GUIDING ONTOLOGY LEARNING AND POPULATION BY KNOWLEDGE SYSTEM GOALS

Rosario Girardi
Computer Science Department, Federal University of Maranhão, São Luís, MA, Brazil

Keywords: Knowledge engineering, Ontology learning, Ontology population, Ontology development.

Abstract: This article discusses the motivation and proposes a new process for learning and population of application ontologies which is entirely guided by the goals of the knowledge system being developed and emphasizes the acquisition of the ontology axioms as a first step in the process.

1 INTRODUCTION

Knowledge representation formalisms, like ontologies, are used by modern knowledge systems, to represent and share the knowledge of an application domain (Russel, 1995). Supporting semantic processing, they allow for more precise information interpretation. Thus, knowledge systems can provide greater usability and effectiveness than traditional information systems.

Traditionally, the development of knowledge bases has been performed manually by domain experts and knowledge engineers. However, this is an expensive and error prone task. An approach for overcoming this problem is the automatic or semi-automatic construction of ontologies, a field of research that is usually referred to as ontology learning and population (Cimiano, 2006).

With few exceptions, existing proposals for ontology learning and population adopt similar processes to the ones used for the manual construction of reusable ontologies (mainly top-level, task and domain ontologies) (Gómez-Pérez, 2004) and therefore, they concentrate on the identification, in this order, of classes, hierarchies and relationships without providing appropriate solutions for the acquisition of axioms. In spite of the valuable contributions of these proposals, we consider that the manual construction of good-quality reusable ontologies is still an open problem and therefore, the feasibility of automating their construction is still limited. For that reason we believe that ontology learning and population techniques and processes should first approach the automatic or semi-automatic construction of application ontologies, that is, non-reusable ontologies to be used as knowledge bases of a particular knowledge system. We argue that reusable ontologies could be better constructed in a bottom-up approach as abstractions of specific application ontologies.

On the other hand, axioms are central components of application ontologies because, along with relationships, they specify the goals and constraints of a knowledge system. Therefore, we critically argue that axioms should be directly derived from the requirements of the knowledge system to be developed and, therefore, should be extracted early in the development process. Moreover, development processes for ontology learning should be integrated or, at least, consider current advances made in development methodologies for modern knowledge systems like agent-oriented systems (Girardi, 2010).

In this paper, we develop the ideas above and propose a first approach for learning and population of application ontologies which considers the extraction of all ontology elements guided by the goals of the knowledge system being constructed.

This paper is structured as follows. Firstly, in Section 2, we distinguish data from information and we discuss how they can be used for knowledge representation. Next, we review some important concepts relating ontologies to current approaches for learning and population. In section 3, we present supporting ideas that would validate our hypothesis about the construction (or the extension) of an ontology in the context of the development of a particular knowledge system. Section 4 concludes the article with some remarks on further work being developed.

Girardi R..
GUIDING ONTOLOGY LEARNING AND POPULATION BY KNOWLEDGE SYSTEM GOALS .
DOI: 10.5220/0003119404800484
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2010), pages 480-484
ISBN: 978-989-8425-29-4

2 KNOWLEDGE AND ITS REPRESENTATION ON ONTOLOGIES

According to their abilities for processing data, information and knowledge, software systems have evolved from data processing to information to knowledge systems.

There is not a consensus of what exactly distinguishes data from information from knowledge (Stenmark, 2001). We consider data as an uninterpreted term and knowledge as derived from information. Information consists of concrete facts, assertions giving a meaning to data terms (for instance, "Socrates is a man") and to relationships between terms (for instance, "Plato wrote about Socrates"). Knowledge is constructed upon logical rules, conditional prepositions which provide the basic factual information from which useful conclusions (axioms) can be derived through some inference procedure (Russel, 1995). Thus, the axiom stated by the rule "If someone is a man then he is mortal" provides the knowledge that "All men are mortal". This is an example of a constraint axiom illustrating how knowledge can be derived from information which can be extracted from similar recurring concrete factual information (patterns). Axioms can also be factual information and could also be derived from other axioms. For instance, the classical silogism "Socrates is a man. All men are mortal. Therefore, Socrates is mortal" illustrates the example of an axiom, knowledge representing the information that "Socrates is mortal" derived from the knowledge that "All men are mortal" and from the information that "Socrates is a man".

