A context-based algorithm for annotating educational content with Linked Data*

Estefanía Otero-García[1], Juan C. Vidal[1], Manuel Lama[1], Alberto Bugarín[1] and José E. Domenech[2]

[1] Depto. de Electrónica e Computación, Universidade de Santiago de Compostela
15782 Santiago de Compostela, Spain
{estefania.otero,juan.vidal,manuel.lama,alberto.bugarin.diz}@usc.es
[2] Netex Knowledge Factory, 15172 Oleiros, A Coruña
jose.domenech@netex.es

Abstract. In this paper we present an approach for annotating and enriching educational contents modeled as Learning Fruits (LFs). LFs are web books described in XML files and created to make more dynamic and flexible the learning process. A way to reduce the cost of creating a LF is to complete its content with information available in the Web. The solution described in this paper combines syntactic and semantic analysis techniques to enrich and annotate the LFs with relevant and reliable data retrieved from the DBpedia repository.

1 Introduction

The education of youth has always been a major concern of society, and a great deal of effort has been spent with the aim of getting the right tools to facilitate the learning process. However, the information society in which we live has brought about the need to adapt traditional methods of learning to new habits and requirements. One of the new trends are the Learning Fruits[1].

Learning Fruits (LFs) are learning pills and interactive activities on topics of the school curriculum. More concretely, LFs are web books that stand out for their innovative format which provide a friendly and interactive interface to facilitate *the access and the navigation* through the course contents. An important feature of LFs is that they provide links to other content that complement and extend the information to the student and the teacher. This step is expensive when implementing a course: *(i)* it involves identifying which parts need to be supplemented with external information, that is, it is necessary to determine the important topics of the LF; and *(ii)* it involves selecting and analyzing the external links in order to determine if they contain accurate information.

In this paper we propose a semantic and context-based approach to minimize the cost of enriching and annotating LFs with external contents. Our solution

* This work was supported by the Ministerio de Educación y Ciencia and the Xunta de Galicia under the projects TSI2007-65677C02-02 and 09SIN065E respectively.

[1] http://www.netex.es/santillana/eng/index.html

is based on the application of semantic technologies *(i)* to identify and annotate the relevant concepts of the LF; and *(ii)* to retrieve the corresponding contents from the web, in our case from Linked Data [1]. With this approach we deal with the drawbacks of other approaches for annotating learning contents [2, 3], because we use semantic data represented through standard and large ontologies. However, since we only want to enrich the LF with relevant data, we need to discriminate important information from which is not. In this work we called *context* of a LF to the set of terms that determine the topics of the course. In the annotation process, this context will establish the degree of relevance of the concepts and relations filtered from the Linked Data repository, in our case the DBpedia [4], and therefore will influence the creation of the the most appropriate graph to annotate the LF terms. In other words, we will use the context of the LF to filter the RDF triples that describe its content.

The paper is structured as follows: in Section 2 we describe the sequence of tasks to obtain the topics that characterize a LF. In section 3 we detail the algorithm we implemented to retrieve the most appropriate (sub)graph of the DBpedia for the topics of the course. Finally, in Section 4 we point out some results and the conclusions.

2 Learning Fruit Context

Figure 1 depicts the sequence of processing that must be executed to obtain the LF context. The first step is to parse the LF XML document in order to get its most representative *fields*, such as title, sections, paragraphs, etc. Once identified, we weight the content according to the field where it has been found, because the relevance of a term vary depending on the field in which it appears. For example, terms in the title or marked in bold are more representative than those of appearing in a paragraph. The result of this step is a document made up of fields whose content is classified and weighted according to where it has been located.

From this document, we analyse the *morphology*, *similarity*, and *frequency* of the terms in order to determine the most relevant ones, and so characterize the context of the LF:

– *Morphological analysis.* The morphological analysis is carried out with the GATE tool [5], and it is used to determine the grammatical category of each word in the document. This analysis affects the LF context creation in two points:

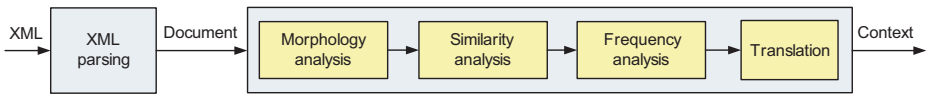

Fig. 1. Sequence of tasks to obtain the context of a Learning Fruit

  - Terms that are not representative to characterize the document content or are not included in the DBpedia will be ruled out. For example verbs, conjunctions, prepositions or determiners are never included in the context.
  - Identification of composite terms. For example, terms like *Ancient Egypt* or *Ramesses II* cannot be considered separately, and they are detected as regular expressions through the JAPE rules of the GATE tool.
- *Similarity analysis.* Since a term may appear in different forms, we create clusters of terminological similarity in order to increase the frequency of occurrence of a word, and also to avoid words that share the same meaning or arise from the same word. In our approach, we use the metrics Monge-Elkan [6] and Jaro-Winkler [7] to calculate the similarity among the document words, because these return adequate values for words with a common root. Thus, two given strings s and t are divided into substrings $s = a_1 \dots a_K$ and $t = b_1 \dots b_L$, and the similarity between those words is calculated as:

