Selected Results of a Comparative Study of Four Ontology Visualization Methods for Information Retrieval tasks

Katifori, Akrivi

IEEE Xplore

http://hdl.handle.net/11728/6474

Downloaded from HEPHAESTUS Repository, Neapolis University institutional repository
Abstract— The need for effective ontology visualization for design, management and browsing has arisen as a result of the progress in the areas of Semantic Web and Personal Information Management. There are several ontology visualizations available through existing ontology management tools, but not as many evaluations to determine their advantages and disadvantages and their suitability for various ontologies and user groups. This work presents selected results of an evaluation of four visualization methods in Protégé.

Index Terms— Visualization, Information retrieval, User interfaces

I. INTRODUCTION

The need for more effective information retrieval has recently lead to the creation of the notions of the semantic web and personalized information management. In many of the proposed solutions in this field, it is common to include the use of an ontology. Consequently, the need for effective ontology visualization for design, management and browsing has arisen.

Numerous research attempts and ontology management tool development efforts have targeted the area of ontology visualization, proposing different approaches to visualizing and facilitating user interaction with ontologies. This work investigates the suitability of four ontology visualization methods for tasks not directly related to ontology management, but rather to information retrieval and for users that are not familiar with the specific content and structure of the visualized ontology. Four commonly used visualization methods, which are representative of the major ontology visualization approaches [8], have been chosen for the presented experiment. In order to assess the appropriateness of each visualization method for different user tasks, we formulated a set of information retrieval tasks and asked a group of users to carry them out, recording task completion times and success rates, as well as other user comments and reactions. The tasks within the set were chosen so as to cover the major task types identified in [8], while their complexity also varied. Tasks involving temporal characteristics (e.g. entity evolution) were also included in the task set, since such tasks often occur in certain contexts, such as information retrieval with the aid of historical archive material ontologies.

The rest of the paper is organized as follows: in section 2 related work is surveyed and useful definitions for ontologies are listed. Section 3 includes brief presentations of the evaluated methods and their characteristics, while section 4 describes the evaluation method and the results obtained from the experiment. Finally, section 6 concludes the paper and outlines future work.

II. BACKGROUND AND DEFINITIONS

An ontology, according to the definition in [1] is a formal explicit description of a domain, consisting of classes, which are the concepts found in the domain. Classes are organized in a specialization/generalization hierarchy through is-a (or inheritance) links, where each class is allowed to have zero, one or multiple parent classes. Each class has properties (or slots) describing various features of the modeled class. Slots are typed, and allowed types are either simple types (strings, numbers, booleans or enumerations) or instances of other classes (references); restriction on the value ranges of slots (e.g. integers from 1 to 10) may also be defined. Finally, instantiation may be applied to classes to produce items corresponding to individual objects in the domain of discourse (instances). Each instance has a concrete value for each property of the class it belongs to. Classes, together with instances are said to constitute the knowledge base.

From the definition above, it is evident that the task of visualizing the full set of ontology features is not an easy one. A number of ontology visualizations exist that have been embedded in ontology management tools (e.g. [2] and [6])