based evaluation
criteria-based evaluation

Evaluations can be performed to assess the (1) correctness at the terminology layer, (2) coverage at the conceptual layer, (3) wellness at the taxonomy layer, and (4) adequacy of the non-taxonomic relations.

(1)*Lexical recall (LR)* measures the number of relevant terms extracted (*erelevant*) divided by the total number of relevant terms in the benchmark (*brelevant*), while *lexical precision (LP)* measures the number of relevant terms extracted (*erelevant*) divided by the total number of terms extracted (*eall*).

$$LP = \frac{e_{relevant}}{e_{all}},$$

$$LR = \frac{e_{relevant}}{b_{relevant}}.$$

$$F_\beta = \frac{(1+\beta^2)(precision \times recall)}{\beta^2 \times precision + recall}.$$

(2)Evaluation measures at the conceptual level are concerned with whether the desired domain-relevant concepts are discovered or otherwise. *Lexical overlap (LO)* measures the intersection between the discovered concepts (*Cd*) and the recommended concepts (*Cm*). LO is defined as

$$LO = \frac{|C_d \cap C_m|}{|C_m|}.$$

Ontological improvement (OI) and *ontological loss (OL)* are two additional measures to account for newly discovered concepts that are absent from the benchmark and for concepts which exist in the benchmark but were not discovered, respectively.

$$OI = \frac{|C_d \backslash C_m|}{|C_m|},$$

$$OL = \frac{|C_m \backslash C_d|}{|C_m|}.$$

(3)The similarity of the concepts' positions in the learned taxonomy and in the benchmark is used to compute the local measure. The global measure is then derived by averaging the local scores for all concept pairs. One of the few measures for the taxonomy layer is the *taxonomic overlap (TO).*
The *semantic cotopy*, that is, the set of all super- and sub- concepts of a term, varies depending on the taxonomy. The local similarity between two taxonomies given a particular term is determined based on the overlap of the term's semantic cotopy. The global taxonomic overlap is then defined as the average of the local overlaps of all the terms in the two taxonomies. The same idea can be applied to compare adequacy non-taxonomic relations.

4. AN OVERVIEW OF PROMINENT ONTOLOGY LEARNING SYSTEMS

System	Output					Technique / Resource				Evaluation Metrics
	Terms	Concepts	Taxonomic relations	Non-taxonomic relations	Axioms	Statistics-based	Linguistics-based	Logic-based		
ASIUM (2000)							Sentence parsing, Syntactic structure analysis, Subcategorization frames			Precision measure for term extraction
						Agglomerative Clustering				
Text-to-Onto (2000)							Part-of-speech tagging, Sentence parsing, Syntactic structure analysis			F-measure and generic relations learning accuracy (RLA) for non-taxonomic relation extraction
						Co-occurrence analysis	Concepts from domain lexicon			
						Agglomerative Clustering	Hypernyms from WordNet, Lexico-syntactic patterns			
						Association rule mining				
TextStorm/ Clouds (2001)							Part-of-speech tagging using WordNet, Syntactic structure analysis, Anaphora resolution			Accuracy measure for binary predicate extraction
									Inductive logic programming	
SYNDIKATE (2001)							Syntactic structure analysis, Anaphora resolution			F-measure for relation extraction
							Use of semantic templates and domain knowledge	Inference engine		Accuracy measure for concept extraction
OntoLearn (2002)						Relevance analysis	Part-of-speech tagging, Sentence parsing			F-measure for term extraction
							Concepts and glossary from WordNet			
							Hypernyms from WordNet			
CRCTOL (2005)						Relevance analysis	Part-of-speech tagging, Sentence parsing, Use of domain lexicon, Word sense disambiguation			F-measure for term and relation extraction
							Lexico-syntactic patterns, Syntactic structure analysis			
OntoGain (2010)							Part-of-speech tagging, Shallow parsing, Relevance analysis			Precision measure for concept extraction, hierarchy construction, and non-taxonomic relation extraction
						Agglomerative Clustering, Formal concept analysis				
						Association rule mining				

4.1. ASIUM

—Preprocess texts and discover subcategorization frames.
—Extract terms and form concepts.
—Construct hierarchy.

4.2. Text-to-Onto
—Preprocess texts and extract terms.
—Form concepts.
—Construct hierarchy.
—Discover non-taxonomic relations and label non-taxonomic relations.

4.3. TextStorm/Clouds
—Preprocess texts and extract terms.
—Construct hierarchy, discover non-taxonomic relations, and label non-taxonomic relations.
—Extract axioms.

4.4. SYNDIKATE
—Extract terms.
—Form concepts, construct hierarchy, discover non-taxonomic relations, and label non-taxonomic relations.

4.5. OntoLearn
—Preprocess texts and extract terms.
—Form concepts.
—Construct hierarchy.

4.6. CRCTOL
—Preprocess texts.
—Extract terms and form concepts.
—Construct hierarchy and discover non-taxonomic relations.

4.7. OntoGain
Unsupervised acquisition of ontologies from unstructured text.
—Preprocess texts. The OpenNLP suite of tools and the WordNet Java Library are first used for tokenization, lemmatization, part-of-speech tagging, and shallow parsing .
—Extract terms and form concepts. OntoGain implements the existing C/NC-value measure for extracting compound or nested multi- word terms. The C-value of a term *t* is given by

$$Cvalue(t) = \begin{cases} \log_2 |t| f_t & \text{if } |t| = g \\ \log_2 |t| \left(f_t - \frac{\sum_{l \in L_t} f_l}{|L_t|} \right) & \text{otherwise,} \end{cases}$$

where |*t*| is the number of words that constitute *t*; *Lt* is the set of potential longer term candidates that contain *t*; *g* is the longest n-gram considered; and *ft* is the frequency of occurrences of *t* in the corpus.
The C-value measure is based upon the notion that a substring of a term candidate is a candidate itself, given that it demonstrates adequate independence from the longer version in which it appears.
—Construct hierarchy and discover non-taxonomic relations.
The authors implemented agglomerative clustering into OntoGain to build a hierarchy. As usual, each term is considered as a cluster initially, and with each step, clusters are merged based on a similarity measure. A lexical-based group average measure similar to the Dice-like coefficient that incorporates the constituents in multi-word terms is used.

$$sum(x, y) = \frac{|C(x_h) \cap C(y_h)|}{|C(x_h)| + |C(y_h)|} + \frac{|C(x) \cap C(y)|}{|C(x)| + |C(y)|},$$

Where *xh* and *yh* are the heads of term *x* and term *y*, respectively, and their set of constituents is denoted by *C*. *Formal concept analysis (FCA)* is also used to build hierarchies in OntoGain. A *formal contexts* matrix containing a set of formal objects, which are the extracted multi-word terms, and also containing attributes, which are the associated verbs identified during shallow parsing, is provided as input to the FCA algorithm.
Association rule mining is used to discover non-taxonomic relations. The predictive apriori algorithm implementation on the Weka platform7 is used for this purpose.