$$sim(s,t) = \frac{1}{K} \sum_{i=1}^{K} \max_{j=1}^{L} sim'(A_i, B_j) \tag{1}$$

  where sim' is a secondary distance function, in our case the two metrics mentioned above.
- *Frequency analysis.* The frequency is a quantitative measure which provides the number of occurrences of a term in the LF document. It indicates the relevance of a term within the document and, therefore, it must be combined with the field weights to obtain a relative weighted frequency for each document term:

$$f_i = \frac{\sum_{j=0}^{K}(n_{ij} * p_j) + s}{N} \tag{2}$$

  where n_{ij} is the number of occurrences of the term i in the field j of the document; p_j is the weight of the field; N is the sample size; and s is the number of similar terms of the term under analysis. It is important to remark that the similarity analysis is used to calculate the relative frequency of each document term.

  As a first approximation, we consider terms whose relative weighted frequency is between 4% and 15%: terms are discarded if this frecuency is less than 4%, because they are not representatives, or greater than 15%, because we consider them too general.

The last step for obtaining the LF context is the translation of terms to the English language. Although some of the properties of concepts in the DBpedia are multi-language, most of them are only in English. Thus, if the LF is in a different language, a translation should be performed to obtain better results. Table 1 shows the context of a LF written in Spanish, whose subject is the *Ancient Egypt*. Note that although all the terms of the Table 1 are included in the context, some of them are too general. For example, the term *land* is not relevant in the domain of the Ancient Egypt.

Table 1. Context of the LF about *Ancient Egypt*

Term	**Translation**	**Relevance**
Osiris	Osiris	0.054814816
Horus	Horus	0.057777777
templo	temple	0.05925926
tumba	tomb	0.062222224
Cleopatra VII	Cleopatra VII	0.06518518
Ra	Ra	0.06962963
sacerdote	priest	0.07111111
dios	god	0.072592594
tierras	**lands**	**0.07703704**
faraón	Pharaoh	0.07851852
Ramsés II	Ramses II	0.08592593
Pirámides de Gizeh	Pyramids of Giza	0.08888889
Nilo	Nile	0.093333334
Egipto	Egypt	0.11111111
Antiguo Egipto	Ancient Egypt	0.14222223

3 Semantic Filtering of Linked Data

The objective of the semantic filtering is to select the DBpedia nodes that are relevant to annotate the terms of the context that characterizes the LF document. As it is depicted in Figure 2, to get this objective the first step is to identify the DBpedia URI (resource) that match each context term. Once this step is executed through the DBpedia lookup service we need to deal with two issues: *(i)* the lookup service may retrieve many URIs for a given keyword, but a term can only be paired with a single URI; and *(ii)* not all the relationships that describe the URI are relevant to annotate the LF context term. For example, if the LF is about the *Ancient Egypt*, we are not interested in relationships with URIs that describe contemporary facts or persons.

To solve these two issues each URI is expanded to asses whether the node deserves to be considered. This expansion process is an *iterative deepening depth-first algorithm* [8], which carries out a detailed search through the semantic DBpedia graph until a depth limit. For each URI we perform a SPARQL query and retrieve all its relationships; that is, all its related RDF triples. Thus, according to the type of the object of those RDF triples, we take the following actions:

– *If the object is a literal*, we analyze the literal to check the relevance of the relationship. We consider that a literal is related to the LF if it contains any of the context terms, and assess its relevance with the similarity measures described in Section 2. This analysis also takes into account the relative frequency of the terms of the context and uses a threshold to consider which relations are relevant or not.

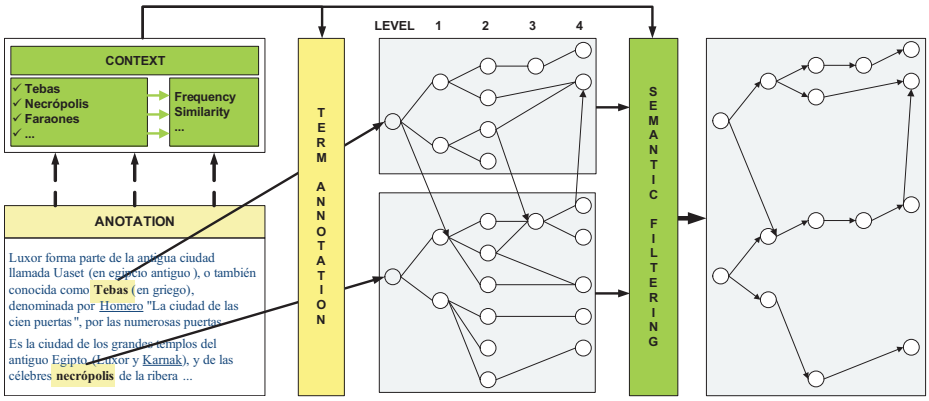

Fig. 2. Filtering process to annotate LF documents

– *If the object is an URI* and we have not reached the depth limit, we continue the exploration through this URI. At this point we have distinguished between two types of URIs:

  - Those that describe a concept. For example, the term *Pharaoh* is represented with the URI *http://dbpedia.org/resource/Pharaoh*.
  - Those that defines a category that allows to classify the resource. For example, *http://dbpedia.org/resource/Category:Ancient_Egypt_titles* specifies the category *Ancient Egypt titles* in which the resource identified by *http://dbpedia.org/resource/Pharaoh* is classified.

