Chapter 6

Using Similarity-Based Approaches for Continuous Ontology Development

Maryam Ramezani
SAP Research, Germany

ABSTRACT

This paper presents novel algorithms for learning semantic relations from an existing ontology or concept hierarchy. The authors suggest recommendation of semantic relations for supporting continuous ontology development, i.e., the development of ontologies during their use in social semantic bookmarking, semantic wiki, or other Web 2.0 style semantic applications. This paper assists users in placing a newly added concept in a concept hierarchy. The proposed algorithms are evaluated using datasets from Wikipedia category hierarchy and provide recommendations.

INTRODUCTION

Ontologies are known in computer science as consensual models of domains of discourse, usually implemented as formal definitions of the relevant conceptual entities (Hepp, 2007). There are two broad schools of thought on how ontologies are created: the first views ontology development akin to software development as a - by and large - one off effort that happens separate from and before ontology usage. The second view is that ontologies are created and used at the same time, that ontologies are continuously developed throughout their use. The second view is exemplified by the Ontology Maturing model (Braun, Kunzmann, & Schmidt, 2010; Braun, Schmidt, & Walter, 2007) and by the ontologies that are developed in the course of the usage of a semantic wiki (Krötzsch, Vrandecic, Völkel, Haller, & Studer, 2007).

Machine learning, data mining and text mining methods to support ontology development have so far focused on the first schools of thought;
have focused on creating an initial ontology from large sets of text or data that is refined in a manual process before it is then used. For example, recently, many researchers have tried to apply data mining techniques to create ontologies from folksonomies (Heymann & Garcia-Molina, 2006; Balby Marinho, Buza, & Schmidt-Thieme, 2008).

In our work, however, we focus on using machine learning techniques to support continuous ontology development. In particular, we focus on one important decision: given the current state of the ontology, the concepts already present and the sub/super concept relations between them - where should a given new concept be placed? Which existing concept(s) can be considered as the super concept(s) of the new concept?

We investigate this question on the basis of applications that use ontologies to aid in the structuring and retrieval of information resources (as opposed to for example the use of ontologies in an expert system). These applications associate concepts of the ontology with information resources, e.g., a concept “Computer Science Scholar” is associated to a text about Alan Turing. Such system can use the background knowledge about the concept to include Alan Turing page in responses to queries like “important British scholars”. Important examples for such systems are:

- **The Floyd Case Management System:** Developed at SAP. In that system, cases and other objects (that are attached to cases, such as documents) can be tagged freely with terms chosen by the user. These terms can also be organized in a semantic network and this can be developed by the users. The Floyd system is usually deployed with a semantic network initially taken from existing company vocabulary.

- **The SOBOLEO System:** (Zacharias & Braun, 2007) Uses a taxonomy developed by the users for the collaborative organization of a repository of interesting web pages.

- **The (Semantic) Media Wiki system:** (Krötzsch et al., 2007) Uses a hierarchy of categories to tag pages. We can view categories as akin to concepts and support the creation and placement of new categories by proposing candidate super-categories.

All these system are “Web 2.0” style semantic applications; they enable users to change and develop the ontology during their use of the system. The work presented in this paper assists users in this task by utilizing machine learning algorithms. The algorithms suggest potential super-concepts for any new concept introduced to the system.

**RELATED WORK**

According to one of the most cited definitions of the Semantic Web literature, an ontology is a formal, explicit specification of a shared conceptualization (Gruber, 1993). Guarino clarifies Gruber’s definition by adding that the AI usage of the term refers to “an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words”.

John Sowa (Sowa, 2009) classifies ontologies into three kinds. A *formal ontology* is a conceptualization in which categories are described by axioms and definitions. They are stated in logic or in some computer-oriented language that could be automatically translated to logic. In contrast, in *prototype-based ontologies*, categories are distinguished by typical instances or prototypes rather than by axioms and definitions in logic. Categories are formed by collecting instances rather than describing the set of all possible instances in an intensional way, and the most typical members are selected for description. The third kind of ontology is *terminological ontology* which is partially specified by subtype-supertype relations. The concepts are described by concept labels or synonyms rather than prototypical in-