Ontologies (Gruber, 1995) are structures particularly appropriate for representing both knowledge and information about a problem or domain in different abstraction levels thus allowing its reuse and easy extension.

2.1 An Ontology Definition

An ontology can be defined as the tuple:

$$O = (C, H, R, P, I, A). \qquad (1)$$

where,

$C = C^C$ U C^I is the set of entities of the ontology. The C^C set consists of classes, i.e., concepts that represent entities (for example, "Person" $\in C^C$) describing a set of objects, class instances in the C^I set (for example "Erik" $\in C^I$).

$H = \{kind\_of(c_1,c_2) \mid c_1 \in C^C,\ c_2 \in C^C\}$ is the set of taxonomic relationships between concepts, which define a concept hierarchy and are denoted by "kind_of(c_1,c_2)", meaning that c_1 is a subclass of c_2, for instance, "kind_of(Lawyer,Person)".

$R = \{rel_k(c_1,c_2,\ldots, c_n) \mid \forall_i,\ c_i \in C^C\}$ is the set of non-taxonomic ontology relationships like "represents(Lawyer, Client)".

$P = \{prop^C(c_k,datatype) \mid c_k \in C^C\}$ is the set of properties of ontology entities. The relationship $prop^C$ defines the basic datatype of a class property. For instance, subject (Case, String) is an example of a $prop^C$ property.

$I = \{is_a\ (c_1,c_2) \mid c_1 \in C^I,\ c_2 \in C^C\} \cup \{prop^I(c_k,value) \mid c_k \in C^I\} \cup \{rel_k(c_1,c_2,\ldots, c_n) \mid \forall_i,\ c_i \in C^I\}$is the set of instance relationships related to the C^C (eg. "is_a (Anne,Client)"), P (eg. "subject (Case12, "adoption")") and R (eg. "represents(Erik, Anne)") sets.

$A = \{condition_x \Rightarrow conclusion_y\ (c_1,c_2,\ldots, c_n) \mid \forall j,\ c_j \in C^C\}$ is a set of axioms, rules that allow checking the consistency of an ontology and infer new knowledge through some inference mechanism. The term $condition_x$ is given by $condition_x = \{(cond_1,cond_2,\ldots,cond_n) \mid \forall z,\ cond_z \in H \cup I \cup R\}$. For instance, "∀Defense_Argument, OldCase, NewCase, applied_to(Defense_Argument, OldCase), similar_to (OldCase, NewCase) ⇒ applied_to (Defense_Argument, NewCase)" is a rule that indicates that if two legal cases are similar then, the defense argument used in one case could be applied to the other one.

As an example, consider a very simple ontology describing the domain of a law firm (Figure 1), which has lawyers responsible for cases of the clients they serve.

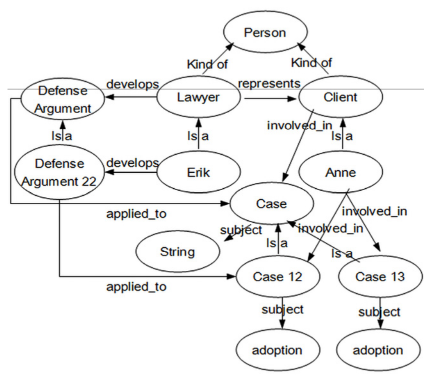

Figure 1: Example of a simples ontology of a law firm.

According to the previous ontology definition, from the ontology in the Figure 1, the following sets can be identified.

C^C = {person, lawyer, client, case}.

C^I = {Erik, Anne, Case12, Case13, DefenseArgument22}.

H = {kind_of(Person, Lawyer), kind_of(Person, Client)}.

I = {is_a(Erik, Lawyer), is_a(Anne, Client), is_a(DefenseArgument22, DefenseArgument), is_a(Case12, Case), is_a(Case13, Case), subject(Case12, "adoption"), subject(Case13, "adoption")}.

R = {represents(Lawyer, Client), applied_to(DefenseArgument, Case), develops (Lawyer, Defense_Argument), involved_in(Client, Case)}.

P = {subject(Case, String)}.

A = ∀Defense_Argument, OldCase,NewCase, applied_to(Defense_Argument, OldCase), similar_to (OldCase, NewCase) ⇒ applied_to (Defense_Argument, NewCase).