  In the case of categories, the algorithm adds a new behavior: if the search process retrieves a resource that shares one of the categories of the resource from which the expansion was realized, we consider this category relevant. Therefore, the category is expanded, which means that URIs with this category will also be processed in our filtering process.

Figure 3 shows the result obtained when this algorithm is applied to the LF about the *Ancient Egypt*. In this example, we have retrieved 1579 RDF triples for the 15 terms that compose the LF context.

4 Conclusions

In this paper we described two of the key processes for enriching the contents of LFs with information extracted from the DBPedia. The first process identifies the main topics of the LF, by means of the combination of frequency, similarity and morphology analysis. From the result of this first process, a filtering process retrieves from the DBpedia the most suitable (sub)graphs to annotate the terms of the LF.

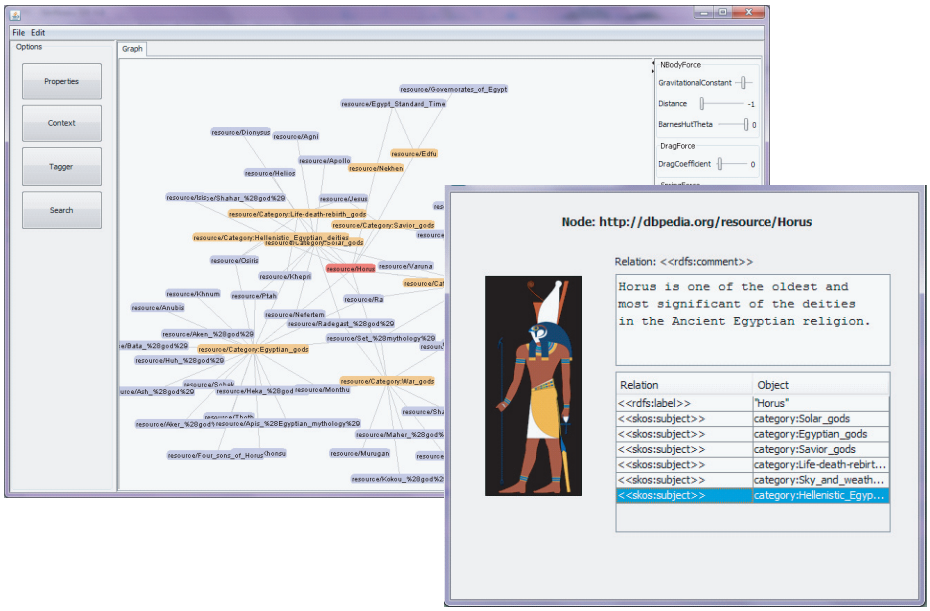

Fig. 3. Screenshot of the application to annotate LF documents

References

1. Bizer, C., Heath, T., Berners-Lee, T.: Linked data: The story so far. International Journal on Semantic Web and Information Systems **5**(3) (2009) 1–22
2. Jovanovic, J., Gasevic, D., Devedzic, V.: Ontology-based automatic annotation of learning content. International Journal on Semantic Web and Information Systems **2**(2) (2006) 91–119
3. Simov, K., Osenova, P.: Applying ontology-based lexicons to the semantic annotation of learning objects. In: Proceedings of the RANLP-Workshop on Natural Language Processing and Knowledge Representation for eLearning Environments, Borovets, Bulgaria (2006) 49–55
4. Bizer, C., Lehmann, J., andSören Auer, G.K., Becker, C., Cyganiak, R., Hellmann, S.: Dbpedia: A crystallization point for the web of data. Journal of Web Semantics **7**(3) (2009) 154–165
5. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: A framework and graphical development environment for robust nlp tools and applications. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistic (ACL'02), Philadephia, USA (2002) 168–175
6. Monge, A., Elkan, C.: An efficient domain-independent algorithm for detecting approximately duplicate database records. In: Proceedings of the SIGMOD-Workshop on Research Issues on Data Mining and Knowledge Discovery, Tucson, USA (2003)
7. Cohen, W.W., Ravikumar, P.D., Fienberg, S.E.: A comparison of string distance metrics for name-matching tasks. In: Proceedings of the IJCAI-Workshop on Information Integration on the Web (IIWeb'03), Acapulco, Mexico (2003) 73–78
8. Russell, S.J., Norvig, P.: Artificial Intelligence: A modern approach. 3rd edn. Prentice Hall (2009)