Knowledge-Based Systems 55 (2014) 29–42

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Graph-based semantic annotation for enriching educational content with linked data

Juan C. Vidal *, Manuel Lama, Estefanía Otero-García, Alberto Bugarín

Centro de Investigación en Tecnoloxías da Información (CITIUS), Universidade de Santiago de Compostela, 15782 Santiago de Compostela, Spain

ARTICLE INFO

Article history:
Received 7 February 2013
Received in revised form 31 August 2013
Accepted 4 October 2013
Available online 16 October 2013

Keywords:
Semantic annotation
Linked data
Semantic web
Ontologies
Technology enhanced learning

ABSTRACT

In this paper, a new approach to semantic annotation with linked data in the field of document enrichment is presented. This application has been developed in the domain of Education and contrary to traditional semantic annotation, which relates each relevant term of the document with an instance of the ontology, in our approach relevant terms are connected to a (sub)graph of the ontology. Specifically, each relevant term is related to an instance which is expanded to a predefined depth limit, so the term is finally annotated with a (sub)graph. During the expansion process, instances unrelated with the document topics are ruled out, so only relevant and contextualized information is finally included. As result of this process, the document is annotated with a set of interconnected (sub)graphs, and students may access and navigate through these contents to deepen the topics described in the document. This approach has several benefits. From the document enrichment perspective, a set of (sub)graphs, provides a better description, moreover considering the semantic nature of linked data. From the computational perspective, this approach is also more suitable, particularly in the domain of Education. Filtering linked data is computationally expensive, and thus this process cannot be performed in real time. Our approach has been validated in the e-Learning domain and compared with similar approaches that also annotate with linked data.

1. Introduction

The development of Linked Data (LD) technology [1] in the last few years has propitiated a second youth to semantic annotation (SA) [2]. New open and machine-processable repositories are now accessible lowering thus the cost of the annotation. An important part of this cost was the creation or the integration of the ontologies to annotate the documents, but now this cost is much lower with LD since related data that were not previously linked are accessible using the Web. As a matter of fact, there are nowadays many LD repositories accessible, for almost every domain of application [3], and third party applications are just starting to use LD to enrich or complement their own contents. In fact, the term *Linked Data* refers to a set of best practices for publishing and connecting structured data on the Web, lowering thus the barrier of linking these data.

However, the use of LD has also brought new challenges to SA. In this paper we deal with one of these challenges: how to take advantage of LD-based SA for the enrichment of documents in the domain of Education. The objective here is to use the annotations to provide additional or complementary information [4,5] to the users, or in our case to students, and not only data that are machine processable. Ontologies provide the right means to do this process of enrichment. In fact, each relation of an instance describes a specific property with a semantics and whose target node can be either a data, such as a string or a date, or another instance that is itself described by other properties. This configures a graph-based structure through which the user can easily navigate to get more information about a specific resource. Fig. 1 provides a visual representation of this process of enrichment. As it is depicted, relevant terms of the document are annotated with RDF (sub)graphs extracted from LD, and through which relations the user can navigate and visualize (e.g., by means of a web page template) the information contained in the different nodes/instances of the graph.

The annotation of documents consists of attaching comments, phrases, or tags to a document or to a selected part of a document [6]. SA extends this concept and goes one level deeper to reduce the gap between natural language and its computational representation. Contrary to annotation or tagging, SA tries to match the terms of the document with its semantic representation, that is, terms are associated to an instance/individual of the ontology, which represents in a formal and structured way the knowledge

* Corresponding author. Tel.: +34 881816388; fax: +34 881813602.
E-mail addresses: juan.vidal@usc.es (J.C. Vidal), manuel.lama@usc.es (M. Lama), estefanianatalia.otero@usc.es (E. Otero-García), alberto.bugarin.diz@usc.es (A. Bugarín).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.10.007