2.2 An Ontology Taxonomy

(Guarino, 1998) classifies ontologies into a hierarchy like the one illustrated in Figure 2, according to their level of dependence on a particular task or point of view. Thick arrows represent specialization relationships. Top-level ontologies describe very general concepts which are independent of a particular problem or domain. Domain ontologies and task ontologies describe, respectively, the vocabulary related to a generic domain (like medicine, or automobiles) or a generic task or activity (like diagnosing or selling), by specializing the terms introduced in the top-level ontology. Application ontologies describe concepts depending both on a particular domain and task, which are often specializations of both the related ontologies. These concepts often correspond to roles played by domain entities while performing a certain task, like the diagnosis made by a medical doctor.

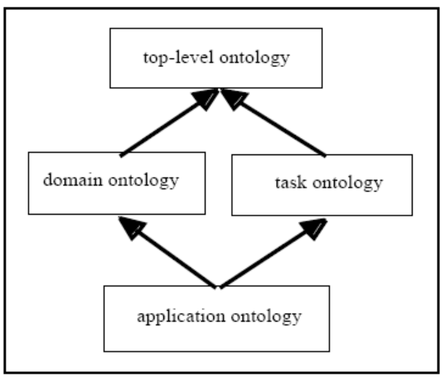

Figure 2: A taxonomy of ontologies (Guarino, 1998).

Considering this taxonomy, ontology-based knowledge systems should be developed by promoting the reuse of already available domain and task ontologies. Therefore, there are currently many research efforts on the development of techniques, methodologies and tools approaching the reuse problems of creating reusable top-level, domain and tasks ontologies as well as their selection, specialization and integration for building application ontologies (Gómez-Pérez, 2004) (Staab, 2009). Thus, the manual construction of good-quality reusable ontologies (and their reuse) is still an open problem. Since this technology is not enough mature to successfully approach the automatic creation of reusable ontologies, we believe that ontology learning and population techniques and processes should first approach the automatic or semi-automatic construction of application ontologies, that is, non-reusable ontologies to be used as knowledge bases of a particular knowledge system and that reusable ontologies could be better constructed in a bottom-up approach as abstractions of specific application ontologies.

2.3 Current approaches for Ontology Learning and Population

Current processes for ontology learning and population from text (Cimiano, 2006) (Shamsfard, 2003) organize their tasks into a set of layers similarly as the one illustrated in Figure 3. Layer tasks looks for acquiring some of the ontology sets in definition 1 by using the sets obtained in the lower layers.

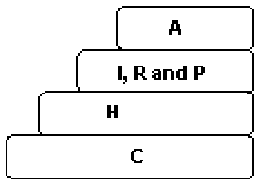

Figure 3: Layers of current ontology learning and population processess.

For years we have been training students on the development of mainly expert systems. It has been difficult for students to identify appropriate classes, hierarchies, properties and relationships without previously stating the goals of the system and considering the system requirements. On the other hand, successful student experiences on the manual construction of knowledge bases have followed an approach rather different than the one of Figure 3 which has been adapted from the knowledge engineering process in first order logic proposed by

(Russel, 1995) emphasizing the early specification of the system goals through the questions that the knowledge base rules needs to support.

Consider, for instance, the construction of the ontology of Figure 1 for building a knowledge system providing decision support for a law firm. A goal of the system could be to recommend a lawyer about defense arguments to be applied in a legal case (the conclusion of the axiom example in Section A: "applied_to (Defense_Argument, NewCase)"). From this goal and considering a strategy that could be undertaken to achieve it: "if two legal cases are similar then, the defense argument used in one case could be applied to the other one" (the axiom example in Section A), several class and relationship candidates could be easily identified, for instance, the "Lawyer", "Defense_Argument" and "Case" classes and the "applied_to" and "similar_to" relationships.

3 A PROCESS FOR ACQUIRING APPLICATION ONTOLOGIES

Figure 4 shows a first approach of a process for learning and population of application ontologies from textual resources. The process is goal-driven, that is, for each system goal corresponding tasks are performed, in this order, for acquiring axioms (A set), relationships and properties (R and P sets), classes (C set), taxonomic relationships (H set) and class-instance relationships (I set), looking for satisfying the goal. However, this task order is not strictly top-down. Bottom-up refinements between layers could happen to improve the effectiveness of the acquired sets. Available domain and tasks ontologies could be reused in each layer.