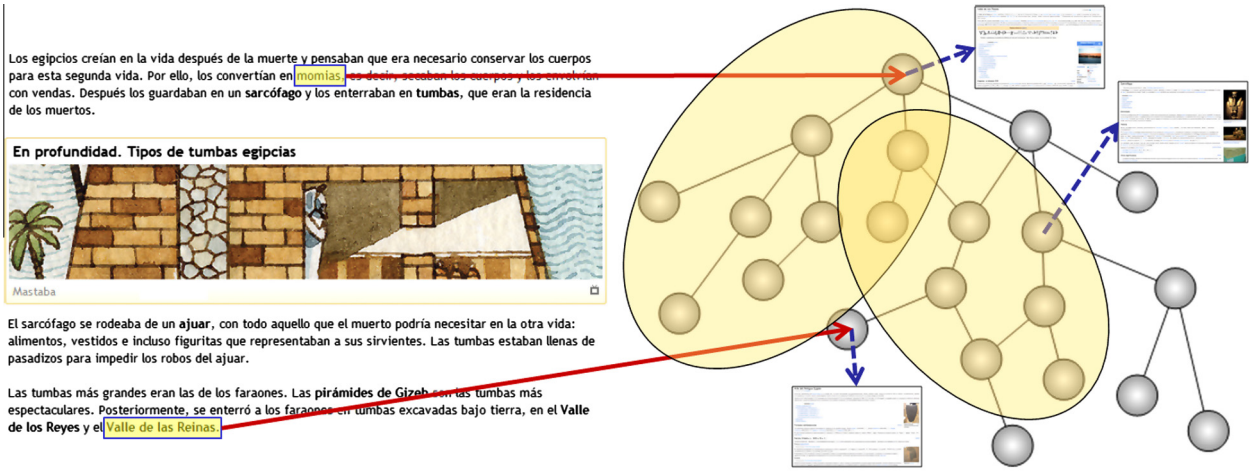

Fig. 1. Semantic enrichment of the relevant terms of a document.

of a domain [2]. However, a same term may be used with different meanings and different terms may also have the same meaning. For instance, the term *Paris* in a document may represent the capital of *France*, a city in Texas (USA), or, for example, the name of a person. Hence, the correct instance in the ontology must be identified to achieve a correct annotation. For this purpose, SA usually exploits the *context* of the document, that is, its relevant terms, to improve the precision during this search process. For example, if the term "France" or the composed word "capital of France" have also been extracted from the document, the identification of the correct instance has an improved probability of success. The advantage of using ontologies here is that the model is defined at the knowledge level [7], and thus data have relations to other data that can be evaluated automatically to reduce the uncertainty of selecting the correct instance. For example, there should be a relation in the ontology that indicates that *Paris* is the capital of *France*. However, the use of LD makes this process even more expensive. LD do have a lot more instances, more data, and more relations, and may be specific to a domain but also the contrary, that is, a cross-domain ontology. Therefore the identification cannot be based on just text-based descriptions or labels, and thus some part of the graph that hangs from the instance must be explored too in order to disambiguate between several instances.

Most of SA approaches using LD annotate terms with *only one instance* of the ontology [8–11]. These approaches make use of text-based data to disambiguate which instance will annotate the term of the document, and some of them even explore relations to other instances and the text-based data of these instances to improve the precision of the annotation. However, the search strategy of these approaches is very limited both in depth, since only few children nodes are visited, and in breadth, since taxonomic relations are not taken into account. Therefore, only a small part of the search space is explored. From the perspective of document enrichment these approaches share another important drawback: the term is directly matched with an only instance. However, in LD an instance may have many information that is not related with the document, which may be an inconvenient in some domains, such as Education, where the complementary information is meant to help students to understand the topics of a document and not to introduce noise in their learning process. For example, if the topic of the document is about "Ancient Egypt" and the term "Egypt" is matched with the instance that represents the country, non-relevant information should be pruned and complementary information should be provided. In this situation, teachers are not interested in providing information regarding contemporary Egypt, but information about egyptian gods, river Nile, and so on. Therefore, it is needed to explore the graph associated to a term to identify the correct information about that term such as Fig. 1 depicts. Notice that this exploration and filtering of information is time-consuming and, therefore, approaches that annotate with only one instance are not well suited since they need to filter this information in runtime. For instance, each time the user wants to explore the contents associated to the document, several SPARQL queries to the LD, in addition to syntactic and semantic analysis, must be performed to filter the information showed to the user.

Taking this into account, more recent approaches start to annotate relevant terms with (sub)graphs of LD [12,13]. These approaches have the working hypothesis that a graph of instances provides a richer semantics than *only one instance*, specifically in document enrichment where these annotations are accessible to users. The novel approach presented in this paper follows this same principle and also annotates the terms of the documents with graphs extracted from the LD. However, our annotation process (i) differs in the way graphs are discovered. Starting from the root nodes of the graphs (the instances that represent some topic of the document), a depth first-based algorithm, called ADEGA, filters each relation considering the frequency of each term in the context, that is, if a data field or an instance is not considered relevant it is pruned from the final graph. The exploration strategy is also extended to (ii) specially include the taxonomic relations, taking thus much more instances into account, such as siblings, parents, or grandparents. Finally, (iii) a method for the assessment of the nodes of the graph is also provided, which considers that not all the relations provide the same information, and thus cannot be weighted the same to determine the relevance of a node, contrary to the other graph-based solutions.

Summarizing, with ADEGA documents are annotated with graphs that are contextualized to the topics of the document they annotate, complementing the document with additional information to facilitate students learning. Furthermore, our approach has two additional benefits. On the one hand, the cost of filtering is during the annotation and thus does not deteriorate the performance during runtime, which is a key factor considering the cost of performing queries and natural language processing in large-sized repositories such as DBpedia. On the other hand, the graphs that annotate the document may even improve the retrieving of this document in the learning environment (or information system) since the instances of the graph can be used to enrich the description of the document. This feature may be particularly interesting if the annotated document is short-sized.