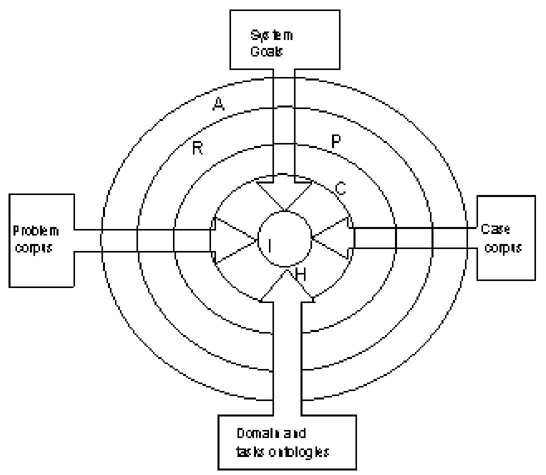

Figure 4: A first proposal of a goal-driven process for learning and population of application ontologies.

We distinguish between two types of corpus used for learning and population purposes. A problem corpus contains a set of documents describing the particular problem to be solved by the knowledge system. For instance, for the development of a decision support system for a law firm specialized in family law, the problem corpus could contain documents in natural language specifying what kind of support the law firm needs and documents about the family law doctrine as well. The problem corpus will be a source for learning all sets excluding the I set. A case corpus contains documents describing problem cases. In the example of the law firm decision support system, a case corpus could be composed of jurisprudence documents, specifying court decisions on family law cases. The case corpus will be the source for acquiring the I set but we are currently also testing its usefulness for acquiring the other ontology sets.

4 CONCLUDING REMARKS

According to our view, ontology learning and population processes should first approach the automatic or semi-automatic construction of application ontologies, that is, non-reusable ontologies to be used as knowledge bases of a particular knowledge system. On the other hand, we critically argue that axioms should be directly derived from the requirements of the knowledge system to be developed and, therefore, should be extracted early in ontology learning processes.

Considering these work hypotheses, we propose a new process for learning and population of application ontologies which is entirely guided by the system goals and emphasizes the acquisition of the ontology axioms as a first step in the process.

Current work looks for improving the process specification taking into account both advances on requirement engineering of multi-agent systems (Girardi, 2010) and ontology and population techniques (Cimiano, 2006) and evaluating the proposal through the development of case studies.

ACKNOWLEDGEMENTS

This work is supported by CNPq, CAPES and FAPEMA.

REFERENCES

Cimiano, P. , 2006. "Ontology Learning and Population from Text: Algorithms, Evaluation and Applications. Springer.

Fernández-López, M., Gómez-Pérez, A., 2002. "Overview and analysis of methodologies for building ontologies," The Knowledge Engineering Review.

Girardi, R., Leite, A., 2010. "Knowledge Engineering Support for Agent-Oriented Software Reuse," In: M. Ramachandran. (Ed.) Knowledge Engineering for Software Development Life Cycles: Support Technologies and Applications. Hershey: IGI Global, in press.

Gómez-Pérez, A., Fernandez-López, M.. Corcho, O. , 2004. "Ontological Engineering," Springer.

Gruber, T. R., 1995. "Toward Principles for the Design of Ontologies used for Knowledge Sharing", International Journal of Human-Computer Studies, nº43, pp. 907-928.

Guarino, N., 1998. "Formal Ontology in Information Systems," Proceedings of the 1st International Conference, Trento, Italy, IOS Press, pp. 3-15.

Russel, S., Norvig, P., 1995. Artificial Intelligence: A Modern Approach, Prentice-Hall.

Shamsfard, M., Barforoush, A. A., 2003. "The state of the art in ontology learning: a framework for comparison," The Knowledge Engineering Review, Vol. 18, pp. 293-316.

Staab, S., Studer, R. (editors), , 2009. "Handbook on Ontologies," Springer Series on Handbooks in Information Systems.

Stenmark, D. , 2001. "The Relationship between Information and Knowledge", in Proceedings of IRIS 24, Ulvik, Norway, August, pp. 11-14..